Promotor
Prof. Dr. B. De Baets
Research Unit Knowledge-based Systems
Department of Applied Mathematics, Biometrics and Process Control
Ghent University

Dean
Prof. Dr. ir. H. Van Langenhove

Rector
Prof. Dr. P. Van Cauwenberge

ir. Ester Van Broekhoven

Monotonicity aspects of linguistic fuzzy models

Thesis submitted in fulfilment of the requirements for the degree of
Doctor (Ph.D.) in Applied Biological Sciences

Dutch translation of the title:
Monotoniteitsaspecten van linguïstische vage modellen

Please refer to this work as follows:
Van Broekhoven, E., 2007. Monotonicity aspects of linguistic fuzzy models. Ph.D. thesis, Ghent University, Gent, Belgium.

ISBN-number: 9789059891609
The author and the promotor give the authorisation to consult and to copy parts of this work for personal use only. Every other use is subject to the copyright laws. Permission to reproduce any material contained in this work should be obtained from the author.

Legende von der Entstehung des Buches Taoteking auf dem Weg des Laotse in die Emigration

1
Als er siebzig war und war gebrechlich
Drängte es den Lehrer doch nach Ruh
Denn die Güte war im Lande wieder einmal schwächlich
Und die Bosheit nahm an Kräften wieder einmal zu.
Und er gürtete den Schuh.
2
Und er packte ein, was er so brauchte:
Wenig. Doch es wurde dies und das.
So die Pfeife, die er immer abends rauchte
Und das Büchlein, das er immer las.
Weißbrot nach dem Augenmaß.
3
Freute sich des Tals noch einmal und vergaß es
Als er ins Gebirg den Weg einschlug.
Und sein Ochse freute sich des frischen Grases
Kauend, während er den Alten trug.
Denn dem ging es schnell genug.
4
Doch am vierten Tag im Felsgesteine
Hat ein Zöllner ihm den Weg verwehrt:
"Kostbarkeiten zu verzollen ?" – "Keine."
Und der Knabe, der den Ochsen führte, sprach: "Er hat gelehrt."
Und so war auch das erklärt.
5
Doch der Mann in einer heitren Regung
Fragte noch: "Hat er was rausgekriegt ?"
Sprach der Knabe: "Daß das weiche Wasser in Bewegung
Mit der Zeit den mächtigen Stein besiegt.
Du verstehst, das Harte unterliegt."
6
Daß er nicht das letzte Tageslicht verlöre
Trieb der Knabe nun den Ochsen an.
Und die drei verschwanden schon um eine schwarze Föhre
Da kam plötzlich Fahrt in unsern Mann
Und er schrie: "He, du ! Halt an !
7
Was ist das mit diesem Wasser, Alter ?"
Hielt der Alte: "Intressiert es dich ?""
Sprach der Mann: "Ich bin nur Zollverwalter
Doch wer wen besiegt, das intressiert auch mich.
Wenn du's weißt, dann sprich !

8
Schreib mir’s auf! Diktier es diesem Kinde !
So was nimmt man doch nicht mit sich fort.
Da gibt’s doch Papier bei uns und Tinte
Und ein Nachtmahl gibt es auch: ich wohne dort.
Nun, ist das ein Wort ? ”
9
Über seine Schulter sah der Alte
Auf den Mann: Flickjoppe. Keine Schuh.
Und die Stirne ein einzige Falte.
Ach, kein Sieger trat da auf ihn zu.
Und er murmelte: “Auch du ? ”
10
Eine höfliche Bitte abzuschlagen
War der Alte, wie es schien, zu alt.
Den er sagte laut:“Die etwas fragen
Die verdienen Antword.“ Sprach der Knabe:“Es wird auch schon kalt.”
“Gut, ein kleiner Aufenthalt.”
11
Und von seinem Ochsen stieg der Weise
Sieben Tage schrieben sie zu zweit.
Und der Zöllner brachte Essen (und er fluchte nur noch leise
Mit den Schmugglern in der ganzen Zeit).
Und dann war’s soweit.
12
Und dem Zöllner händigte der Knabe
Eines Morgens einundachtzig Sprüche ein
Und mit Dank für eine kleine Reisegabe
Bogen sie um jene Föhre ins Gestein.
Sagt jetzt: kann man höfflicher sein ?
13
Aber rühmen wir nicht nur den Weisen
Dessen Name auf dem Buche prangt !
Denn man muß dem Weisen seine Weisheit erst entreißen.
Darum sei der Zöllner auch bedankt:
Er hat sie ihm abverlangt.
(Bertolt Brecht, 1938)

LEGENDE VAN HET ONTSTAAN VAN HET BOEK TAOTEKING
TIJDENS DE EMIGRATIETOCHT VAN LAOTSE

1
Toen hij zeventig was en zwak op zijn benen
Wou de leraar op rust, hij was moe,
Want het goede in het land was haast weer eens verdwenen
En het kwaad nam gaandeweg weer eens in krachten toe.
En hij bond zijn schoenen toe.

2
En hij pakte in wat hij nodig vond:
Weinig. Wat toch een en ander was.
Ook het pijpje dat hij rookte elke avond
En het boekje dat hij telkens las.
Ook wat witbrood, net van pas.

3
Blij keek hij nog eens het dal in en vergat het
Nu hij de weg naar de bergen insloeg.
Zijn os genoot van het frisse gras, hij at het
Traag terwijl hij de oude man droeg.
Want hem ging het snel genoeg.

4
In het gebergte echter verscheen,
Op de vierde dag, een tollenaar.
"Kostbaarheden aan te geven ?" – "Geen."
En de jongen die de os geleidde, zei: "Hij is leraar."
Dat was dan verklaard zowaar.

5
Maar de man, in opgewekte stemming,
Vroeg toen nog: " Vond hij wel al iets uit ?"
De jongen zei:"Het zachte water in beweging
Haalt de sterkste steen ooit onderuit.
Je begrijpt dat hardheid niets beduidt."

6
Omdat de zon al zwak begon te schijnen
Spoorde de jongen de os nu aan.
Toen het drietal achter een zwarte den zou verdwijnen
Liep onze man plots achter hen aan
En hij riep:"Hé jij! Blijf staan!

7
Hoe zit dat met dat water nou ?"
De oude stopte:"Heb j'er oren naar ?"
De man zei:"Wie van wie wint wil ik wel van jou
Vernemen, al ben ik slechts een tollenaar.
Als jij het weet, verklaar !

8
Schrijf het op, dicteer het aan dit kind hier !
Zoiets neem je toch niet met je mee.
Daar woon ik: je vindt er inkt en papier
En een maaltijd ook, is dat geen goed idee ?
Jij zegt toch niet nee ?"
9
De oude keek om en zag een simpele
Stakker: blootsvoets. Verstelde kledij.
En zijn voorhoofd was een en al rimpel.
Ach, geen winnaar kwam hier naderbij.
En hij mompelde:"Ook jij ? "
10
Om wie iets hoffelijks vraagt te mishagen
Was de oude, naar het leek, te oud.
En toen sprak hij luid: "Zij die iets vragen
Verdienen een antwoord." De jongen zei:"Het wordt ook al koud."
"Goed, een luttel oponthoud."
11
En de wijze stapte af. Zij schreven
Samen zeven dagen na elkaar.
En de tollenaar bracht eten (en hij vloekte amper even
In die dagen op een smokkelaar.)
En toen was het klaar.
12
En de jongen gaf dan op een morgen
Eenentachtig spreuken aan de tollenaar weg.
En met dank voor wat reisgeld en de goede zorgen
Trokken zij omheen die zwarte den op weg.
Kan het echt nog hoffelijker, zeg ?
13
Maar, laten wij niet slechts de wijze prijzen
Wiens naam op het titelblad mocht !
Wijsheid moet men eerst afhandig maken van de wijze.
Danken wij dus ook de tollenaar nog:
Hij vroeg ernaar, zo is het toch.
(Bertolt Brecht, 1938, vertaling: Koen Stassijns en Ivo van Strijten, 1998)

Acknowledgements

One mark of a great educator is the ability to lead students out to new places where even the educator has never been. (Thomas Groome)

First of all, I wish to acknowledge my supervisor Prof. Dr. Bernard De Baets. He gave me the opportunity to work at his research group and arranged a scholarship and additional financial support which allowed me to complete this dissertation. Being first one of two and then the only fuzzy modeller at the search group, I'm very grateful for the numerous chances he gave me to present my work and meet other researchers from the fuzzy modelling domain at international summer schools, conferences and during two research visits abroad. I also wish to thank him for the freedom he gave me in my research and for the thorough reviews of our proceedings and articles as well as this dissertation.

Next, I would like to express my gratitude to Prof. Dr. Ulrich Bodenhofer for hosting me twice at his (former) company Software Competence Center Hagenberg in Austria. Our discussions convinced me that there was a need within the fuzzy modelling community for research on monotonicity and that this interesting topic had hardly been addressed yet. It was his enthusiasm that made me change subject halfway my PhD scholarship and start working on the monotonicity of linguistic fuzzy models. A decision which I never regretted.

I wish to thank the other members of the examination committee, Prof. Dr. ir. D. Botteldooren, Prof. Dr. J. Casillas, Prof. Dr. H. De Meyer, Prof. Dr. ir. V. Pauwels and Prof. Dr. ir. W. Steurbaut for the valuable feedback and suggestions to improve this dissertation.

Voor dwaze dingen heb je aan één hoofd genoeg, om te denken zijn meerdere hoofden nodig. — For foolish things you need only one head, to think you need several. (Jan Keteleer, 2004)

Thanks should go to the people with whom I had the pleasure to cooperate. The ecological case study described in this dissertation was a joined work with Veronique

Adriaenssens. Without the ecological knowledge and data she supplied there would have been no ecological case study. To Sofie Piepers I like to say once more that I very much enjoyed working with her on her Master thesis. I also retain good memories of the discussions on fuzzy modelling I had with Ans Mouton.

Next, I would like to thank the professors, the administrative and technical staff, PhD and Master students at the Department of Applied Mathematics, Biometrics and Process Control for the agreeable atmosphere. I owe special thanks to the colleagues with whom I shared offices. At the beginning of my PhD I enjoyed the company of Saskia Janssens, Koen Maes and Hilde Vernieuwe. After two years and a half I changed offices and joined Michaël Rademaker whom I'm greatly indebted for being such an excellent office companion: one who knows when it is a time for silent working and when it is a time for chatting. I also like to thank those who kept us company for a shorter period, Peter Dawyndt, Sven Degroeve, Karel De Loof, Susana Díaz Vázquez, Pekka Marttinen, Yanet Rodríguez Sarabia, Ramiro Sanchez Lopez, Bram Slabbinck and Willem Waegeman. The people of the Departement of Forest and Water Management, in particular Eva De Clercq and Frieke Vancoillie, should not be missing in this list. A lunch break with them was always a real break !

Also during my stay in Hagenberg, Austria, I was surrounded by nice colleagues. I would like to thank Mario Drobics, Peter Haslinger, Bettina Heise, Johannes Himmelbauer, Prof. E.P. Klement, Felix Kossak, Leila Muresan, Susanne Saminger, Holger Schöner, Gerrit Steinöcker (thanks for the daily lifts from and to Linz !) and Burkhard Zimmermann for their hospitality.

> *One of the symptoms of an approaching nervous breakdown is the belief that one's work is terribly important.* (*Conquest of Happiness*, Bertrand Russell, 1930)

Graag had ik m'n vrienden bedankt om zich vijf jaar lang tevreden te stellen met de uitleg dat vaaglogica de vertaling is van die holle FUZZY LOGIC slogan in reclamefolders voor wasmachines en *iets* is dat gebruikt maakt van grijstinten in plaats van zwart-wit. Zo kon ik, als ik bij jullie was, dit doctoraat volledig uit m'n hoofd zetten en m'n vrije tijd besteden aan wat me ècht nauw aan het hart ligt. Mazurka's dansen. Ontgoocheling en vreugde delen. Frambozenvariëteiten bespreken. Nieuwe plaatsen en culturen ontdekken (in het echt of in fotoboeken). Verjaardagstaart eten. Merci !

Tot slot wil ik hen, die eigenlijk niet met woorden te bedanken zijn, bedanken. Thierry, Liefste, bedankt voor zoveel liefde, voor alle steun, voor de heerlijke momenten samen, voor de aanmoedigende woorden wanneer het onderzoek of het schrijven wat minder vlotte, . . . Mama en papa, bedankt om meer dan een kwarteeuw lang alles in het werk gesteld te hebben om mij een zorgeloos leven te bezorgen en steeds, hoe onbezonnen m'n plannen ook waren, voor me te supporteren.

Ester Van Broekhoven
Gent, maart 2007

CONTENTS

List of Abbreviations

AD	Average Deviation
ATL	AT Least
ATM	AT Most
CCI	Correctly Classified Instances
CFCI	Correctly Fuzzy Classified Instances
COG	Centre Of Gravity
EKOO	Ecologische Karakterisering van Oppervlaktewateren in Overijssel
EU	European Union
MA	Mamdani–Assilian
MOM	Mean Of Maxima
RMSE	Root Mean Square Error
wAD	weighted Average Deviation
WFD	Water Framework Directive

CHAPTER 1
Introduction

"Would you tell me, please, which way I ought to go from here?"
"That depends a good deal on where you want to get to," said the Cat.
"I don't much care where–" said Alice.
"Then it doesn't matter which way you go," said the Cat.
"–so long as I get ***somewhere****," Alice added as an explanation.*
"Oh, you're sure to do that," said the Cat, "if you only walk long enough."
(*Alice's Adventures in Wonderland*, Lewis Carroll, 1865)

1.1 Setting

Linguistic fuzzy modelling is an attractive mathematical framework to formally represent systems for which qualitative knowledge, *i.e.* a linguistic description, is available. In linguistic fuzzy models the knowledge about the system is expressed in words, more specifically in if-then rules such as 'IF the slope is very large AND the coverage by vegetation is low THEN the expected soil loss by erosion is high'. Hence the term *linguistic* fuzzy models. They are referred to as linguistic *fuzzy* models since fuzzy sets are used to incorporate the uncertainty in the definition of the linguistic values 'very large', 'low' and 'high' of the linguistic variables 'slope', 'coverage by vegetation' and 'expected soil loss by erosion' in the model. In contrast to classical set theory where one or zero is assigned to an object (*e.g.* a real value) depending on whether the object is in or not in a set, a fuzzy set is characterized by a membership function which assigns a grade ranging between zero and one to each object to reflect the degree to which an object is 'a member' of the fuzzy set.

The components of a linguistic fuzzy model, *i.e.* the if-then rules, membership functions and mathematical operations used to obtain a model output from an input, can all be based on knowledge from an expert familiar with the system, or can — either partially or completely — be derived from data. The first linguistic fuzzy models, reported in the 1970's and mainly applied as controllers to replace manual control by human operators, were completely designed based on expert knowledge. Later, data-driven model identification of linguistic fuzzy models gained importance. With this shift from knowledge-based to data-driven model identification, the model accuracy, *i.e.* the degree to which the output returned by the model resembles the output in the data set, became the principal model performance measure, while the underlying meaning of the different model components was neglected. Most of these early data-driven identification methods resulted for instance in models applying fuzzy sets with such strange shapes that no meaningful labels as 'very low', 'medium' or 'rather high' could be assigned to them. However, in the last decade their interpretable model structure, *i.e.* the fact that a simple reading of the if-then rules gives insight in the system's behaviour and that a meaning can be assigned to the fuzzy sets, is no longer solely regarded as the property that sets linguistic fuzzy models apart from other modelling techniques, but is also considered their greatest asset. Awareness has grown that the interpretability of a model should be safeguarded or at least be balanced against the model accuracy in the model identification process. A good trade-off between accuracy and interpretability can be obtained by including as much qualitative knowledge as possible, how little this may be, in the data-driven model identification process. When a data-driven identification method results in interpretable linguistic fuzzy models, this method can be used for data mining purposes since the obtained if-then rules and fuzzy sets give insight in the system's behaviour.

Monotonicity is the type of qualitative knowledge that plays a central role in this dissertation. Monotone is hereby interpreted as order-preserving. Man often uses ordered linguistic values when describing a system and human decision making frequently involves monotonicity. A garden is for instance described as 'small', 'medium' or 'large' and a location is considered 'very easy', 'fairly easy' or 'difficult' to reach by public transport, *i.e.* the linguistic values assigned to variables such as 'garden size' and 'accessibility by public transport' are ordered. Consequently, an environmentally conscious person with a green thumb shall be willing to pay more for a house with a large garden in easy reach of a main train station than for an house with a small patio two blocks away from a bus stop. Or, expressed more mathematically, the price this person is willing to pay for a house increases with increasing garden size and increasing accessibility by public transport. Formulated more generally, it is said that the price is monotone in the garden size and the accessibility by public transport.

In the ecological case study described in this dissertation habitat suitability models were developed. Fuzzy ordered classifiers were used to assign fuzzy labels to river sites expressing their suitability as a habitat for a certain macroinvertebrate taxon, given up to three abiotic properties of the considered river site. Ordered linguistic values were assigned to both input and output variables, but the output variable, *i.e.* the habitat suitability, was not necessarily monotone in the input variables. The models were built using expert knowledge and evaluated on data collected in the Province of Overijssel in the Netherlands. In literature only performance measures for (fuzzy) classifiers were found that indicate to which degree objects are assigned to a same class or a different class by the model and in the data set. With these measures a same performance is assigned to models assigning 'small' objects to the class of 'medium' objects and to models assigning 'small' objects to the class of 'very large' objects, *i.e.* these measures do not incorporate the available qualitative knowledge that the output classes are ordered. Therefore, a new performance measure for fuzzy ordered classifiers was introduced, referred to as the average deviation (AD) as it takes **the order of the output classes** into account by returning the average deviation between the position of the class obtained with the model and the position of the class stored in the data set. Furthermore an interpretability-preserving genetic optimization of the membership functions in the input domains, applying once binary-coded and once real-coded genetic algorithms, was carried out.

The second, more methodological half of this dissertation discusses the monotonicity of linguistic fuzzy models. In monotone models, ordered linguistic values are assigned to both input and output variables and **the model output is monotone in all input variables**. Linguistic fuzzy models applying different inference procedures, *i.e.* different procedures to determine the model output corresponding to a given input, were considered. Apart from two existing inference procedures, the Mamdani–Assilian and the implicator-based inference, that can be used but are not specifically designed for monotone models, a new inference procedure for models with a monotone rule base, called ATL–ATM inference, is introduced. This new inference procedure is based on a cumulative interpretation of the rule base. For each inference procedure the model behaviour was investigated for the most commonly applied mathematical operators. Combinations of inference procedures and operators were selected that result in a monotone input-output behaviour for all sets of if-then rules describing a monotone relation between the input variables and the output variable. This selection could be used as a guideline by designers of interpretable monotone linguistic fuzzy models.

1.2 A road map to this dissertation

This dissertation consists of three main parts as shown in Fig 1.1. The first part includes introductions to fuzzy rule-based models and genetic algorithms. In the second part the identification and optimization of a fuzzy ordered classifier for an ecological modelling problem is described. The final part discusses the monotonicity of linguistic fuzzy models.

Part I consists of three chapters. Chapter 2 starts with an introduction to fuzzy sets, one of the main components of fuzzy models. This is followed by the description of the two main types of fuzzy rule-based models: the linguistic fuzzy models, including the Mamdani–Assilian models and the models applying implicator-based inference, on the one hand and the Takagi–Sugeno models, on the other hand. The goal of this chapter is twofold: to familiarize fuzzy modellers with the notation used in this dissertation and to provide other readers with a sufficient stock-in-trade concerning fuzzy modelling. Note that reading the tough Section 2.3.1 is not required to comprehend the inference procedures applied in linguistic fuzzy models. In Chapter 3 a computationally attractive and accurate implementation of the Center of Gravity (COG) defuzzification method, applied in the final step of the Mamdani–Assilian inference procedure,

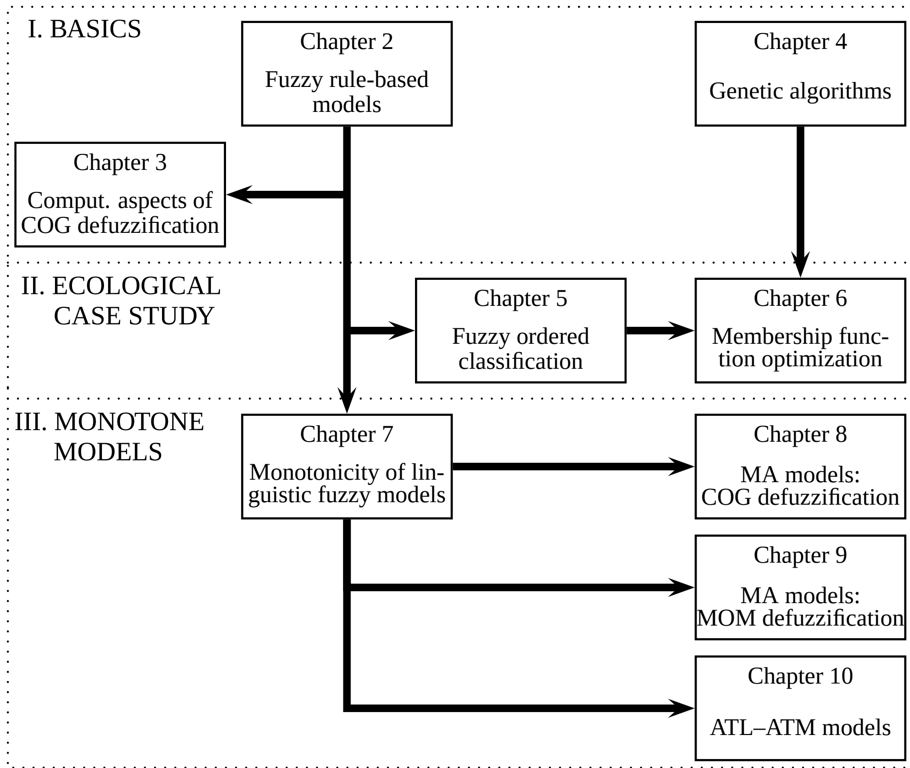


Figure 1.1: A road map to this dissertation.

is introduced and compared to two other implementations. Even if the results described in Chapter 3 were essential for the research written down in Chapter 8, reading Chapter 3 is not essential to comprehend Chapter 8. Chapter 4 discusses the fundamentals of both binary-coded and real-coded genetic algorithms. These optimization algorithms were applied in the ecological case study to optimize the membership functions in the input domains of the habitat suitability models.

Part II deals with the ecological case study carried out in the framework of this dissertation. In Chapter 5 the habitat suitability models, built using expert knowledge described in literature, are described and the data on which the models were evaluated are discussed. Next, the measures used to evaluate the models as well as the results of the model evaluation are presented. Chapter 6 starts with a description of the genetic algorithm applied to optimize the membership functions of 48 selected habitat suitability models, with special attention to the applied representation of candidate solutions and fitness function. Futhermore, the optimization results are discussed.

Part III of this dissertation, consisting of Chapters 7–10, is dedicated to my work on the monotonicity of linguistic fuzzy models. In Chapter 7 some general aspects are discussed, as the applicability of monotone linguistic fuzzy models, the model properties assumed in this work, the applied representation of if-then rules and the issue of incomparable fuzzy model outputs. In Chapters 8–10 the monotonicity of linguistic fuzzy models under different inference procedures is discussed. Chapters 8–9 deal with Mamdani–Assilian models applying the t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ combined with respectively the COG and MOM defuzzification method. Chapter 10 focusses on models applying either plain implicator-based inference or ATL–ATM inference, one of the three basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$, one of the three residual implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ and the MOM defuzzification method. For each inference procedure, combinations of t-norm, implicator or defuzzification method were selected, resulting in a monotone input-output behaviour for any monotone rule base, or at least for any smooth rule base.

The dissertation concludes with general conclusions and suggestions for future research in Chapter 11.

Part I

Basics

CHAPTER 2

Fuzzy rule-based models

Associer le mot flou avec le mot logique est choquant. La logique, au sens vulgaire du mot, est une conception des mécanismes de la pensée qui ne devrait être jamais floue, mais toujours rigoureuse et formelle. [...] La pensée humaine, superposition d'intuition et de rigueur, c'est-à-dire d'une prise en compte globale ou parallèle (nécessairement floue) et d'une prise en compte logique ou séquentielle (nécessairement formelle), est un mécanisme flou. Les lois de la pensée que nous pouvons faire entrer dans les programmes des ordinateurs sont obligatoirement formelles, les lois de la pensée dans le dialogue homme-homme sont floues.
(*Introduction à la Théorie des Sous-Ensembles Flous - Vol. 1: Eléments Théoriques de Base*, Arnold Kaufmann, 1973)

2.1 Introduction

Modelling the behaviour of a system can be done in various ways. The most traditional approach is white-box modelling, which assumes that the system's behaviour is fully known, and there exists a suitable mathematical scheme, for instance a set of differential equations, to represent this behaviour. The requirement for a good understanding of the system shows to be a severe limiting factor in practice, when complex and poorly understood systems are considered. In white-box modelling difficulties can arise from, for instance, poorly understanding the underlying phenomena, inaccurate values of various system parameters, or from the complexity of the resulting model (Casillas et al., 2003a).

In black-box modelling the system under study is represented by a mathematical structure that is sufficiently general to correctly capture the dynamics and the nonlinearity of the system. In this modelling approach, the model identification consists of the selection of an appropriate mathematical structure followed by the estimation of its parameters. If representative data are available, black-box models usually can be developed quite easily, without requiring system-specific knowledge. A severe drawback of this approach is that the structure and parameters of these models usually do not have any physical significance, in other words that these models are not interpretable (Babuška, 1998).

A third, intermediate approach, called grey-box modelling, attempts to combine the advantages of the white-box and black-box approaches, such that the known parts of the system are modelled using a priori knowledge, and the unknown or partially known parts are identified with black-box procedures. A common drawback of most standard modelling approaches is that they cannot make effective use of extra information, such as knowledge of persons who are familiar with the system, information which is often imprecise and qualitative in its nature (Babuška, 1998). The type of grey-box models used in this dissertation, the fuzzy rule-based models, can be identified using quantitative as well as qualitative information (Casillas et al., 2003a).

The main component of fuzzy rule-based models is the fuzzy rule base, containing rules of the form

IF antecedent part THEN consequent part

These if-then rules describe relations between the variables of the system. A fuzzy controller of a heater could for instance contain the following rule: IF temperature is low AND change in temperature is negative THEN strongly increase the power of the heater. The antecedent defines when the rule holds and the consequent describes the corresponding conclusion (in fuzzy models) or desired action (in fuzzy control). The if-then rule of the heat controller contains the linguistic variables ‘temperature’, ‘temperature change’ and ‘power change’. These linguistic variables take linguistic values such as ‘low’, ‘OK’, ‘zero’ and ‘strong increase’. The rule-based nature of the model allows for a linguistic description of the knowledge, which is captured in the model (Sousa and Kaymak, 2002). Studies have been carried out to prove that fuzzy systems are universal approximators, *i.e.* they can uniformly approximate any continuous real function on a compact domain to any degree of accuracy (Buckley, 1993; Campello and do Amaral, 2006; Perfilieva and Kreinovich, 2002; Ying et al., 1999).

Depending on the structure of the rules, two main types of fuzzy rule-based models can be distinguished:

- linguistic fuzzy models, where both the antecedent and consequent contain linguistic values, and
- Takagi–Sugeno models, where the antecedent contains linguistic values and the consequent contains a crisp function of the antecedent variables.

The linguistic values in the rules are defined by fuzzy sets, a concept which is introduced in Section 2.2. The fuzzy sets serve as an interface between the linguistic variables in the model, and the input and output numerical variables.

The rules and fuzzy sets can be identified from data using various techniques such as fuzzy clustering, neural learning methods or genetic algorithms (see (Guillaume, 2001) for an overview). Takagi–Sugeno models are mostly obtained by a

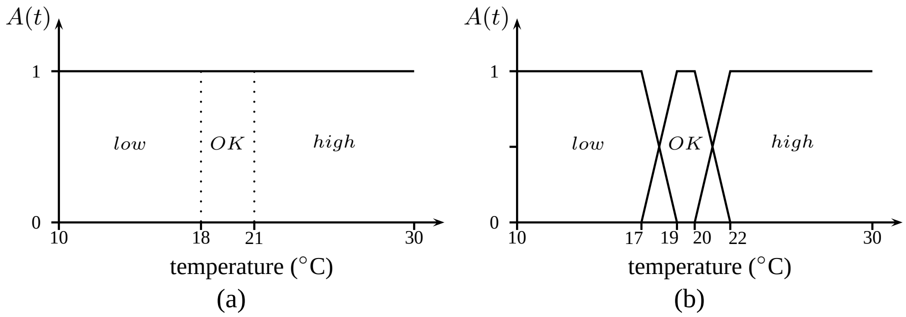


Figure 2.1: Definition of the three linguistic values assigned to temperature by means of (a) crisp and (b) fuzzy sets.

purely data-driven identification, whereas when developing linguistic fuzzy models a knowledge-based identification approach is generally adopted. Data-driven identification methods for fuzzy models used to be focussed on increasing the model's accuracy, paying little attention to the interpretability of the final model. Recently, however, obtaining a good balance between the interpretability and accuracy is gaining importance in fuzzy modelling. Several mechanisms have been proposed to either guarantee the interpretability of a model obtained by purely data-driven identification (Espinosa and Vandewalle, 2000), improve the interpretability of accurate fuzzy models (Casillas et al., 2003a) or improve the accuracy of linguistic fuzzy models with a good interpretability (Casillas et al., 2003b). Linguistic fuzzy models and Takagi–Sugeno models are respectively discussed in more detail in Sections 2.3 and 2.4.

2.2 Fuzzy sets

2.2.1 Crisp sets versus fuzzy sets

In classical set theory, an element either belongs to a set (it has membership degree one to the set) or it does not (it has membership degree zero to the set). In the fuzzy modelling field, the sets used in classical set theory are referred to as *crisp* sets in order to avoid confusion with the *fuzzy* sets used in fuzzy models. To the three linguistic values, 'low', 'OK' and 'high' of the linguistic variable temperature, for instance, crisp sets can be assigned as in Fig. 2.1(a). In words, temperatures below 18°C are considered 'low', temperatures between 18°C and 21°C 'OK' and temperatures higher than 21°C 'high'.

Such assignment does not correspond to the way temperature is experienced. When applying crisp sets a temperature of 17.9°C is considered completely 'low'. If the temperature increases with 0.2°C it becomes completely 'OK' and a different set of rules in the rule base would be fired if temperature would be an input variable of model controlling a heater. When manually adjusting the power of a heater, however, one will never change ones behaviour in such an abrupt way as one will consider a temperature around 18°C to a certain extent 'low' as well as 'OK'. Note that, the richer one's vocabulary is, the more precise one will be able to linguistically describe a situation or value. In general, however, man seldom assigns more than nine linguistic values to a variable (Miller, 1956).

The terms 'low', 'OK' and 'high' temperature are fuzzy concepts. Describing them by means of crisp sets is therefore an arduous task. It is far more straightforward to define them by fuzzy sets (Zadeh, 1965), as fuzzy sets allow a gradual transition between not belonging and completely belonging to a set. Mathematically speaking, a fuzzy set is defined as a function from the domain $\mathbf{X}$ to the unit interval $[0, 1]$ that maps an element x to $A(x)$

$$A : \mathbf{X} \rightarrow [0, 1] : \, x \mapsto A(x)\,. \tag{2.1}$$

If the value of the membership function A in x, called the membership degree of x to A, is one, x completely belongs to the fuzzy set. If it is equal to zero, x does not belong to the fuzzy set. If the membership degree is between 0 and 1, x partially belongs to the fuzzy set. A crisp set is a particular fuzzy set with membership degrees restricted to $\{$0,1$\}$. In Fig. 2.1(b) the three linguistic values 'low', 'OK' and 'high' of the linguistic variable temperature are defined by membership functions in the domain $\mathbf{T}$ = [10,30].

2.2.2 Characteristics of fuzzy sets

In principle any function of the form $A : \mathbf{X} \rightarrow [0, 1]$ describes a membership function associated with a fuzzy set A. In most applications, however, fuzzy sets are represented by a parameterized function. Popular types of membership functions, commonly used in fuzzy models determined based upon expert knowledge, are trapezial and triangular fuzzy sets. A trapezial fuzzy set (Fig. 2.2(a)) can be characterized by four parameters (a_1, a_2, a_3, a_4) and be defined as

$$A(x) = \begin{cases} 0 & \text{if } x < a_1\,, \\ \frac{x-a_1}{a_2-a_1} & \text{, if } x \in [a_1, a_2]\,, \\ 1 & \text{if } x \in [a_2, a_3]\,, \\ \frac{a_4-x}{a_4-a_3} & \text{, if } x \in [a_3, a_4]\,, \\ 0 & \text{if } x > a_4\,. \end{cases} \tag{2.2}$$

A triangular membership function is obtained if a_2 is equal to a_3 (Fig. 2.2(b)). In this dissertation, membership functions defining the linguistic values of a certain input or output variable of a model are assumed to be trapezial (including triangular) and to form a fuzzy partition in the sense of Ruspini (1969), which guarantees an interpretable description of the linguistic values (Bodenhofer and Bauer, 2005; Jin, 2003). A family $(A_i)_{i=1}^n$ of membership functions forms a fuzzy partition of a domain $\mathbf{X}$ if for each element x the sum of its n membership degrees to all membership functions equals one

$$\left(\forall x \in \mathbf{X}\right)\left(\sum_{i=1}^{n} A_i(x) = 1\right). \tag{2.3}$$

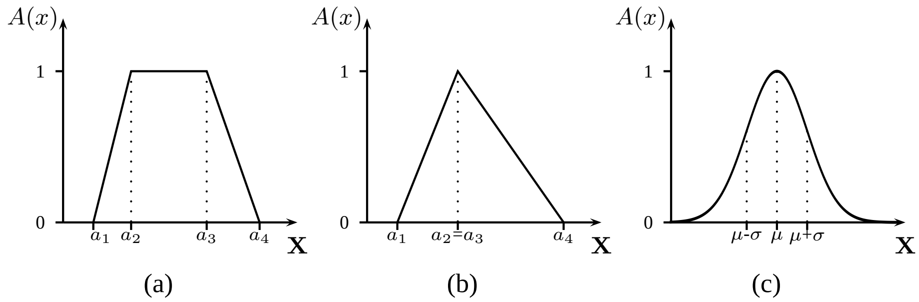


Figure 2.2: Representation of a (a) trapezial, (b) triangular and (c) symmetric Gaussian membership function.

The membership functions defining the three linguistic values 'low', 'OK' and 'high' of the linguistic variable temperature in Fig. 2.1(b) form a fuzzy partition.

Other commonly used types of membership functions are Gaussian membership functions, as for instance the symmetric Gaussian function presented in Fig. 2.2(c), determined by two parameters μ and σ

$$A(x) = e^{-\frac{(x-\mu)^2}{2\sigma^2}} \,, \tag{2.4}$$

as well as, (piece-wise) exponential and polynomial functions (Pedrycz and Gomide, 1998). In contrast to trapezial membership functions, these types of membership functions have the advantage that they are differentiable in the whole domain on which they are defined, which can be of importance in data-driven identification procedures.

Fuzzy sets can be characterized in more detail by referring to the features of the membership functions that describe them. Below the concepts normality, support, kernel, core, (weak) α-cut and strict α-cut of a fuzzy set are defined (Fig. 2.3).

Definition 2.1 *A fuzzy set A is normal if there exists an element x of $\mathbf{X}$ that completely belongs to A*

$$(\exists x \in \mathbf{X})(A(x) = 1)\,.$$

Fuzzy sets A which are not normal, are called subnormal.

Definition 2.2 *By the support of a fuzzy set A, denoted by supp(A), all elements of $\mathbf{X}$ are meant that belong to A to a nonzero degree*

$$\textit{supp}(A) = \{x \in \mathbf{X} \mid A(x) > 0\}\,.$$

Definition 2.3 *The set of elements that completely belong to a fuzzy set A is called the kernel of A, denoted by kern(A)*

$$\textit{kern}(A) = \{x \in \mathbf{X} \mid A(x) = 1\}\,.$$

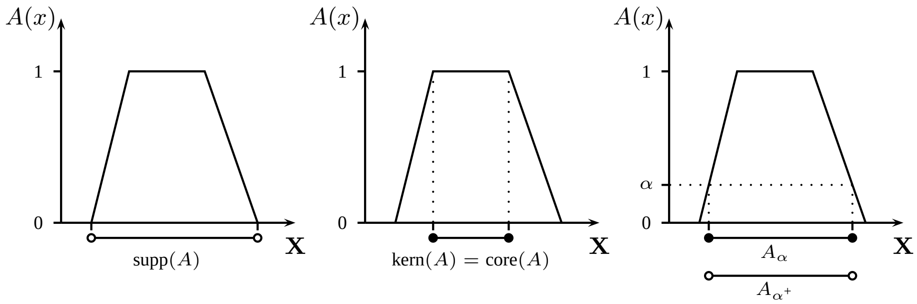


Figure 2.3: Support, kernel and α-cut of a trapezial membership function.

Definition 2.4 *The set of elements having the largest degree of membership in a fuzzy set A is called the core of A*

$$\textit{core}(A) = \{x_1 \in \mathbf{X} \mid (\forall x_2 \in \mathbf{X})(A(x_2) \leq A(x_1))\} \,.$$

Definition 2.5 *The (weak) α-cut ($\alpha \in [0,1]$) of a fuzzy set A, denoted by A_α, is a set consisting of those elements of the domain $\mathbf{X}$ whose membership degrees exceed or are equal to the threshold level α*

$$A_\alpha = \{x \in \mathbf{X} \mid A(x) \geq \alpha\} \,.$$

Definition 2.6 *The strict α-cut ($\alpha \in [0,1[$) of a fuzzy set A, denoted by A_{α^+}, is a set consisting of those elements of the domain $\mathbf{X}$ whose membership degrees exceed the threshold level α*

$$A_{\alpha^+} = \{x \in \mathbf{X} \mid A(x) > \alpha\} \,.$$

In this dissertation there will be referred to two special fuzzy sets: the empty set and the universal set. The empty set is defined as the fuzzy set A to which all elements x of a domain $\mathbf{X}$ have membership degree zero

$$(\forall x \in \mathbf{X})(A(x) = 0) \,, \tag{2.5}$$

whereas the universal set is defined as the fuzzy set A to which all elements x of a domain $\mathbf{X}$ have membership degree one

$$(\forall x \in \mathbf{X})(A(x) = 1) \,. \tag{2.6}$$

2.2.3 Operations on fuzzy sets

In the following paragraphs the basic operations of *fuzzy* set theory — intersection, union and complement — are introduced. These operations are extensions of the operations used in classical set theory. For fuzzy sets A and B defined in a domain $\mathbf{X}$, the intersection of A and B is defined by

$$A \cap B(x) = T(A(x), B(x)) \,, \tag{2.7}$$

where T is a triangular norm, t-norm for short. The union of A and B is defined by

$$A \cup B(x) = S(A(x), B(x))\,, \tag{2.8}$$

where S is a triangular co-norm, t-conorm for short, and the complement of A is defined by

$$\mathrm{co}A(x) = 1 - A(x)\,. \tag{2.9}$$

Dozens of definitions have been suggested for t-norms and t-conorms. In this dissertation the three most commonly applied t-norms and t-conorms, illustrated in Fig. 2.4, are considered: the minimum t-norm $T_{\mathbf{M}}$, the product t-norm $T_{\mathbf{P}}$ and the Łukasiewicz t-norm $T_{\mathbf{L}}$

$$T_{\mathbf{M}}(a,b) = \min(a,b)\,, \tag{2.10}$$
$$T_{\mathbf{P}}(a,b) = a \cdot b\,, \tag{2.11}$$
$$T_{\mathbf{L}}(a,b) = \max(0, a+b-1)\,, \tag{2.12}$$

and the corresponding t-conorms, the maximum $S_{\mathbf{M}}$, the algebraic sum $S_{\mathbf{P}}$ and the Łukasiewicz t-conorm $S_{\mathbf{L}}$

$$S_{\mathbf{M}}(a,b) = \max(a,b)\,, \tag{2.13}$$
$$S_{\mathbf{P}}(a,b) = a + b - a \cdot b\,, \tag{2.14}$$
$$S_{\mathbf{L}}(a,b) = \min(1, a+b)\,. \tag{2.15}$$

Formally, a t-norm is defined as a binary operation T on the unit interval [0,1], *i.e.* a function T: $[0,1]^2 \to [0,1]$, satisfying the following requirements

- commutativity : $T(x,y) = T(y,x)\,,$ (2.16)
- associativity : $T(x, T(y,z)) = T(T(x,y), z)\,,$ (2.17)
- monotonicity : $T(x,y) \leq T(x,z)$ whenever $y \leq z\,,$ (2.18)
- neutral element 1 : $T(x,1) = x\,.$ (2.19)

From Eqs. (2.18–2.19) is follows that 0 is the absorbing element, $T(x,0) = 0$.

To fuzzy sets defining linguistic values, such as ‘low’, ‘OK’ and ‘high’, apart from the intersection, union and complement, a group of operators can be applied that do not have a counterpart in classical set theory. These operators are referred to as linguistic hedges or linguistic modifiers. Examples of linguistic hedges are ‘very’, ‘extremely’, ‘greatly’ and ‘at least’. The application of a linguistic hedge modifies the shape of the membership function of a fuzzy set, transforming one fuzzy set into another. The meaning of the transformed set (*e.g.* ‘very high’) can easily be interpreted from the meaning of the original set (*e.g.* ‘high’) and that embedded in the hedge applied (*e.g.* ‘very’). The definition of hedges has more to do with common sense knowledge in a domain than with mathematical theory. For a fuzzy set defined by a trapezial membership function characterized by four parameters (a_1, a_2, a_3, a_4) (Eq. (2.2)) the hedge very can for instance either be defined by

$$\mathrm{very}A(x) = A(x)^2\,, \tag{2.20}$$

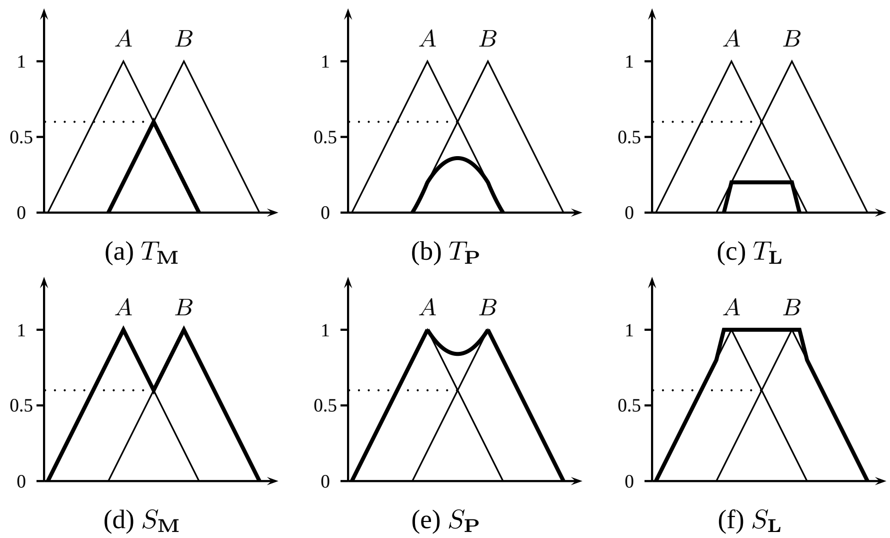


Figure 2.4: Intersection and union of two fuzzy sets A and B.

or by

$$a'_2 = \frac{1}{2}(a_2 + a_3) - \frac{1}{2}\left(\frac{1}{2}(a_2 + a_3) - a_2\right) \tag{2.21}$$

$$a'_3 = \frac{1}{2}(a_2 + a_3) + \frac{1}{2}\left(a_3 - \frac{1}{2}(a_2 + a_3)\right) \tag{2.22}$$

$$a'_1 = a'_2 - \frac{1}{2}(a_2 - a_1) \tag{2.23}$$

$$a'_4 = a'_3 + \frac{1}{2}(a_4 - a_3) \tag{2.24}$$

where the fuzzy set '$\mathrm{very}A$' is defined by a trapezial membership function characterized by four parameters (a'_1, a'_2, a'_3, a'_4) (Marín-Blázquez and Shen, 2002).

In Chapter 10 the modifiers 'at least' (ATL) and 'at most' (ATM) introduced by Bodenhofer (1999) and illustrated in Fig. 2.5 are applied

$$\mathrm{ATL}(A)(x) = \sup\{A(t) \mid t \leq x\}\,, \tag{2.25}$$

$$\mathrm{ATM}(A)(x) = \sup\{A(t) \mid t \geq x\}\,. \tag{2.26}$$

2.3 Linguistic fuzzy models

The main component of linguistic fuzzy models is a rule base consisting of rules in which both the antecedent and the consequent part contain fuzzy sets. The rule base and

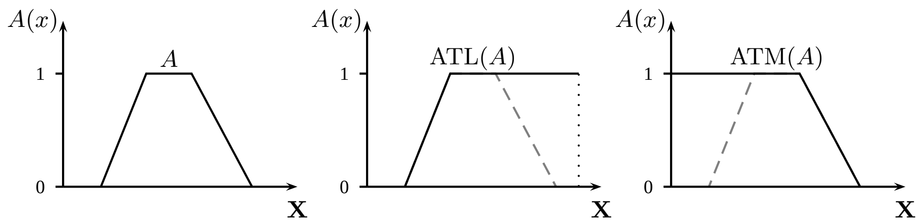


Figure 2.5: Illustration of the modifiers 'at least' (ATL) and 'at most' (ATM).

the fuzzy sets are generally determined by expert knowledge. The rules of a linguistic fuzzy model with m input variables X_l ($l \in L = \{1, \ldots, m\}$) and one output variable Y are of the form

$$R_s\text{: IF } X_1 \text{ IS } B^1_{j_{1,s}} \text{ AND } \ldots \text{ AND } X_m \text{ IS } B^m_{j_{m,s}} \text{ THEN } Y \text{ IS } A_{i_s}$$

where $B^l_{j_{l,s}}$ (resp. A_{i_s}) are linguistic values of variable X_l (resp. Y) in the domain $\mathbf{X}_l$ (resp. $\mathbf{Y}$) ($s \in S = \{1, \ldots, r\}$). The input vector is denoted by $\mathbf{x} = (x_1, \ldots, x_m)$. In this dissertation two kinds of linguistic fuzzy models are distinguished: models applying t-norm-based inference and models applying implicator-based inference. These two inference methods correspond to two fundamentally different interpretations of if-then rules, which are discussed in Section 2.3.1. Next, in Sections 2.3.2 and 2.3.3, the inference methods are described.

2.3.1 Interpretation of if-then rules

Crisp inputs Let us consider an if-then rule 'IF X is B THEN Y is A' with X (resp. Y) a variable in the domain $\mathbf{X}$ (resp. $\mathbf{Y}$) and B (resp. A) a fuzzy set in $\mathbf{X}$ (resp. $\mathbf{Y}$). Regardless of the interpretation given to this if-then rule, it is modelled as a fuzzy relation R from $\mathbf{X}$ to $\mathbf{Y}$. The direct image $A'(y)$ of a fuzzy set B' in $\mathbf{X}$ under a fuzzy relation R from $\mathbf{X}$ to $\mathbf{Y}$ is the fuzzy set in $\mathbf{Y}$ defined by

$$A'(y) = \sup_{x \in \mathbf{X}} T(B'(x), R(x, y))\,. \tag{2.27}$$

As only crisp inputs $B'(x)$ are considered in this study, for which the following equation holds

$$B'(x) = \begin{cases} 1 & \text{, if } x = x^*, \\ 0 & \text{, otherwise,} \end{cases} \tag{2.28}$$

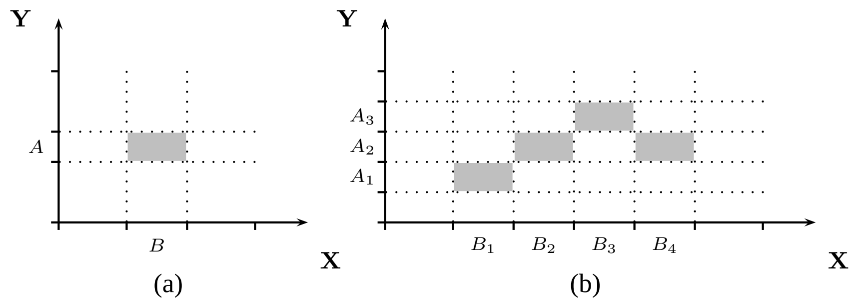


Figure 2.6: Illustration of the interpretation given to crisp if-then rules when applying a t-norm-based inference procedure. The shaded regions are the pairs (x, y) for which the rule 'IF X is B THEN Y is A', respectively the four rules 'IF X is B_s THEN Y is A_s' hold.

and, as $T(0, x) = 0$ and $T(1, x) = x$ (Eq. (2.19)), Eq. (2.27) can be simplified to

$$\begin{aligned} A'(y) &= \max(\sup_{x \in \mathbf{X} \setminus \{x^*\}} T(B'(x), R(x, y)), T(B'(x^*), R(x^*, y))) \\ &= \max(\sup_{x \in \mathbf{X} \setminus \{x^*\}} T(0, R(x, y)), T(1, R(x^*, y))) \\ &= \max(0, R(x^*, y)) \\ &= R(x^*, y)\,. \end{aligned} \tag{2.29}$$

T-norm-based inference For the sake of simplicity, let us first consider a crisp if-then rule 'IF X is B THEN Y is A' to illustrate the first interpretation given to fuzzy if-then rules, with X (resp. Y) a variable in the domain $\mathbf{X}$ (resp. $\mathbf{Y}$) and B (resp. A) a subset of $\mathbf{X}$ (resp. $\mathbf{Y}$). When applying t-norm-based inference the crisp rule is modelled as

$$(x, y) \in R \text{ with } R = B \times A\,,$$

i.e. the expression 'IF X is B THEN Y is A' is said to hold only for those (x, y) for which x is a member of B and y is a member of A. These pairs (x, y) are indicated in gray in Fig. 2.6(a). Note that strictly mathematically speaking, it is incorrect to interpret if-then rules in this way. When A and B are fuzzy sets, the rule is modelled as

$$(X, Y) \in R \text{ with } R(x, y) = T(B(x), A(y))\,,$$

with T being a t-norm. The fuzzy output A', given a crisp input x^*, is obtained by

$$A'(y) = T(B(x^*), A(y))\,.$$

Rule bases of fuzzy linguistic models do not consist of a single rule, but are a collection of r if-then rules 'IF X is B_s THEN Y is A_s'. When applying t-norm-based inference the global fuzzy model output $A'(y)$ is derived from the individual outputs $A'_s(y)$ by

$$A'(y) = \max_{s=1}^{r} A'_s(y) \;\text{ with }\; A'_s(y) = T(B_s(x^*), A_s(y))\,, \tag{2.30}$$

or, alternatively, by

$$A'(y) = R(x^*, y) \;\text{ with }\; R(x,y) = \max_{s=1}^{r} T(B_s(x), A_s(y))\,, \tag{2.31}$$

as illustrated in Fig. 2.6(b) for the four rules 'IF X is B_1 THEN Y is A_1', 'IF X is B_2 THEN Y is A_2', 'IF X is B_3 THEN Y is A_3' and 'IF X is B_4 THEN Y is A_2'.

In literature, if-then rules interpreted and fuzzy models applying if-then rules according to the interpretation above, are referred to as possibility rules ('the more X is B, the more possible A is a range for Y') (Dubois and Prade, 1996), pessimistic modelling (De Baets, 1996) or Mamdani-type constructive linguistic models (Yager and Filev, 1994).

Implicator-based inference When applying an implicator-based inference procedure, a crisp if-then rule 'IF X is B THEN Y is A' is modelled as

$$(x,y) \in R \;\text{ with }\; R = (B \times A) \cup (\mathrm{co}B \times \mathbf{Y})\,,$$

i.e. the expression 'IF X is B THEN Y is A' is implemented as a logical implication: if x is a member of B, y is a member of A, but if x is not a member of B, y can take any value in the domain $\mathbf{Y}$. These pairs are indicated in gray in Fig. 2.7(a). Mathematically speaking, this is the only correct interpretation of an if-then rule. When A and B are fuzzy sets, the above rule is modelled as

$$(X,Y) \in R \;\text{ with }\; R(x,y) = I(B(x), A(y))\,,$$

with I being an implicator, *i.e.* a function $I : [0,1]^2 \to [0,1]$ coinciding with the Boolean implication on $\{0,1\}^2$ (*i.e.* $I(0,0) = I(1,1) = I(0,1) = 1$ and $I(1,0) = 0$) and having decreasing first and increasing second partial functions

$$(\forall\, x,y,z \in [0,1])(x \le y \Rightarrow I(x,z) \ge I(y,z))\,, \tag{2.32}$$
$$(\forall\, x,y,z \in [0,1])(y \le z \Rightarrow I(x,y) \le I(x,z))\,. \tag{2.33}$$

In the work by De Baets and Kerre (1993) two other representations of the crisp rule 'IF X is B THEN Y is A' are derived

$$(x,y) \in \{C \mid C \text{ is a subset of } \mathbf{X} \times \mathbf{Y} \text{ and } (B \times \mathbf{Y}) \cap C \subseteq \mathbf{X} \times A\} \quad \text{, and} \tag{2.34}$$
$$(x,y) \in (\mathrm{co}B \times \mathbf{Y}) \cup (\mathbf{X} \times A)\,, \tag{2.35}$$

corresponding to the following implicators

$$I_T(x,y) = \sup\{z \mid T(x,z) \le y\}\,, \tag{2.36}$$
$$I_{T,\mathcal{N}}(x,y) = S(1-x, y)\,. \tag{2.37}$$

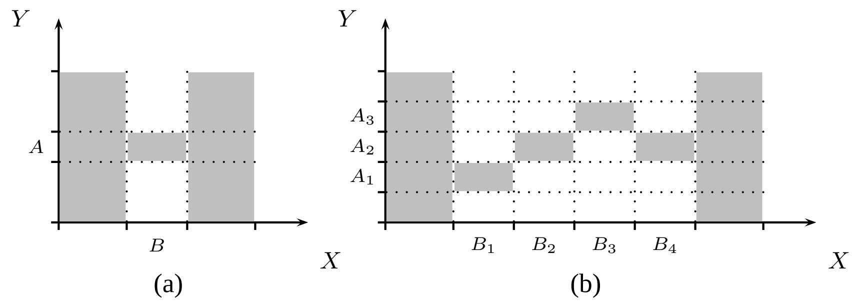


Figure 2.7: Illustration of the interpretation given to crisp if-then rules when applying a implicator-based inference procedure. The shaded regions are the pairs (x, y) for which the rule 'IF X is B THEN Y is A', respectively the four rules 'IF X is B_s THEN Y is A_s' hold.

The implicators defined in Eq. (2.36) are called R-implicators: the implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$ are obtained by replacing T by $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ respectively. The implicators defined in Eq. (2.37) are called S-implicators: the implicators $I_{\mathbf{M},\mathcal{N}}$, $I_{\mathbf{P},\mathcal{N}}$ and $I_{\mathbf{L},\mathcal{N}}$ are obtained by replacing S by $S_{\mathbf{M}}$, $S_{\mathbf{P}}$ and $S_{\mathbf{L}}$ respectively (note that $I_{\mathbf{L},\mathcal{N}}$ = $I_{\mathbf{L}}$).

For a crisp input x^*, the fuzzy output $A'(y)$ is obtained as

$$A'(y) = I(B(x^*), A(y)) \,, \tag{2.38}$$

or in case of r fuzzy rules

$$A'(y) = \min_{s=1}^{r} A'_s(y) \;\text{ with }\; A'_s(y) = I(B_s(x^*), A_s(y)) \,, \tag{2.39}$$

or

$$A'(y) = R(x^*, y) \;\text{ with }\; R(x, y) = \min_{s=1}^{r} I(B_s(x), A_s(y)) \,, \tag{2.40}$$

illustrated in Fig. 2.7(b) for the same four rules used to illustrate t-norm-based inference procedures in Fig. 2.6(b). If the rule base of a model contains one rule for each B_s of a set of crisp sets forming a partition of the input domain, both interpretations result in the same global relation R. Applying if-then rules according to the first interpretation in fuzzy models can therefore be considered *mathematically defensible* if fuzzy partitions are assigned to the linguistic values of all input variables X_l ($l \in L = \{1, \ldots, m\}$) and if the rule base contains one rule for each combination of fuzzy sets $(A^1_{j_1}, \ldots, A^m_{j_m})$ with $A^l_{j_l}$ ($j_l \in J_l = \{1, \ldots, n_l\}$) membership functions of an input variable X_l.

In the work by Dubois and Prade (1996), fuzzy if-then rules modelled by R-implicators are called gradual rules as they correspond to statements of the form 'the more X is B, the more Y is A', whereas the term certainty rules is used in case of S-implicators, modelling statements as 'the more X is B, the more certain Y is A'. Modelling applications defining if-then rules as implications are referred to as optimistic modelling (De Baets, 1996) or logical-type destructive linguistic models (Yager and Filev, 1994).

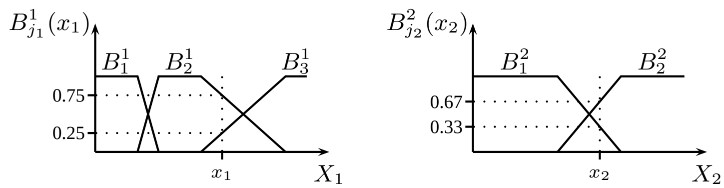


Figure 2.8: Determining the membership degrees of the model input vector $\mathbf{x} = (x_1, x_2)$ to the linguistic values of the input variables X_1 and X_2.

2.3.2 Mamdani–Assilian inference

Linguistic fuzzy models applying t-norm-based inference are called Mamdani–Assilian models (Assilian, 1974; Mamdani, 1974). When determining the model output via Mamdani–Assilian inference, first the membership degrees $B^l_{j_{l,s}}(x_l)$ of the model input vector $\mathbf{x}$ to the linguistic values in the antecedents of the rules are determined. In Fig. 2.8 the membership degrees of the input values x_1 and x_2 ($\mathbf{x} = (x_1, x_2)$) to the corresponding linguistic values of the input variables X_1 and X_2 are

$$\begin{array}{lll} B^1_1(x_1) = 0 & B^1_2(x_1) = 0.75 & B^1_3(x_1) = 0.25 \\ B^2_1(x_2) = 0.33 & B^2_2(x_2) = 0.67 & \end{array}$$

In the following step, the fulfilment degrees β_s of the r rules ($s \in S = \{1, \ldots, r\}$) are computed from the membership degrees $B^l_{j_{l,s}}(x_l)$ of the model input vector $\mathbf{x}$ to the linguistic values in the antecedents of the rules . For the t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ this results in

$$\beta_s = \begin{cases} \min\limits_{l=1}^{m} B^l_{j_{l,s}}(x_l) & \text{, if } T = T_{\mathbf{M}} \,, \\ \prod\limits_{l=1}^{m} B^l_{j_{l,s}}(x_l) & \text{, if } T = T_{\mathbf{P}} \,, \\ \max\left(\sum\limits_{l=1}^{m} B^l_{j_{l,s}}(x_l) - (m-1), 0 \right) & \text{, if } T = T_{\mathbf{L}} \,. \end{cases} \tag{2.41}$$

Next, the adapted membership functions $B'_s(y)$ are computed using the same t-norm T as for the fulfilment degrees β_s (see Fig. 2.9)

$$A'_{i_s}(y) = \begin{cases} \min(\beta_s, A_{i_s}(y)) & \text{, if } T = T_{\mathbf{M}} \,, \\ \beta_s \cdot A_{i_s}(y) & \text{, if } T = T_{\mathbf{P}} \,, \\ \max(\beta_s + A_{i_s}(y) - 1, 0) & \text{, if } T = T_{\mathbf{L}} \,, \end{cases} \tag{2.42}$$

and the global fuzzy output $A(y)$ is determined as follows

$$A(y) = \max_{s=1}^{r} A'_{i_s}(y) \,. \tag{2.43}$$

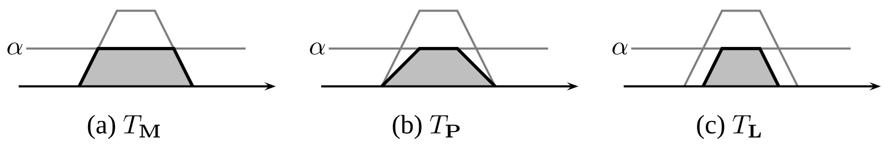


Figure 2.9: Adapted membership functions (in black) obtained by applying Eq. (2.42) with $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ to the membership function in grey.

Finally, the crisp model output y^* is obtained by defuzzifying the fuzzy output. In this dissertation the Center of Gravity (COG) defuzzification resulting in the crisp model output y^*_{COG} and the Mean of Maxima (MOM) defuzzification resulting in the crisp model output y^*_{MOM} (Kruse et al., 1994) are considered

$$y^*_{\mathrm{COG}} = \frac{\int\limits_{\mathbf{Y}} yA(y)\,dy}{\int\limits_{\mathbf{Y}} A(y)\,dy}\,, \tag{2.44}$$

$$y^*_{\mathrm{MOM}} = \frac{\int\limits_{\mathrm{core}(A)} y\,dy}{\int\limits_{\mathrm{core}(A)} dy}\,. \tag{2.45}$$

When the core of the fuzzy model output A is a set of discrete values, the integrals in the expression for the crisp output y^*_{MOM} in Eq. (2.45) vanish. In this case the crisp output y^*_{MOM} is defined as the average of these discrete values.

In practice Eqs. (2.42–2.43) are implemented in a slightly different way. From the fulfilment degrees β_s of the r rules, a fulfilment degree α_i is computed for each linguistic output value A_i,

$$\alpha_i = \max\{\beta_s \mid i_s = i\}\,. \tag{2.46}$$

For each linguistic output value an adapted membership function A'_i is determined with the corresponding fulfilment degree α_i

$$A'_i(y) = \begin{cases} \min(\alpha_i, A_i(y)) & \text{, if } T = T_{\mathbf{M}}\,, \\ \alpha_i \cdot A_i(y) & \text{, if } T = T_{\mathbf{P}}\,, \\ \max(\alpha_i + A_i(y) - 1, 0) & \text{, if } T = T_{\mathbf{L}}\,, \end{cases} \tag{2.47}$$

and the global fuzzy output $A(y)$ is determined as follows

$$A(y) = \max_{i=1}^{n} A'_i(y)\,. \tag{2.48}$$

In Fig. 2.10 the Mamdani–Assilian inference procedure is illustrated for a model with two input variables X_1 and X_2 and one output variable Y. The linguistic values of all three variables are described by membership functions forming a fuzzy partition.

The linguistic values are ‘low’, ‘medium’ and ‘high’ for X_1 and Y, and ‘low’ and ‘high’ for X_2. The fulfilment degree β_s of each of the six rules is the minimum of the membership degree of x_1 and x_2 to the corresponding linguistic value in the antecedent of the rule and the membership functions in the consequent part of the rules are truncated according to this fulfilment degree β_s ($T = T_{\mathbf{M}}$). The global fuzzy output is the union, based on the maximum, of all these truncated fuzzy sets. Finally, the crisp model output y^*_{COG} is obtained by the COG defuzzification method.

2.3.3 Implicator-based inference

When applying implicator-based inference, the fulfilment degrees β_s and α_i are calculated as described in Eqs. (2.41) and (2.46), but the adapted membership functions A'_i are computed using an implicator instead of a t-norm. In this dissertation the three R-implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$ are considered. Note that the adapted membership functions A'_i do not necessarily have to be computed with the corresponding implicator I_T of the t-norm T used for the conjunction when computing the fulfilment degrees.

For $I_{\mathbf{M}}$ the adapted membership functions are obtained by

$$A'_i(y) = \begin{cases} 1 & \text{, if } \alpha_i \leq A_i(y)\,, \\ A_i(y) & \text{, otherwise}\,, \end{cases} \tag{2.49}$$

for $I_{\mathbf{P}}$ by

$$A'_i(y) = \begin{cases} 1 & \text{, if } \alpha_i \leq A_i(y)\,, \\ \frac{A_i(y)}{\alpha_i} & \text{, otherwise}\,, \end{cases} \tag{2.50}$$

and for $I_{\mathbf{L}}$ by

$$A'_i(y) = \min(1 - \alpha_i + A_i(y), 1)\,. \tag{2.51}$$

Figs. 2.9 and 2.11 show how membership functions are changed when applying respectively a t-norm or an implicator, given a fulfilment degree α. It nicely illustrates the two different interpretations of fuzzy if-then rules: the better rule corresponds to the current situation, *i.e.* the higher the fulfilment degree, the more the adapted membership function is restricted when applying an implicator, but the more the adapted membership function is extended when applying a t-norm. While $T_{\mathbf{M}}$ is less restrictive than $T_{\mathbf{P}}$, which on his turn is less restrictive than $T_{\mathbf{L}}$, the reverse order applies for the implicator operators. The implicator $I_{\mathbf{L}}$ is the least restrictive implicator, followed by $I_{\mathbf{P}}$ and $I_{\mathbf{M}}$.

The global fuzzy output A is the intersection, based on the minimum, of the n adapted membership functions A'_i

$$A(y) = \min_{i=1}^{n} A'_i(y)\,. \tag{2.52}$$

The only specific defuzzification method for models applying implicator-based inference known to the author is the defuzzification method introduced by Dvořák and

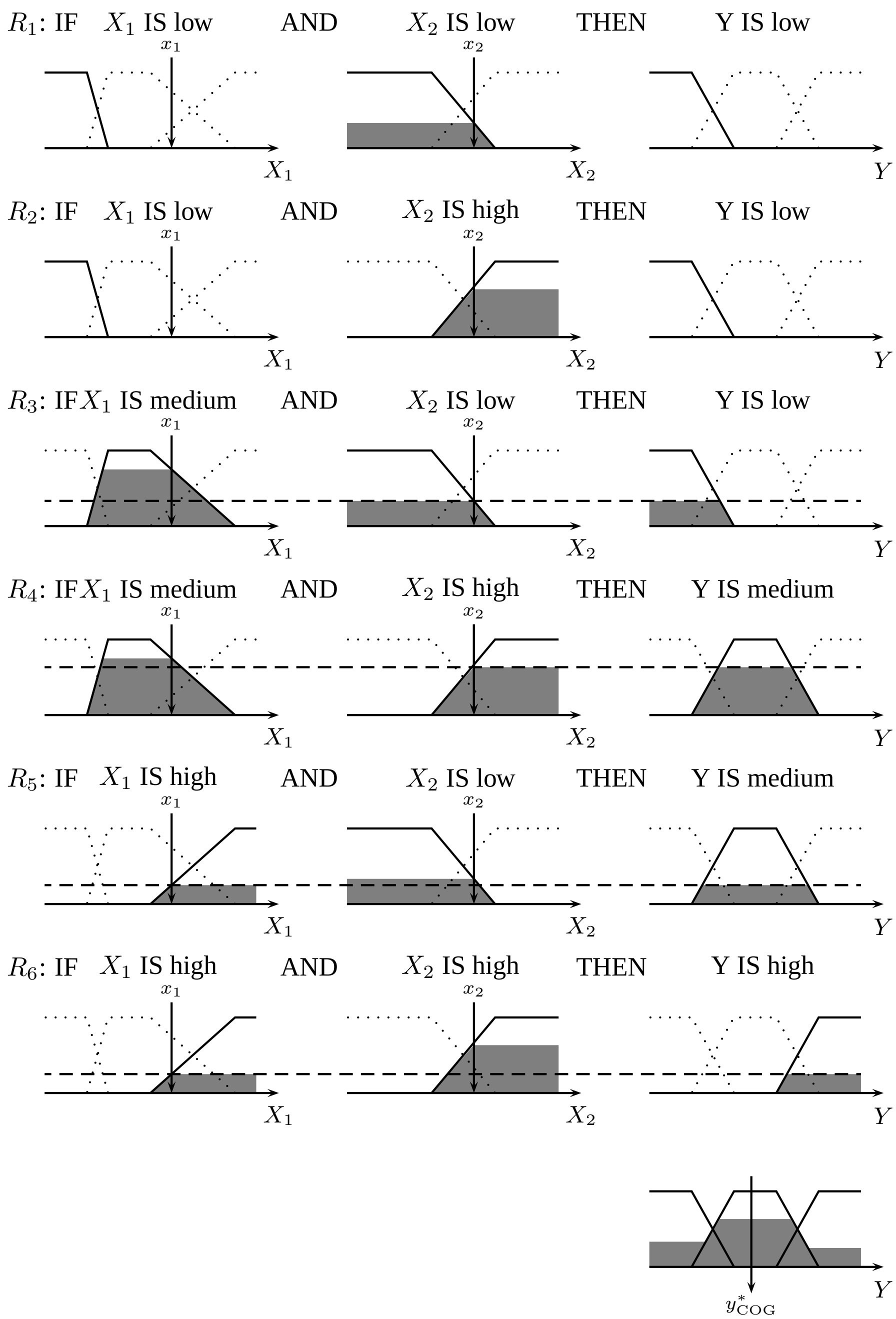
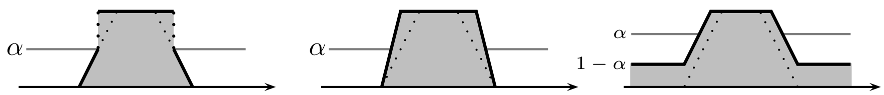


Figure 2.10: Illustration of Mamdani–Assilian inference ($T = T_{\mathrm{M}}$, COG defuzzification) applied to a model with six rules.



(a) $I_{\mathbf{M}}$ (b) $I_{\mathbf{P}}$ (c) $I_{\mathbf{L}}$

Figure 2.11: Adapted membership functions (solid black line) obtained by applying Eqs. (2.49–2.51) to the membership function represented by the dotted black line.

Jedelský (1999) for models applying $I_{\mathbf{L}}$ resulting in the crisp output y^*_{COGDJ} ($\mathbf{Y} = [y_0, y_{\mathrm{end}}]$)

$$y^*_{\mathrm{COGDJ}} = \begin{cases} \frac{1}{2}(y_0 + y_{\mathrm{end}}) & \text{, if } \min\limits_{\mathbf{Y}} A(y) = \max\limits_{\mathbf{Y}} A(y)\,, \\ \frac{\int_{\mathbf{Y}} y \cdot (A(y) - \min_{\mathbf{Y}} A(y))\, dy}{\int_{\mathbf{Y}} (A(y) - \min_{\mathbf{Y}} A(y))\, dy} & \text{, otherwise}\,. \end{cases} \tag{2.53}$$

If the smallest membership degree $\min\limits_{\mathbf{Y}} A(y)$ obtained in the output domain to the fuzzy output A is equal to zero, the defuzzification method introduced by Dvořák and Jedelský (1999) coincides with the COG defuzzification method defined in Eq. (2.44). In models applying implicator-based inference also the MOM defuzzification method defined in Eq. (2.45) can be applied.

In Fig. 2.12 implicator-based inference is illustrated for a model with two input variables X_1 and X_2 and one output variable Y. The linguistic values of all three variables are described by membership functions forming a fuzzy partition. The linguistic values are ‘low’, ‘medium’ and ‘high’ for X_1 and Y, and ‘low’ and ‘high’ for X_2. The fulfilment degree β_s of each of the six rules is the minimum of the membership degrees of x_1 and x_2 to the corresponding linguistic value in the antecedent of the rule ($T = T_{\mathbf{M}}$) and the membership functions in the consequent part of the rules are adapted according to this fulfilment degree β_s using the implicator $I_{\mathbf{L}}$. The global fuzzy output is the intersection of all these adapted fuzzy sets. Finally, the crisp model output y^*_{MOM} is obtained by the MOM defuzzification method.

2.4 Takagi–Sugeno models

A different type of fuzzy models is the Takagi–Sugeno model, introduced by Takagi and Sugeno (Takagi and Sugeno, 1985). The Takagi–Sugeno model differs from the linguistic model on this point that its consequent parts are zero-, first- or higher-order polynomial functions of the input variables. Rules in Takagi–Sugeno models with m input variables X_l ($l \in L = \{1, \ldots, m\}$) and one output variable Y can be expressed in following general form

$$R_s\text{: IF } X_1 \text{ IS } B^1_{j_{1,s}} \text{ AND } \ldots \text{ AND } X_m \text{ IS } B^m_{j_{m,s}} \text{ THEN } Y = f_s(X_1, \ldots, X_m)$$

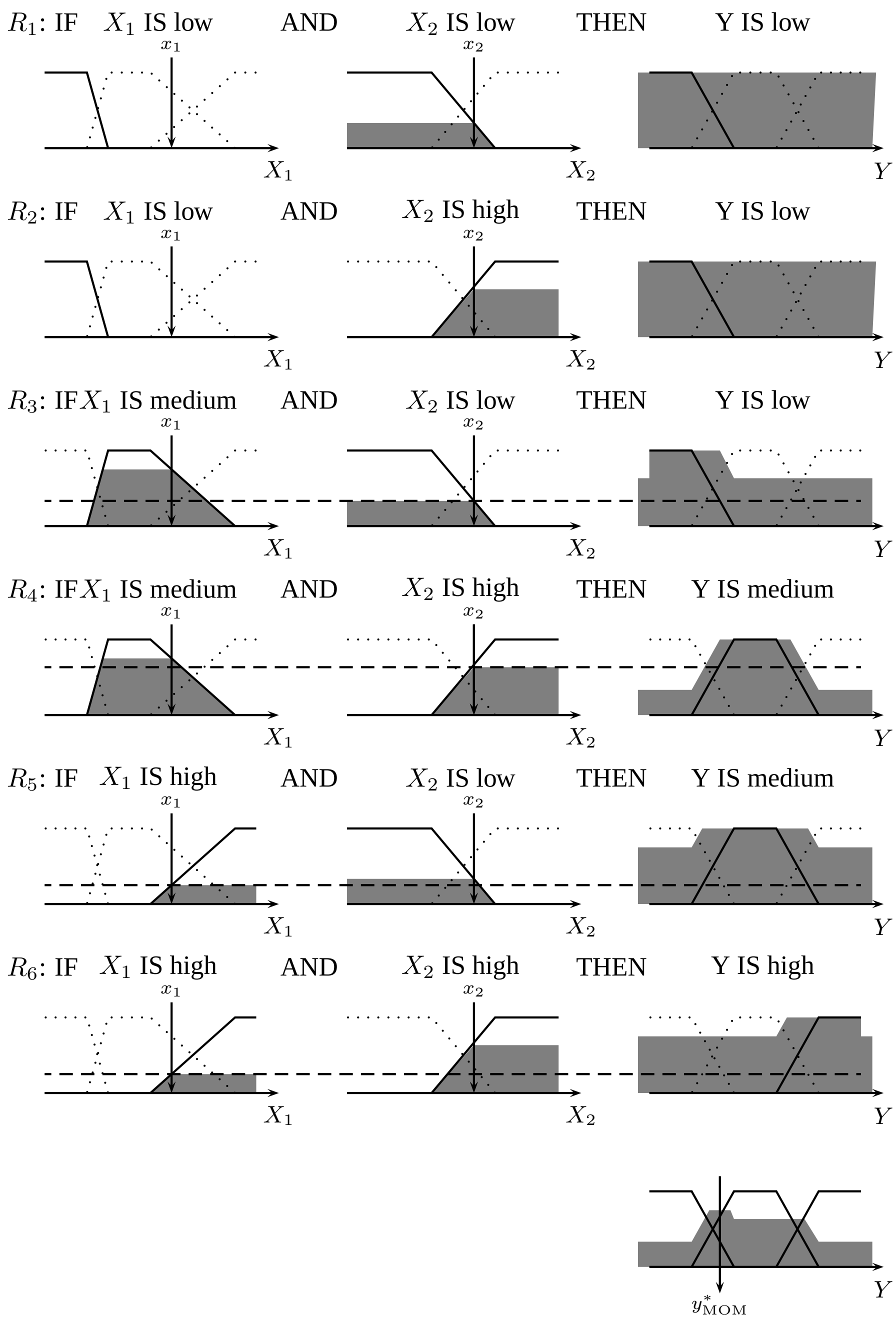


Figure 2.12: Illustration of implicator-based inference ($T = T_{\mathbf{M}}$, $I = I_{\mathbf{L}}$, MOM defuzzification) applied to a model with six rules.

where $B^l_{j_{l,s}}$ are linguistic values of variable X_l ($s \in S = \{1, \ldots, r\}$). The input vector is denoted by $\mathbf{x} = (x_1, \ldots, x_m)$.

The model output y is obtained as

$$y = \frac{\displaystyle\sum_{s=1}^{r} \beta_s(\mathbf{x})\, f_s(\mathbf{x})}{\displaystyle\sum_{s=1}^{r} \beta_s(\mathbf{x})}\,, \tag{2.54}$$

with $\beta_s(\mathbf{x})$ the fulfilment degree of rule R_s

$$\beta_s(\mathbf{x}) = \mathop{T}_{l=1}^{m} B^l_{j_{l,s}}(x_l)\,. \tag{2.55}$$

The most commonly used Takagi–Sugeno models are first-order Takagi–Sugeno models, where the function f_s is a linear function of the input variables (Cordón et al., 2001; Jin, 2003; Sousa and Kaymak, 2002)

$$\begin{array}{lll} R_s\colon & \text{IF} & X_1 \text{ IS } B^1_{j_{1,s}} \text{ AND } \ldots \text{ AND } X_m \text{ IS } B^m_{j_{m,s}} \\ & \text{THEN} & Y = a_{1,s}X_1 + a_{2,s}X_2 + \ldots + a_{m,s}X_m + b_s \end{array}$$

The first-order Takagi–Sugeno model approximates a nonlinear function by means of local linear models, represented in the consequent parameters. By computing a weighted average of the individual rule outputs, *i.e.* the linear functions, the nonlinear function can be approximated, and a smooth transition between the consequent functions is established, which is different from an ordinary piecewise linear approximation method (Takagi and Sugeno, 1985). The structure of a first-order Takagi–Sugeno model with one input variable X and one output variable Y is illustrated in Figure 2.13. Three fuzzy sets A_1, A_2 and A_3, the antecedent fuzzy sets, are assigned to the input variable. This results in three fuzzy rules of the form

$$R_s : \text{IF } X \text{ is } A_{i_s} \text{ THEN } Y = a_s X + b_s \tag{2.56}$$

with a_s and b_s the parameters of the consequent part of rule R_s.

By using functions instead of linguistic values in the consequent parts of the rules, the human interpretation of the phenomenon described by a rule is garbled, but on the other hand this rule structure significantly increases the approximation capability of the model (Casillas et al., 2003b). Mostly, the identification of Takagi–Sugeno models consists of the determination of the number of rules and the parameters of the antecedent and the consequent parts of these rules and is mostly carried out using a data-driven approach (Sousa and Kaymak, 2002). The antecedent part of the rules are generally identified by means of clustering algorithms (Babuška and Verbruggen, 1997) and neural networks (Jang, 1993) and the consequent parameters by a least squares method (Babuška, 1998).

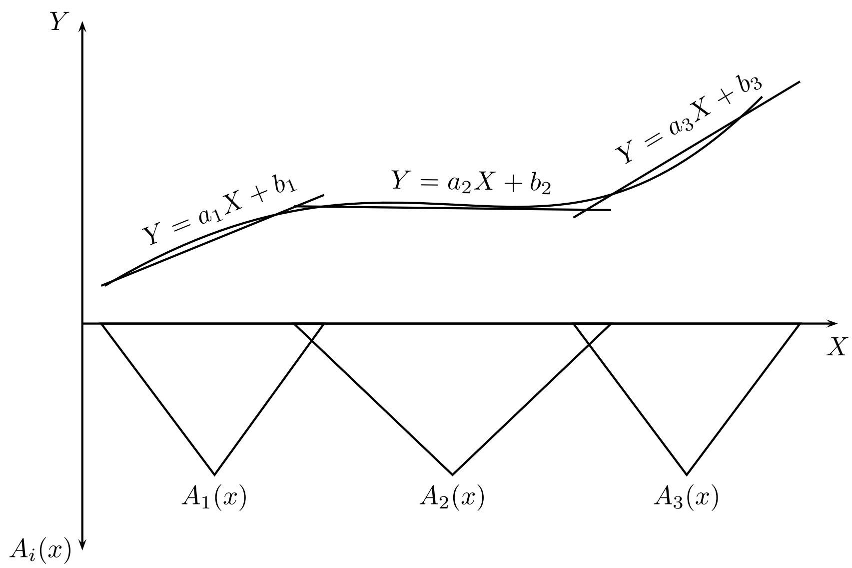


Figure 2.13: Schematic representation of a Takagi–Sugeno model.

CHAPTER 3

Computational aspects of COG defuzzification

Good, Fast, Cheap: Pick any two.
(Sign in Print Shop)

3.1 Introduction

As most modelling and control applications require crisp outputs, when applying fuzzy inference systems, the fuzzy system output A usually has to be defuzzified, *i.e.* to be converted into a crisp output y^*. The most popular defuzzification methods for linguistic fuzzy models applying t-norm-based inference are the *Center Of Gravity* (COG) and the *Mean Of Maxima* (MOM) methods. More general frameworks have been proposed, in which the COG and MOM defuzzification methods have their place, such as the parametric BADD (BAsic Defuzzification Distribution) and SLIDE (Semi-LInear DEfuzzification) methods of Yager and Filev (Filev and Yager, 1991; Yager and Filev, 1993). They are essentially based on the transformation of a possibility distribution into a probability distribution based on Klir's principle of uncertainty invariance. The main emphasis is on the learning of the parameters involved, which is treated as an optimization problem (Jiang and Li, 1996; Roychowdhury and Wang, 1996; Song and Leland, 1996). This issue falls outside the scope of this dissertation. Note that in literature the terms for describing different defuzzification methods vary from source to source. The terms Center of Gravity defuzzification, Center of Area defuzzification and Center of Sum defuzzification, for instance, refer to different methods in some sources and are used as synonyms in other sources. Therefore one should pay attention to the formal definitions of the defuzzification methods rather than to their names. In (Roychowdhury and Pedrycz, 2001; Van Leekwijck and Kerre, 1999) comprehensive overviews are given on defuzzification methods. In this dissertation the same terminology is used as by Van Leekwijck and Kerre (1999).

This chapter deals with the computational aspects of the COG defuzzification method. When applying the COG defuzzification method, the crisp output y^* of the system will change continuously when the input values change continuously, a desirable property in modelling and control applications. However, the COG defuzzification method has a high computational burden (Driankov et al., 1993; Patel and Mohan, 2002), which is a considerable disadvantage in control and model identification, and in tuning applications. This high cost is often circumvented by introducing new defuzzification methods that intend to approximate the center of gravity (Patel and Mohan, 2002; Sakly and Benrejeb, 2003). In this study, however, the definition of the center of gravity is sticked to and other ways are introduced to compute the crisp output in case the membership functions of the output variable are trapezial and form a fuzzy partition (Eq. (2.3)). Two computational methods, the *slope-based method* and the *modified transformation function method*, are introduced and compared to the well-known *discretization method*. The accuracy, straightforwardness of implementation and computational burden of the three mentioned techniques are examined for the three most commonly applied t-norms: the minimum $T_{\mathbf{M}}$, the product $T_{\mathbf{P}}$ and the Łukasiewicz t-norm $T_{\mathbf{L}}$.

In this work the linguistic output values are assumed to be described by trapezial membership functions forming a fuzzy partition. A fuzzy partition guarantees that a value in the domain characterized by a full membership to a certain linguistic value is completely excluded from all other linguistic values. Trapezial fuzzy partitions are used in many applications of fuzzy set theory, including modelling and control, pattern recognition and classification. Although they are based on intuitively plausible grounds and have become popular due to their striking simplicity of the membership functions, there exist deeper motivations for using them. Pedrycz (1994), for instance, showed that, based on a specific linear notion of entropy of fuzzy sets, suitably designed trapezial fuzzy partitions can guarantee uniformly excited fuzzy rule bases, in accordance with the distribution of the input variables. Since the work described in this chapter is about defuzzification, it essentially only requires a trapezial fuzzy partition of the domain of the output variable. In the same work (Pedrycz, 1994), Pedrycz provides an additional argument, based on a specific defuzzification procedure involving modal values, that the use of trapezial fuzzy partitions can guarantee an error-free inversion of the defuzzification strategy considered. The influence of the form of membership functions on the accuracy of fuzzy rule-based systems was also studied by Chang et al. (1991), although Delgado et al. (1998) enunciated that trapezial membership functions might adequately approximate other, *e.g.* Gaussian or exponential-shaped, membership functions, presenting the advantage of their simplicity as well (Cordón et al., 2001).

This chapter is organized as follows. After a short introduction on the COG defuzzification method, the three computational methods are presented in Section 3.2. Experimental results showing that the newly introduced methods exhibit excellent accuracy at an extremely low computational cost compared to the widely applied discretization method, are described in Section 3.3. Conclusions and further work are summarized in Section 3.4.

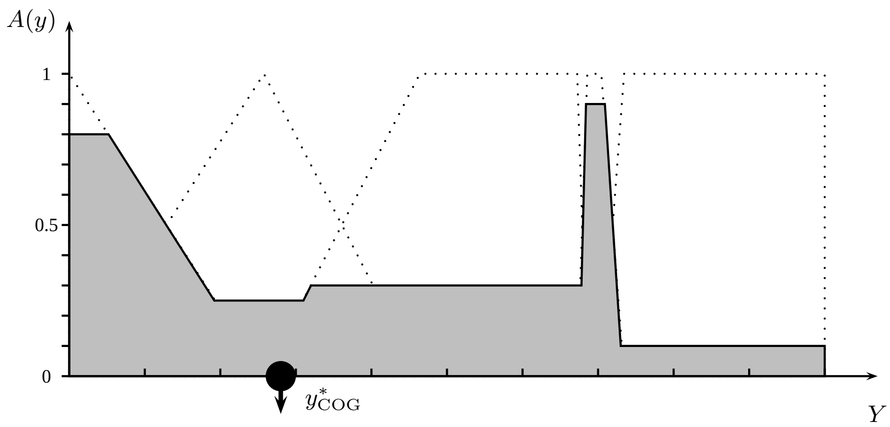


Figure 3.1: Center of gravity defuzzification method.

3.2 Computational methods for the COG defuzzification

3.2.1 COG defuzzification and related methods

The procedure applied in Mamdani–Assilian models to determine the model output is described in detail in Section 2.3.2. First the fulfilment degrees β_s ($s \in \{1, \ldots, r\}$) of the r rules R_s in the rule base are computed. In a next step the fulfilment degrees α_i ($i \in \{1, \ldots, n\}$) of the n linguistic output values A_i are determined and used to define the membership functions of the adapted membership functions A'_i. The global fuzzy output A is the union of the n adapted membership functions A'_i. Finally, the crisp model output y^* is obtained by defuzzifying the fuzzy output A, for instance with the COG defuzzification method. As illustrated in Fig. 3.1, the crisp output y^* obtained with the COG defuzzification method is the abscissa of the center of gravity of the surface F described by the fuzzy output A. The crisp output y^*_{COG} is defined by

$$y^*_{\mathrm{COG}} = \frac{\iint\limits_F y \, du \, dy}{\iint\limits_F \, du \, dy} \,. \tag{3.1}$$

In case the fuzzy output is the empty set, the midpoint of the domain (fixed beforehand) is returned as crisp output. Some authors (Cordón et al., 2001; Sakly and Benrejeb, 2003) refer to the above strategy as *Mode A* (aggregation first, defuzzification after) and propose converse procedures that consist of defuzzifying the individual adapted membership functions, and averaging the resulting crisp values in one way or another. The latter approach could be called *Mode B* (defuzzification first, aggregation after), but is not based on a solid theoretical basis.

In practice, y^*_{COG} is approximated by means of numerical methods. Although

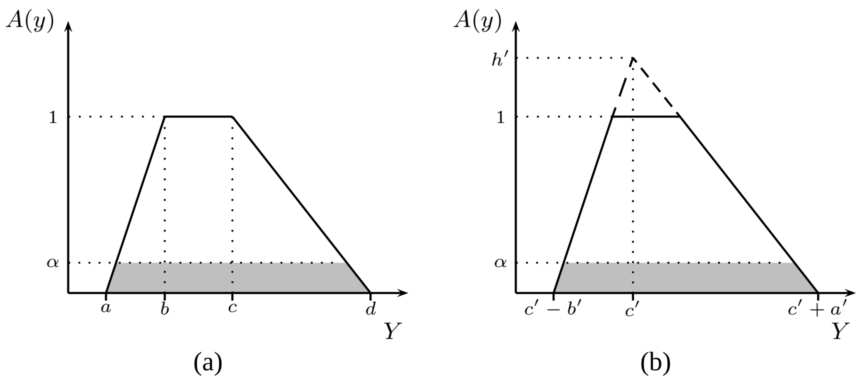


Figure 3.2: In this chapter representation (a) is used to describe trapezial membership functions, except in Section 3.2.4 where representation (b) is applied.

Eq. (3.1) is formally identical to the problem of determining the expected value of an unnormalized probability density function A, the functions A considered here are atypical for probability theory, and no results from that field can be drawn upon. Throughout this chapter, except in Section 3.2.4 (for reasons that will become clear), a trapezial membership function with support $[a, d]$ and kernel $[b, c]$ is represented by the four parameters a, b, c and d. As the modified transformation function method described in Section 3.2.4 is inspired by the transformation function method by Patel and Mohan (2002), their membership function representation by means of the parameters a', b', c' and h' is used. Both ways to represent trapezial membership functions are illustrated in Fig. 3.2.

3.2.2 Discretization method

The discretization method is the most straightforward implementation of the COG defuzzification strategy. The fuzzy output in the interval [y_{min},y_{max}] is approximated by k rectangles of equal width ($= (y_{max} - y_{min})/k$) and height $A(y_j)$. Eq. (3.1) is then converted into Eq. (3.2), in which $A(y_j)$ is the membership degree of y_j to the global fuzzy output and $A'_i(y_j)$ its membership degree to the adapted membership function of the i^{th} linguistic output value

$$y^*_{\mathrm{COG}} \approx \frac{\sum_{j=0}^{k-1} y_j \cdot A(y_j)}{\sum_{j=0}^{k-1} A(y_j)} \,, \tag{3.2}$$

$$\text{with} \qquad y_j = y_{min} + j \cdot \frac{y_{max} - y_{min}}{k} \,, \tag{3.3}$$

or, explicitly

$$y^*_{\mathrm{COG}} \approx \frac{\sum_{j=0}^{k-1} y_j \cdot \max\limits_{i=1}^{n} A'_i(y_j)}{\sum_{j=0}^{k-1} \max\limits_{i=1}^{n} A'_i(y_j)} \,. \tag{3.4}$$

The greater k is, the narrower the rectangles are in which the fuzzy output is divided and the closer the exact y^*_{COG} is approximated. However, Eq. (3.4) shows that not only the accuracy, but also the computational cost increases with increasing values of k.

When applying the discretization method to defuzzify a fuzzy output, the following calculations are executed:

1. construction of a vector with discretization point values y_j (Eq. (3.3)),
2. calculation of membership degrees $A_i(y_j)$ of all discretization points y_j in the vector for the n original membership functions,
3. calculation of membership degrees $A'_i(y_j)$ using the $A_i(y_j)$-values, the fulfilment degrees α_i and the appropriate t-norm (Eq. (2.47)),
4. determination of the maximum of the n membership degrees for each discretization point y_j (Eq. (2.48)) and
5. calculation of y^*_{COG} (Eq. (3.2)).

3.2.3 Slope-based method

3.2.3.1 Introduction

Like the other techniques presented in this work, the slope-based method is based on the fact that the moment of a surface about an axis equals the sum of the moments about the same axis of the surfaces obtained by *partitioning* this surface. The moment M_{ax} of a surface about an axis equals the product of its area O and the distance d_{ax} of its center of gravity to the axis

$$M_{\mathrm{ax}} = d_{\mathrm{ax}} \cdot O = d_{\mathrm{ax}} \cdot \sum_{j=1}^{k} O_j = \sum_{j=1}^{k} d_{\mathrm{ax},j} \cdot O_j = \sum_{j=1}^{k} M_{\mathrm{ax},j} \,. \tag{3.5}$$

In the slope-based method, the surface F described by the fuzzy output is partitioned such that the slope of the fuzzy output is constant within each part and different in two adjacent parts. When the output values are described by trapezial membership functions, as assumed in this work, linguistic fuzzy models with t-norm-based inference using the t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ always result in piecewise linear fuzzy outputs. A general representation of the partitioning of the fuzzy output is given in Fig. 3.3.

Note that the intersection of two parts obtained with the above-described method is always empty. This implies that the centers of gravity y^*_j and areas O_j of the k parts ($j = \{1, \ldots, k\}$) allow an exact computation of the crisp output y^*_{COG} as shown in

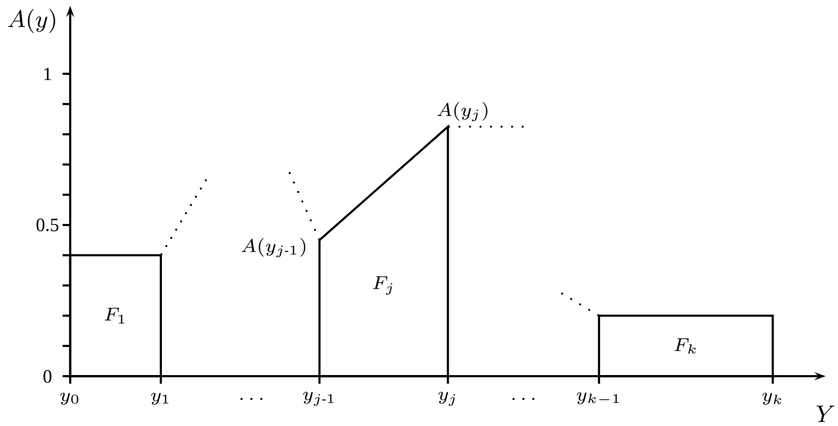


Figure 3.3: Partitioning of the fuzzy output obtained when applying the slope-based method.

Eq. (3.6). Furthermore, the centers of gravity y_j^* and areas O_j of the obtained parts are easy to compute

$$y^*_{\text{COG}} = \frac{\sum_{j=1}^{k} y_j^* \cdot O_j}{\sum_{j=1}^{k} O_j}\,, \tag{3.6}$$

$$\text{with} \qquad y_j^* = \frac{\iint\limits_{F_j} y\, du\, dy}{\iint\limits_{F_j} du\, dy}\,, \tag{3.7}$$

$$\text{and} \qquad O_j = \iint\limits_{F_j} du\, dy\,. \tag{3.8}$$

In the following lines, Eqs. (3.7) and (3.8) of the center of gravity and area of the parts are derived for a surface as depicted in Fig. 3.3. In the interval $[y_{j\text{-}1}, y_j]$ the fuzzy output is computed as follows

$$A(y) = \frac{A(y_j) - A(y_{j\text{-}1})}{y_j - y_{j\text{-}1}}(y - y_{j\text{-}1}) + A(y_{j\text{-}1})\,. \tag{3.9}$$

First the numerator and denominator in Eq. (3.7) are worked out separately

$$\iint\limits_{F_j} y\, du\, dy$$

$$= \int_{y_{j\text{-}1}}^{y_j} \int_0^{A(y)} y\, du\, dy$$

$$
\begin{aligned}
&= \int_{y_{j\text{-}1}}^{y_j} \left[\frac{A(y_j) - A(y_{j\text{-}1})}{y_j - y_{j\text{-}1}} (y - y_{j\text{-}1}) + A(y_{j\text{-}1}) \right] y\, dy \\
&= \frac{A(y_j) - A(y_{j\text{-}1})}{y_j - y_{j\text{-}1}} \left[\frac{y_j^3}{3} - \frac{y_{j\text{-}1}\, y_j^2}{2} + \frac{y_{j\text{-}1}^3}{6} \right] + \frac{A(y_{j\text{-}1})}{2} \left[y_j^2 - y_{j\text{-}1}^2 \right] \\
&= \frac{1}{6} (y_j - y_{j\text{-}1}) [(2A(y_j) + A(y_{j\text{-}1}))\, y_j + (A(y_j) + 2A(y_{j\text{-}1}))\, y_{j\text{-}1}] \,, \qquad (3.10)
\end{aligned}
$$

and

$$
\begin{aligned}
&\iint\limits_{F_j} du\, dy \\
&= \int_{y_{j\text{-}1}}^{y_j} \int_0^{A(y)} du\, dy \\
&= \int_{y_{j\text{-}1}}^{y_j} \left[\frac{A(y_j) - A(y_{j\text{-}1})}{y_j - y_{j\text{-}1}} (y - y_{j\text{-}1}) + A(y_{j\text{-}1}) \right] dy \\
&= \frac{A(y_j) - A(y_{j\text{-}1})}{y_j - y_{j\text{-}1}} \left[\frac{y_j^2}{2} - y_{j\text{-}1}\, y_j + \frac{y_{j\text{-}1}^2}{2} \right] + A(y_{j\text{-}1}) \left[y_j - y_{j\text{-}1} \right] \\
&= \frac{1}{2} (y_j - y_{j\text{-}1})(A(y_j) + A(y_{j\text{-}1})) \,. \qquad (3.11)
\end{aligned}
$$

This results in Eqs. (3.12) and (3.13) for the center of gravity y_j^* and area O_j of the parts

$$
\begin{aligned}
y_j^* &= \frac{\iint\limits_{F_j} y\, du\, dy}{\iint\limits_{F_j} du\, dy} \\
&= \frac{1}{3}(y_j + y_{j\text{-}1}) + \frac{1}{3} \frac{A(y_j)\, y_j + A(y_{j\text{-}1})\, y_{j\text{-}1}}{A(y_j) + A(y_{j\text{-}1})} \,, \qquad (3.12) \\
O_j &= \iint\limits_{F_j} du\, dy \\
&= \frac{1}{2}(y_j - y_{j\text{-}1})(A(y_j) + A(y_{j\text{-}1})) \,. \qquad (3.13)
\end{aligned}
$$

3.2.3.2 Transition points

In order to apply this new method, apart from the formulae for the centers of gravity and areas of the parts, also a method is needed to partition the fuzzy output. When determining the *transition points*, as the points defining the parts will be called in the following, it is assumed that the membership functions form a fuzzy partition. As the shape of the fuzzy output depends on the t-norm used to adapt the original membership functions (Eq. (2.47)), the procedure to determine the transition points is different for $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$. In the formulae of the slope-based method, parameters a, b, c and d are used to characterize the trapezial membership functions as illustrated in Fig. 3.2a.

Table 3.1: Co-ordinates of the potential transition points in Fig. 3.5.

point	co-ordinates
$pt_{\mathbf{M},1}$	$(d_i - \alpha_i\,(d_i - c_i), \alpha_i)$
$pt_{\mathbf{M},2}$	$(c_i + \alpha_i\,(d_i - c_i), \alpha_i)$
$pt_{\mathbf{M},3}$	$(d_i - \alpha_{i+1}\,(d_i - c_i), \alpha_{i+1})$
$pt_{\mathbf{M},4}$	$(c_i + \alpha_{i+1}\,(d_i - c_i), \alpha_{i+1})$
$pt_{\mathbf{M},5}$	$(\frac{1}{2}(c_i + d_i), 0.5)$
$pt_{\mathbf{P},1}$	(c_i, α_i)
$pt_{\mathbf{P},2}$	$(c_i + \frac{\alpha_i}{\alpha_i+\alpha_{i+1}}\,(d_i - c_i), \frac{\alpha_i \cdot \alpha_{i+1}}{\alpha_i+\alpha_{i+1}})$
$pt_{\mathbf{P},3}$	(d_i, α_{i+1})
$pt_{\mathbf{L},1}$	(c_i, α_i)
$pt_{\mathbf{L},2}$	$(c_i + \alpha_i\,(d_i - c_i), 0)$
$pt_{\mathbf{L},3}$	$(c_i + \frac{1}{2}(\alpha_i - \alpha_{i+1} + 1)(d_i - c_i), \frac{1}{2}(\alpha_i + \alpha_{i+1} - 1))$
$pt_{\mathbf{L},4}$	$(d_i - \alpha_{i+1}\,(d_i - c_i), 0)$
$pt_{\mathbf{L},5}$	(d_i, α_{i+1})

The case of $T_{\mathbf{M}}$ In Fig. 3.4 all possible configurations are depicted that may occur for two adjacent membership functions A_i and A_{i+1} when $T = T_{\mathbf{M}}$. At each overlap of the two membership functions A_i and A_{i+1}, the following five points, whose co-ordinates are listed in Table 3.1, should be taken in consideration as potential transition points:

- $pt_{\mathbf{M},1}$, the intersection of $A = \alpha_i$ with the line through (c_i,1) and (d_i,0),
- $pt_{\mathbf{M},2}$, the intersection of $A = \alpha_i$ with the line through (c_i,0) and (d_i,1),
- $pt_{\mathbf{M},3}$, the intersection of $A = \alpha_{i+1}$ with the line through (c_i,1) and (d_i,0),
- $pt_{\mathbf{M},4}$, the intersection of $A = \alpha_{i+1}$ with the line through (c_i,0) and (d_i,1),
- $pt_{\mathbf{M},5}$, the point of intersection of A_i and A_{i+1}.

The five points are indicated in Fig. 3.4. Note that in some configurations some of the points coincide. Actual transition points are coloured black. One can see that at each overlap up to three of the five points are added to the list of transition points. As a consequence, when $T = T_{\mathbf{M}}$ the total number of transition points for n membership functions is at least two (namely the first point (y_0,α_1) and the last point (y_k,α_n)) and at most $3n-1$ ($= 2+3(n-1)$). The rules used for the selection of transition points are visualized in Fig. 3.5a. If both degrees of fulfilment are larger than 0.5, the points $pt_{\mathbf{M},1}$, $pt_{\mathbf{M},4}$ and $pt_{\mathbf{M},5}$ are selected as transition points (Fig. 3.4a). If this is not the case and α_i is larger than α_{i+1}, the transition points are $pt_{\mathbf{M},1}$ and $pt_{\mathbf{M},3}$ (Fig. 3.4b,c,f,k); if α_i and α_{i+1} are equal no transition points are selected (Fig. 3.4e,j) and finally, if α_i is smaller than α_{i+1}, the transition points are $pt_{\mathbf{M},2}$ and $pt_{\mathbf{M},4}$ (Fig. 3.4d,g–i).

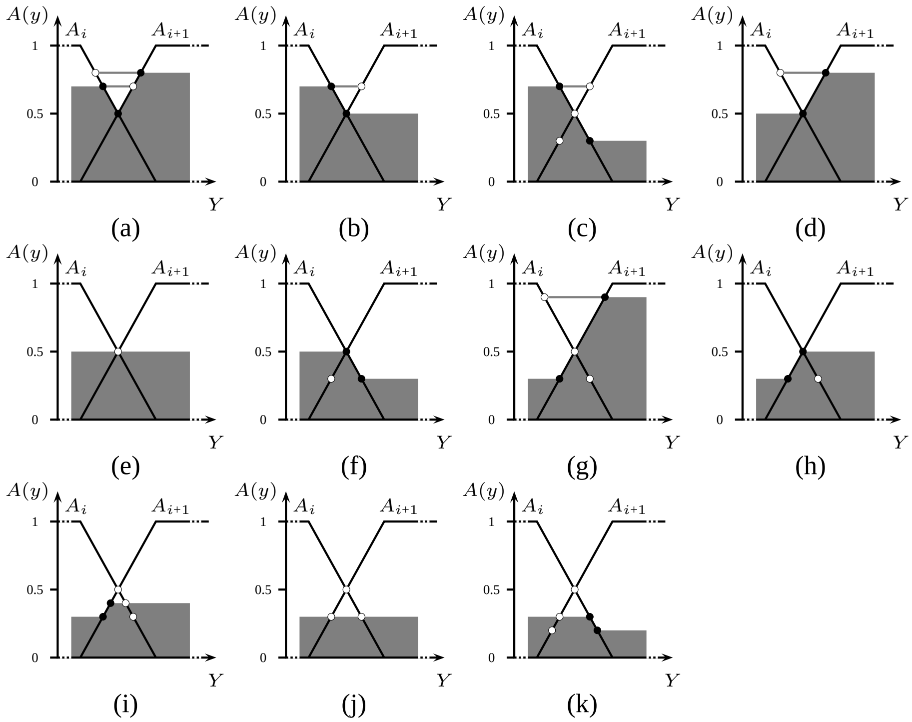


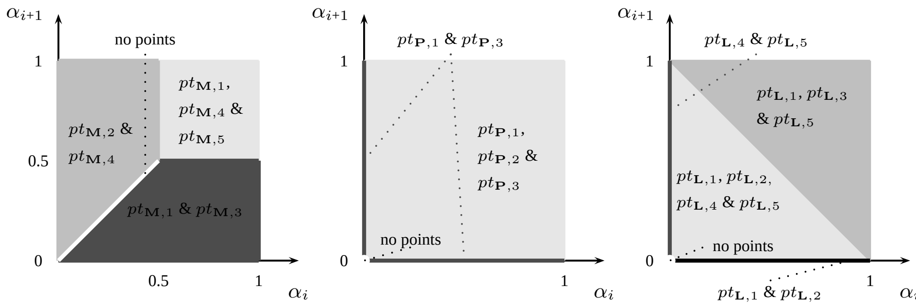
Figure 3.4: Different configurations at the intersection of two membership functions when $T = T_{\mathbf{M}}$.

The case of $T_{\mathbf{P}}$ The selection of transition points for $T_{\mathbf{P}}$ is more straightforward. The four typical configurations occurring at the intersection of two membership functions are shown in Fig. 3.6.

As long as both degrees of fulfilment α_i and α_{i+1} are strictly positive (Fig. 3.6a), the following three points are added to the list of transition points (see Table 3.1 for their co-ordinates):

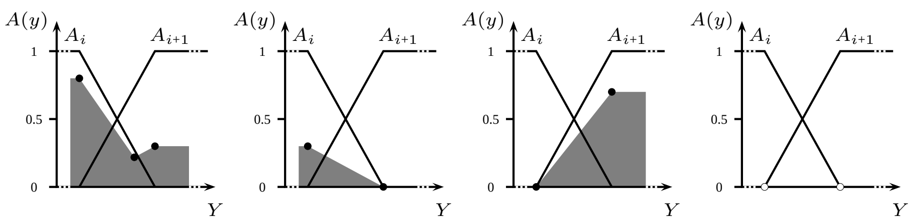
- $pt_{\mathbf{P},1}$, the point with the maximum of the kernel of A_i as abscissa and the membership degree in this point to A'_i as ordinate,
- $pt_{\mathbf{P},2}$, the point of intersection of the two adapted membership functions A'_i and A'_{i+1},
- $pt_{\mathbf{P},3}$, the point with the minimum of the kernel of A_{i+1} as abscissa and the membership degree in this point to A'_{i+1} as ordinate.

If one of the degrees of fulfilment is zero, only $pt_{\mathbf{P},1}$ and $pt_{\mathbf{P},3}$ are selected (Fig. 3.6b–c). No transition points are added to the list if both degrees of fulfilment are zero (Fig. 3.6d). Fig. 3.5b summarizes which points are selected as transition points



(a) case of $T_{\mathbf{M}}$ (b) case of $T_{\mathbf{P}}$ (c) case of $T_{\mathbf{L}}$

Figure 3.5: Selection of the transition points as a function of α_i and α_{i+1} in the slope-based method for the three t-norms.



(a) (b) (c) (d)

Figure 3.6: Different configurations at the intersection of two membership functions when $T = T_{\mathbf{P}}$.

given the degrees of fulfilment α_i and α_{i+1}. As for $T_{\mathbf{M}}$, the total number of transition points is at least two and at most $3n - 1$ for n membership functions.

The case of $T_{\mathbf{L}}$ When using $T_{\mathbf{L}}$, the following five points are the potential transition points (see Table 3.1 for their co-ordinates):

- $pt_{\mathbf{L},1}$, the point with the maximum of the kernel of A_i as abscissa and the membership degree in this point to A'_i as ordinate,
- $pt_{\mathbf{L},2}$, the intersection of the right non-parallel side of the adapted membership function A'_i with $A = 0$,
- $pt_{\mathbf{L},3}$, the point of intersection of the adapted membership functions A'_i and A'_{i+1},
- $pt_{\mathbf{L},4}$, the intersection of the left non-parallel side of the adapted membership function A'_{i+1} with $A = 0$,
- $pt_{\mathbf{L},5}$, the point with the minimum of the kernel of A_{i+1} as abscissa and the membership degree in this point to A'_{i+1} as ordinate.

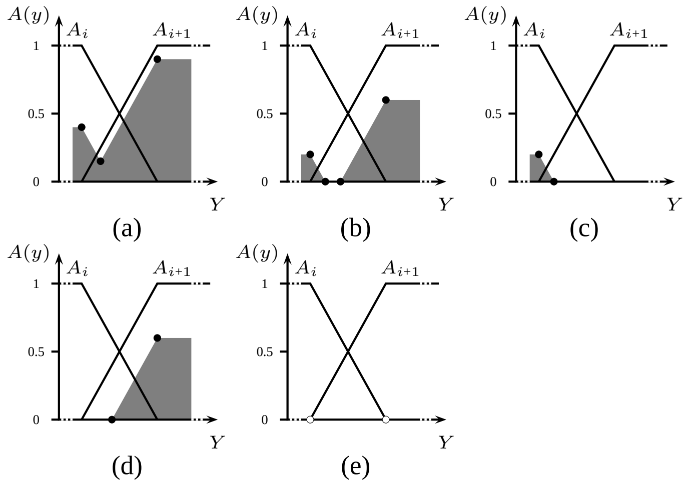


Figure 3.7: Different configurations at the intersection of two membership functions when $T = T_{\mathbf{L}}$.

If the sum of the fulfilment degrees α_i and α_{i+1} is larger than or equal to one, as shown in Fig. 3.7a, the transition points are $pt_{\mathbf{L},1}$, $pt_{\mathbf{L},3}$ and $pt_{\mathbf{L},5}$. If the sum of the fulfilment degrees α_i and α_{i+1} is strictly positive and smaller than 1: all points except $pt_{\mathbf{L},3}$ are selected if both fulfilment degrees are strictly positive (Fig. 3.7b), $pt_{\mathbf{L},1}$ and $pt_{\mathbf{L},2}$ are selected if α_{i+1} is zero (Fig. 3.7c) and $pt_{\mathbf{L},4}$ and $pt_{\mathbf{L},5}$ are selected if α_i is zero (Fig. 3.7d). Finally no points are selected if both fulfilment degrees are zero (Fig. 3.7e). The selection of transition points for $T = T_{\mathbf{L}}$ is summarized in Fig. 3.5c. The total number of transition points for n membership functions is at least two and at most $4n - 2$ ($= 2 + 4(n - 1)$) for $T_{\mathbf{L}}$.

3.2.3.3 Implementation

The practical implementation of the slope-based method consists of the following steps:

1. the transition points are determined (according to the rules visualized in Fig. 3.5),
2. y_i^* and O_i are calculated for each part with Eqs. (3.12–3.13),
3. y^*_{COG} is obtained with Eq. (3.6).

3.2.4 Modified transformation function method

The modified transformation function method is based on the transformation function method presented by Patel and Mohan. In their joint article (Patel and Mohan, 2002), they claim their method to be a computationally attractive technique to compute the

Table 3.2: Formulae for the transformation functions $f(h'_i, \alpha_i)$ and areas S_i of the adapted membership functions in Eqs. (3.15) and (3.16).

t-norm	$f(h'_i, \alpha_i)$	S_i
$T_{\mathbf{M}}$	$\frac{3{h'_i}^2 - 3\alpha_i h'_i + \alpha_i^2}{h'_i \cdot (2h'_i - \alpha_i)}$	$\frac{(a'_i + b'_i) \cdot \alpha_i \cdot (2h'_i - \alpha_i)}{2h'_i}$
$T_{\mathbf{P}}$	$\frac{3{h'_i}^2 - 3h'_i + 1}{h'_i \cdot (2h'_i - 1)}$	$\frac{(a'_i + b'_i) \cdot \alpha_i \cdot (2h'_i - 1)}{2h'_i}$
$T_{\mathbf{L}}$	$\frac{\alpha_i^2 + 3\alpha_i h'_i - 3\alpha_i + 3{h'_i}^2 - 6h'_i + 3}{h'_i \cdot (\alpha_i + 2h'_i - 2)}$	$\frac{(a'_i + b'_i) \cdot \alpha_i \cdot (2h'_i + \alpha_i - 2)}{2h'_i}$

center of area defuzzification, which they use as a synonym for the center of gravity defuzzification defined in Eq. (3.1), for triangular membership functions. In a more recent article, Patel (2004) correctly states that their method is not valid for the COG defuzzification, but for the computationally less demanding (Driankov et al., 1993, Section 3.6) *Center Of Sum* (COS) defuzzification, and extends the transformation function method to trapezial membership functions. The definition of the COS defuzzification is given by

$$y^*_{\mathrm{COS}} = \frac{\int\limits_Y y \cdot \sum_{i=1}^n A'_i(y)\, dy}{\int\limits_Y \sum_{i=1}^n A'_i(y)\, dy} \,. \tag{3.14}$$

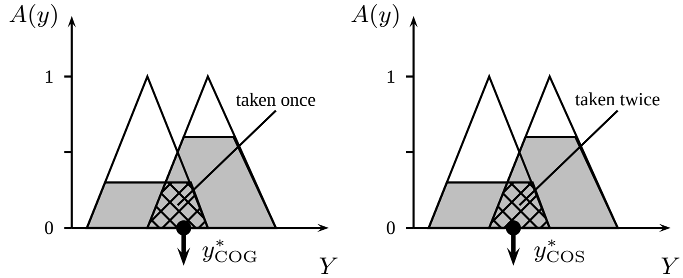
The crisp output y^*_{COS} is computed from the centers of gravity y^*_i and areas S_i of the adapted membership functions

$$y^*_{\mathrm{COS}} = \frac{\sum_{i=1}^n y^*_i \cdot S_i}{\sum_{i=1}^n S_i} \,, \tag{3.15}$$

$$\text{with} \qquad y^*_i = c'_i + \frac{a'_i - b'_i}{3} f(h'_i, \alpha_i) \,. \tag{3.16}$$

Given a trapezial fuzzy partition, the function f appearing in Eq. (3.16) only depends on the t-norm used to adapt the original membership functions (Eq. (2.47)). This function is called ‘transformation function’ by Patel and Mohan, whence the name of this method. In (Patel, 2004) the formulae for the transformation functions $f(h'_i, \alpha_i)$ and areas S_i of the adapted membership functions A'_i are derived for 12 t-norms. In Table 3.2 the formulae are shown for the basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$. The parameters a'_i, b'_i, c'_i and h'_i characterize the shape of the membership function of the i^{th} linguistic output value A_i. The parameter α_i is the corresponding fulfilment degree. The meaning of the parameters used in Eq. (3.16) and Table 3.2 is illustrated in Fig. 3.2b.

However, the center of sum defuzzification method is rarely applied. Based on the transformation function method, a new computational method is presented for the commonly used COG defuzzification method. The COS and COG defuzzification methods differ in the number of times overlapping parts of the adapted membership functions are taken into account: only once with the COG defuzzification method and more than once (twice in case of a fuzzy partition) with the COS defuzzification method. The difference between both defuzzification methods is illustrated in



(a) COG defuzzification (b) COS defuzzification

Figure 3.8: Center of gravity and center of sum defuzzification methods (after (Driankov et al., 1993, Section 3.6)).

Fig. 3.8. Sakly and Benrejeb (2003) have suggested yet another defuzzification method that amounts to the replacement of the areas S_i in Eq. (3.15) by the respective fulfilment degrees α_i. This approach fits into the *Mode B* strategy mentioned before, but does obviously not result in the true COG defuzzification.

The center of gravity y^*_{COG} of the surface defined by the global fuzzy output is obtained by taking not only the centers of gravity y_i^* and areas S_i of the n adapted membership functions into account, but also the centers of gravity $y^*_{\mathrm{op},i}$ and areas $S_{\mathrm{op},i}$ of the $n-1$ overlapping parts, which results in the following formula

$$y^*_{\mathrm{COG}} = \frac{\sum_{i=1}^{n}(y_i^* \cdot S_i) - \sum_{i=1}^{n-1}(y^*_{\mathrm{op},i} \cdot S_{\mathrm{op},i})}{\sum_{i=1}^{n} S_i - \sum_{i=1}^{n-1} S_{\mathrm{op},i}} \,. \tag{3.17}$$

This expression is formally similar to that of Wang and Luoh (2000) who also treat the COG defuzzification problem as a COS defuzzification problem accounting for overlapping parts. However, their approach requires the explicit computation of the co-ordinates of the vertices of the adapted membership functions as well as of their overlapping parts (both viewed as 2D-objects), and is therefore computationally not at all attractive.

When trapezial membership functions forming a fuzzy partition are adapted according to $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$, the overlapping areas are always trapezial or triangular. The formula for $y^*_{\mathrm{op},i}$ and $S_{\mathrm{op},i}$ are given in Table 3.3. In case of $T_{\mathbf{M}}$, the bases of the trapezial overlapping parts coincide with the projection of the non-parallel sides of the membership functions on the Y-axis as illustrated in Fig. 3.9. The height $h_{\mathrm{op},i}$ of the trapezium varies between 0 and 0.5 and depends on the fulfilment degrees of the two adjacent linguistic output values

$$h_{\mathrm{op},i} = \min(\alpha_i, \alpha_{i+1}, 0.5)\,. \tag{3.18}$$

As the membership functions form a fuzzy partition, the trapezia always have a vertical axis of symmetry through the intersection of the two adjacent membership functions. The center of gravity $y_{\mathrm{op},i}$ of the overlapping areas therefore has the same abscissa as the intersection of the two adjacent membership functions

$$y^*_{\mathrm{op},i} = c'_i + a'_i\left(1 - \frac{1}{2h'_i}\right). \tag{3.19}$$

The formula for the area $S_{\mathrm{op},i}$ can easily be derived given the co-ordinates of the overlapping parts in Fig. 3.9

$$\begin{aligned} S_{\mathrm{op},i} &= \frac{h_{\mathrm{op},i}}{2}\left(-c'_i - a'_i\left(1 - \frac{1}{h'_i}\right) - c'_i - a'_i\left(1 - \frac{1 - h_{\mathrm{op},i}}{h'_i}\right) + c'_i\right.\\ &\qquad \left. + a'_i\left(1 - \frac{h_{\mathrm{op},i}}{h'_i}\right) + c'_i + a'_i\right)\\ &= \frac{a'_i h_{\mathrm{op},i}(1 - h_{\mathrm{op},i})}{h'_i}. \end{aligned} \tag{3.20}$$

For $T_{\mathbf{P}}$ the centers of gravity $y^*_{\mathrm{op},i}$ and areas $S_{\mathrm{op},i}$ of the triangular overlapping parts are given by (Fig. 3.10)

$$\begin{aligned} y^*_{\mathrm{op},i} &= \frac{1}{3}\left(c'_i + a'_i\left(1 - \frac{1}{h'_i}\right) + c'_i + a'_i\left(1 - \frac{\alpha_{i+1}}{h'_i(\alpha_i + \alpha_{i+1})}\right) + c'_i + a'_i\right)\\ &= c'_i + a'_i\left(1 - \frac{\alpha_i + 2\alpha_{i+1}}{3h'_i(\alpha_i + \alpha_{i+1})}\right), \end{aligned} \tag{3.21}$$

$$\begin{aligned} S_{\mathrm{op},i} &= \frac{1}{2}\,\frac{\alpha_i\,\alpha_{i+1}}{\alpha_i + \alpha_{i+1}}\,\left(c'_i + a'_i - c'_i - a'_i\left(1 - \frac{1}{h'_i}\right)\right)\\ &= \frac{a'_i\,\alpha_i\,\alpha_{i+1}}{2h'_i(\alpha_i + \alpha_{i+1})}. \end{aligned} \tag{3.22}$$

For $T_{\mathbf{L}}$, overlapping parts are only obtained if the sum of the fulfilment degrees α_i and α_{i+1} of the two consecutive membership functions A_i and A_{i+1} is larger than 1. In this case, the formulae for the centers of gravity $y^*_{\mathrm{op},i}$ and areas $S_{\mathrm{op},i}$ are given by (Fig. 3.11)

$$\begin{aligned} y^*_{\mathrm{op},i} &= \frac{1}{3}\left(c'_i + a'_i\left(1 - \frac{\alpha_{i+1}}{h'_i}\right) + c'_i + a'_i\left(1 - \frac{1 - \alpha_i + \alpha_{i+1}}{2h'_i}\right) + c'_i\right.\\ &\qquad \left. + a'_i\left(1 - \frac{1 - \alpha_i}{h'_i}\right)\right)\\ &= c'_i + a'_i\left(1 - \frac{1 - \alpha_i + \alpha_{i+1}}{2h'_i}\right), \end{aligned} \tag{3.23}$$

$$\begin{aligned} S_{\mathrm{op},i} &= \frac{1}{4}\left(\alpha_i + \alpha_{i+1} - 1\right)\left(c'_i + a'_i\left(1 - \frac{1 - \alpha_i}{h'_i}\right) - c'_i - a'_i\left(1 - \frac{\alpha_{i+1}}{h'_i}\right)\right)\\ &= \frac{a'_i}{4h'_i}(\alpha_i + \alpha_{i+1} - 1)^2. \end{aligned} \tag{3.24}$$

The practical implementation of the modified transformation function method consists of the following steps:

Table 3.3: Formulae for the centers of gravity $y^*_{\mathrm{op},i}$ and areas $S_{\mathrm{op},i}$ of the overlapping parts in Eq. (3.17).

t-norm	$y^*_{\mathrm{op},i}$	$S_{\mathrm{op},i}$
$T_{\mathbf{M}}$	$c'_i + a'_i \cdot (1 - \frac{1}{2h'_i})$	$\frac{a'_i \cdot \min(\alpha_i,\alpha_{i+1},0.5)\cdot(1-\min(\alpha_i,\alpha_{i+1},0.5))}{h'_i}$
$T_{\mathbf{P}}$	$c'_i + a'_i \cdot (1 - \frac{\alpha_i+2\alpha_{i+1}}{3h'_i\cdot(\alpha_i+\alpha_{i+1})})$	$\frac{a'_i\cdot\alpha_i\cdot\alpha_{i+1}}{2h'_i\cdot(\alpha_i+\alpha_{i+1})}$
$T_{\mathbf{L}}$	$c'_i + a'_i \cdot (1 - \frac{1-\alpha_i+\alpha_{i+1}}{2h'_i})$	$\frac{a'_i\cdot(\max(\alpha_i+\alpha_{i+1}-1,0))^2}{4h'_i}$

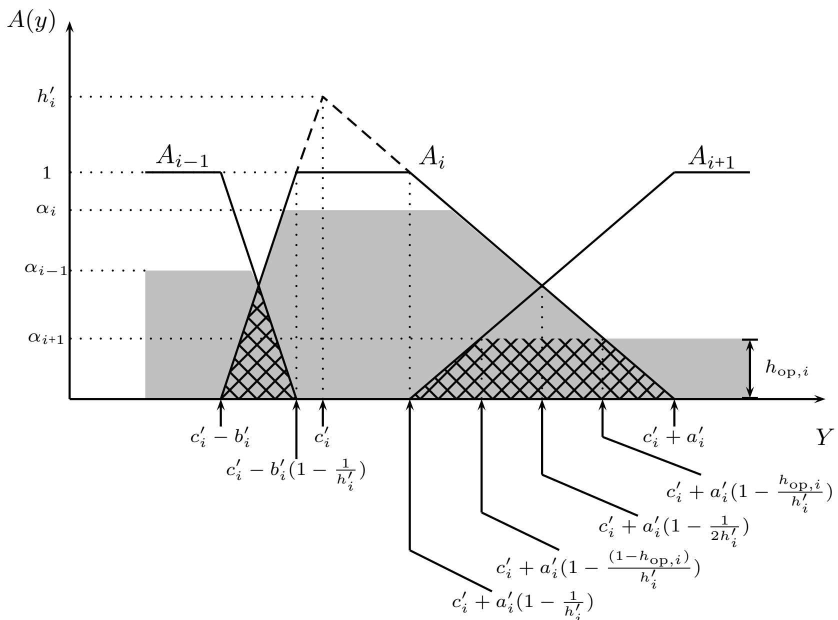


Figure 3.9: Co-ordinates of the trapezial membership functions and the triangular overlapping parts for $T_{\mathbf{M}}$.

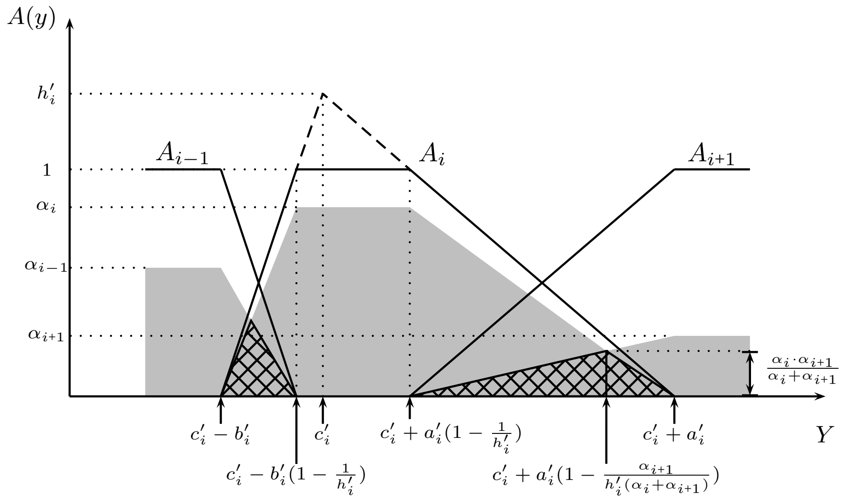


Figure 3.10: Co-ordinates of the trapezial membership functions and the triangular overlapping parts for $T_{\mathbf{P}}$.

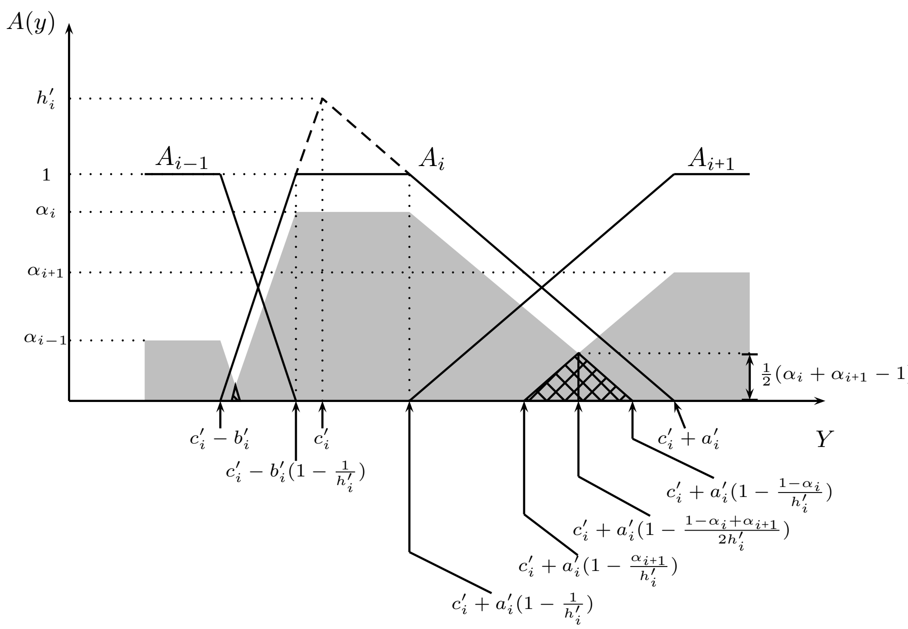


Figure 3.11: Co-ordinates of the trapezial membership functions and the triangular overlapping parts for $T_{\mathbf{L}}$.

1. the transformation function value $f(h'_i, \alpha_i)$, the center of gravity $y^*_{\mathrm{op},i}$ and the areas S_i and $S_{\mathrm{op},i}$ are calculated for all linguistic values (Tables 3.2 and 3.3),

2. y^*_i is calculated for all linguistic values (Eq. (3.16)),

3. y^*_{COG} is computed using Eq. (3.17).

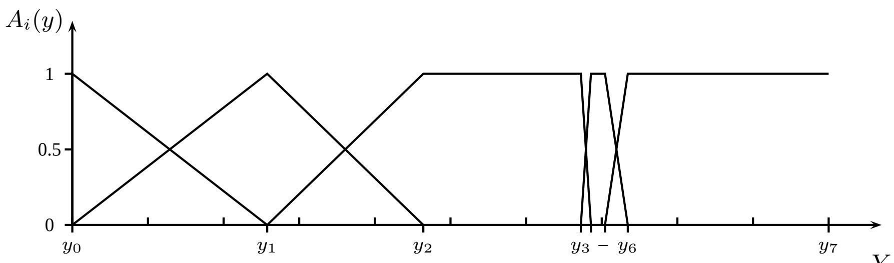
3.3 Experiments and results

3.3.1 Implementation

As the defuzzification function was meant to be called by the objective function of a model optimization algorithm, requiring, at once, the calculation of the corresponding crisp outputs of N (typically a few hundreds) training examples, the numerical methods were implemented as functions taking Nn fulfilment degrees defining N fuzzy outputs, $4n$ membership function parameters defining n linguistic values, a label indicating the t-norm and, in case of the discretization method, the number of discretization steps k as input and returning N crisp outputs. Note that, when defuzzifying N fuzzy outputs of a same model, or more generally, N fuzzy outputs defined on the same fuzzy partition of trapezial output membership functions, some calculation steps should be carried out only once. This concerns for instance the calculation of the discretization point values y_j and membership degrees $A_i(y_j)$ in the first and second step of the discretization method and the calculation of the α_i-independent factors for each of the n linguistic values in Tables 3.2 and 3.3 when applying the modified transformation function method. For the discretization method, the time complexity is $\mathcal{O}(Nnk)$ and the space complexity is $\mathcal{O}(nk)$. For the two other methods to execute the COG defuzzification and the transformation function method by Patel (2004), the time complexity is $\mathcal{O}(Nn)$ and the space complexity is $\mathcal{O}(n)$. All programs were written in MATLAB and executed on a 1,8 GHz AMD Athlon with 512 Mb RAM.

3.3.2 Experimental setup

To illustrate the differences in accuracy and computational cost of the methods presented in Section 3.2, the centers of gravity of 1000 fuzzy outputs were calculated for $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ via the three computational methods for the COG defuzzification method and via the transformation function method presented by Patel (2004). For the discretization method the number of discretization steps was varied: 50, 100, 250, 500, 1000, 2500 and 5000 discretization steps were used. The same five membership functions shown in Fig. 3.12 and the same randomly generated set of 1000 times 5 fulfilment degrees were used during all computations. Note that some of the used membership functions have a particular shape which often occurs in optimization processes. When for instance two parameter values coincide, a triangular instead of the more general trapezial membership function is obtained or small differences between successive parameters result in membership functions with a very narrow support. This particularly



y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7
0.000	25.768	46.407	67.237	68.574	70.429	73.444	100.000

Figure 3.12: Membership functions of the output variable Y used in the experiment.

shaped output membership functions often give rise to low accuracy when carrying out the defuzzification with the discretization method.

The RMSE was used as a measure for the accuracy

$$\text{RMSE} = \sqrt{\frac{\sum_{z=1}^{N}(y_z - y_{z,\text{MTF}})^2}{N}} \,. \tag{3.25}$$

We considered the results obtained with the modified transformation function (MTF) method as reference values. When applying the modified transformation function method, apart from round-off errors by the computer, no approximations are made. The computational burden of a method was assumed to be proportional to the time needed to compute the crisp outputs for the 1000 fuzzy outputs.

3.3.3 Results

The results obtained during 50 repetitions of the experiment are shown in Table 3.4. In the first column the RMSE-values are listed. Further, the table contains the average absolute calculation times t_a as well as the relative average calculation times $t_{\text{r,MTF}}$ to the calculation time needed with the modified transformation function method.

As expected, the accuracy and computational cost of the discretization method increases with increasing discretization steps (Fig. 3.13). When examining the results in Table 3.4, its easy implementation appears to be the only advantage of the discretization method. The two other methods to compute the COG defuzzification method are not as straightforward to implement but allow for both a quicker and more accurate computation. The same accuracies are obtained with the slope-based and modified transformation function method, but due to the fact that no transition points have to be determined in the modified transformation function method, the latter is faster. Finally, the results obtained with the transformation function method (Patel, 2004) show that taking the overlapping areas twice into account instead of once results in an error of more than respectively 3%, 2% and 0.5% for $T_\mathbf{M}$, $T_\mathbf{P}$ and $T_\mathbf{L}$, reflecting the decreasing amount of overlapping.

Table 3.4: RMSE-values and average computation times (over 50 runs) for the different methods and t-norms.

	$T = T_{\mathbf{M}}$			$T = T_{\mathbf{P}}$			$T = T_{\mathbf{L}}$		
Method	RMSE (-)	t_{a} (s)	$t_{\mathrm{r,MTF}}$ (-)	RMSE (-)	t_{a} (s)	$t_{\mathrm{r,MTF}}$ (-)	RMSE (-)	t_{a} (s)	$t_{\mathrm{r,MTF}}$ (-)
discret. (50)	1.03	0.13	2	1.21	0.11	1	1.43	0.13	0.7
discret. (100)	0.51	0.16	3	0.58	0.14	2	0.67	0.17	0.9
discret. (250)	0.20	0.29	5	0.23	0.24	3	0.26	0.32	2
discret. (500)	0.10	0.51	9	0.11	0.39	5	0.13	0.52	3
discret. (1000)	0.05	1.02	20	0.06	0.79	10	0.06	1.07	6
discret. (2500)	0.02	2.64	40	0.02	2.31	30	0.03	2.90	20
discret. (5000)	0.01	5.09	90	0.01	4.64	60	0.01	5.67	30
slope-based	0.00	0.15	3	0.00	0.15	2	0.00	0.16	0.8
transf. funct.	3.51	0.03	0.5	2.53	0.02	0.2	0.79	0.14	0.7
m. transf. funct.	—	0.06	—	—	0.08	—	—	0.19	—

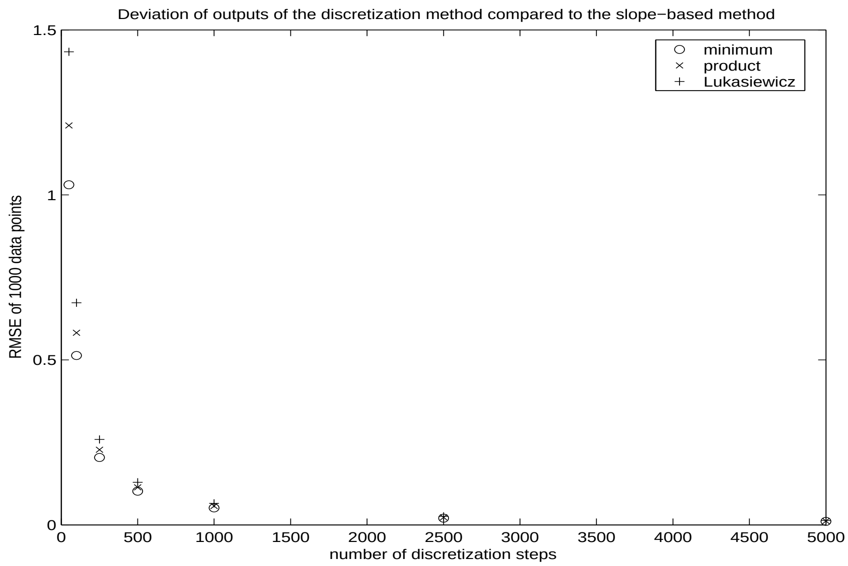


Figure 3.13: RMSE as a function of the number of discretization steps.

3.4 Conclusion

In this chapter two computational methods, the slope-based method and the modified transformation function method, were introduced for the center of gravity defuzzification method for trapezial membership functions forming a fuzzy partition. The accuracy, computational cost and implementational complexity of these two methods and the commonly applied discretization method were discussed for the basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$. Its easy implementation appears to be the only advantage of the discretization method. The two other methods to compute the COG defuzzification method are not as straightforward to implement but allow both a quicker and more accurate computation. Of the three methods presented, the modified transformation function method has the smallest computational cost while being as accurate as the slope-based method. Note that in this study the linguistic output values were assumed to be described by trapezial membership functions forming a fuzzy partition.

Future investigations could imply attractive computational methods for the defuzzification of fuzzy rule-based models applying implicator-based inference. Furthermore, it should be checked whether the computational methods could be further simplified if not only the membership functions of the output variable, but also those of the input variables are assumed to form a fuzzy partition or if the rule base is assumed to be smooth. A rule base is called smooth if every set of two rules differing in only one input variable in their antecedent and containing adjacent values for this variable, have equal or adjacent values in their consequent as defined in Definition 7.3 in Section 7.2.2.

CHAPTER 4

Genetic algorithms

Once upon a time a fire broke out in a hotel, where just then a scientific conference was held. It was night and all guests were sound asleep. As it happened, the conference was attended by researchers from a variety of disciplines. The first to be awakened by the smoke was a mathematician. His first reaction was to run immediately to the bathroom, where, seeing that there was still water running from the tap, he exclaimed: "There is a solution!". At the same time, however, the physicist went to see the fire, took a good look and went back to his room to get an amount of water, which would be just sufficient to extinguish the fire. The engineer was not so choosy and started to throw buckets and buckets of water on the fire. Finally, when the biologist awoke, he said to himself: "The fittest will survive" and went back to sleep.
(Anecdote originally told by C.L. Liu)

4.1 Introduction

Genetic algorithms (Goldberg, 1989; Holland, 1975) are one of the four main types of evolutionary algorithms, as the general class of search and optimization methods which imitate the principles of natural evolution is called. The three other main groups of evolutionary algorithms are (Cordón et al., 2001; Eiben and Smith, 2003): evolutionary programming (Fogel et al., 1966), evolution strategies (Rechenberg, 1973) and genetic programming (Koza, 1993). The computer science field dealing with evolutionary algorithms is referred to as evolutionary computation. That some computer scientists have chosen natural evolution as a source of inspiration is not surprising. The power of evolution in nature is evident in the diverse species on earth, all being well-adjusted to survive in their specific niche (Eiben and Smith, 2003).

The adaptation of species to a specific niche has occurred due to selective pressure from the environment. Species that are more successful at avoiding death have the opportunity to produce more offspring than those that die young. This offspring inherits some of the beneficial features from the parents, allowing it to survive even better under the environmental conditions. On average, the survival fitness of a generation increases due to this selective pressure. Furthermore, because of the recombination of the genetic material of the parents, the offspring develops new features that were not present in one of the parents. Mutation once in a while throws in a wild card. This wild card is sometimes bad causing early death, but occasionally it creates a somewhat different feature that allows an individual to be even more successful than would have been the case after simple recombination of the parents' genetic material. In fact mutation increases the diversity in a population. As the environment (the predators for example) is dynamic, there is a constant struggle of all species to stay at the edge of the current 'genetic' technology. Species that adapt too slowly to new environmental conditions, will get extinct some time or another (Ducheyne, 2003).

In evolutionary algorithms a population of candidate solutions of the optimization problem (individuals) is evolved. The fitness of the individuals is obtained by an objective function. Operators mimicking natural selection (survival of the fittest) cause a rise in the fitness of the population. The general scheme of an evolutionary algorithm is given in pseudocode in Alg. 1. The optimization process starts with a population of either randomly generated or previously known candidate solutions. Based on their fitness, some of the better candidates are chosen to seed the next generation by applying recombination and/or mutation to them. Recombination is an operator applied to two or more selected candidates (the so-called parents) and results in one or more new candidates (the children). Mutation is applied to one candidate and results in one new candidate. Executing recombination and mutation leads to a set of new candidates (the offspring) that compete — based on their fitness (and possibly age) — with the old ones for a place in the next generation. This process can be iterated until a candidate with sufficient quality (a solution) is found or a previously set computational limit is reached (Eiben and Smith, 2003).

Algorithm 1: The general scheme of an evolutionary algorithm in pseudocode

$t \leftarrow 0$
Initialize Population P_t at random
Evaluate P_t
while *stopping criterion not met* **do**
  Select parents from P_t
  Recombine parents
  Mutate the resulting offspring
  Evaluate new candidates P_{t+1}
  Select individuals for the next generation
  $t \leftarrow t + 1$
end

The four variants of evolutionary algorithms differ in the data structure used to represent a candidate solution, the relative importance of recombination and mutation as variation operator as well as in the procedures applied to select individuals as parents or as individuals of the next generation. Typically, the candidate solutions are represented by strings over a finite alphabet in genetic algorithms, real-valued vectors in evolution strategies, finite state machines in evolutionary programming, and trees in genetic programming. In genetic algorithms and genetic programming, recombination and mutation are respectively the primary and secondary variation operators, whereas in evolution strategies the reverse order applies (Eiben and Smith, 2003).

In this chapter, genetic algorithms, the type of evolutionary algorithms applied in Chapter 6 for the optimization of membership functions of a Mamdani–Assilian model, are described. First, the different elements and operators of a genetic algorithm and the biologically inspired terminology used to refer to them, are introduced by means of a simple example. Then, following issues that should be addressed when setting up a genetic algorithm, are discussed:

- the representation of a candidate solution,
- the parent selection procedure,
- the variation operators,
- the replacement procedure, and
- the parameter setting

A sixth important component that should be specified in order to define a genetic algorithm is the fitness function. This issue is highly related to the optimization problem under consideration. It is therefore not addressed in this general chapter on genetic algorithms, but discussed in detail in Section 6.2.2.

4.2 Terminology

In the example below, after Eiben and Smith (2003), one selection-reproduction cycle of a genetic algorithm is illustrated. The objective of the considered optimization problem is to find the integer value $X \in \{0, \ldots, 15\}$ for which the fitness function is maximum

$$\text{fitness}(X) = 2 + \sin(\frac{\pi X}{7} - \frac{3}{4})\,. \tag{4.1}$$

The fitness of the 16 candidate solutions $\{0, \ldots, 15\}$ is shown in Fig. 4.1.

The data structure used in a genetic algorithm to represent a candidate solution is referred to as a *chromosome*. In the example, the chromosome $C = (c_1, \ldots, c_n)$ is a binary string of four bits and the corresponding integer value x is obtained by ($x_{\min} = 0, x_{\max} = 15, n = 4$)

$$x = x_{\min} + \frac{x_{\max} - x_{\min}}{2^n - 1} \sum_{i=1}^{n} c_i\, 2^{i-1}\,. \tag{4.2}$$

The binary string is called the *genotype* and the corresponding integer value the *phenotype* of a candidate solution. All candidate solutions are hereby represented: chromosome 0000 for instance represents $x = 0$ and chromosome 1111 represents $x = 15$. In

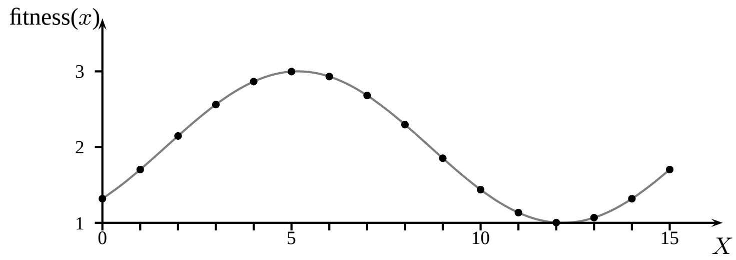
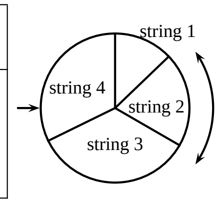


Figure 4.1: Fitness of the 16 candidate solutions in the genetic algorithm example.

Table 4.1: Illustration of a genetic algorithm: initiation, evaluation and parent selection.

string no.	initial population	x	fitness	selection prob.	times selected
1	1 1 0 1	13	1.07	0.13	0
2	0 0 0 1	1	1.70	0.20	1
3	0 1 0 0	4	2.87	0.34	2
4	0 1 1 1	7	2.68	0.32	1



this case the string represent only one variable, but a binary string or a string of reals $\mathbf{x} = (x_1, \ldots, x_n)$ can represent n variables or *genes* with each gene taking several values or *alleles*.

In the first step of a genetic algorithm, a *population* of *individuals* is initialized, either generated randomly or seeded by previously known solutions. Here the population size is equal to four. The four genotypes of the initial population are shown in Table 4.1 with the corresponding phenotypes and fitness values. The genetic algorithm applies fitness proportionate selection as parent selection procedure. The probability that an individual is selected as a parent is given in the column 'selection prob.' and is given by the quotient of its fitness and the average fitness of all individuals in the population. The population is mapped to a roulette wheel such that the slot size of each individual corresponds to its selection probability. The wheel is spun four times and the number of copies of an individual in the *mating pool* corresponds to the number of times the pointer pointed to the segment corresponding to the individual when the wheel stopped. Note that in the column 'times selected' *one* possible outcome of the parent selection procedure is shown.

As *crossover* and *mutation* procedures respectively one-point crossover and bit flip are applied. The selected individuals are paired at random, and for each pair a random point along the string is chosen. The children are created by splitting both parents at this point and exchanging the tails. In Table 4.2 the results of crossover on the given mating pool are given for crossover points after the first and third bit respectively. Next, a random number (from a distribution uniform over the range [0,1])

Table 4.2: Illustration of a genetic algorithm: recombination.

string no.	parents	crossover point	offspring after xover	mutation mask	offspring after mutation
1	0 | 0 0 1	1	0 1 0 0	0 0 0 0	0 1 0 0
2	0 | 1 0 0	1	0 0 0 1	0 0 0 0	0 0 0 1
3	0 1 0 | 0	3	0 1 0 1	0 0 0 0	0 1 0 1
4	0 1 1 | 1	3	0 1 1 0	1 0 0 0	1 1 1 0

Table 4.3: Illustration of a genetic algorithm: replacement procedure and evaluation of the next generation.

string no.	offspring	x	fitness	worst offspring	generation 1	x	fitness
1	0 1 0 0	4	2.87		0 1 0 0	4	2.87
2	0 0 0 1	1	1.70		0 0 0 1	1	1.70
3	0 1 0 1	5	3.00		0 1 0 1	5	3.00
4	1 1 1 0	14	1.32	$\times$	0 1 0 0	4	2.87

is generated for each bit position. Positions for which this random number is smaller than a fixed low (*e.g.* 0.001) value, the *mutation probability*, are indicated by a one in the *mutation mask* and the corresponding bits are flipped.

In Table 4.3 the genotypes, phenotypes and fitness values of the offspring obtained after crossover and mutation are shown. In the example, the next *generation* is obtained by *generational replacement* with *elitism*. Generational replacement means that the whole population is replaced by the offspring. Elitism guarantees that the best individual of a generation is never worse that the best individual of the preceding generation, for instance by replacing the worst offspring by the best individual of the current population. Although manually engineered, this example shows a typical progress: the average fitness increases from 2.08 to 2.61, and the best fitness in the population from 2.87 to 3.00 after crossover and mutation.

A genetic algorithm as the one applied in the example using a binary representation, fitness proportionate selection, a low probability of mutation, and an emphasis on genetically inspired recombination as a means of generating new candidate solutions, is referred to as a canonical or simple genetic algorithm. The theoretical foundation why genetic algorithms work, is usually illustrated using a simple genetic algorithm and can be found in the textbooks by Goldberg (1989) and Michalewicz (1996).

4.3 Binary and real-valued representation

The way a candidate solution is represented might be critical for the success or failure of the optimization process (Eiben and Smith, 2003; Michalewicz, 1996). The success of a representation can be evaluated by the best fitness value obtained after running the genetic algorithm with the different representation. In Chapter 6 the performances of a binary-coded and a real-coded genetic algorithm are compared for a membership function optimization problem.

Fixed-length and binary-coded strings for the representation of candidate solutions, as used in the simple genetic algorithm in Section 4.2, tend to dominate in research and applications of genetic algorithms. The use of a binary representation is mainly inspired by the outcome of the theoretical analysis of genetic algorithms by Holland (1975) and Goldberg (1989), recommending the use of alphabets of low cardinality. In a binary representation the alphabet with the lowest possible cardinality is applied, as bits only take values from the alphabet $\{0, 1\}$ with cardinality two. More recently, however, the use of genetic algorithms applying strings of real values, *i.e.* with large alphabets, is rising. Real-coded genetic algorithms, as genetic algorithms applying strings of real values are called, showed to outperform binary-coded genetic algorithms in the optimization problems presented by Wright (1991); Michalewicz (1996); Herrera et al. (1998). Furthermore theoretical foundations were established on the reason why and the way in which real-coded genetic algorithms are suitable optimization algorithms (Antonisse (1989) and Eshelman and Schaffer (1993) in Herrera et al. (1998) as well as Goldberg (1991) and Radcliffe (1991)).

The binary representation allows the use of straightforward variation operators, but has some drawbacks as illustrated below with two examples from the fuzzy model optimization field. A first disadvantage of binary-coded genetic algorithms is the difficulties they meet when dealing with continuous search spaces where a great numerical precision is required, for instance when searching the set of variables $\mathbf{a} = (a_2, \ldots, a_{2n\text{-}1})$ defining n membership functions as shown in Fig. 4.2, which maximize a certain fitness. If a binary representation is used, each variable $a_k \in [a_{\min,k}, a_{\max,k}]$ ($k \in \{2, \ldots, 2n\text{-}1\}$) is represented by a binary string C_k of $n_{\text{bit},k}$ bits

$$C_k = (c_{k,1}, \ldots, c_{k,n_{\text{bit},k}}) \qquad \text{with} \qquad c_{k,i} \in \{0, 1\}\,, \tag{4.3}$$

and a candidate solution is represented by a chromosome C

$$C = (C_2, \ldots, C_{2n\text{-}1})\,. \tag{4.4}$$

The phenotype $\mathbf{a}$ of a chromosome C is obtained by

$$\mathbf{a} = (a_2, \ldots, a_{2n\text{-}1})\,, \tag{4.5}$$

$$\text{with} \quad a_k = a_{\min,k} + \frac{a_{\max,k} - a_{\min,k}}{2^{n_{\text{bit},k}} - 1} \sum_{i=1}^{n_{\text{bit},k}} c_{k,i}\, 2^{i-1}\,. \tag{4.6}$$

A variable a_k encoded by a 2-bit string, can take one of the values of the four-element set

$$\{a_{\min,k}, \frac{2}{3}a_{\min,k} + \frac{1}{3}a_{\max,k}, \frac{1}{3}a_{\min,k} + \frac{2}{3}a_{\max,k}, a_{\max,k}\}\,.$$

If the variable should be defined by a higher precision, the number of bits used to encode it should be increased, resulting in a larger search space. For example, for 100

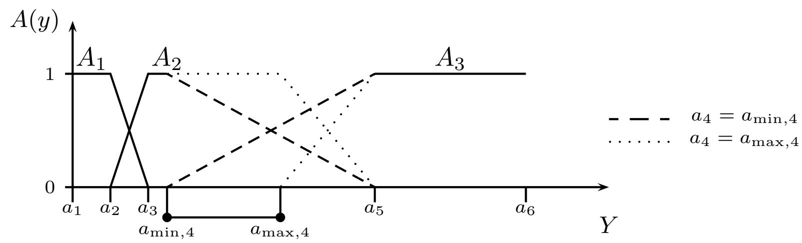


Figure 4.2: Optimization of membership functions used in a fuzzy model.

variables with domains in the range [-500,500] where a precision of six digits after the decimal point is required

$$\frac{a_{\max,k} - a_{\min,k}}{2^{n_{\mathrm{bit},k}} - 1} = 10^{-6} ,$$

the length of the binary solution vector is 3000. This, in turn, generates a search space of about 10^{900} ($= 2^{3000}$). For such problems genetic algorithms perform poorly. During the first generations, the algorithm wastes efforts evaluating the less significant digits of the binary coded variables. However, their optimum values depend on the most significant digits. As long as the most significant digits are not converged, the manipulation of less significant digits is useless. When convergence of the most significant digits is achieved, it is not necessary to waste more efforts on them. However, this ideal behaviour is not achieved by the genetic algorithm since all digits are handled in a similar way (Herrera et al., 1998).

Both Michalewicz (1996) and Eiben and Smith (2003) call a string of real values the most sensible way to represent variables originating from a continuous distribution. Real-coded genetic algorithms offer the advantage that continuous parameters can gradually adapt to the fitness landscape over the entire search space whereas parameter values in binary implementations are limited to a certain interval and resolution. The real-valued representation is sometimes referred to as the floating-point representation as the precision of these real values are actually limited by that of the computer on which the algorithm is executed. By using a real-valued representation the distinction between genotype and phenotype is blurred, since in many problems the real-number vector already embodies a solution in a natural way. This is also the case in the membership function optimization problem considered above where the chromosome C and phenotype $\mathbf{a}$ are obtained by

$$C = (c_1, \ldots, c_{n_{\mathrm{bit}}}) \qquad \text{with} \qquad c_i \in [c_{\min,i}, c_{\max,i}] \subset \mathbb{R} , \tag{4.7}$$

$$\mathbf{a} = (a_2, \ldots, a_{2n\text{-}1}) \qquad \text{with} \qquad a_i = c_{i\text{-}1} . \tag{4.8}$$

The straightforward variation operators for binary representation cannot be applied in case of real-valued representation, which forces, but also allows, the designer of the genetic algorithm to design operators that are more problem specific. Furthermore,

Table 4.4: Two coding strategies applied to binary representations: binary and Gray coding.

integer	0	1	2	3	4	5	6	7
binary	000	001	010	011	100	101	110	111
Gray	000	001	011	010	110	111	101	100

...

IF	X_1 IS LOW	AND	X_2 IS HIGH	THEN	Y IS a_3
IF	X_1 IS MEDIUM	AND	X_2 IS LOW	THEN	Y IS a_4
IF	X_1 IS MEDIUM	AND	X_2 IS MEDIUM	THEN	Y IS a_5

...

with $a_k \in \{$ NEGATIVE LARGE, NEGATIVE SMALL, ZERO, POSITIVE SMALL, POSITIVE LARGE $\}$

Figure 4.3: Identification of the rule consequents of a linguistic fuzzy model.

real-valued representation has the property that two points close to each other in the representation space are also close in the problem space, and vice versa. This is not generally true in the binary approach, where the distance in a representation is normally defined by the number of different bit positions. This discrepancy can however be omitted by using a binary representation with Gray coding. As illustrated in Table 4.4 for integer values in the interval [0,7] two points next to each other in the problem space differ by one bit only when using Gray coding. Procedures to convert binary coding into Gray coding and vice versa are described in Michalewicz (1996, p. 98).

Another shortcoming of binary representation is the problem of redundancy. When a binary alphabet is used to represent variables belonging to a discrete set with a cardinality different from a power of two, some codes may be redundant, *i.e.* their phenotypes correspond to values that do not belong to the discrete set. When for instance identifying the consequents of the rules of a Mamdani–Assilian model, where the consequents can contain one of the five linguistic values of the set $\{$ NEGATIVE LARGE, NEGATIVE SMALL, ZERO, POSITIVE SMALL, POSITIVE LARGE $\}$ as illustrated in Fig. 4.3, a binary string of two bits cannot represent all possible candidate solutions. When using a binary string of three bits however, three of the eight binary strings are redundant as they do not correspond to a linguistic value. When applying crossover or mutation to genotypes that correspond to a linguistic value, a redundant binary string can be obtained. One can either overcome this problem by replacing redundant binary strings by valid binary strings, assigning a very low fitness value to redundant binary strings or remapping the redundant binary strings to valid binary strings. A more straightforward solution however is the application of integer representation. Eiben and Smith (2003) give an overview of variation operators suited to evolve integer representations, as well as a fourth type of representation, the permutation representation.

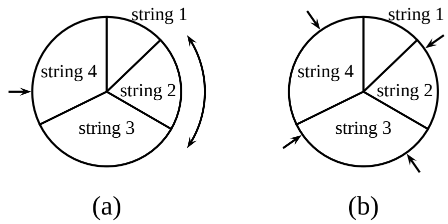


Figure 4.4: Fitness proportional and rank-based selection with (a) roulette wheel sampling or (b) stochastic universal sampling.

4.4 Parent selection

The role of parent selection or mating selection is to distinguish among individuals based on their fitness, in particular, to allow the better individuals to become parents of the next generation. Highly fit individuals get a higher chance to become parents than those with a low fitness. Nevertheless, also unfit individuals can be selected as parent in order to prevent the search of becoming too greedy and getting stuck in a local optimum. Three main selection schemes can be distinguished: fitness proportional (Holland, 1975), rank-based (Baker, 1985) and tournament selection (Blickle and Thiele, 1995).

Both fitness proportional and rank-based selection can be graphically represented by a biased roulette wheel on which each slot corresponds to an individual. The roulette wheel is biased as the slot size is proportional to the selection probability of the corresponding individual. The N parents are either obtained by spinning the wheel N times and including as many copies of an individual in the mating pool as the number of times the point pointed to the corresponding segment, or by spinning the wheel once and including as many copies of an individual in the mating pool as the number of the N equally spaced points pointing to the corresponding segment when the wheel comes to a halt. These two selection procedures are respectively referred to as roulette wheel sampling and stochastic universal sampling and are illustrated in Fig. 4.4. As the roulette wheel is called N independent times in roulette wheel sampling, this may result in a high variance in the number of copies made from each individual. Baker (1987) developed stochastic universal sampling to reduce the variance. In this case the number of copies of an individual is bounded by the integer floor and ceiling of the expected number of copies, which is the product of N and the selection probability of the individual (Note that the sum of the selection probabilities of all individuals of a population is equal to one).

In fitness proportional selection the selection probability is a function of the fitness. In the original procedure proposed by Holland (1975), the selection probability is given by the quotient of the fitness of the individual to the sum of the fitnesses of all individuals of the population. In this case, if one individual has a much better fitness that the others, that individual will tend to be selected much more often than the others. If this *superindividual* represents a local optimum, there will be a premature convergence for that suboptimal solution. If on the other hand, most individuals have about the same fitness, those individuals will have about the same probability of being selected, so that the selection will be almost random. These shortcomings can partially be compensated by scaling the fitness (Goldberg, 1989) or by applying rank-based selection (Baker, 1985). In rank-based selection the population is sorted on the basis of fitness, and the selection probabilities of the individuals are a (linear or exponential) function of their rank. Drawback of rank-based selection is that information about the magnitude of fitness differences between individuals is not taken into account.

In tournament selection (Blickle and Thiele, 1995) k individuals are randomly selected with replacement from the population and the best individual of this group is selected as parent. This procedure is repeated N times in order to obtain a mating pool of N parents. Usually tournaments are held between two individuals ($k = 2$) (Freitas, 2002). The larger the value of k, the more the selection procedure will be in favour of highly fit individuals. Tournament selection is perhaps the most widely used selection operator in modern applications of genetic algorithms, due to its simplicity of implementation and its time complexity of $\mathcal{O}(N)$ because no sorting of the population is required (Eiben and Smith, 2003).

4.5 Variation operators

Variation operators are applied to create new individuals from the parents in the mating pool, leading to exploration of new regions of the search space and exploitation of the knowledge available in the current population about the optimization problem. Two groups of variation operators can be distinguished: recombination (or crossover), merging information of two (or more) parent genotypes into one or two offspring genotypes and mutation, altering the genotype of a parent in a rather random way to create one child. In this section a selection is given of recombination and mutation procedures applied to binary and real-valued representations. The parent genotypes will be represented by $C_1 = (c_1^1, \ldots, c_{n_{\mathrm{bit}}}^1)$ and $C_2 = (c_1^2, \ldots, c_{n_{\mathrm{bit}}}^2)$ and the offspring genotypes by $H_1 = (h_1^1, \ldots, h_{n_{\mathrm{bit}}}^1)$ and $H_2 = (h_1^2, \ldots, h_{n_{\mathrm{bit}}}^2)$.

4.5.1 Recombination

Recombination operators are usually applied stochastically according to a crossover rate P_c. For each pair of parents, selected (without replacement) from the mating pool, a value is uniformly drawn from [0,1]. If the value is lower than P_c, two children are created via recombination of the two parents. Otherwise, two children are obtained by copying the parents. By many genetic algorithm theorists and practitioners recombination is considered the most important feature of genetic algorithms, whereas mutation is regarded as a background search operator. Regardless of the merits (or otherwise) of this viewpoint, recombination is certainly one of the features that most distinguishes genetic algorithms from other global optimization algorithms (Eiben and Smith, 2003).

Binary representation Three forms of recombination are generally applied to binary representations. One-point crossover, proposed by Holland (1975), is illustrated in the introductory example. In this case, the genotypes of the children H_1 and H_2, are obtained by

$$H_1 = (c_1^1, \ldots, c_i^1, c_{i+1}^2, \ldots, c_{n_{\text{bit}}}^2)\,, \tag{4.9}$$

$$H_2 = (c_1^2, \ldots, c_i^2, c_{i+1}^1, \ldots, c_{n_{\text{bit}}}^1)\,, \tag{4.10}$$

with i a random number from $\{1, \ldots, n_{\text{bit}} - 1\}$. A generalization of one-point crossover is n-point crossover (Spears and De Jong, 1991), where the parent strings are broken in more than two segments of contiguous genes and the offspring are created by taking alternative segments from the two parents. This means that n random numbers have to be selected from $\{1, \ldots, n_{\text{bit}} - 1\}$. For $n = 2$ the genotypes of the children are

$$H_1 = (c_1^1, \ldots, c_{i_1}^1, c_{i_1+1}^2, \ldots, c_{i_2}^2, c_{i_2+1}^1, \ldots, c_{n_{\text{bit}}}^1)\,, \tag{4.11}$$

$$H_2 = (c_1^2, \ldots, c_{i_1}^2, c_{i_1+1}^1, \ldots, c_{i_2}^1, c_{i_2+1}^2, \ldots, c_{n_{\text{bit}}}^2)\,, \tag{4.12}$$

with $i_1, i_2 \in \{1, \ldots, n_{\text{bit}} - 1\}$. Syswerda (1989) introduced uniform crossover. It is implemented by generating a mask, a (random) binary string of n_{bit} bits. The first offspring inherits the genes of the first parent in the positions where the mask contains a zero and the genes of the second parent in the positions where the mask contains a one. The second offspring is created using the inverse mapping. Given a mask M

$$M = (m_1, \ldots, m_{n_{\text{bit}}})\,, \tag{4.13}$$

the genotypes of the children $H_1 = (h_1^1, \ldots, h_{n_{\text{bit}}}^1)$ and $H_2 = (h_1^2, \ldots, h_{n_{\text{bit}}}^2)$, are obtained by

$$h_i^1 = \begin{cases} c_i^1 & \text{, if } m_i = 0 \\ c_i^2 & \text{, if } m_i = 1 \end{cases} \qquad h_i^2 = \begin{cases} c_i^2 & \text{, if } m_i = 0 \\ c_i^1 & \text{, if } m_i = 1 \end{cases} \quad . \tag{4.14}$$

As it tends to keep together genes that are located close to each other in the representation, n-point crossover (including one-point crossover) is said to suffer from positional bias. The third crossover, uniform crossover, does not exhibit any positional bias, but does have a strong tendency towards transmitting half of the genes of each parent. This is known as distributional bias. It is however impossible to state that one of these operators performs better than the others on any given problem (Eiben and Smith, 2003).

Real-valued representation Real-valued strings can be recombined using the same procedures as those described above for binary representations. The real counterparts of one-point crossover and uniform crossover are respectively called simple crossover (Wright, 1991; Michalewicz, 1996) and discrete crossover (Mühlenbein and Schlierkamp-Voosen, 1993). These recombination procedures however, do not lead to exploration of the search space in the neighbourhood of the parents, since the allele value for gene i is equal to the allele value of one of the parents, *i.e.* $h_i \in \{c_i^1, c_i^2\}$. In literature a wide range of recombination procedures is available where the allele values of the offspring lies between or within a certain distance from those of the parents. An extensive literature review is given by Rademaker (2004). Below only the two recombination procedures applied in this dissertation are described.

In arithmetic recombination (Michalewicz, 1996) the genotypes of the children $H_1 = (h_1^1, \ldots, h_{n_{\text{bit}}}^1)$ and $H_2 = (h_1^2, \ldots, h_{n_{\text{bit}}}^2)$, are obtained by

$$h_i^1 = \lambda c_i^1 + (1-\lambda) c_i^2 \,, \tag{4.15}$$

$$h_i^2 = (1-\lambda) c_i^1 + \lambda c_i^2 \,. \tag{4.16}$$

The parameter λ can be constant (uniform arithmetic recombination) or change as a function of the generation of the genetic algorithm (non-uniform arithmetic recombination).

Heuristic crossover, introduced by Wright (1991), is a unique crossover since it uses values of the objective function in determining the direction of the search and it produces only one offspring. The operator generates the genotype of a single offspring $H_1 = (h_1^1, \ldots, h_{n_{\text{bit}}}^1)$ according to the following rule

$$h_i^1 = c_i^1 + r(c_i^1 - c_i^2) \,, \tag{4.17}$$

with C_1 the genotype of the best performing parent and r a random value from [0,1].

4.5.2 Mutation

The most common mutation operator used in binary encodings, bit flip mutation (Goldberg, 1989), is illustrated in the introductory example. It considers each gene separately and allows each bit to change with a probability P_m, called the mutation probability. In its real counterpart, uniform mutation (Michalewicz, 1996), allele values h_i of the offspring are uniformly drawn from the interval $[c_{\min,i}, c_{\max,i}]$. Each allele value is replaced by a random value with a probability P_m.

Real-coded genetic algorithms also commonly apply non-uniform mutation with a fixed distribution (Eiben and Smith, 2003). This operator adds to the allele of all genes of a parent chromosome a value sampled from a distribution that is symmetric about zero, and is more likely to generate small changes that large ones. The distribution is for instance normal with mean zero and a user-specified standard deviation. If necessary, the obtained allele h_i of the offspring is curtailed to the interval $[c_{\min,i}, c_{\max,i}]$. It is normal practice to apply this operator with probability one per gene, and use the parameter P_m as standard deviation of the distribution instead.

4.6 Replacement procedure

Replacement can be regarded as the complementary operator to parent selection. It determines which individuals among the current population and the offspring will be included in the next generation. The simple genetic algorithm presented in the introductory example applies the most common replacement procedure, generational replacement, in which the entire population is replaced by the offspring. A steady-state GA operates on overlapping populations in which only a subset of the current population is replaced in each generation. In fitness-based replacement the individuals of the current generation and the offspring compete for a place in the next generation using one of the procedures mentioned earlier when discussing parent selection (Section 4.4). Finally, elitism can be applied in conjunction with any of the replacement procedures above in order to prevent the loss of the current fittest member of the population. Elitism is for instance obtained if the worst offspring is discarded and replaced by the best individual of the current population.

4.7 Parameter setting

Apart from choosing a representation of the candidate solutions, a parent selection procedure, variation operators and a replacement procedure, one also has to set the values of the various parameters: the population size, the crossover probability P_c and the mutation probability P_m. Furthermore a stopping criterion needs to be defined.

The values of the population size, the crossover probability P_c and the mutation probability P_m greatly determine whether the algorithm will find an optimal or near-optimal solution, and whether it will find such a solution efficiently (Eiben and Smith, 2003). If the population size is for instance too small, the genetic algorithm may converge to a local minimum because the diversity in the population is too low. On the other hand, if the population size is too large, the genetic algorithm may waste computational resources, which means that the waiting time for an improvement is too long. The crossover probability P_c is also a very important parameter and its influence on the results is similar to that of the population size. A higher crossover probability allows more exploration in the search space and reduces the chances of converging to a local minimum. On the other hand, a crossover probability which is too high, results in wastage of computation time in exploring unpromising regions of the search space. The mutation probability P_m on its turn controls the rate at which new genes are introduced into the population. If the mutation probability is too low, many genes that might be useful are never tried out. On the other hand, if the mutation probability is too high, there will be much random perturbation and the offspring will lose their resemblance to the parents. This means that the genetic algorithm will lose its ability to learn from the history of the search (Osyczka, 2002).

Unfortunately, even though genetic algorithms have quite a long history, few heuristics are available for determining the values of the parameters of genetic algorithms (Michalewicz, 1996). Several researchers (De Jong (1975), Grefenstette (1986) and Schaffer et al. (1989) in Michalewicz and Fogel (2000)) found parameters values that were good for a number of test problems, but as their recommendations are based solely on experimental evidence, their generalizability is limited. At the time when that research was carried out, genetic algorithms used to be seen as robust problem solvers that exhibit approximately the same performance over a wide range of problems (Goldberg, 1989, p. 6). The contemporary view on evolutionary algorithms, however, acknowledges that specific problems require specific evolutionary algorithm setups for satisfactory performance (Bäck et al. (1997) in Eiben and Smith (2003)). Thus, the scope of 'optimal' parameter settings is necessarily narrow. There are also theoretical arguments that any quest for generally good evolutionary algorithms, thus generally good parameter settings, is lost a priori (Wolpert and Macready, 1997). For real-world applications the parameter values are mostly sought through trial and error (Osyczka, 2002), a hard task which is considered more 'an art than a science' (Michalewicz, 1996). In most genetic algorithm applications, the population size stays between 50 and 100, the probability of crossover between 0.65 and 1.00 and the probability of mutation between 0.001 and 0.01.

Eiben and Smith (2003) remark that as the search carried out by an evolutionary algorithm is a dynamic, adaptive process, different values of parameters might be optimal at different stages of the evolutionary process. For instance, large mutation steps can be good in the early generations, helping the exploration of the search space, and small mutation steps might be needed in the late generation to help fine-tune the suboptimal chromosomes. Therefore, they argue that the fact that the values of the parameters of the genetic algorithm remain fixed during the whole search process, can itself be a cause of inferior algorithm performance. An overview of procedures to adapt the parameters of genetic algorithms during the evolutionary process is given in the textbooks by Michalewicz (1996, Section 4.4, dealing with the population size only), Michalewicz and Fogel (2000, Section 10.4) and Eiben and Smith (2003, Chapter 8).

Stopping at a predefined number of generations or function evaluations is a quite common stopping criteria and has the advantage that one knows how long it will take to achieve a solution. The genetic algorithm can also stop searching when there is no significant improvement of the fitness of the population. The search can either be terminated if the number of converged chromosomes in the population is greater than some predefined percentage of the population or if the improvement in the average or best fitness in the last t^* generations is smaller than an preset value (Osyczka, 2002).

A fifth but implicit parameter that can largely influence the behaviour of a genetic algorithm is the initial seed for the random population (Osyczka, 2002). Running any genetic algorithm with a different random starting seed might produce very different results and this should be kept in mind when making comparisons between algorithms. For real-world applications this means that repetitions of experiments are needed in order to remove the random effect.

Part II

Ecological case study

CHAPTER 5

Fuzzy ordered classification

Knowledge is indivisible. When people grow wise in one direction, they are sure to make it easier for themselves to grow wise in other directions as well. On the other hand, when they split up knowledge, concentrate on their own field, and scorn and ignore other fields, they grow less wise — even in their own field.
(*The Roving Mind*, Isaac Asimov, 1983)

5.1 Introduction

According to European Union (EU) standards and objectives, ecological water quality in EU Member States is still far from satisfactory, both in terms of nutrient management and habitat degradation (Chave, 2001). Within the last decade, the industrial pollution load has significantly decreased, but household and agricultural pollution still causes a high load of organic substances and nutrients (Hering et al., 2004). New requirements at the EU level, mainly covered by the Water Framework Directive (EU, 2000) in which Member States are hold to reach good ecological quality for their surface waters by 2015 (Chave, 2001), urge the Member States to extend their assessment methodologies to implement the desired river management. A methodology of interest in this context is the modelling of habitat suitability. Habitat suitability models describe which abiotic conditions are appropriate for a certain taxon or species to establish a population (Guisan and Zimmerman, 2000).

Ecological models that are meant to be used in river management can differ in biological endpoint. The choice of the endpoint can depend on the conservation value of a specific group of organisms as well as on the functionality as a biological indicator of river conditions. The biological endpoints for rivers as set by the Water Framework Directive (EU, 2000) include phytoplankton, phytobenthos and macrophytes, macroinvertebrates and fish. In this study benthic macroinvertebrates are considered. Benthic macroinvertebrates are invertebrate organisms that inhabit mainly bottom substrates of freshwater habitats (Rosenberg and Resh, 1993). The term ‘macro’ assumes that they are large enough to be seen without magnification and that they are retained in a net with mesh size of 500μm. Macroinvertebrate communities are made up of species that constitute a broad range of trophic levels and pollution tolerances. Furthermore, they show limited migration patterns and are therefore well suited for assessing site-specific impacts, they are abundant in most streams and they are easily sampled. Because of their central role in aquatic ecosystems, macroinvertebrates are widely used as indicators for assessing the quality of freshwater (De Pauw and Vanhooren, 1983; Wiederholm, 1980; Sládecek et al., 1982; Metcalfe, 1989; Rosenberg and Resh, 1993).

The development of habitat suitability models is not an easy task. When developing ecological models to support decisions in river management, one should compromise between the policy relevance of the variables, the ecological processes incorporated in the model and the accuracy of the model. Furthermore, the available knowledge is usually only verbally described, with terminology and meaning differing from source to source. On the other hand, data available are not only scarce, but insufficiently representative for all river conditions, and can therefore play at most a role in model optimization, but not in model identification (Casillas et al., 2003a,b). Taking into account these limitations and the ultimate use of these models in decision support, requiring understandability to the end user (Ehrlich and Daily, 1993; Ludwig et al., 1993; Parsons and Norris, 1996; Omlin and Reichert, 1999; Elith et al., 2002; Holling and Allen, 2002; Regan et al., 2002; Borsuk, 2003; Poff and Allan, 1995), it was opted for linguistic fuzzy models and a knowledge-based design approach, described in this chapter, followed by an interpretability-preserving data-driven optimization of the membership functions, discussed in Chapter 6.

The models developed in this study describe the habitat suitability for macroinvertebrates in springs up to small rivers in the eco-region of the Central and Western Plains of Europe (Illies, 1978). As will be explained further on, this modelling problem asks for a model that gives a shaded indication of a certain river site's suitability as habitat for a certain macroinvertebrate species. Therefore, fuzzy classifiers were applied, instead of classical models with crisp outputs or crisp classifiers. A more detailed description of the habitat suitability models, built using expert knowledge described in literature, is given in Section 5.2. In Section 5.3, the data collected in the Province of Overijssel in the Netherlands (Verdonschot, 1990) on which the models were evaluated, referred to in this work as the EKOO data set, are discussed. The measures used to evaluate the models, percentage of correctly classified instances (% CCI) and the percentage of correctly fuzzy classified instances (% CFCI) as well as the results of the model evaluation are presented in Section 5.4. The chapter concludes with some remarks on the use of knowledge-based model identification and fuzzy modelling for habitat suitability modelling in Section 5.5.

5.2 Habitat suitability models

5.2.1 Knowledge base

The knowledge base, used during the model design process, is described in detail in Adriaenssens (2004). It summarizes observations of several ecological studies (Mauch, 1976; Moller Pillot and Buskens, 1990; Verdonschot, 1990; Usseglio-Polatera, 1994; De Loose et al., 1995; Bayerisches Landesamt für Wasserwirtschaft, 1996; RIZA, 2000; Verdonschot, 2000a,b; Tachet et al., 2000) regarding univariate preferences as well as tolerances of 86 macroinvertebrate species for a limited set of environmental variables. In Appendix A the names of the 86 macroinvertebrate species are listed. Among them, 30 species are regarded as characteristic for the reference conditions of the river types included in this study (high ecological quality; based on (Verdonschot, 2000a,b)) and will be referred to as indicator species. The other 56 species are so-called common species, observed at river sites of diverse ecological quality. The information in the knowledge base applies to springs up to small rivers within the limnological eco-regions Central and Western Plains of Europe as defined by Illies (1978). The ecological variables addressed in the knowledge base are: river dimension (stream width), stream velocity, saprobic conditions, habitat *sensu stricto* and habitat diversity.

5.2.2 Input and output variables

As discussed in detail by Adriaenssens (2004), the selected input variables should be of high ecological importance to the macroinvertebrate species under study as well as to the whole macroinvertebrate community and should be of importance to river management. Furthermore, knowledge about their preferences for certain environmental conditions needs to be available and the variables need to be included in the EKOO data set. Physical variables do provide effective assessment criteria when rivers are not affected by physical-chemical degradation (Karr et al., 1986). However, in the Central and Western Plains of Europe, the main threats for biological communities in rivers are the deteriorated physical-chemical water quality conditions. This is mainly due to increased nutrient and organic loading mainly caused by agricultural activities and pollution originating from households.

Therefore, apart from stream width and stream velocity, two variables determining the river type and reflecting the water quantity conditions, an additional input variable is used, expressing the physical-chemical conditions at a river site. The knowledge base contains preferences and tolerances for the saprobic status at a river site, which can be represented by the ammonium concentration. During the model design process, the information in the knowledge base concerning the preferences and tolerances for the saprobic condition is interpreted in a more general way. For each macroinvertebrate species, four different models were built: apart from an A-model including the stream width, stream velocity and ammonium concentration as input variables, also an N- and a P-model were constructed including respectively nitrate and phosphate concentration (trophic status) and a C-model in which electrical conductivity (ionic status) was selected as third input variable. This allowed us to evaluate to which extent knowledge concerning the saprobic status of the water column is also valid for its trophic and ionic status. The occurrence of some of the 86 considered macroinvertebrate species is independent of the stream width. In these models stream width is not included and only two input variables are used.

Due to the different context of the studies described in the eight publications used as a source of expert knowledge, meanings given to the used linguistic terms are not identical in all eight publications. However, in all considered studies, a similar number of linguistic values is assigned to variables as stream width, stream velocity and nutrient and organic loading and in most cases similar expressions are applied to refer to the different situations distinguished. The linguistic values assigned to the variables in the developed models are listed in Table 5.1. The number of linguistic values distinguished for a certain variable ranges from three to five. All values are defined by membership functions forming a fuzzy partition, as illustrated in Fig. 5.1(a) for the five linguistic values for ammonium concentration (in order of increasing organic load): $oligosaprobic$, β, α-$oligosaprobic$, β-$mesosaprobic$, α-$mesosaprobic$ and $polysaprobic$ conditions. All membership functions are of the trapezial type, characterized by four parameters (a_1, a_2, a_3, a_4): the membership degree linearly increases from 0 to 1 for values between a_1 and a_2, remains constant for values between a_2 and a_3 and linearly decreases from 1 to 0 for values between a_3 and a_4. A triangular membership function is obtained if a_2 is equal to a_3. The values of the membership function parameters of all variables, given in Table 5.1, are based on crisp boundaries found in literature. The kernel of each of the membership functions is the intersection of the crisp intervals used in the different literature sources to define the corresponding linguistic term. As fuzzy partitions were opted for, the supports of the membership functions are determined by the kernels of the membership functions of the adjacent linguistic values and the lower and upper bounds of the underlying domain.

A site's suitability as a habitat for macroinvertebrates cannot be measured directly. As output variable of the developed habitat suitability models, the abundance of a macroinvertebrate species at a river site is used. The abundance is a measure for habitat suitability: the higher the abundance of a species, the higher the site's suitability as a habitat. Furthermore the EKOO data set contains the number of sampled individuals of the 86 species considered at all investigated river sites. In the developed models four linguistic values were assigned to the variable: $absent$, low, $moderate$ and $high$. They are defined by the membership functions shown in Fig. 5.1(b) with the help of the same experts assigning the membership functions of the input variables. In order to take into account the non-linear response of macroinvertebrate species to environmental conditions (Statzner et al., 1988), the abundance values were log-transformed. When comparing abundance values, relative differences rather than absolute differences should be considered, since the difference between 1 and 2 individuals found at a river site is more significant than the difference between 101 and 102 recorded individuals. We also want to stress that these abundance values are not equal to the exact number of individuals present at a site, but are proportional to the number of individuals present at a site (see the sampling procedures in Section 5.3).

Table 5.1: Linguistic values assigned to the input and output variables of the habitat suitability models. The values between brackets characterize the corresponding trapezial membership functions.

	stream width (m)		stream velocity (m/s)
1	spring / small stream (0,0,0,2)	1	low (0,0,0,0.25)
2	upper course stream (0,2,2,4)	2	moderate (0,0.25,0.25,0.5)
3	middle course stream (2,4,4,6)	3	high (0.25,0.5,1.2,1.2)
4	lower course stream / small river (4,6,201,201)		
	ammonium concentration (mg NH_4^+-N/L)		nitrate concentration (mg NO_3^--N/L)
1	oligosaprobic (0,0,0,0.1)	1	oligotrophic (0,0,0,0.15)
2	β,α-oligosaprobic (0,0.1,0.1,0.15)	2	β-mesotrophic (0,0.15,0.15,0.30)
3	β-mesosaprobic (0,0.15,0.15,4.5)	3	α-mesotrophic (0.15,0.3,0.3,0.4)
4	α-mesosaprobic (4,5,8,10)	4	eutrophic (0.30,0.4,0.4,0.45)
5	polysaprobic (8,10,30,30)	5	hypertrophic (0.40,0.45,112,112)
	phosphate concentration (mg PO_4^{3-}-P/L)		conductivity (μS/cm)
1	oligotrophic (0,0,0,0.0080)	1	oligoionic (0,0,150,250)
2	β-mesotrophic (0,0.0080,0.0080,0.0150)	2	β-mesoionic (150,250,450,550)
3	α-mesotrophic (0.0080,0.0150,0.0150,0.0250)	3	mesoionic (450,550,750,850)
4	eutrophic (0.0150,0.0250,0.0250,0.0450)	4	α-mesoionic (750,850,1050,1150)
5	hypertrophic (0.0250,0.0450,5.45,5.45)	5	polyionic (1050,1150,2880,2880)
	$\log_{10}$(abundance + 1) (-)		corresponding abundance (-)
1	absent (0,0,0,0.477121)	1	absent (0,0,0,2)
2	low (0,0.477121,0.477121,0.778151)	2	low (0,2,2,5)
3	moderate (0.477121,0.778151,1.041393, 1.322219)	3	moderate (2,5,10,20)
4	high (1.041393,1.322219,3.602169, 3.602169)	4	high (10,20,4000,4000)

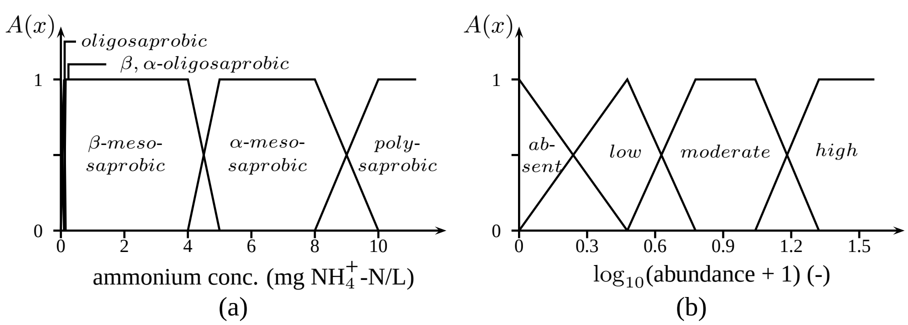


Figure 5.1: Definition of the five linguistic values assigned to ammonium concentration and the four fuzzy abundance classes through membership functions.

5.2.3 Rule bases

Based on the knowledge base, rule bases were built describing the preferences of macroinvertebrates with regard to the environmental variables stream width and stream velocity in combination with the saprobic status represented by the ammonium concentration. The rule bases of the 86 species can be consulted in Appendix B. As the preferences for habitat structure and habitat diversity are far too complex to be represented in such a compact way, these variables were not included in the rule bases. The four linguistic values of stream width, the three linguistic values of stream velocity and the five linguistic values of the variables describing the nutrient and organic concentration, define 60 environmental situations. The following procedure was followed during the rule base development, *i.e.* the assignment of a linguistic abundance value to this 60 environmental situations. First of all, a two-dimensional rule base with stream width and stream velocity as input variables and abundance as output variable was constructed, based on the univariate preferences for stream width and stream velocity. The development of the three-dimensional rule base was initiated by assigning the corresponding abundance values for the 12 combinations of stream width and stream velocity in the two-dimensional rule base to all situations with an optimal saprobic condition according to the univariate preference in the knowledge base. In a next step, the rule base was completed for situations with suboptimal conditions being less saprobic than the optimal saprobic condition(s). For all combinations of stream width and stream velocity, a lower abundance value than the corresponding abundance value in the two-dimensional rule base was assigned, the difference between both classes being equal to the difference between the univariate preference for the saprobic condition under consideration and the abundance value 'high'. Note that the abundance value 'absent' is the smallest linguistic abundance value and can therefore not be further decreased. Finally, abundance values were assigned in case of sub-optimal saprobic conditions being more saprobic than the optimal saprobic conditions. For low and moderate stream velocities, the same procedure was followed as for less saprobic conditions. For fast running waters the abundance values were lowered less fast as a function of the univariate preference for the saprobic condition, reflecting the lower effect on the water chemistry and the related lower uptake of toxic substances due to the lower residence time of organic components in the water.

The rule base development is illustrated by means of the rule bases of the crustacean *Proasellus meridianus* and the mollusc *Stagnicola palustris* (Fig. 5.2). The univariate preference of *Proasellus meridianus* concerning the saprobic status of its habitat varies from low, low, moderate, high and low for oligosaprobic to polysaprobic conditions (Adriaenssens, 2004). As such, the most optimal condition for this species is α-mesosaprobic. In this situation, *Proasellus meridianus* will have its optimal distribution that is completely determined by (stream width and) stream velocity. At the other saprobic levels, this macroinvertebrate will have a diminished distribution based on its univariate preference for the saprobic status. For example, in oligosaprobic conditions, the univariate preference of *Proasellus meridianus* is low. As such, the resulting abundance level will be two classes lower than the optimal level, and only river sites with moderate stream velocities will have a low abundance level. The methodology of rule

(a)

Proasellus meridianus — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Low
moderate	Moderate	Moderate	Moderate	Moderate
high	Low	Low	Low	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	Moderate	Moderate	Moderate	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Low	Low	Low	Low

(b)

Stagnicola palustris — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Low	Moderate	Moderate
moderate	Low	Moderate	High	High
high	Absent	Low	Moderate	Moderate
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Low	Moderate	Moderate
moderate	Low	Moderate	High	High
high	Absent	Low	Moderate	Moderate
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Low	Low
moderate	Absent	Low	Moderate	Moderate
high	Absent	Low	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Low
high	Absent	Absent	Low	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Figure 5.2: Rule base of the four models describing the habitat suitability for *Proasellus meridianus* and *Stagnicola palustris*.

base development is applied in a similar way for higher saprobic levels, although tolerance and subsequent abundance are lowered more gradually for fast running rivers, because of the smaller chance of negative effects at high nutrient or organic loading and high current velocity. This can be seen in the rule base of *Stagnicola palustris* (Fig. 5.2(b)) which univariate preference concerning the saprobic status of its habitat varies from high, high, moderate, low and absent for oligosaprobic to polysaprobic conditions. At mesosaprobic conditions, for which *Stagnicola palustris* has a moderate univariate preference, lower abundance values are assigned than at β,α-oligosaprobic conditions in case of low and moderate stream velocities, whereas equal abundances are used for high velocities for both saprobic levels. The abundances at high velocities are however not maintained for all sub-optimal, more saprobic conditions. In the saprobic level (α-mesosaprobic in case of *Stagnicola palustris*) preceding a saprobic level for which the univariate preference of the species is absent (polysaprobic in case of *Stagnicola palustris*), the abundances at high velocities are lowered with one class compared to the abundances assigned in the preceding saprobic condition (mesosaprobic in case of *Stagnicola palustris*).

5.2.4 Fuzzy classifiers

In the A-, N-, P- and C-models of the 86 macroinvertebrate species, including respectively ammonium concentration, nitrate concentration, phosphate concentration and electrical conductivity as input variables, the same membership functions are used. The rule bases of the models of the different species differ, but are identical for the four models of a certain species (Appendix B) as the information in the knowledge base concerning the preferences and tolerances for the saprobic condition are interpreted in a more general way and extended towards trophic and ionic conditions. All constructed rule bases are complete, *i.e.* each rule base contains a rule for each combination of linguistic values of the m input variables. The 60 rules are of the following type

$$
\begin{array}{lll}
\text{IF} & & \mathit{width}\ \text{IS}\ \mathit{upper\ course\ stream} \\
 & \text{AND} & \mathit{velocity}\ \text{IS}\ \mathit{low} \\
 & \text{AND} & \mathit{nitrate\ concentration}\ \text{IS}\ \mathit{eutrophic} \\
\text{THEN} & & \mathit{abundance}\ \text{IS}\ \mathit{moderate}
\end{array}
$$

The if-part of the rule (the antecedent) describes in which situations the then-part of the rule (the consequent) holds.

The rule bases show that the abundance of some of the considered macroinvertebrate species is independent of the stream width or identical for two consecutive linguistic values of an input variable. The rule bases, modelling the abundance of these species, were simplified by removing redundant input variables or redundant linguistic values, as these would slightly distort the model output.

The occurrence of *Proasellus meridianus*, for instance, is independent of stream width, as one can see from the rule base in Fig. 5.2(a). Furthermore, according to the rules derived from the eight consulted knowledge sources, its abundance is the same in oligosaprobic (resp. oligotrophic and oligoionic) conditions as in β,α-oligosaprobic

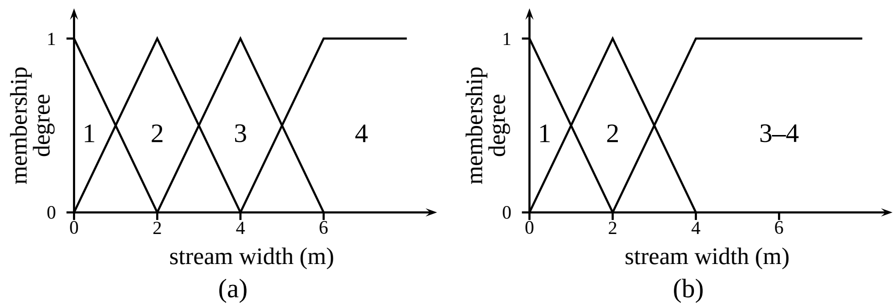


Figure 5.3: Membership functions of (a) the original stream width values and (b) the three width values used in the models of *Stagnicola palustris* obtained by combining the third and fourth original linguistic value.

(resp. β-mesotrophic and β-mesoionic) conditions. If two consecutive linguistic values of a variable yield the same model output for all combinations of linguistic values of the other input variables, then the corresponding rules are merged and a new linguistic value is introduced defined as the convex hull of the membership functions of the original linguistic values. Therefore, in the reduced model the variables ammonium, nitrate and phosphate concentration and conductivity, take four values instead of five, for ammonium concentration these linguistic values are 'oligosaprobic to β,α-oligosaprobic, 'β-mesosaprobic', 'α-mesosaprobic' and 'polysaprobic' conditions. The linguistic value 'oligosaprobic to β,α-oligosaprobic' conditions is defined as the convex hull of the membership function of 'oligosaprobic' conditions and the membership function of 'β,α-oligosaprobic' conditions.

The creation of new linguistic values and their corresponding membership functions is illustrated in Fig. 5.3 for the variable stream width in the models of *Stagnicola palustris* (Fig. 5.2(b)). For this species, the same abundance values are assigned to middle course and lower course streams and small rivers for all combinations of stream velocity and saprobic (respectively trophic and ionic) status. The two linguistic values 'middle course stream' and 'lower course stream / small river' were therefore replaced by one linguistic value 'middle course stream to small river' defined by the convex hull of the fuzzy sets describing the two original linguistic values.

As a result of the reduction of input variables and linguistic values, the number of rules in the rule base decreases. The rule base of the resulting, fully reduced model for *Proasellus meridianus* is shown in Fig. 5.4. This model reduction procedure is carried out for the models of all 86 species, resulting in models with different numbers of input variables, membership functions and number of rules.

Given the available qualitative expert knowledge and uncertainty in the definitions of the used linguistic expressions, linguistic fuzzy models are the most appropriate model types for the modelling problem. Given crisp input values x_w, x_v and x_a for the three input variables width, velocity and for instance ammonium concentration in case of an A-model, the fuzzy model output is obtained by the following procedure. In

Proasellus meridianus		stream velocity		
		low	moderate	high
saprobic / trophic / ionic status	oligosaprobic to β,α-oligosaprobic / oligotrophic to β-mesotrophic / oligoionic to β-mesoionic	Absent	Low	Absent
	mesosaprobic / α-mesotrophic / mesoionic	Low	Moderate	Low
	α-mesosaprobic / eutrophic / α-mesoionic	Moderate	High	Moderate
	polysaprobic / hypertrophic / polyionic	Absent	Low	Low

Figure 5.4: Reduced rule base of the four models describing the habitat suitability for *Proasellus meridianus*.

a first step, the membership degrees of the input values to the linguistic values of the input variables are determined. The membership degrees of x_w = 2.4 m, x_v = 0.25 m/s and x_a = 4.7 NH_4^+-N mg/l to the linguistic values in the antecedents of the rules of the A-model of *Gammarus pulex* (model index = 42) are

width:	spring/small stream to upper course stream	$A_{\mathrm{width},1}(x_w)$ = 0.8,
	middle course stream	$A_{\mathrm{width},2}(x_w)$ = 0.2,
	lower course stream/small river	$A_{\mathrm{width},3}(x_w)$ = 0,
velocity:	low	$A_{\mathrm{velocity},1}(x_v)$ = 0,
	moderate	$A_{\mathrm{velocity},2}(x_v)$ = 1,
	high	$A_{\mathrm{velocity},3}(x_v)$ = 0,
ammonium conc.:	oligosaprobic	$A_{\mathrm{ammon},1}(x_a)$ = 0,
	β,α-oligosaprobic to β-mesosaprobic	$A_{\mathrm{ammon},2}(x_a)$ = 0.3,
	α-mesosaprobic	$A_{\mathrm{ammon},3}(x_a)$ = 0.7 and
	polysaprobic	$A_{\mathrm{ammon},4}(x_a)$ = 0.

Next, the degree of fulfilment is calculated for each rule as the minimum of the fulfilment degrees in its antecedent. For the example above, the following four rules have a non-zero fulfilment degree

IF w = $A_{\mathrm{width},1}$ AND v = $A_{\mathrm{velocity},2}$ AND a = $A_{\mathrm{ammon},2}$ THEN abundance = moderate
(0.3 = min(0.8,1,0.3))
IF w = $A_{\mathrm{width},1}$ AND v = $A_{\mathrm{velocity},2}$ AND a = $A_{\mathrm{ammon},3}$ THEN abundance = absent
(0.7 = min(0.8,1,0.7))
IF w = $A_{\mathrm{width},2}$ AND v = $A_{\mathrm{velocity},2}$ AND a = $A_{\mathrm{ammon},2}$ THEN abundance = low
(0.2 = min(0.2,1,0.3))
IF w = $A_{\mathrm{width},2}$ AND v = $A_{\mathrm{velocity},2}$ AND a = $A_{\mathrm{ammon},3}$ THEN abundance = absent
(0.2 = min(0.2,1,0.7))

Finally, to each linguistic output value a fulfilment degree is assigned given by the maximum fulfilment degree obtained for all rules containing the linguistic output value under consideration in their consequent. In the given example the following fulfilment degrees are obtained: 0.7 for absent, 0.2 for low, 0.3 for moderate and 0 for high.

Up to this point, the procedure is the same as the one applied in Mamdani–Assilian models (see Section 2.3.2). In Mamdani–Assilian models the procedure continues by adapting the membership functions of the output variable according to the corresponding fulfilment degree, constructing the union of all adapted membership functions and deriving the model output, a crisp value, by defuzzifying this union, for instance, by computing its center of gravity. It is, however, not the purpose of a habitat suitability model to predict a precise numerical value for the occurrence of a given species. No ecologist is interested in or would even trust a model stating an occurrence of, *e.g.* 37 individuals. It is rather the magnitude of the abundance which is of interest. Therefore, a different kind of fuzzy model was applied: a fuzzy classifier. The model output of the developed models is fuzzy. The model output $\mathbf{y}_{\text{model}}$ is a set of four values between zero and one and summing up to one, [($absent$, $A_1(\mathbf{y}_{\text{model}})$), ($low$, $A_2(\mathbf{y}_{\text{model}})$), ($moderate$, $A_3(\mathbf{y}_{\text{model}})$), ($high$, $A_4(\mathbf{y}_{\text{model}})$)], expressing the degree to which the considered river site is respectively regarded not (abundance value 'absent'), lowly, moderately or highly suitable as a habitat for the species. The output is obtained by normalizing the fulfilment degrees of the abundance (output) classes, which results in $\mathbf{y}_{\text{model}}$ = [7/12,1/6,1/4,0] for the numerical example. Note that the abundance values included in the validation data set are crisp values (integers). When comparing the fuzzy model outputs with the information in the validation data set, the membership degrees of the crisp abundance values to the four linguistic abundance values are used (Table 5.1).

5.3 EKOO data set

5.3.1 Data collection

The data used in this study to evaluate and optimize the habitat suitability models were collected in running waters in the Province of Overijssel in the Netherlands. They are part of a larger data set described by Verdonschot (1990), which apart from the 445 data points collected along running waters and used in this study, also includes data collected in pools and lakes, canals and large standing waters.

The sampling dates were spread over the four seasons as well as over several years (from 1981 to 1985). The objective was to capture the majority of species present at a given site, and assess their relative abundances. At each site, 70 abiotic variables were measured, as stream width, depth, temperature, transparency of the water column, bank shape, substratum, dissolved oxygen concentration, pH, nitrate concentration and phosphate concentration, and samples were taken of the major habitats, the water body and the bottom habitat to collect macroinvertebrates. In shallow sites, habitats with vegetation were sampled by sweeping a hand net (20 $\times$ 30 cm, mesh size 500 μm) several times over a length of 0.5 to 1 m through each vegetation type. Bottom habitats were sampled by vigorously pushing the hand net through the upper few centimeters of each type of substratum over a length of 0.5 to 1 m. All habitat samples collected at a site were combined in a single sample with a standard area of 1.5 m^2 (1.2 m^2 of vegetation and 0.3 m^2 of bottom). At sites lacking vegetation, the standard sampling was confined to the bottom habitats. In deeper sites, five samples from the bottom habitats were taken with an Ekman-Birge sampler. These five grab-samples were equivalent to one hand net bottom sample. Habitats with vegetation were sampled with a hand net as described above. Again the total sampling area was standardized to 1.5 m^2. Macroinvertebrate samples were taken to the laboratory, sorted by eye, counted and identified to species level, except for chironomids.

In this work the term 'EKOO data set' (Ecologische Karakterisering van Oppervlaktewateren in Overijssel, *ecological characterisation of surface waters in Overijssel*) does not refer to the complete data set described by Verdonschot (1990), but only to those data used in this study: the values of the six abiotic variables, stream width, stream velocity, ammonium concentration, nitrate concentration, phosphate concentration and electrical conductivity, and the number of sampled individuals of the 86 macroinvertebrate species listed in Appendix A at 445 sites along running waters.

5.3.2 Data distribution over input and output space

When applying a model to data, one should examine the data set in order to be able to interpret the scores obtained by performance measures (Fielding and Bell, 1997; Boone and Krohn, 1999; Cowley et al., 2000; Manel et al., 2001). When a data set contains examples of all situations covered by the model, the whole model will be assessed. If, however, some if-then rules in the rule base of a model apply to none of the examples in the data set, one cannot draw any conclusion about the correctness of these rules. Therefore, the distribution of the validation data set over the input and output space has to be taken into account.

As a measure for the uniformity of this distribution, the Shannon entropy measure was used (Shannon and Weaver, 1963). The fuzzy sets were converted into crisp ones to calculate this measure. The boundaries of these crisp sets are the points having membership degree 0.5 to the corresponding adjacent fuzzy sets. The entropy is given by (convention $0 \cdot \log_2 0 = 0$)

$$\text{entropy} = -\frac{1}{\log_2 n} \sum_{i=1}^{n} p_i \cdot \log_2 p_i \,, \tag{5.1}$$

where n is the number of classes, N is the number of data points and p_i is the proportion of data points belonging to class i. The entropy is 1 for a uniform distribution and 0 if all data points are assigned to the same abundance class as is the case for *Odontomesa fulva*. Note that entropy is a non-linear concept. In Table 5.2 entropy values for some species are given. When a distribution is highly non-uniform, as for *Agabus affinis*, the shift of 1 data point from the most frequent class to a less frequent class results in an entropy increase of at least 0.009. Given a more uniform initial distribution, a larger shift towards a more uniform distribution, gives a smaller entropy increase, for instance

Table 5.2: Distributions of data points over four crisp abundance classes and the corresponding entropy.

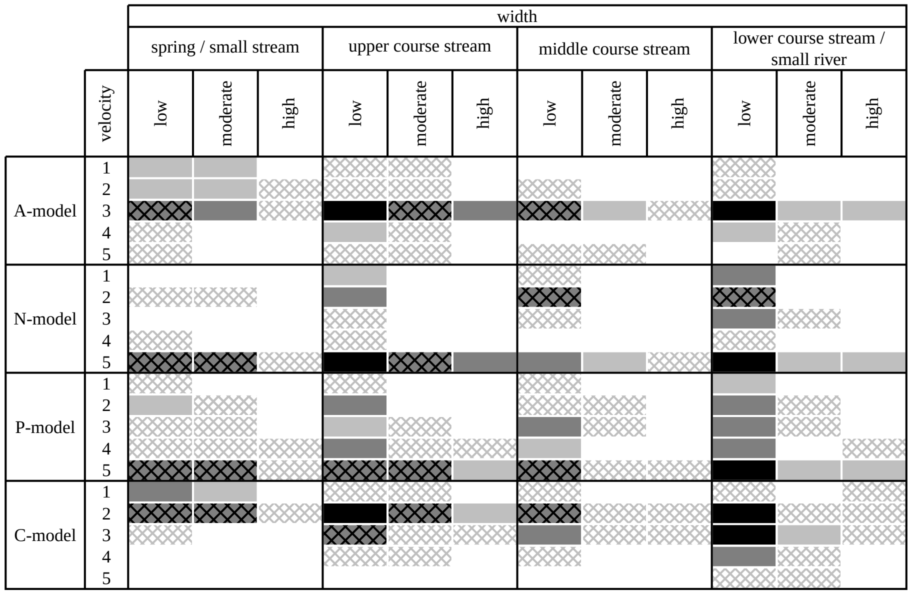
	number of data points classified as				
Species	$absent$	low	$moderate$	$high$	entropy
Odontomesa fulva	445	0	0	0	0.000
Agabus affinis	444	1	0	0	0.012
Elmis aenea	443	2	0	0	0.021
Plectronemia conspersa	399	15	15	16	0.322
Proasellus meridianus	247	78	80	40	0.835
Erpobdella octoculata	237	106	64	38	0.841

an entropy increase with 0.006 for *Erpobdella octoculata* compared to the entropy for *Proasellus meridianus*.

The EKOO data set is characterized by a highly non-uniform distribution of the data points in the input space. As the same input values and the same membership functions are used in respectively all A-, N-, P- and C-models, the distribution of the input values over the different regions of the input space is the same for all models of a given type. By replacing the fuzzy sets describing the linguistic values of the input variables by crisp sets, each data point can be assigned to one environmental condition. The crisp sets are bounded by the points having membership degree 0.5 to the corresponding fuzzy sets. The distribution of the data points over the 60 'crisp' environmental conditions considered by the habitat suitability models is given in Fig. 5.5 for the four model types. This table gives an indication of the usefulness of the data set for the validation of the developed habitat suitability models over a range of environmental conditions that can be found in the Province of Overijssel.

When only considering 'stream width', the distribution of the sites is relatively balanced over the four linguistic values used in the habitat suitability models. For 'stream velocity', fast running streams are underrepresented in the data set. For the saprobic status characterised by the ammonium concentration (A-model), most sites are classified into the β-mesosaprobic class, although other saprobic classes are also present at the sampling sites. For nitrate and phosphate concentration (N- and P-models), dominance of polytrophic conditions is obvious. When focussing on conductivity (C-model), most of the sites are at β-mesoionic conditions.

The sampling sites included in the EKOO data set were chosen in such a way that a rather uniform geographical distribution was obtained, while trying to include a similar number of examples of the different environmental conditions present in the region. In other words, during the selection of the sampling sites, a maximal input entropy was strived for. However, due to the fact that the exact conditions at the sites are unknown before the sampling, that some environmental conditions described by the model are underrepresented in the considered region due to human impact (*e.g.* reference conditions) and that the four model types include different input variables, no perfectly uniform distribution among the 60 environmental situations described by the models is obtained. As shown in Table 5.3, input entropies ranging between 0.63 and



Legend: 0 sites ( ), 1 to 5 sites ( ), 6 to 10 sites ( ), 11 to 20 sites ( ), 21 to 60 sites ( ) and more than 50 sites ( ).

Figure 5.5: Distribution of the sites included in the EKOO data set over the 60 environmental situations considered in the habitat suitability models. The linguistic values assigned to the variables ‘ammonium concentration’, ‘nitrate concentration’, ‘phosphate concentration’ and ‘electrical conductivity’ corresponding to the numbers 1 to 5 are given in Table 5.1.

Table 5.3: Entropy of the distribution of the data points of the EKOO data set over the input space over the crisp classes derived from the fuzzy sets of the non-simplified A-, N-, P- and C-models.

A-model	N-model	P-model	C-model
0.6311	0.6622	0.7189	0.7044

0.72 were obtained for the four model types. As the sampling sites were selected with great care, distributions with entropy values larger than or equal to the ones obtained for the distributions of the data points over the input space will be regarded as sufficiently uniform to allow for an objective validation. Therefore an entropy threshold of 0.7 was adopted in this study to distinguish not-sufficiently uniform from sufficiently uniform distributions.

In Fig. 5.6, the entropy of the distribution of the data points over the crisp abundance values as well as the mean presence of the species at the sampling sites is plotted.

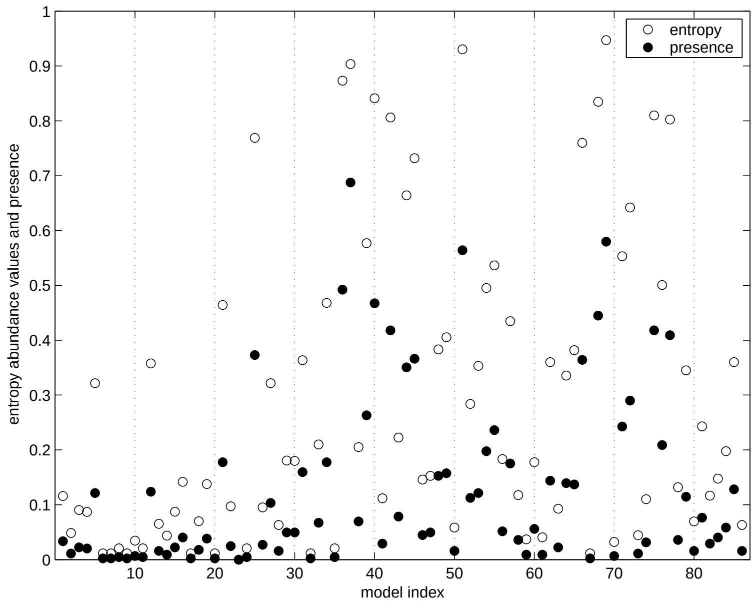


Figure 5.6: Entropy of the distribution of the abundance values in the EKOO data set over the four crisp abundance classes, as well as the mean presence for the 86 species. The names corresponding to the model index in the horizontal axis are given in Appendix A.

Mean presence, *i.e.* the relative number of data points with a non-zero abundance, is expressed on a scale from 0 to 1, the extreme values indicating respectively 'absent from all sites' (0) and 'present at all sites' (1). This means that for N data points, presence can be formulated as follows

$$\text{presence} = \frac{1}{N}\sum_{i=1}^{N}\min(\text{abundance}_i, 1)\,, \tag{5.2}$$

where abundance$_i$ is the number of individuals collected at site i. As shown by the low abundance entropy values in Fig. 5.6 the observed abundances of the species are distributed in a highly non-uniform way over the different abundance classes. Moreover, the indicator species considered are absent from a large number of sites, as indicated by their low presence value. The entropy and presence values for the majority of the indicator species indicate that the correctness of the models was hardly tested for sites at which the species occur quite abundantly. On the other hand, for a significant part of the non-indicator species a more uniform distribution over the four abundance values is recorded. Moreover, these species are observed at more sites in comparison to the indicator species, making a more relevant evaluation of these models possible.

Only the models of the 12 species for which the abundance entropy, *i.e.* the en-

Table 5.4: Distributions of 445 abundances over four abundances classes for the 12 species, for which an abundance entropy larger than 0.7 was obtained.

model index	species name	number of data points classified as absent	low	moderate	high	entropy
25	*Physa fontinalis*	279	67	47	52	0.77
36	*Anisus vortex*	226	53	61	105	0.87
37	*Asellus aquaticus*	139	49	66	191	0.90
40	*Erpobdella octoculata*	237	106	64	38	0.84
42	*Gammarus pulex*	259	53	43	90	0.81
45	*Glossiphonia heteroclita*	282	85	56	22	0.73
51	*Helobdella stagnalis*	194	108	82	61	0.93
66	*Planorbis planorbis*	283	62	52	48	0.76
68	*Proasellus meridianus*	247	78	80	40	0.83
69	*Radix peregra*	187	101	82	75	0.95
75	*Sigara striata*	259	84	60	42	0.81
77	*Valvata piscinalis*	263	82	56	44	0.80

tropy of the distribution of the abundances over the four abundance classes, is larger than 0.7 were considered to be evaluated in an objective way by the EKOO data set: one indicator species and 11 non-indicator species. The selected species are: *Physa fontinalis*, *Anisus vortex*, *Asellus aquaticus*, *Erpobdella octoculata*, *Gammarus pulex*, *Glossiphonia heteroclita*, *Helobdella stagnalis*, *Planorbis planorbis*, *Proasellus meridianus*, *Radix peregra*, *Sigara striata* and *Valvate piscinalis*. For these 12 species, the distributions of the abundance values over the four abundance classes are given in Table 5.4. Moreover in Fig. 5.7 for one of the 12 selected species, namely for *Proasellus meridianus*, the distribution of the data belonging to the four crisp abundance classes over the input space of the corresponding A-model is given. The reduced habitat suitability models for *Proasellus meridianus* have only two input variables, stream velocity and ammonium concentration in case of the A-model, and the number of linguistic values assigned to ammonium concentration is reduced from five to four, as mentioned earlier when discussing the reduced rule base for *Proasellus meridianus* in Fig. 5.4. Thus, the 0.5-cuts of the membership functions defining the three velocity values and the four ammonium concentration values divide the 2-dimensional input space in 12 parts. Fig. 5.7 clearly illustrates that the data belonging to the crisp abundance class absence, coloured in black, largely outnumber the data belonging to the three other abundance classes and that data holding similar values for the considered environmental variables show highly variable registered abundances. Therefore, the EKOO data set cannot be expected to reveal an unambiguous relationship between the selected abiotic variables and macroinvertebrate abundance.

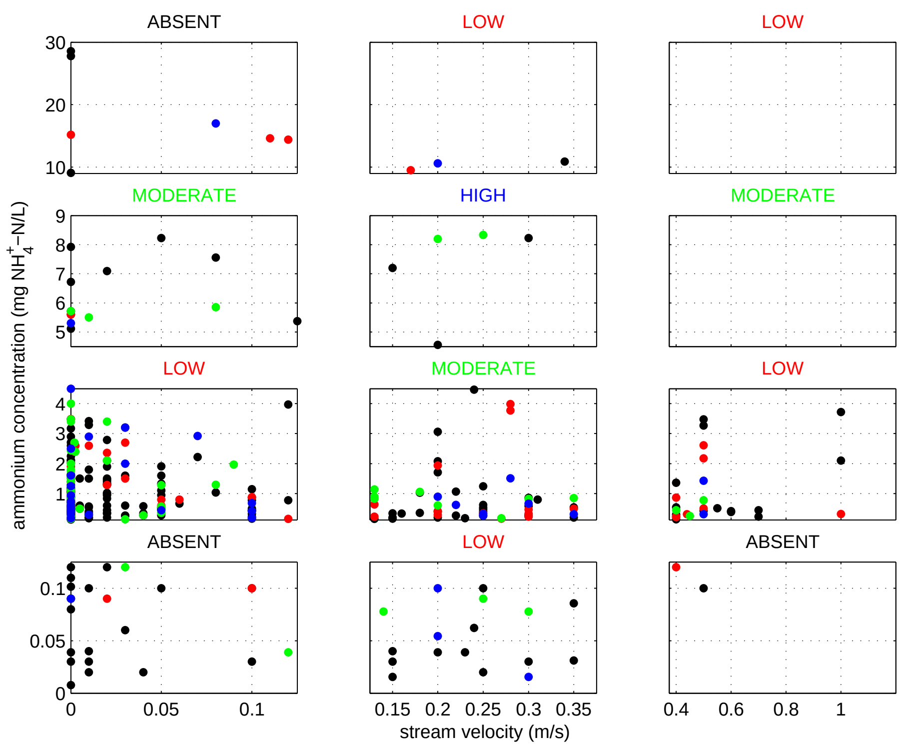


Figure 5.7: Data points in the different parts of the input space defined by the 0.5-cuts of the membership functions of velocity and ammonium concentration. The points are coloured according to the crisp (see Eq. (5.5) for the defuzzification procedure) abundance classes to which the measured abundance of *Proasellus meridianus* belongs.

5.4 Model evaluation

5.4.1 Performance measures

In order to compare the output obtained with the fuzzy ordered classifiers to the information in the EKOO data set, model and reference output should have the same format. In this study the membership degrees of the crisp abundance values in the data set to the linguistic abundance values, defined by membership functions shown in Fig. 5.1(b), are used as reference output. Two measures were used to evaluate the performance of the models: the percentage of correctly fuzzy classified instances (% CFCI) and the percentage of correctly classified instances (% CCI).

In ecology, % CCI is frequently used to compare the performance of species distribution models (Manel et al., 2001). Note that the calculation of % CCI, a performance measure for crisp classifiers, requires the defuzzification of the output of a fuzzy classifier, *i.e.* the output of the fuzzy classifier has to be turned into a crisp counterpart. In this study the fuzzy classifiers are defuzzified by assigning an object y to the smallest linguistic output value for which the maximum membership degree was obtained. As fuzzy classifiers are dealt with, a new performance measure, inspired by the % CCI and similar to the measure presented by Bodenhofer and Klement (2001), was defined: the percentage of correctly fuzzy classified instances (% CFCI). For N data points and a classification into n fuzzy classes, the % CFCI and % CCI are calculated as follows

$$\%\ \text{CFCI} = \frac{100}{N}\sum_{j=1}^{N}\left(1-\frac{1}{2}\sum_{i=1}^{n} \mid A_i(\mathbf{y}_{\text{data},j}) - A_i(\mathbf{y}_{\text{model},j}) \mid\right), \tag{5.3}$$

$$\%\ \text{CCI} = \frac{100}{N}\sum_{j=1}^{N}\left(1-\frac{1}{2}\sum_{i=1}^{n} \mid A_{\text{crisp},i}(\mathbf{y}_{\text{data},j}) - A_{\text{crisp},i}(\mathbf{y}_{\text{model},j}) \mid\right), \tag{5.4}$$

with

$$A_{\text{crisp},i}(\mathbf{y}) = \begin{cases} 1 & \text{, if } i = \min\{k \mid A_k(\mathbf{y}) = \max\limits_{l=1}^{n} A_l(\mathbf{y})\}, \\ 0 & \text{, otherwise.} \end{cases} \tag{5.5}$$

where $A_i(\mathbf{y}_{\text{data},j})$ is the membership degree of the j^{th} output to the i^{th} linguistic output value and $A_i(\mathbf{y}_{\text{model},j})$ is the membership degree to the i^{th} linguistic output value obtained as model output for the j^{th} input of the data set.

The two performance measures % CFCI and % CCI are illustrated by classification examples in Table 5.5. In case of a crisp classification, as in example 1, the two performance measures coincide. If an instance is classified in the correct class, it has a contribution of 100 to the global percentage of correctly (fuzzy) classified instances. If an instance is classified in a wrong class it has a contribution of zero to the global percentage of correctly (fuzzy) classified instances. Examples 2 to 4 in Table 5.5 are examples of fuzzy classifications, where the output is a set of degrees between 0 and 1 summing up to one. As long as there are classes to which both model output and reference output have a non-zero membership degree, the corresponding data point has

Table 5.5: Percentage of correctly classified instances (% CCI) and percentage of correctly fuzzy classified instances (% CFCI) for a crisp and three fuzzy classification examples given the measured output $\mathbf{y}_{\text{data}}$ and the modelled output $\mathbf{y}_{\text{model}}$.

	$\mathbf{y}_{\text{data}}$				$\mathbf{y}_{\text{model}}$					
	A_1	A_2	A_3	A_4	A_1	A_2	A_3	A_4	% CCI	% CFCI
1	0	1	0	0	0	0	1	0	0	0
2	0	0.2	0.8	0	0.8	0.2	0	0	0	20
3	0	0.2	0.8	0	0	0.4	0.6	0	100	80
4	0	0.2	0.8	0	0	0.2	0.8	0	100	100

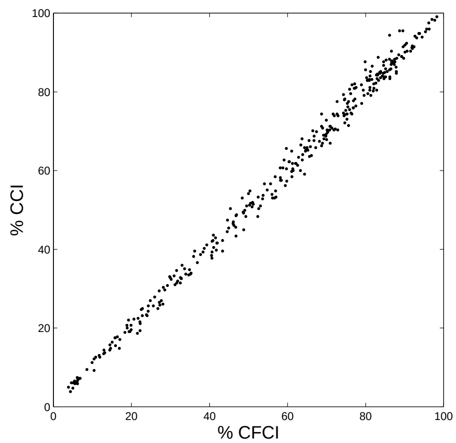


Figure 5.8: Comparison of the two performance measures % CCI and % CFCI for the 344 models (4 models for 86 species) validated on the EKOO data set.

a positive contribution to the global percentage of correctly fuzzy classified instances. Only if there exists no class to which both model output and reference output have a non-zero membership degree, the corresponding data point has a contribution of zero to the global percentage of correctly fuzzy classified instances. When determining the % CCI for fuzzy classifications, both reference and model output are first converted into crisp classifications and the % CCI is derived from those crisp classifications.

Similar values are obtained for % CCI as for its fuzzy alternative % CFCI, when evaluating the 344 models on the EKOO data set (Fig. 5.8). Both % CFCI and % CCI values obtained in this study can therefore be used for direct comparison with % CCI values found in literature for other applications. As the models designed in this study are fuzzy models, only the measure % CFCI was considered further in this study.

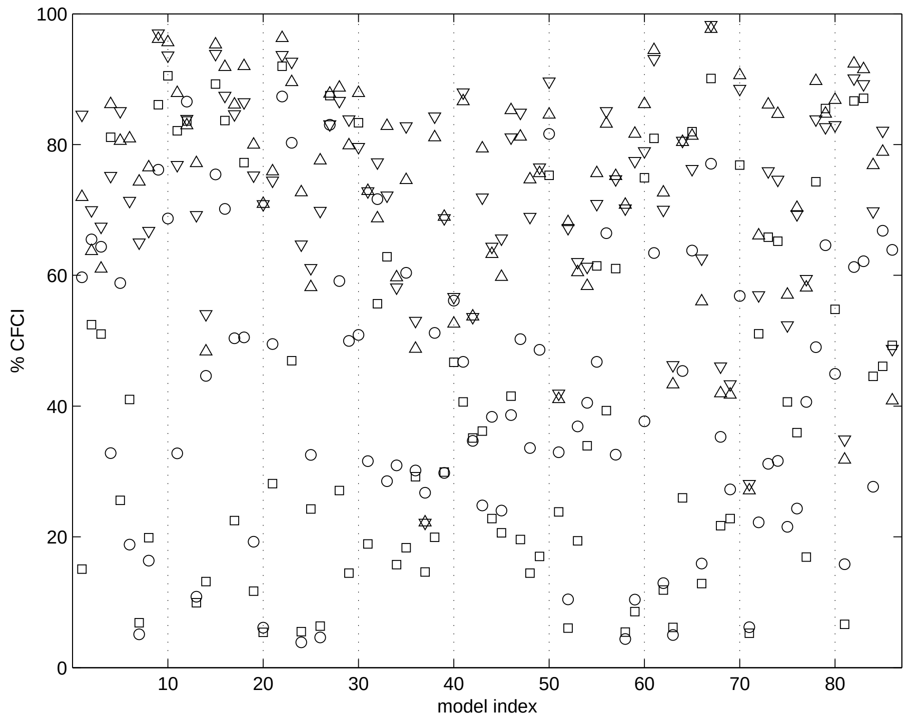


Figure 5.9: Percentage correctly fuzzy classified instances for the A-model (□), N-model (▽), P-model (△) and C-model (◯) for the 86 macroinvertebrate species. The names corresponding to the model index in the horizontal axis are given in Appendix A.

5.4.2 Results

The performances of the four models are given in Fig. 5.9. The N- and P-models have a relative high % CFCI ($>$ 50%) for most of the macroinvertebrate species in contrast to the A- and C-models, which have a low to moderate performance for a significant number of species. The obtained performances (% CFCI) should, however, be interpreted in the light of the EKOO data set on which the models were evaluated. Only the models of the 12 species listed in Table 5.4, with an abundance entropy larger than 0.7, were considered to be evaluated in an objective way by the EKOO data set. Among these 12 species, only one is an indicator species. One should therefore not conclude from the relatively high model performances obtained for most of the indicator species (Fig. 5.9) that these models really resemble the situation on the field to such a large extent.

When comparing the performances (% CFCI) of the four model types for the 12 selected species by means of box-whisker plots (Fig. 5.10), one can see that in

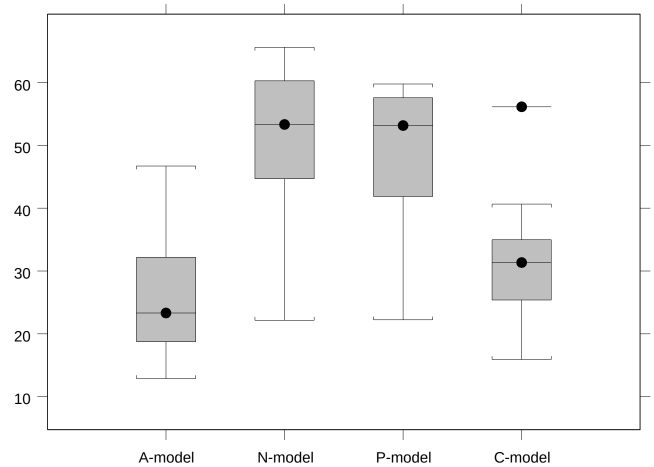


Figure 5.10: Box-whisker plots of the % CFCI obtained for the four model types for the 12 species for which a validation by the EKOO data set was considered objective. The box stretches from the 25^{th} percentile to the 75^{th} percentile. The median is shown as a line across the box. Any individual observation that is more than 1.5 $\times$ interquartile range from the box is identified separately with a horizontal line. The whiskers extend to the maximal and minimal observations that are not potential outliers.

general, the A- and C-models perform worse than the N- and P-models, even though the latter models still have some extremely low performance values for some species. This difference in performance over the different model types is due to the high number of sites having high N- and P-concentrations combined with a rather low presence.

In Table 5.6 those models are listed which have a good performance (% CFCI > 50) and which are evaluated in an objective way as the corresponding abundance entropy, *i.e.* the entropy of the distribution of the abundance values over the four abundances classes, was larger than 0.7. Note that for those species, for which an abundance entropy larger than 0.7 was obtained, a presence larger than 25% was recorded. Some species with a presence larger than 25%, however, have an abundance entropy smaller than 0.7. The corresponding species belong to the taxonomic groups Mollusca (*Physa fontinalis*, *Anisus vortex*, *Planorbis planorbis*, *Valvata piscinalis*), Hirudinea (*Erpobdella octoculata*, *Glossiphonia heteroclita*), Crustacea (*Gammarus pulex*) and Hemiptera (*Sigara striata*), none of them, except for *Physa fontinalis*, being characteristic for reference conditions. Most of these species are widely distributed ubiquitous species found in eutrophied very slow flowing and stagnant water bodies. The optimal

Table 5.6: Models selected based on a % CFCI larger than 50 and an abundance entropy larger than 0.7.

model index	species name	model type	% CFCI	presence	entropy
25	*Physa fontinalis*	N	61	0.37	0.77
		P	58		
36	*Anisus vortex*	N	53	0.49	0.87
40	*Erpobdella octoculata*	N	57	0.47	0.84
		P	53		
		C	56		
42	*Gammarus pulex*	N	54	0.42	0.81
		P	54		
45	*Glossiphonia heteroclita*	N	66	0.37	0.73
		P	60		
66	*Planorbis planorbis*	N	63	0.36	0.76
		P	56		
75	*Sigara striata*	N	52	0.42	0.81
		P	57		
77	*Valvata piscinalis*	N	59	0.41	0.80
		P	58		

environmental conditions for these species are quite different, as shown by the corresponding rule bases in Appendix B. This means that the selection results from an objective evaluation of the respective habitat suitability models. These 16 good performing and objectively evaluated models are all, except one model, N- or P-models.

Model performance comparison with other macroinvertebrate habitat suitability models remains difficult because most of the models developed have different output variables, for example a presence/absence proportion or a bio-assessment index such as the prediction of river health in (Walley and Dzeroski, 1995). Few studies even use validation measures (Rykiel, 1996; Fielding and Bell, 1997; Manel et al., 2001). However, due to the high correlation of the model performance measures % CFCI and % CCI in this study (Fig. 5.8), the obtained results can be compared with % CCI expressed results from more common presence-absence habitat suitability models. Habitat suitability models based on an Artificial Neural Network model structure predicting macroinvertebrate taxa in the Zwalm river basin in Belgium (Dedecker et al., 2002, 2004), obtained CCI-values of 60% for Gammaridae and 85% for Asellidae, taking into account a set of 15 variables and based on a data set collected in the Zwalm river basin. Although the predictive success was relative high, the large number of input variables needed (15) and the fact that these presence/absence models do not distinguish between different abundance levels, makes them less useful for the aim of river management. Moreover a certain degree of overfitting of non-causal relationships (Vaughan and Ormerod, 2005) is expected from the very small data set used for model validation (60 sites in (Dedecker et al., 2002); 60 sites measured yearly over 2 years in (Dedecker et al., 2004)).

Sometimes, knowledge-based models do not have quantitative expressions as outputs, but are qualitative models returning a certain degree of suitability that cannot be evaluated by means of exact monitoring data. For example (Kerle et al., 2001), (Schneider, 2001) and (Baptist et al., 2002) developed fish habitat suitability models (CASIMIR) based on a fuzzy rule base taking the vegetation and hydraulic river conditions as input and returning a degree of habitat suitability for the considered fish species. This output is not further translated into any measurable indicator such as abundance, diversity or presence/absence, which makes validation more difficult. In the present study, the suitability levels were translated to abundance levels to facilitate validation, but the translation itself can form a source of uncertainty and the reliability of this translation requires further research.

5.5 Conclusion

Fuzzy classifiers were applied to a modelling problem concerning the habitat suitability of river sites along springs to small rivers in the Central and Western Plains of Europe for 86 macroinvertebrate species. For each species, four models were developed, an A-, N-, P-, and C-model. The fuzzy classifiers take a certain width, velocity and either ammonium (A), nitrate (N) or phosphate (P) concentration or electrical conductivity (C) as input and return four values between 0 and 1 as output, indicating the degree to which the river site is concerned 'not suitable' respectively 'lowly', 'moderately' and 'highly suitable' for the species to establish a population. With the developed models the influence on the habitat suitability can be assessed for the stream width and stream velocity, two variables determining the river type and reflecting the water quantity conditions at a river site, as well as for one aspect of the impact of human activities, *i.e.* the nutrient and organic load.

Field data collected at 445 sites in the Province of Overijssel (the Netherlands) allowed for an objective evaluation of the four developed models for 12 selected species. The fact that among them only one is an indicator for reference conditions, indicates that given the present environmental conditions of rivers in EU Member States, shifts in abundance levels of more common species are more suitable to detect gradual changes in water quality. With an improving water quality, the follow-up of indicator species with more narrow niches will gain importance. This topic is addressed in more detail in Van Broekhoven et al. (2006). Of these 48 models, 16 models turned out to have a good model performance expressed by the performance measure % CFCI. These 16 good performing and objectively evaluated models are all, except one model, N- or P-models.

The usefulness of fuzzy rule-based models in ecosystem management was confirmed earlier by different studies (Adriaenssens et al., 2004; Bock and Salski, 1998; Meesters et al., 1998; Steinhardt, 1998; Kampichler et al., 2000; Mackinson, 2000; Kerle et al., 2001; Baptist et al., 2002). The question remains whether the fuzzy rule-based model structure, with its fuzzy sets, its if-then rules and its inference method, is appropriate for habitat suitability modelling in particular. In this study the requirement of interpretability was the decisive factor when determining the number of fuzzy sets assigned to the input variables and the output variable. On the one hand, the number of fuzzy sets is high enough to capture the different influence of the variables on the model output at different values. On the other hand, the number of fuzzy sets is not too high, and still allows for the formulation of an if-then rule for each combination of fuzzy sets of the different input variables. Moreover, the labels attached to the fuzzy sets are relevant for river management as they were inspired by the existing classifications used nowadays in bio-assessment and river typologies required by the Water Framework Directive. The fuzzy sets allow working with vague information which makes them very suitable for the variables and criteria used in this application field. The structure of a fuzzy rule base allows for the incorporation of the information summarized in the knowledge base into an inference system for habitat suitability modelling, by expressing non-linear relations in terms of if-then rules. The degrees of membership to the different output classes provide the end-user with a quantification of the uncertainty associated with the model output. This information has an added value in decision support. Hence, fuzzy rule-based modelling can be of great value as a knowledge-based habitat suitability modelling technique in river management.

A disadvantage of fuzzy rule-based models is that the shape and overlap of the fuzzy sets, having an important impact on the model output, are determined rather subjectively (Kompare et al., 1994) and more complex models, incorporating a higher number of input variables and fuzzy sets, are hard to develop following a purely knowledge-based design approach. Alternative input variables that could be considered when developing habitat suitability models, such as specific habitat structures, a specific habitat diversity level or a parameter reflecting the hydraulic events in a watercourse, are discussed in Van Broekhoven et al. (2006). Because of the subjectivity involved in the fuzzy rule-based model development, data-based techniques are often combined with knowledge-based models, for either the optimization of the rule base, the membership functions, or for the total fuzzy system. Likewise, in this study, there is certainly a need for a more rigid basis for model construction and optimization, mainly for the construction of membership functions. In Chapter 6 the application is discussed of genetic algorithms in the optimization of the membership function parameters for the four models of the 12 selected species.

CHAPTER 6

Membership function optimization

Not everything that can be counted counts,
and not everything that counts can be counted.
(Albert Einstein)

6.1 Introduction

Two main approaches to combine genetic algorithms and concepts from fuzzy logic or fuzzy systems can be distinguished (Cordón et al., 2001). In fuzzy genetic algorithms, a first type of hybrid approach, the performance of genetic algorithms is enhanced by fuzzy tools. Fuzzy models are for instance used to adapt the values of the parameters of the genetic algorithm, like the crossover probability P_c and mutation probability P_m, during the search (Herrera and Lozano, 2003) or the genetic algorithm applies fuzzified variation operators, as the fuzzy connectives based crossover (Herrera et al., 1997). In the second type of hybrid approach, called genetic fuzzy systems, a genetic algorithm evolves a fuzzy system. The most extended genetic fuzzy system type is the genetic fuzzy rule-based system, where an evolutionary algorithm is employed to optimize or identify different components of a fuzzy rule-based system. Other types of genetic fuzzy systems include genetic fuzzy clustering systems, genetic fuzzy neural systems and genetic fuzzy decision trees. An overview of the latter types of genetic fuzzy systems, which are not considered in this dissertation, is given in (Cordón et al., 2001, Chapter 10). Genetic fuzzy rule-based systems encompass both optimization and identification of membership functions and rules. In optimization problems the objective is to find optimal values for a set of parameters, for instance membership function parameters, whereas in identification problems model components, for instance the rule base, are designed from scratch. The flexible data structure used in evolutionary algorithms to represent a candidate solution and their ability to explore a large search space for suitable solutions only requiring a simple scalar performance measure, make evolutionary algorithms suitable search techniques for a partial optimization or identification of the model structure (Cordón et al., 2004). Information about known model properties, such as the shape of the membership functions, the rules or the number of rules, can be easily incorporated in the evolutionary search process.

In this study the accuracy is tried to be improved of the habitat suitability models obtained by the knowledge-based design process described in Chapter 5, for the region where the EKOO data set was collected, while maintaining the interpretability, *i.e.* the descriptive power of the models (Casillas et al., 2003b; Mencar et al., 2005). In the framework of this study, interpretability means that the river manager consulting the models is familiar with all components of the designed models and is able to get insight in the models just by looking at the different components. Given the uniformity of the qualitative information in the eight consulted knowledge sources, the rules in the rule bases of the developed models can be considered generally applicable to the Central and Western Plains of Europe. The knowledge sources also clearly reveal that the definition of linguistic values of environmental variables slightly differ from one river basin to another. Therefore the rule bases were kept unchanged, yet only the membership functions of the input variables were optimized in such a way that after optimization all fuzzy sets still represent the meaning assigned by experts to the corresponding linguistic values. As no straightforward relation exists between the membership functions and the output of a linguistic fuzzy model, a genetic algorithm was used as optimization method as it works on the complete solution of the optimization problem, in this case being the whole set of membership function parameters. In literature examples can be found of membership function parameter optimization with genetic algorithms for all common types of membership functions, *i.e.* for triangular (using binary encoding in (Arslan and Kaya, 2001; Chiou and Lan, 2005), using real encoding in (Casillas et al., 2005; Ishigami et al., 1995; Rojas et al., 2001) and using a special encoding in (Kinzel et al., 1994)), trapezial (using binary encoding in (Ascia et al., 2006; Bodenhofer and Klement, 2001; Karr and Gentry, 1993) and using real encoding in (Lau et al., 2005; Suzuki et al., 2001)) and Gaussian functions (using binary encoding in (Damousis et al., 2002; Shu and Burn, 2004; Surmann et al., 1993) and using real encoding in (Damousis et al., 2002; Kim and Roschke, 2006; Kim et al., 2005; Suzuki et al., 2001)). Other techniques used to optimize membership functions are gradient descent (Shi and Mizumoto, 2000; Simon, 2002; Vishnupad and Shin, 1999), algorithms inspired on those used in the neural networks field (Nauck and Kruse, 1997; Paiva and Dourado, 2004; Tanaka et al., 1995), the Levenberg-Marquardt algorithm (Botzheim et al., 2004), Kalman filters (Simon, 2002; Sun, 1994), genetic programming (Bastian, 2000), evolution strategies (Jin et al., 1999), tabu search (Bağiş, 2003) and simulated annealing (Guély et al., 1999; Sánchez et al., 2001).

The membership function optimization was only carried out for the A-, N-, P- and C- models of the 12 species whose performance could be evaluated in an objective way on the EKOO data set, as defined in Section 5.3.2. The 12 selected species are *Anisus vortex*, *Asellus aquaticus*, *Erpobdella octoculata*, *Gammarus pulex*, *Glossiphonia heteroclita*, *Helobdella stagnalis*, *Physa fontinalis*, *Planorbis planorbis*, *Proasellus meridianus*, *Radix peregra*, *Sigara striata* and *Valvata piscinalis*. The membership functions of the input variables were optimized using a classic genetic algorithm with binary chromosomes, as well as a real-coded genetic algorithm. The properties of the genetic algorithm are described in Section 6.2, with special attention to the applied representation of candidate solutions and fitness function. The optimization results are

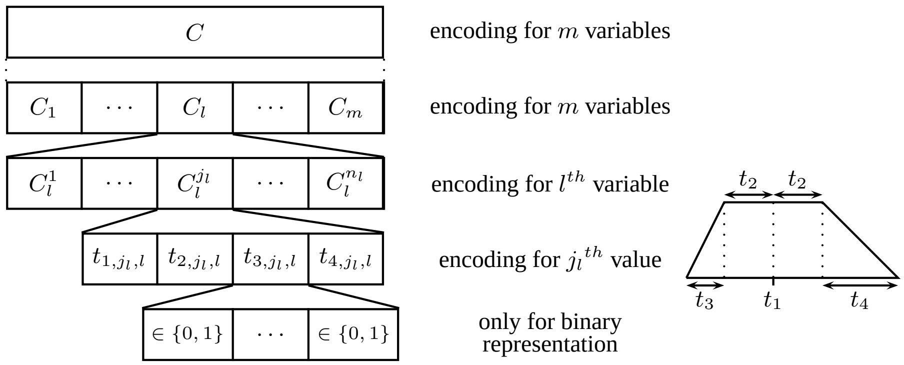


Figure 6.1: Encoding of membership function parameters which does not ensure interpretability.

discussed in Section 6.3 and conclusions are summarized in Section 6.4.

6.2 Properties of the genetic algorithm

6.2.1 Representation of a candidate solution

In the habitat suitability models obtained by the knowledge-based design described in Chapter 5 the linguistic values assigned to each individual variable are defined in a meaningful way by membership functions forming a fuzzy partition. In order to maintain the interpretability of the definition of the linguistic values during the optimization, the encoding of the membership function parameters should be well considered.

In Fig. 6.1 a straightforward encoding of the trapezial membership functions of m variables is given. To the l^{th} variable n_l linguistic values are assigned and each linguistic value is characterized by four parameters t_1, t_2, t_3 and t_4. These parameters are real values in a real-valued representation and further encoded in a binary string in a binary representation. However, if this chromosome is used, meaningless membership functions as shown in Fig. 6.2 might be obtained.

To decide if a certain set of membership functions is interpretable or not is a difficult and subjective task. Nevertheless, several properties ensuring good interpretability of membership functions have been proposed (Valente de Oliveira, 1999). The most important properties in the framework of membership function optimization are (Casillas et al., 2003a)

- the σ-*completeness* property, requiring for each point x the existence of a fuzzy set A_i to which x has a membership degree larger than σ

  $$(\forall x \in \mathbf{X})(\exists i \in \{1, \ldots, n\})(A_i(x) \geq \sigma > 0)\,,$$

  with A_i a fuzzy set defined on the domain X of x, and σ a given completeness degree,

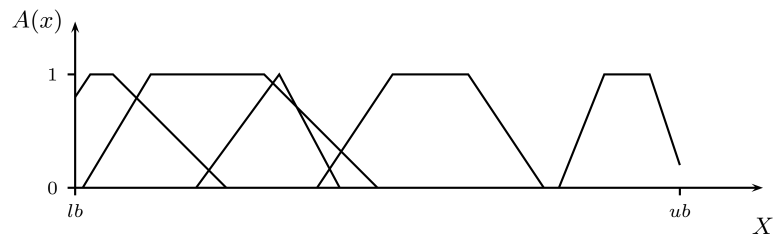


Figure 6.2: Interpretability might be lost when optimizing membership functions using an inappropriate encoding.

- the *normality* property, satisfied if all linguistic values are defined by normal fuzzy sets such that each linguistic value exhibits full matching with, at least, a value of the variable's domain, and
- the *distinguishability* property, asking for membership functions that are distinct enough from each other such that each linguistic value has a clear meaning and the corresponding fuzzy set clearly defines a range in the variable's domain.

When optimizing membership functions, one can either represent the membership functions is such a way that all candidate solutions satisfy (some of) the three above properties (Ascia et al., 2006; Casillas et al., 2005; Chiou and Lan, 2005; Shu and Burn, 2004), or include interpretability measures in the objective functions, thus guiding the search to good solutions (Jin et al., 1999; Kim et al., 2005; Surmann et al., 1993). As fuzzy partitions satisfy all three properties, encoding the membership functions in such a way that all candidate solutions are fuzzy partitions, is very common in membership function optimizations ensuring the semantic integrity of the linguistic values (Casillas et al., 2003a,b). Also in this study an encoding which always result in fuzzy partitions was opted for.

The n_l membership functions of an input variable X_l of the considered models are characterized by a vector of $2n_l$ reals (Fig. 6.3), $\mathbf{a}_l = (a_{1,l}, \ldots, a_{2n_l,l})$, satisfying the following two constraints

$$(\forall\, j \in \{1, \ldots, n_l\})(a_{2j\text{-}1,l} \leq a_{2j,l})\,, \tag{6.1}$$

$$(\forall\, j \in \{1, \ldots, n_l - 1\})(a_{2j,l} < a_{2j+1,l})\,. \tag{6.2}$$

In this study both a binary-coded as well as a real-coded genetic algorithm is applied. They evolve chromosomes as presented in Fig. 6.4. The representation of the membership function parameters by a binary vector (using Gray encoding), restricts the values the parameters can take to a limited set of values defined by the upper and lower bound of the optimization interval and the length of the binary string, but has the advantage that it allows the use of very straightforward crossover and mutation strategies. The real-coded genetic algorithm is directly applied to a vector containing the real values of the optimized parameters, which allows for a finer tuning of the parameters.

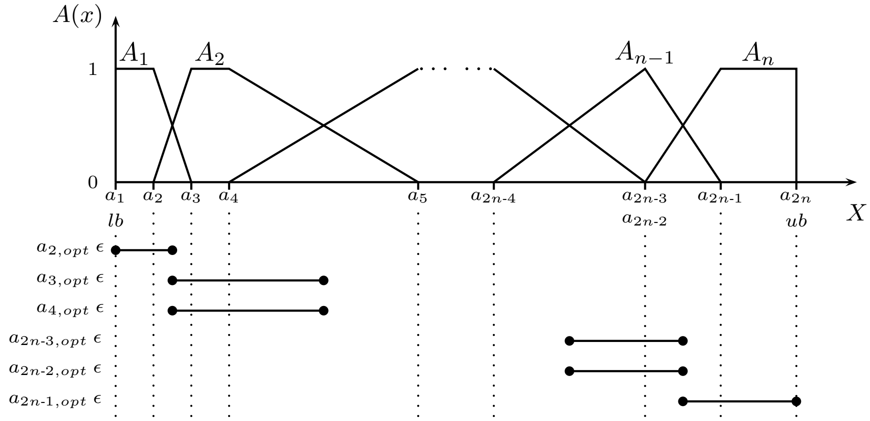


Figure 6.3: Illustration of the optimization intervals used for the membership function parameters during the bounded simulation.

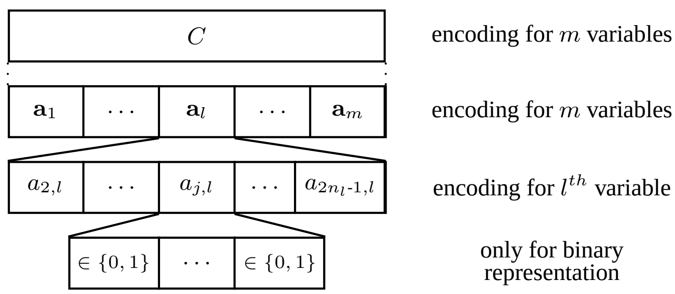


Figure 6.4: Encoding of membership function parameters which does ensure interpretability.

Two optimizations were carried out: a bounded and a free optimization. During the bounded optimization the kernels of the optimized membership functions are always subsets of the 0.5-cuts of the corresponding original membership functions (as illustrated in Fig. 6.3), whereas during the free optimization only the number of membership functions of the fuzzy partition is fixed for each input variable. The free optimization was carried out to investigate how the optimization process evolves if no constraints are set. The membership function parameters were coded as binary strings of 7 and 10 bits per parameter respectively for the bounded and free optimization respectively.

Table 6.1: Four fuzzy classification examples and their corresponding performances expressed by % CFCI and AD.

	$\mathbf{y}_{\text{data}}$				$\mathbf{y}_{\text{model}}$					
	A_1	A_2	A_3	A_4	A_1	A_2	A_3	A_4	% CFCI	AD
a	0	0.2	0.8	0	0.8	0.2	0	0	20	1.6
b	0	0.2	0.8	0	0	0.4	0.6	0	80	0.2
c	0	0.2	0.8	0	0	0.1	0.8	0.1	90	0.2
d	0	0.2	0.8	0	0	0	0.8	0.2	80	0.4

6.2.2 Fitness

The % CFCI, presented in Section 5.4.1, has the advantage that it can be understood intuitively. For N data points and a classification into n fuzzy classes, the % CFCI is obtained by

$$\%\ \text{CFCI} = \frac{100}{N}\sum_{j=1}^{N}\left(1-\frac{1}{2}\sum_{i=1}^{n}\left|A_i(\mathbf{y}_{\text{data},j})-A_i(\mathbf{y}_{\text{model},j})\right|\right), \tag{6.3}$$

where $A_i(\mathbf{y}_{\text{data},j})$ is the membership degree of the j^{th} output to the i^{th} linguistic output value and $A_i(\mathbf{y}_{\text{model},j})$ is the membership degree to the i^{th} linguistic output value obtained as model output for the j^{th} input of the data set.

However, it is not an appropriate objective function for the optimization of a fuzzy ordered classifier, as % CFCI is not sensitive to the position of the classes where the wrong classification occurs. When visually comparing the reference output in Table 6.1 with the model outputs b and d and given the fact that the output classes are ordered from A_1 to A_4, one would certainly say that model output b approximates the reference output better than model output d. However, the same % CFCI is assigned to examples b and d, as the sum of the absolute differences in membership degree in the reference and model output to the four individual classes is identical, as shown in Fig. 6.5.

Therefore another performance measure for fuzzy classifiers with an ordered set of classes is introduced, returning the average deviation (AD) between the position of the class obtained with the model and the position of the class stored in the reference data set. The AD varies from 0 to $n-1$ and is calculated as follows for N data points and a classification into n fuzzy classes

$$\text{AD} = \frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{n-1}\left|\sum_{k=1}^{i}A_k(\mathbf{y}_{\text{data},j})-\sum_{k=1}^{i}A_k(\mathbf{y}_{\text{model},j})\right|, \tag{6.4}$$

where $A_i(\mathbf{y}_{\text{data},j})$ is the membership degree of the j^{th} output to the i^{th} linguistic output value and $A_i(\mathbf{y}_{\text{model},j})$ is the membership degree to the i^{th} linguistic output value obtained as model output for the j^{th} input of the data set. In Fig. 6.6(a) the % CFCI-

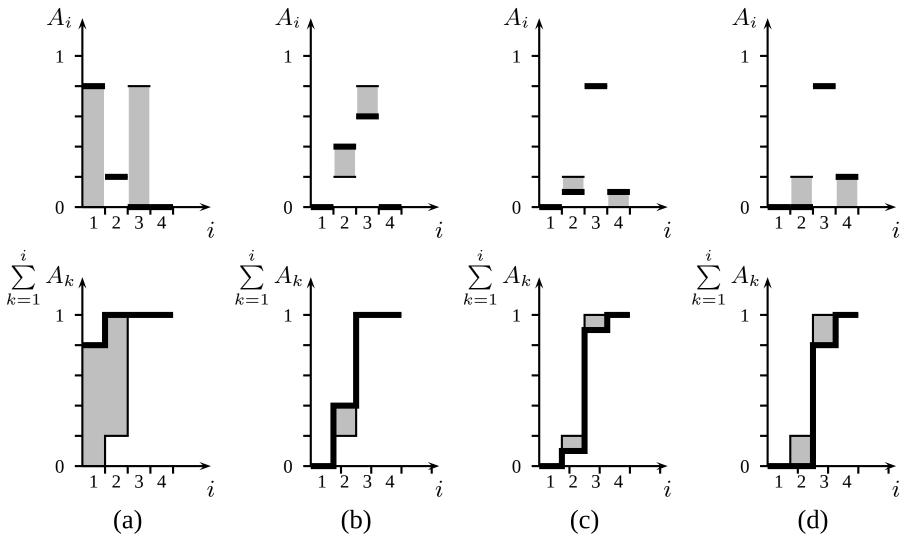


Figure 6.5: Illustration of the performance measures % CFCI and AD for the fuzzy classification examples in Table 6.1. In the figures in the top row, illustrating % CFCI, the thin and thick lines indicate the reference and model output respectively. In the figures in the second row, illustrating AD, the thin and thick lines are the cumulative functions of the reference and model output respectively.

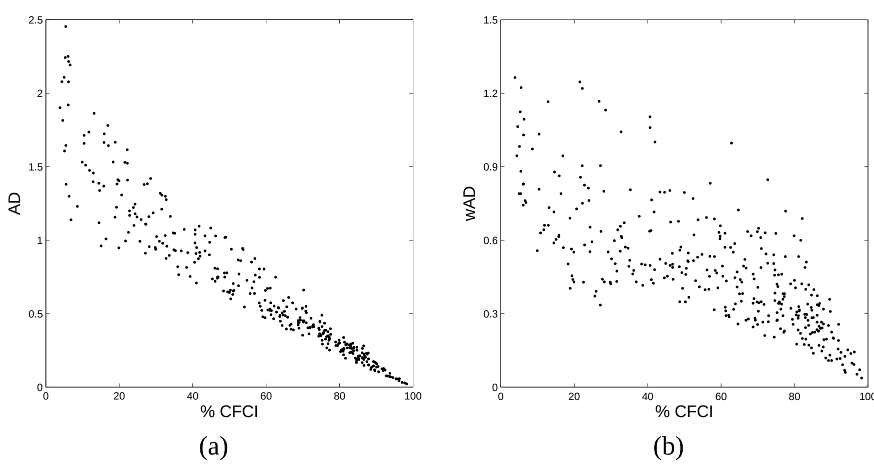


Figure 6.6: Comparison of the % CFCI-values to the AD- and wAD-values.

and the AD- values obtained for the four models of the 86 macroinvertebrate species are plotted. One sees that AD tends to decrease with increasing % CFCI.

The measure AD is illustrated in Table 6.1 on the same examples as the two other performance measures. At first sight it seems hard to get insight in AD. When considering the cumulative membership degrees, *i.e.* the sum of the membership degrees to a class and its lower classes as in Fig. 6.5, instead of the membership degrees, one sees that the AD is nothing else but the area between the cumulative functions of model and reference output.

The AD is zero if the model output equals the reference output and increases with increasing distance between the reference output and the model output. The AD distinguishes between examples b and d, whereas the % CFCI does not. On the other hand, the same AD, but a different % CFCI, is obtained for examples b and c. In example b the membership degree assigned to class A_2 is 0.2 too high. This surplus of membership degree should in fact be assigned to the adjacent class A_3. In example c the membership degree assigned to class A_4 is 0.1 too high and this surplus of membership degree should in fact have been assigned to class A_2, *i.e.* two classes lower. The distance between the reference output is therefore 1×0.2 for example b and 2×0.1 for example c. The % CFCI however is a measure of the sum of the errors made for each individual class. For example b the error in membership degree is 0.2 for the two classes A_2 and A_3, whereas in example d the errors are 0.1 for the two classes A_2 and A_4. Note that the AD is insensitive to the direction of the wrong classification as the absolute values of the differences are taken. If classifying an instance in a too high class is worse (or better) than classifying it in a too low class, the formula in Eq. (6.4) should be slightly altered.

During the search, each candidate solution was evaluated on each of the 445 data points, using a weighted average deviation (wAD) in which the weights guarantee that each region of the input space defined by the 0.5-cuts of the membership functions of the non-optimized models has the same contribution to the fitness

$$\text{wAD} = \sum_{j=1}^{N} w_j \cdot \sum_{i=1}^{n-1} \left| \sum_{k=1}^{i} A_k(\mathbf{y}_{\text{data},j}) - \sum_{k=1}^{i} A_k(\mathbf{y}_{\text{model},j}) \right|, \tag{6.5}$$

with

$$w_j = \frac{1}{N_j \cdot n_{regions}} .$$

In the definition of the weights w_j, N_j is the number of data points in the same region of the input space as the j^{th} input of the data set and $n_{regions}$ is the number of regions in which the input space is divided.

6.2.3 Algorithm

The structure of the genetic algorithm is shown in Algorithm 2. A thorough investigation of the influence on the genetic algorithm performance of different mutation,

Algorithm 2: Genetic algorithm

```
t ← 0
Initialize Population P_t at random
foreach Individual of P_t do
    Decode chromosome
    if chromosome represents unfeasible solution then
        Try to restore chromosome
    end
    if chromosome represents feasible solution then
        Calculate fitness of the individual
    else
        Assign very bad fitness value to the individual
    end
end
while stop criterion not reached do
    Select individuals by tournament selection
    Recombine individuals by crossover and mutation
    foreach Child of P_t do
        Decode chromosome
        if chromosome represents unfeasible solution then
            Try to restore chromosome
        end
        if chromosome represents feasible solution then
            Calculate fitness of the individual
        else
            Assign very bad fitness value to the individual
        end
    end
    Replace worst individual of P_{t+1} by best individual of P_t
    P_t ← P_{t+1}
    t ← t + 1
end
```

crossover and selection procedures and the optimization of their parameters was outside the scope of this study. We carried out some fragmentary investigation of the parameter settings of the selected mutation and crossover procedures with some of the 48 models and applied the best setting obtained to optimize the membership functions of all 48 models.

The same procedure was followed by the binary-coded and real-coded algorithm, except for the recombination and mutation. Each optimization starts with a population of 100 randomly generated strings, which, in case they do not represent a feasible solution, are tried to be restored by replacing them by (the binary representation of) a vector consisting of substrings of sorted real values of the unfeasible string for each variable. Note that this restoration procedure does not always result in a string satisfying Eq. (6.2).

At each generation step, 100 parents were selected by tournament selection. Two by two the parents were recombined and mutated, resulting in two children. In the binary-coded algorithm, uniform crossover is applied (crossover probability = 0.95). Each bit of the strings obtained after recombination, or, in case no crossover was carried out, the strings of the parents, were changed with a mutation probability being the reverse of the length of the binary string. In the real-coded algorithm, one child is created with heuristic crossover and one with arithmetical crossover (crossover probability = 0.95). The procedure of the heuristic crossover described by Michalewicz (1996) was slightly adapted to guarantee that each real value $a_{child_1,l}$ in the string of the child derived from the corresponding values $a_{parent_b,l}$ and $a_{parent_w,l}$ of the best and, respectively, the worst performing parent of the two parents, is an element of the optimization interval [b_l,B_l]. In Eq. (6.7), r_1 is a random number between 0 and 1 and identical for all values of a string during a recombination:

$$a_{interval,l} = \max(b_l, \min(B_l, 2a_{parent_b,l} - a_{parent_w,l}))\,, \tag{6.6}$$

$$a_{child_1,l} = \min(a_{parent_b,l}, a_{interval,l}) + r_1(\max(a_{parent_b,l}, a_{interval,l}) - \min(a_{parent_b,l}, a_{interval,l}))\,, \tag{6.7}$$

$$a_{child_2,l} = \frac{1}{2}(a_{parent_b,l} + a_{parent_w,l})\,. \tag{6.8}$$

The real strings of the children, or, in case no recombination was carried out, the strings of the parents, were mutated as described in Eq. (6.9). Each value a_l is replaced by a randomly selected (uniform probability distribution) value a_l' from an interval around a_l being at most as large as p_{mut}% of the interval [b_l,B_l] ($p_{mut} = 3$ and $p_{mut} = 0.4$ for the bounded and, respectively, the free optimization). In Eq. (6.9), r_2 is a random number between 0 and 1 and r_3 a random binary digit, both being identical for all values of a string during a recombination:

$$a_l' = \begin{cases} \min(a_l + \frac{1}{200}r_2 p_{mut}(B_l - b_l), B_l) & \text{, if } r_3 \text{ is } 0\,, \\ \max(a_l - \frac{1}{200}r_2 p_{mut}(B_l - b_l), b_l) & \text{, if } r_3 \text{ is } 1\,. \end{cases} \tag{6.9}$$

Children not satisfying Eqs. (6.1–6.2) are tried to be restored following the same procedure as during the initialization of the population. Furthermore, elitism is applied in the algorithm: the worst offspring is replaced by the best individual of the current population. The genetic algorithm was stopped if only small improvements of the fitness of the best individual ($\triangle$ fitness $<$ 0.001) were obtained during the last 50 consecutive generations as illustrated in Fig. 6.7 or if the 1000th generation was reached. Hundred repetitions were carried out for each optimization and the model with the highest % CFCI among the 100 candidate models was retained as result of the optimization.

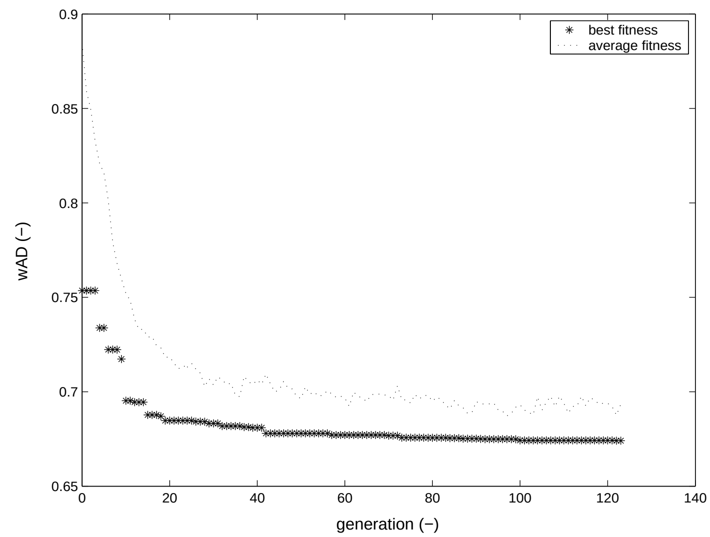


Figure 6.7: Average fitness and best fitness as a function of the generation in the genetic algorithm.

6.3 Optimization results

The results obtained for the four models of the 12 selected species are summarized in Figs. 6.8–6.9. One expects the models obtained with the real-coded genetic algorithm to perform at least as good as the corresponding models obtained with the binary-coded genetic algorithm as the search space of the binary-coded genetic algorithm is a subset of the search space of the real-coded genetic algorithm. Furthermore, the model obtained through free optimization is expected to outperform the corresponding model obtained through bounded optimization, which on its turn is expected to score better than the original model. Strictly speaking, the performance of the genetic algorithms can only be compared based on the performance of the original and optimized models according to the performance measure wAD, used as fitness function. In Fig. 6.8 the wAD of the original models are shown, together with the smallest wAD-value obtained for the best individual of the last population of the 100 repetitions of each optimization. In Fig. 6.9, however, the % CFCI of the original and optimized models are given, as % CFCI can be understood intuitively and resembles the performance measure % CCI commonly used in ecology. When analyzing the results in Fig. 6.9, one should always keep in mind the variability of the relationship between the two performance measures wAD and % CFCI . As shown in Fig. 6.6(b) wAD tends to decrease with increasing % CFCI, but it also shows that a model M_1 scoring better than a model M_2 according to the wAD, might score worse according to the % CFCI. Therefore, the performance of the genetic algorithms can only be really judged by the values obtained for wAD, the fitness.

When considering the performance measure wAD (Fig. 6.8),

- optimized models perform better than the corresponding original models,
- models obtained with the real-coded genetic algorithms do not perform worse than those obtained with the corresponding binary-coded genetic algorithms, except for the A- and N-models of *Erpobdella octoculata* obtained by free optimization, the A-model of *Physa fontinalis* obtained by bounded optimization, the N-models of *Gammarus pulex*, *Glossiphonia heteroclita*, *Sigara striata* and *Valvata piscinalis* obtained by bounded optimization, as well as the N-model of *Sigara striata* obtained by free optimization,
- models obtained with free optimizations of the binary-coded genetic algorithm perform better than the corresponding models obtained by bounded optimization, except for the N-models of *Anisus vortex*, *Asellus aquaticus*, *Physa fontinalis* and *Radix peregra* and,
- models obtained with free optimizations of the real-coded genetic algorithm perform better than the corresponding models obtained by bounded optimization.

When comparing the wAD of the models obtained with the corresponding binary and real-coded genetic algorithms, one sees that the models obtained by bounded optimization are generally equally good for both types of genetic algorithms. The fact that eight of the 96 real-coded genetic algorithms do not return a better solution than their binary-coded counterpart, indicates that the implemented control structures were maladjusted to these eight membership function optimization problems. The recorded reversed order of the wAD-values obtained for the four N-models with the binary-coded genetic algorithms might be caused by the binary coding, restricting the values taken by the membership function parameters in the optimized models to a limited set of values. Thus, when using binary encoding the search space of the binary-coded genetic algorithm applied during the free optimization might simply not contain a solution outperforming the solution returned by the bounded optimization. The fact that all wAD-values obtained by the real-coded genetic algorithms respect the expected order, supports the above argument.

When considering the % CFCI, the models obtained with the real-coded genetic algorithms do not perform worse than those obtained with the binary-coded genetic algorithms, except for the A-model for *Erpobdella octoculata* obtained through free optimization. For this model, the optimized model obtained with the real-coded genetic algorithm shows a negligible worse performance of 0.1% compared to the model obtained with the binary-coded genetic algorithm (Fig. 6.9(a)). For the models obtained with the binary-coded genetic algorithm, the expected order of the % CFCI-values of respectively the original model and the models obtained through bounded and free optimization, is not respected by the results recorded for the A-model of *Radix peregra*, the N-models of *Anisus vortex*, *Erpobdella octoculata*, *Gammarus pulex*, *Glossiphonia heteroclita*, *Helobdella stagnalis*, *Physa fontinalis*, *Planorbis planorbis* and *Radix peregra*, nor for the P-models of *Anisus vortex*, *Glossiphonia heteroclita* and *Physa*

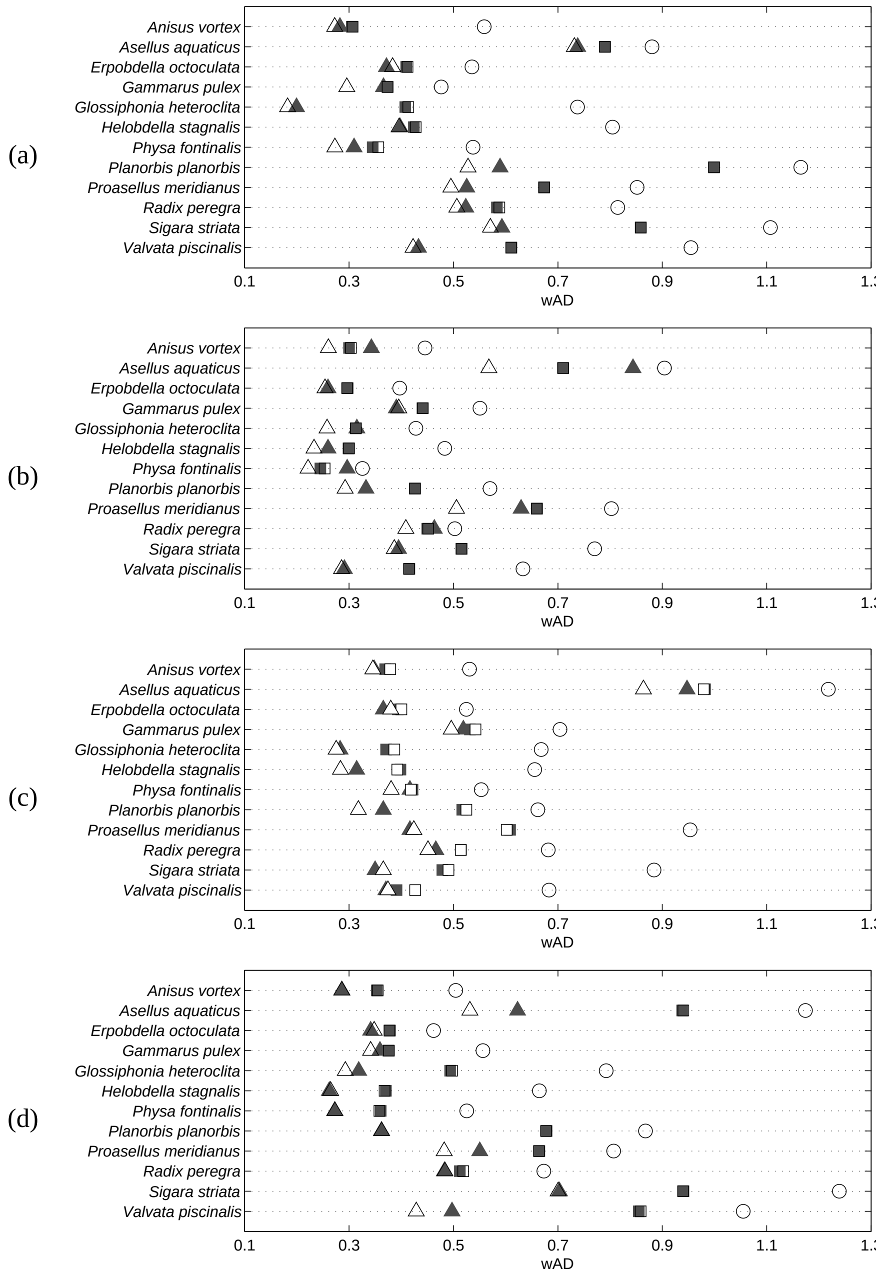


Figure 6.8: Weighted average deviation for the original models (○) and the models obtained through bounded optimization with the binary-coded genetic algorithm (GA) (■), free optimization with the binary-coded GA (▲), bounded optimization with the real-coded GA (□) and free optimization with the real-coded GA (△) for the 12 selected species: (a) A-models, (b) N-models, (c) P-models and (d) C-models.

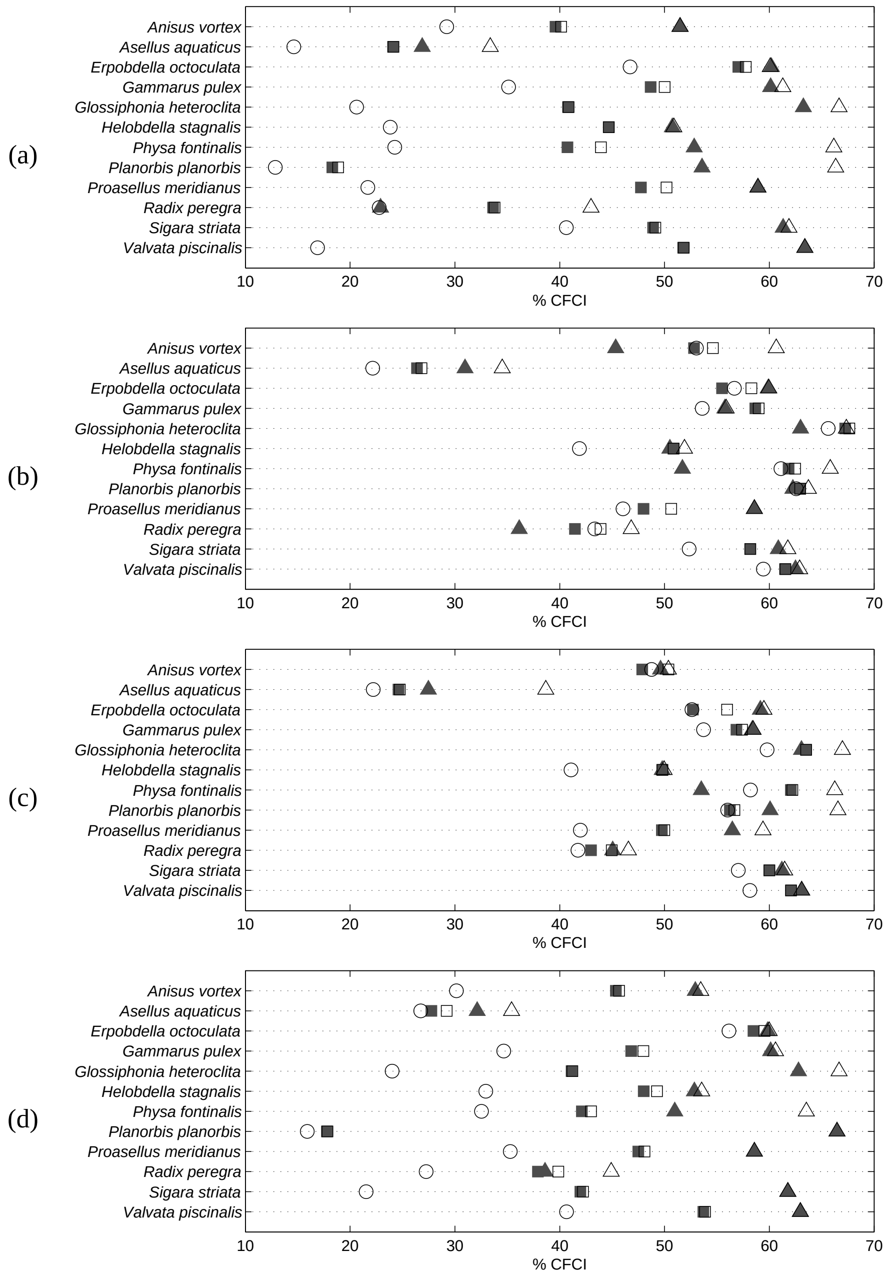


Figure 6.9: Percentage of correctly fuzzy classified instances for the original models (○) and the models obtained through bounded optimization with the binary-coded genetic algorithm (GA) (■), free optimization with the binary-coded GA (▲), bounded optimization with the real-coded GA (□) and free optimization with the real-coded GA (△) for the 12 selected species: (a) A-models, (b) N-models, (c) P-models and (d) C-models.

fontinalis. When applying the real-coded genetic algorithm only the % CFCI-values of the original, bounded and freely optimized N-models of *Gammarus pulex* and *Glossiphonia heteroclita* do not respect the expected order.

In Figs. 6.10–6.11 the results obtained for the A-model of *Proasellus meridianus* are shown. Note that the membership function describing the $oligosaprobic$ to β, α-$oligosaprobic$ conditions (hereafter called $oligosaprobic$) in the original model has such a small support that it can hardly be noticed in Fig. 6.10(a). For the A-model of *Proasellus meridianus*, as for most models of the other selected species, the results obtained with the real-coded genetic algorithm are very similar to the results obtained with the binary-coded genetic algorithm. This is especially true in case of the bounded optimization where the membership function parameters of the optimized models obtained with both algorithms are often equal to the lower or upper bound, or the second or next-to-last value of the corresponding optimization interval.

In Fig. 6.10 one sees that the membership functions of the velocity value low and the $oligosaprobic$ conditions are extended towards higher velocities and ammonium concentrations respectively. The membership functions in Figs. 6.10(c) and 6.10(e) no longer reflect the meaning given by the experts to the linguistic values. During the bounded optimization the extension is however limited by the constraints described in Section 6.2, which guarantees the interpretability of the fuzzy partitions of the optimized models. In Fig. 6.11 the number of data points belonging to the four defuzzified abundance classes $A_{\text{crisp},i}$ (see Eq. (5.5) for the defuzzification procedure) in the different regions of the input space are given and visualized by means of histograms for the original models and the two models obtained with the binary-coded genetic algorithm. No histograms are shown for the models obtained with the real-coded genetic algorithm, as similar membership functions were obtained with the binary-coded and real-coded genetic algorithm. One sees that, by extension of the support of the velocity value low and the $oligosaprobic$ conditions, more data points and in particular more data points belonging to the abundance class $absent$, fire the rule

IF vel IS low AND $ammon$ IS $oligotrophic$ THEN $abundance$ IS $absent$,

instead of the rules

IF vel IS low AND $ammon$ IS β-$mesotrophic$ THEN $abundance$ IS low,
IF vel IS $moderate$ AND $ammon$ IS $oligotrophic$ THEN $abundance$ IS low,
IF vel IS $moderate$ AND $ammon$ IS β-$mesotrophic$ THEN $abundance$ IS $moderate$,

which results in a better score for the used fitness wAD as well as for the other performance measures % CCI, % CFCI and AD.

The differences between the results obtained with the bounded and free optimizations illustrate that one should not only focus on the accuracy of a model when evaluating its performance, but that the global performance of a model implies a balance between its interpretability and its accuracy. In the framework of this study, interpretability means that the river manager consulting the models is familiar with all components of the designed models and is able to get insight in the models just by looking at the different components. In order to guarantee interpretability, the definition of the linguistic values, *i.e.* the membership functions, should correspond to those

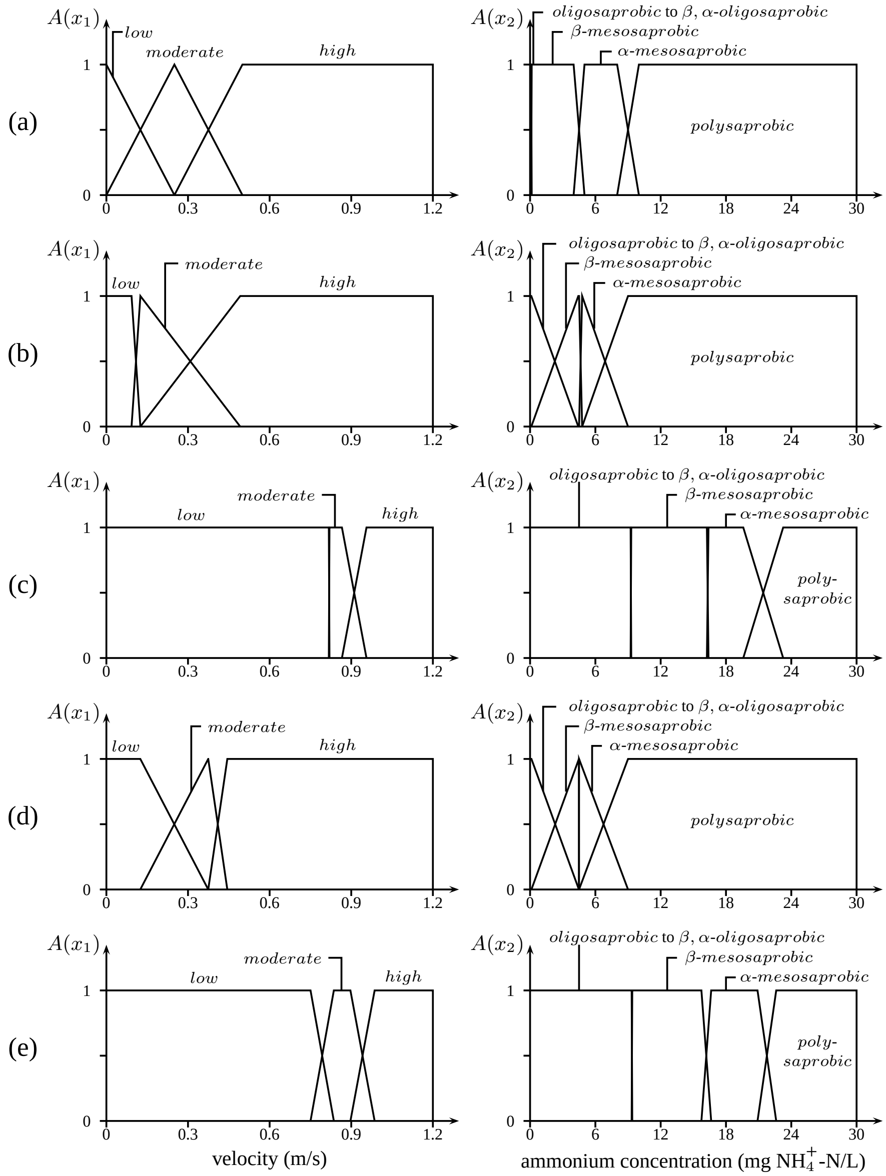


Figure 6.10: Membership functions of the A-model of *Proasellus meridianus*: (a) original model and models obtained through (b) bounded optimization with the binary-coded genetic algorithm (GA), (c) free optimization with the binary-coded GA, (d) bounded optimization with the real-coded GA and (e) free optimization with the real-coded GA.

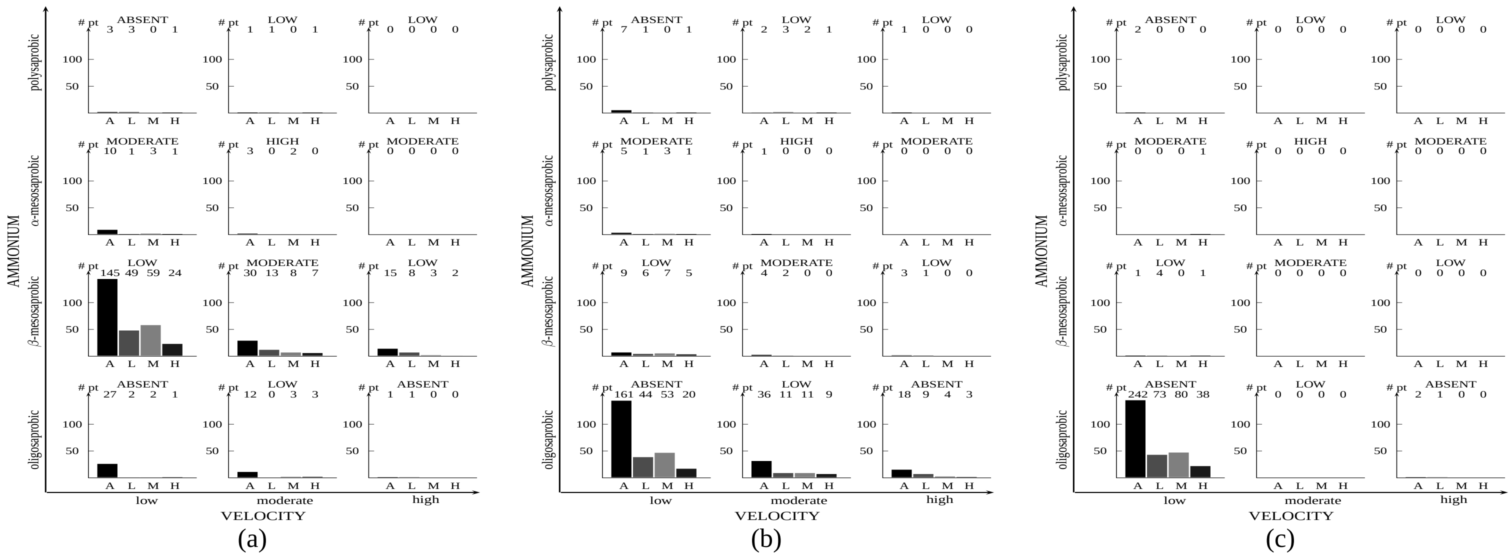


Figure 6.11: Distribution of the data points over the abundance classes in the different regions of the input space defined by 0.5-cuts of the membership functions of (a) the original model, (b) the model obtained through bounded optimization with the binary-coded genetic algorithm and (c) free optimization with the binary-coded genetic algorithm of the A-model of *Proasellus meridianus*.

used in the domain of biological water quality assessment. Therefore, the models obtained with bounded optimization are considered to have a better performance than those obtained with free optimization, even if higher accuracies are obtained for the latter.

6.4 Conclusion

In this chapter the optimization of the membership functions of the input variables of the habitat suitability models obtained by the knowledge-based design process described in Chapter 5, was discussed. One type of interpretability-preserving data-driven optimization, as well as an accuracy-oriented optimization, were applied using both a binary-coded and a real-coded genetic algorithm. As fitness function the average deviation (AD), a new performance measure for fuzzy ordered classification, was used.

For four models the binary-coded genetic algorithms returned less accurate solutions for the accuracy-oriented optimization than for the constrained optimization, due to the fact that the optimized membership function parameters only take values from a limited set of values. A shortcoming which, as shown by the experiments, can be remedied by applying a real-valued representation instead of a binary representation. The real-coded genetic algorithms applied in this study, however, showed maladjusted to eight of the 96 addressed membership function optimization problems, as an exhaustive investigation of the control structures of the genetic algorithms was outside the scope of this study.

A purely accuracy-oriented optimization is no option when one wants to preserve the interpretability of the habitat suitability models under study with the EKOO data set. In this case, expert knowledge is a prerequisite to build interpretable models in order to define the rule bases and determine the optimization intervals of the membership function parameters. The accuracy-oriented optimization, however, gives a better insight in the driving force during the bounded optimization, *i.e.* the tendency to classify as much data points as possible in the abundance class $absent$ by increasing the regions where the input is mapped to $absent$, and stresses the importance of uniformly distributed and unambiguous training data for model optimization.

Part III

Monotone models

CHAPTER 7

Monotonicity of linguistic fuzzy models

The worthwhile problems are the ones you can really solve or help solve, the ones you can really contribute something to.
(*Letter to Koichi Mano,* Richard Feynman, 1966)

7.1 Introduction

When identifying models of real-world systems, one is often confronted with a small number of data points. In such cases it is very important to fully exploit the additional non-quantitative knowledge about the system, in order to obtain meaningful, interpretable models (Carmona et al., 2005; Jin, 2003). Moreover, taking the qualitative knowledge about the system into account renders the model identification process less vulnerable to noise and inconsistencies in the data and suppresses overfitting (Sill, 1998). An example of this additional qualitative information is the monotonicity of the model output with respect to an input variable, *i.e.* the model output is either increasing or decreasing in the variable for all combinations of values of other input variables. Without loss of generality, in this study only increasing model outputs are considered and a fuzzy model is called monotone if it satisfies the following definition.

Definition 7.1

(i) *A fuzzy model is called monotone in an input variable X_l if for any two input vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ such that $x_{1,j} = x_{2,j}$ for any $j \in L \setminus \{l\}$ and $x_{1,l} \leq x_{2,l}$ it holds that $y^*(\mathbf{x}_1) \leq y^*(\mathbf{x}_2)$.*

(ii) *A fuzzy model is called monotone if it is monotone in each input variable.*

Monotonicity is a common property of evaluation and selection procedures. In loan acceptance for instance, the decision rule should be monotone with respect to income, *i.e.* it would be an unacceptable policy that a high-income applicant is rejected, whereas a low-income applicant with otherwise equal characteristics is accepted. In

Section 7.5 four (potential) applications of monotone linguistic fuzzy models in the bioscience engineering field are described.

This work focusses on linguistic fuzzy models as their framework allows for the design of interpretable models for non-experts. The monotonicity of Takagi–Sugeno models is discussed in detail in the work by Koo et al. (2004) and Won et al. (2002), while the work by Schott and Whalen (1996) addresses the influence of the height of the overlap between triangular membership functions on the monotonicity of the input-output behaviour of Mamdani–Assilian models. The design of monotone models has also been investigated for other modelling techniques, such as neural networks (Sill, 1998), decision trees (Ben-David, 1995; Cao-Van and De Baets, 2003; Daniels and Velikova, 2006) and instance-based classification techniques (Lievens et al., in press).

In Chapters 8–10 the monotonicity of linguistic fuzzy models under different inference procedures is discussed. The properties assumed to hold for the linguistic fuzzy models are described in Section 7.2. Section 7.3 deals with the representation used in Chapters 8–10 of if-then rules fired by a given input vector. In Section 7.4 the issue of incomparable fuzzy model outputs is addressed. Finally, this chapter concludes in Section 7.6 with an overview of the objectives of the work described in Chapters 8–10.

7.2 Assumed model properties

The investigated linguistic fuzzy models have m input variables X_l ($l \in L = \{1, \ldots, m\}$) and a single output variable Y. Their rule base contains r rules of the form

$$R_s\text{: IF } X_1 \text{ IS } B^1_{j_{1,s}} \text{ AND } \ldots \text{ AND } X_m \text{ IS } B^m_{j_{m,s}} \text{ THEN } Y \text{ IS } A_{i_s}$$

where $B^l_{j_{l,s}}$ (resp. A_{i_s}) are linguistic values of variable X_l (resp. Y) in the domain $\mathbf{X}_l$ (resp. $\mathbf{Y}$) ($s \in S = \{1, \ldots, r\}$). The input vector is denoted by $\mathbf{x} = (x_1, \ldots, x_m)$.

7.2.1 Linguistic values

The linguistic values A_{i_s} in the consequents of the r rules are selected among n membership functions A_i ($i \in I = \{1, \ldots, n\}$). These membership functions have a trapezial shape, form a fuzzy partition (Eq. (2.3)) and are characterized by $2n$ parameters as shown in Fig. 7.1. The extreme membership functions have one vertical side at respectively the lower bound lb_{output} or the upper bound ub_{output} of the output domain. The midpoints c_i and o_i and the lengths k_i and l_i of the kernel of the membership function A_i, respectively the interval where the membership functions A_i and A_{i+1} overlap, are given by ($l_0 = l_n = 0$),

$$\begin{aligned} c_i &= \frac{1}{2}(a_{2i-1} + a_{2i}) \quad & k_i &= a_{2i} - a_{2i-1} \qquad & &\text{, for all } i \in I\,, \\ o_i &= \frac{1}{2}(a_{2i} + a_{2i+1}) \quad & l_i &= a_{2i+1} - a_{2i} \qquad & &\text{, for all } i \in I \setminus \{n\}\,. \end{aligned} \tag{7.1}$$

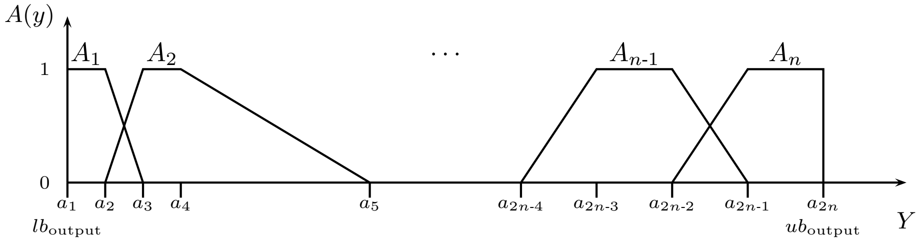


Figure 7.1: Parameters used to characterize the output membership functions forming a fuzzy partition.

The linguistic values $B^l_{j_{l,s}}$ of variable X_l ($l \in L$) in the antecedents of the r rules are selected among n_l membership functions $B^l_{j_l}$ ($j_l \in J_l = \{1, \ldots, n_l\}$). The membership functions in the m input domains are also assumed to be trapezial and to form a fuzzy partition.

The use of a fuzzy partition of trapezial membership functions imposes a natural order on the linguistic values of a variable as the intersection of the kernels of any pair of membership functions is empty. A linguistic value A_a is then said to be smaller than a linguistic value A_b in the same fuzzy partition if the upper bound of the kernel of A_a is smaller than the lower bound of the kernel of A_b

$$A_a \text{ is smaller than } A_b \Leftrightarrow \max(\text{kern}(A_a)) < \min(\text{kern}(A_b))\,. \tag{7.2}$$

In the setting of this work, a linguistic value A_a (resp. B^l_a) is smaller than a linguistic value A_b (resp. B^l_b) if and only if the index a is smaller than b

$$A_a \text{ is smaller than } A_b \Leftrightarrow a < b\,, \tag{7.3}$$

$$B^l_a \text{ is smaller than } B^l_b \Leftrightarrow a < b\,. \tag{7.4}$$

7.2.2 Rule base

The rule base $\mathcal{R}$ is assumed to be either complete, consistent and monotone or complete, consistent, smooth and monotone. Completeness and consistency are commonly required properties of rule bases in fuzzy models (Cordón et al., 2001). A model has a *complete* rule base if for any input vector $\mathbf{x}$ at least one rule is fired. When using fuzzy partitions of trapezial membership functions in all input domains, the rule base is complete if and only if it contains a rule for each combination $(j_1, \ldots, j_m) \in J_1 \times \ldots \times J_m$ of linguistic values of the m input variables. A set of IF-THEN rules is *consistent* if it does not contain contradictory rules. This concept is clear when using classical logical rules but is more difficult to grasp in the case of fuzzy rule bases. Therefore, there are many different interpretations of this property (Driankov et al., 1993). In this dissertation the strictest definition is adopted: a fuzzy rule base is said to be inconsistent if it contains at least two rules with the same antecedent but a different consequent. Therefore in the models considered in Chapters 8–10 the rule base contains exactly one rule for each combination $(j_1, \ldots, j_m) \in J_1 \times \ldots \times J_m$ and the number of rules r is equal to the product of the number of linguistic values assigned to the m input variables

$$r = \prod_{l=1}^{m} n_l \,. \tag{7.5}$$

In the definitions below a monotone rule base is defined for models whose model output is expected to increase with increasing model input.

Definition 7.2 *A rule base $\mathcal{R} = \{R_1, \ldots, R_r\}$ is called monotone if for any two rules R_{s_1} and R_{s_2} it holds that $(j_{1,s_1}, \ldots, j_{m,s_1}) \leq (j_{1,s_2}, \ldots, j_{m,s_2})$ implies $i_{s_1} \leq i_{s_2}$.*

Proposition 7.1 *If a rule base $\mathcal{R}$ is complete and consistent, then it is monotone if and only if for any $(j_1, \ldots, j_m) \in J_1 \times \ldots \times J_m$ and any $l \in \{1, \ldots, m\}$ such that $j_l < n_l$ it holds that $i_{s_1} \leq i_{s_2}$, with $(j_{1,s_1}, \ldots, j_{m,s_1}) = (j_1, \ldots, j_m)$ and $(j_{1,s_2}, \ldots, j_{m,s_2}) = (j_1, \ldots, j_{l\text{-}1}, j_l{+}1, j_{l+1}, \ldots, j_m)$.*

Definition 7.3 *A complete consistent rule base $\mathcal{R}$ is called smooth if for any $(j_1, \ldots, j_m) \in J_1 \times \ldots \times J_m$ and any $l \in \{1, \ldots, m\}$ such that $j_l < n_l$ it holds that $i_{s_2} = i_{s_1}{+}p$, with $p \in \{-1, 0, 1\}$ and $(j_{1,s_1}, \ldots, j_{m,s_1}) = (j_1, \ldots, j_m)$ and $(j_{1,s_2}, \ldots, j_{m,s_2}) = (j_1, \ldots, j_{l\text{-}1}, j_l{+}1, j_{l+1}, \ldots, j_m)$.*

Corollary 7.1 *A smooth complete consistent rule base is monotone if and only if, using the notations of Definition 7.3, it always holds that $p \in \{0, 1\}$.*

7.2.3 Inference procedure

In Chapters 8–9 the monotonicity of models applying Mamdani–Assilian inference is investigated, whereas in Chapter 10 the monotonicity of models applying either plain implicator-based inference or ATL–ATM inference is discussed.

Mamdani–Assilian inference The procedure applied in Mamdani–Assilian models to determine the model output is described in detail in Section 2.3.2. First the fulfilment degrees β_s of the r rules R_s are computed. In a next step the fulfilment degrees α_i of the n linguistic output values A_i are determined and used to define the membership functions of the adapted membership functions A'_i. The global fuzzy output A is the union, based on the maximum, of the n adapted membership functions A'_i. Finally, the crisp model output y^* is obtained by defuzzifying the fuzzy output A. In this study the three most commonly applied t-norms are considered: the minimum $T_{\mathbf{M}}$, the product $T_{\mathbf{P}}$ and the Łukasiewicz t-norm $T_{\mathbf{L}}$. In Chapter 8, models applying the Center of Gravity (COG) defuzzification method are discussed, whereas Chapter 9 deals with models applying the Mean of Maxima (MOM) defuzzification method.

Plain implicator-based inference In Section 10.2 the monotonicity is discussed of models applying implicator-based inference as described in detail in Section 2.3.3. This inference procedure will be referred to as *plain* implicator-based inference in order to avoid confusion with the second implicator-based inference procedure considered in this dissertation, *i.e.* the ATL–ATM inference described in the following paragraph. As in the Mamdani–Assilian inference procedure first the fulfilment degrees β_s of the r rules R_s are computed and afterwards the fulfilment degrees α_i of the n linguistic output values A_i are determined. However, in implicator-based inference procedures, the adapted membership functions A'_i are computed using an implicator instead of a t-norm. In this study the three R-implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$ are considered. The global fuzzy output A is the intersection, based on the minimum, of the n adapted membership functions A'_i. In this work no specific defuzzification method is considered for models applying plain implicator-based inference.

ATL–ATM inference In this dissertation a new inference procedure for linguistic fuzzy models with a monotone rule base is introduced. It is an implicator-based inference procedure in which the modifiers 'at least' (ATL) and 'at most' (ATM) defined in Eqs. (2.25–2.26) play an important part, hence the name ATL–ATM inference. In ATL–ATM models, as the linguistic fuzzy models applying the newly introduced inference procedure are called, the fuzzy model output A is the intersection, based on the minimum, of the fuzzy model outputs A_{ATL} and A_{ATM} of an ATL model and an ATM model

$$A(y) = \min(A_{\mathrm{ATL}}(y), A_{\mathrm{ATM}}(y))\,. \tag{7.6}$$

The ATL and ATM models are derived from a linguistic fuzzy model as defined in Sections 7.2.1–7.2.2. For each rule in the rule base of this linguistic fuzzy model, the rule base of the ATL model contains a corresponding rule obtained by applying the modifier ATL to all linguistic values in the original rule

$$\begin{array}{ll} R_s\text{: IF} & X_1 \text{ IS } \mathrm{ATL}(B^1_{j_{1,s}}) \text{ AND } \ldots \text{ AND } X_m \text{ IS } \mathrm{ATL}(B^m_{j_{m,s}}) \\ \text{THEN} & Y \text{ IS } \mathrm{ATL}(A_{i_s}) \end{array}$$

and the rule base of the ATM model contains a corresponding rule obtained by applying the modifier ATM to all linguistic values of the original rule

$$\begin{array}{ll} R_s\text{: IF} & X_1 \text{ IS } \mathrm{ATM}(B^1_{j_{1,s}}) \text{ AND } \ldots \text{ AND } X_m \text{ IS } \mathrm{ATM}(B^m_{j_{m,s}}) \\ \text{THEN} & Y \text{ IS } \mathrm{ATM}(A_{i_s}) \end{array}$$

The linguistic values $B^l_{j_l}$ and A_i are defined by the same membership functions as in the linguistic fuzzy model. The fuzzy model output of the ATL and ATM model are obtained by implicator-based inference as described in Section 2.3.3. In Chapter 10, the monotonicity of ATL–ATM models applying the Mean of Maxima (MOM) defuzzification method is discussed.

7.3 Rules determining the fuzzy model output

7.3.1 Models applying Mamdani–Assilian inference or plain implicator-based inference

When defining the linguistic values of a variable by membership functions as described in Fig. 7.1, a given crisp value partially belongs to at most two linguistic values. As these fuzzy partitions are used in all input domains, a crisp input x_l either completely belongs to one linguistic value, *i.e.*

$$(\exists! j_1 \in J_l)(B^l_{j_1}(x_l) = 1 \wedge (\forall j_2 \in J_l \setminus \{j_1\})(B^l_{j_2}(x_l) = 0))\,,$$

or partially belongs to two adjacent linguistic values, *i.e.*

$$\begin{aligned}(\exists! j_1 \in J_l \setminus \{j_{n_l}\})(B^l_{j_1}(x_l) \in\ ]0,1[\ \wedge\ B^l_{j_1+1}(x_l) = 1 - B^l_{j_1}(x_l)\\ \wedge (\forall j_2 \in J_l \setminus \{j_1, j_1{+}1\})(B^l_{j_2}(x_l) = 0))\,.\end{aligned} \tag{7.7}$$

As a consequence, for a given input vector $\mathbf{x}$ at most 2^m rules are fired, *i.e.* at most 2^m rules R_s have a non-zero fulfilment degree β_s. All inputs belonging, in all input domains, to the kernel of the same linguistic value $B^l_{j_l}$, are always mapped to the same (fuzzy) model output. Therefore, a model always shows a monotone input-output behaviour within these parts of the input space. In order to obtain a monotone input-output behaviour for all input vectors $\mathbf{x}$, a monotone input-output behaviour should also be obtained in all regions of the input space corresponding to the intersections of the supports of the respective input membership functions $B^l_{j_l}$ and $B^l_{j_l+1}$ defined in Eq. (7.7). In order to avoid an overloaded notation, the variable γ_l is introduced

$$\gamma_l = 1 - B^l_{j_l}(x_l) = B^l_{j_l+1}(x_l)\,. \tag{7.8}$$

7.3.1.1 Models with a single input variable

For models with a single input variable, monotonicity is guaranteed for any monotone *smooth* rule base if in any interval $[b_{2j_1}, b_{2j_1+1}]$ ($j_1 \in J_1 \setminus \{j_{n_1}\}$) of the input domain, with b_{2j_1} the upper bound of the kernel of a linguistic value $B^1_{j_1}$ and b_{2j_1+1} the lower bound of the kernel of the next linguistic value $B^1_{j_1+1}$, a monotone input-output behaviour is obtained not only if the linguistic values $B^1_{j_1}$ and $B^1_{j_1+1}$ are mapped to a same linguistic output value A_i

IF	X_1 IS $B^1_{j_1}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	THEN	Y IS A_i

but also if the linguistic values $B^1_{j_1}$ and $B^1_{j_1+1}$ are mapped to two consecutive linguistic output values A_i and A_{i+1}

IF	X_1 IS $B^1_{j_1}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	THEN	Y IS A_{i+1}

For models with a single input variable, monotonicity is guaranteed for any monotone rule base if a monotone input-output behaviour is obtained in the two cases above and in the case when linguistic values $B^1_{j_1}$ and $B^1_{j_1+1}$ are mapped to two non-consecutive output values A_i and A_{i+p} ($p \in \mathbb{N}$, $p > 1$).

Thus, given the general representation of the rules containing a linguistic value $B^1_{j_1}$ and the subsequent linguistic value $B^1_{j_1+1}$ in their antecedent, *i.e.*

$$\begin{array}{llll}
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{THEN} & Y \text{ IS } A_i \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{THEN} & Y \text{ IS } A_{i+p}
\end{array}$$

the values to be considered for p are 0 and 1 when investigating the monotonicity of models with a monotone smooth rule base. The investigation of the monotonicity of models with a monotone rule base also requires considering values of p larger than 1.

7.3.1.2 Models with two input variables

For models with two input variables, the set of four rules that might be fired by a given input vector $\mathbf{x} = (x_1, x_2)$ and whose consequents might therefore contribute to the model output can be represented as

$$\begin{array}{llllll}
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{THEN} & Y \text{ IS } A_i \\
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{THEN} & Y \text{ IS } A_{i+p''_2} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{THEN} & Y \text{ IS } A_{i+p''_1} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{THEN} & Y \text{ IS } A_{i+p''_3}
\end{array}$$

with $p''_1, p''_2, p''_3 \in \mathbb{N}$ and, in order to obtain a monotone rule base, $p''_1 \le p''_3$ and $p''_2 \le p''_3$. As monotonicity of a model requires monotonicity in all of its input variables, if monotonicity is guaranteed for $p''_1 \le p''_2$ it follows by permutation of X_1 and X_2 that monotonicity is also guaranteed for $p''_1 \ge p''_2$. Thus, the investigation of the monotonicity of models with two input variables only requires the verification of the monotonicity of all situations included in the following general representation

$$\begin{array}{llllll}
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{THEN} & Y \text{ IS } A_i \\
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{THEN} & Y \text{ IS } A_{i+p'_1+p'_2} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{THEN} & Y \text{ IS } A_{i+p'_1} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{THEN} & Y \text{ IS } A_{i+p'_3}
\end{array}$$

with $p'_1, p'_2, p'_3 \in \mathbb{N}$ and, in order to obtain a monotone rule base, $p'_1 + p'_2 \le p'_3$. In Chapters 8–9 the following, more straightforward representation will be used, incorporating all constraints the indices of the output membership functions should satisfy

$$\begin{array}{llllll}
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{THEN} & Y \text{ IS } A_i \\
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{THEN} & Y \text{ IS } A_{i+p_1} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_3}
\end{array}$$

with $p_1, p_2, p_3 \in \mathbb{N}$. The four rules are represented schematically in Fig. 7.2.

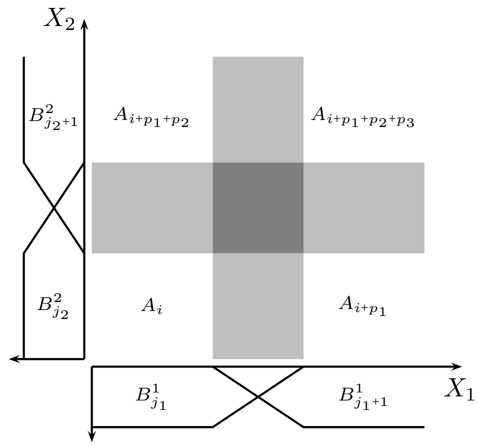


Figure 7.2: General representation of the rules fired for a model with two input variables.

When the rule base is also smooth, the values of p_1, p_2 and p_3 in the rules above are restricted to

$$(p_1, p_2, p_3) \in \{(0,0,0), (0,0,1), (0,1,0), (1,0,0), (1,0,1)\}\,. \tag{7.9}$$

When investigating the monotonicity of models with a monotone rule base, eight situations where p_1, p_2 and p_3 are either equal to zero or strictly positive, should be considered when applying a defuzzification method, like for instance the MOM defuzzification method, which allows for the application of the same procedure in the occurrence and absence of consecutive linguistic values among the fired output values. When applying a defuzzification method, like for instance the COG defuzzification method, requiring a different procedure whether or not there are consecutive linguistic output values among the fired output values, 27 situations should be considered where p_1, p_2 and p_3 are either equal to zero, equal to 1 or larger than 1.

7.3.1.3 Models with three or more input variables

For models with three input variables the set of eight rules that might be fired to a nonzero fulfilment degree and whose consequents might therefore contribute to the model output can be represented in a general way as

$$
\begin{array}{llllllll}
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{AND} & X_3 \text{ IS } B^3_{j_3} & \text{THEN} & Y \text{ IS } A_i \\
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{AND} & X_3 \text{ IS } B^3_{j_3+1} & \text{THEN} & Y \text{ IS } A_{i+p''_4} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{AND} & X_3 \text{ IS } B^3_{j_3} & \text{THEN} & Y \text{ IS } A_{i+p''_2} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{AND} & X_3 \text{ IS } B^3_{j_3+1} & \text{THEN} & Y \text{ IS } A_{i+p''_6} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{AND} & X_3 \text{ IS } B^3_{j_3} & \text{THEN} & Y \text{ IS } A_{i+p''_1} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{AND} & X_3 \text{ IS } B^3_{j_3+1} & \text{THEN} & Y \text{ IS } A_{i+p''_5} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{AND} & X_3 \text{ IS } B^3_{j_3} & \text{THEN} & Y \text{ IS } A_{i+p''_3} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{AND} & X_3 \text{ IS } B^3_{j_3+1} & \text{THEN} & Y \text{ IS } A_{i+p''_7}
\end{array}
$$

with $p''_1, p''_2, p''_3, p''_4, p''_5, p''_6, p''_7 \in \mathbb{N}$ and, in order to obtain a monotone rule base, $p''_1 \leq p''_3$, $p''_2 \leq p''_3$, $p''_1 \leq p''_5$, $p''_4 \leq p''_5$, $p''_2 \leq p''_6$, $p''_4 \leq p''_6$, $p''_3 \leq p''_7$, $p''_5 \leq p''_7$ and $p''_6 \leq p''_7$.

Any set of eight rules originating from a monotone rule base either satisfies, or can, by permuting the input variables, be converted into eight rules satisfying

$$p''_1 \leq p''_2 \leq p''_3 \,. \tag{7.10}$$

Therefore, the representation shown in Fig. 7.3 will be used when investigating the monotonicity of models with three input variables

$$
\begin{array}{lllllll}
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND } X_2 \text{ IS } B^2_{j_2} & \text{AND } X_3 \text{ IS } B^3_{j_3} & \text{THEN} & Y \text{ IS } A_i \\
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND } X_2 \text{ IS } B^2_{j_2} & \text{AND } X_3 \text{ IS } B^3_{j_3+1} & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_3} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND } X_2 \text{ IS } B^2_{j_2+1} & \text{AND } X_3 \text{ IS } B^3_{j_3} & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND } X_2 \text{ IS } B^2_{j_2+1} & \text{AND } X_3 \text{ IS } B^3_{j_3+1} & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_3+p_5} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND } X_2 \text{ IS } B^2_{j_2} & \text{AND } X_3 \text{ IS } B^3_{j_3} & \text{THEN} & Y \text{ IS } A_{i+p_1} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND } X_2 \text{ IS } B^2_{j_2} & \text{AND } X_3 \text{ IS } B^3_{j_3+1} & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_3+p_6} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND } X_2 \text{ IS } B^2_{j_2+1} & \text{AND } X_3 \text{ IS } B^3_{j_3} & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_4} \\
\text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND } X_2 \text{ IS } B^2_{j_2+1} & \text{AND } X_3 \text{ IS } B^3_{j_3+1} & \text{THEN} & Y \text{ IS } A_{i+p'_7}
\end{array}
$$

with $p_1, p_2, p_3, p_4, p_5, p_6, p_7 \in \mathbb{N}$ and $p'_7 = p_1 + p_2 + \max(p_4, p_3 + p_5, p_3 + p_6) + p_7$.

In a *smooth* rule base the parameters p_i are either zero or one and satisfy the following inequalities

$$p_1 + p_2 + p_3 \leq 1 \,, \tag{7.11}$$

$$p_3 + p_5 \leq 1 \,, \tag{7.12}$$

$$p_2 + p_3 + p_6 \leq 1 \,, \tag{7.13}$$

$$p_2 + p_4 \leq 1 \,, \tag{7.14}$$

$$\max(p_4, p_3 + p_5, p_3 + p_6) + p_7 - p_3 - p_5 \leq 1 \,, \tag{7.15}$$

$$\max(p_4, p_3 + p_5, p_3 + p_6) + p_7 - p_3 - p_6 \leq 1 \,, \tag{7.16}$$

$$\max(p_4, p_3 + p_5, p_3 + p_6) + p_7 - p_4 \leq 1 \,. \tag{7.17}$$

There exists 26 vectors $(p_1, \ldots, p_7)$ satisfying Eqs. (7.11–7.17). Among them four sets of three vectors correspond to rule bases that are identical after permuting input variables. Of each of these sets only one vector is selected, which reduces the number of rule bases that should be investigated to 18. The 18 vectors $(p_1, \ldots, p_7)$ are listed in Table 7.1.

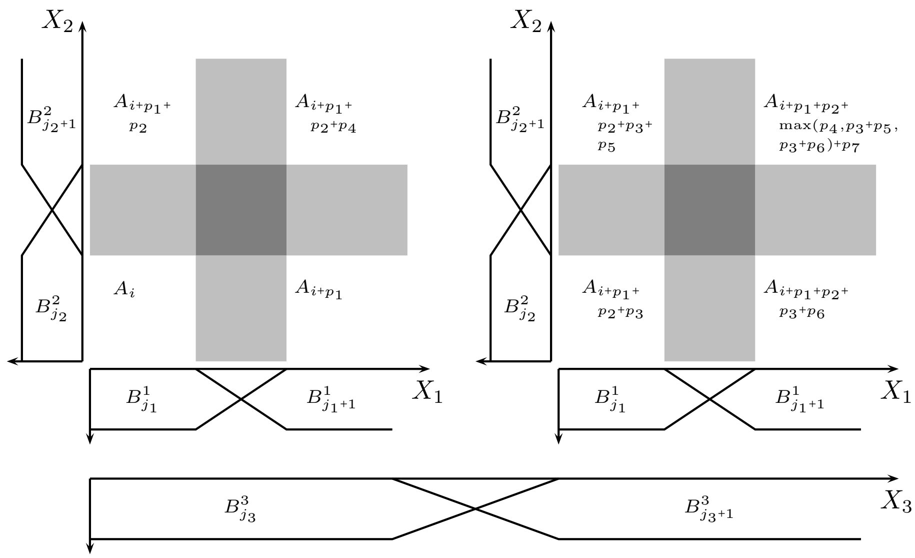


Figure 7.3: General representation of the rules fired for a model with three input variables.

General representations of the rules fired for models with three input variables and a monotone, but non-smooth rule base, or models with more than three input variables, are not included in this section as these models are not explicitly discussed in Chapters 8–9.

7.3.2 ATL–ATM models

In this section it is shown that the fuzzy output of an ATL model (resp. ATM model) for a given input vector $\mathbf{x}$ is determined by the rules derived from the rules fired by the input vector $\mathbf{x}$ under consideration in case of Mamdani–Assilian or plain implicator-based inference, even more rules of the ATL model (resp. ATM model) are fired than of the corresponding linguistic model when applying Mamdani–Assilian or plain implicator-based inference.

7.3.2.1 Models with a single input variable

As illustrated in Fig 7.4, an input x_1 of a model with a single input variable, with

$$\gamma_1 = 1 - B^1_{j_1}(x_1) = B^1_{j_1+1}(x_1)\,, \tag{7.18}$$

Table 7.1: Combinations of values that should be considered for the parameters $p_i (i \in \{1, \ldots, 7\})$ in Fig. 7.3 when investigating the monotonicity of models with three input variables and a monotone smooth rule base.

Case	p_1	p_2	p_3	p_4	p_5	p_6	p_7
I	0	0	0	0	0	0	0
II	0	0	0	0	0	0	1
III	0	0	0	0	1	1	0
IV	0	0	0	1	0	0	0
V	0	0	0	1	1	1	0
VI	0	0	0	1	1	1	1
VII	0	0	1	0	0	0	0
VIII	0	0	1	1	0	0	0
IX	0	0	1	1	0	0	1
X	0	1	0	0	0	0	0
XI	0	1	0	0	0	0	1
XII	0	1	0	0	1	0	0
XIII	1	0	0	0	0	0	0
XIV	1	0	0	0	0	0	1
XV	1	0	0	0	1	1	0
XVI	1	0	0	1	0	0	0
XVII	1	0	0	1	1	1	0
XVIII	1	0	0	1	1	1	1

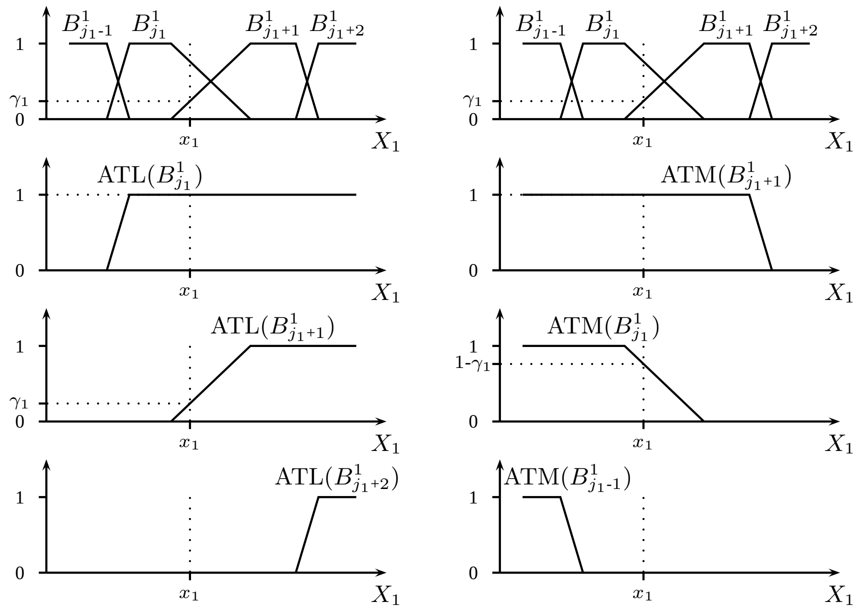


Figure 7.4: Membership degrees of an input x_1 to the linguistic values in the rule antecedents of an ATL and an ATM model.

has the following membership degrees to the linguistic values in the antecedents of the rules of the ATL and ATM model:

$$\mathrm{ATL}(B^1_j)(x_1) = \begin{cases} 1 & \text{, if } j \leq j_1 \,, \\ \gamma_1 & \text{, if } j = j_1 + 1 \,, \\ 0 & \text{, if } j > j_1 + 1 \,, \end{cases} \tag{7.19}$$

$$\mathrm{ATM}(B^1_j)(x_1) = \begin{cases} 0 & \text{, if } j < j_1 \,, \\ 1 - \gamma_1 & \text{, if } j = j_1 \,, \\ 1 & \text{, if } j \geq j_1 + 1 \,. \end{cases} \tag{7.20}$$

Since in a model with a single input variable the fulfilment degree of a rule is identical to the membership degree of the input x_1 to the linguistic value in the rule's antecedent, an input x_1 fires j_1 rules in the ATL model, *i.e.* the rules containing the linguistic values $\mathrm{ATL}(B^1_1)$ to $\mathrm{ATL}(B^1_{j_1})$ in their antecedent, if $\gamma_1 = 0$; and fires $j_1 + 1$ rules, *i.e.* the rules containing the linguistic values $\mathrm{ATL}(B^1_1)$ to $\mathrm{ATL}(B^1_{j_1+1})$ in their antecedent, if $\gamma_1 > 0$. In the ATM model it fires $n_1 - j_1$ rules, *i.e.* the rules containing the linguistic values $\mathrm{ATM}(B^1_{j_1+1})$ to $\mathrm{ATM}(B^1_{n_1})$ in their antecedent, if $\gamma_1 = 1$; and $n_1 - j_1 + 1$ rules, *i.e.* the rules containing the linguistic values $\mathrm{ATM}(B^1_{j_1})$ to $\mathrm{ATM}(B^1_{n_1})$ in their antecedent, if $\gamma_1 < 1$.

The fuzzy output of the ATL model is the intersection of the individual adapted output membership functions of the r rules, *i.e.*

$$\begin{aligned} A_{\mathrm{ATL}}(y) &= \min_{s=1}^{r} (\mathrm{ATL}(A_{i_s}))'(y) \\ &= \min_{s=1}^{r} I_T(\beta_s, \mathrm{ATL}(A_{i_s})(y)) \\ &= \min_{s=1}^{r} I_T(\mathrm{ATL}(B^1_{j_{1,s}})(x_1), \mathrm{ATL}(A_{i_s})(y))\,. \end{aligned} \tag{7.21}$$

In the following paragraphs the linguistic output values in the consequents of the rules of the ATL model containing respectively $\mathrm{ATL}(B^1_{j_1})$ and $\mathrm{ATL}(B^1_{j_1+1})$ in their antecedent will be noted by $\mathrm{ATL}(A_i)$ and $\mathrm{ATL}(A_{i+p})$, *i.e.*

IF	X_1 IS $\mathrm{ATL}(B^1_{j_1})$	THEN	Y IS $\mathrm{ATL}(A_i)$
IF	X_1 IS $\mathrm{ATL}(B^1_{j_1+1})$	THEN	Y IS $\mathrm{ATL}(A_{i+p})$

The fulfilment degrees of the linguistic output values $\mathrm{ATL}(A_i)$ and $\mathrm{ATL}(A_{i+p})$ are equal to 1 and γ_1 respectively. Two groups can be distinguished among the $r-2$ other rules: rules containing in their antecedent a linguistic value smaller than $B^1_{j_1}$ to which the modifier ATL is applied and which is fired to a fulfilment degree equal to 1, *i.e.*

$$S_1 = \{s \in S \mid j_{1,s} < j_1\}\,, \tag{7.22}$$

and rules containing in their antecedent a linguistic value larger than $B^1_{j_1+1}$ to which the modifier ATL is applied and which are not fired, *i.e.*

$$S_2 = \{s \in S \mid j_{1,s} > j_1 + 1\}\,. \tag{7.23}$$

When describing the linguistic output values A_i as defined in Section 7.2.1, for all output values the membership degree to $\mathrm{ATL}(A_{i'})$ is larger than or equal the membership degree to $\mathrm{ATL}(A_{i''})$ when $A_{i'}$ is smaller than $A_{i''}$, *i.e.*

$$(\forall y \in \mathbf{Y})(\forall i', i'' \in I)(i' < i'' \Rightarrow \mathrm{ATL}(A_{i'})(y) \geq \mathrm{ATL}(A_{i''})(y))\,. \tag{7.24}$$

Furthermore, implicators have increasing second partial functions (Eq. (2.33)). Thus, the membership degree to the adapted linguistic value $(\mathrm{ATL}(A_{i'}))'$ is larger than or equal to the membership degree to the adapted linguistic value $(\mathrm{ATL}(A_{i''}))'$ for all output values if $A_{i'}$ is smaller than $A_{i''}$ and if they are fired to the same fulfilment degree, *i.e.*

$$(\forall y \in \mathbf{Y})(\forall i', i'' \in I)(i' < i'' \Rightarrow I_T(\alpha, \mathrm{ATL}(A_{i'})(y)) \geq I_T(\alpha, \mathrm{ATL}(A_{i''})(y)))\,. \tag{7.25}$$

As the rule base from which the rule base of the ATL model is derived is monotone, the linguistic output values in the consequents of the first group of rules are linguistic output values smaller than or equal to A_i, to which the modifier ATL is applied, *i.e.*

$$(\forall s \in S_1)(i_s \leq i)\,, \tag{7.26}$$

and as the fulfilment degrees of the first group of rules are all equal to 1 and thus equal to the fulfilment degree of the linguistic output value $\mathrm{ATL}(A_i)$, *i.e.*

$$(\forall s \in S_1)(\beta_s = \alpha_{\mathrm{ATL},i} = 1)\,, \tag{7.27}$$

it follows from Eq. (7.25) that the minimum of the membership degree of an output value to the adapted linguistic output value of a rule from the first group and its membership degree to the adapted linguistic value $(\mathrm{ATL}(A_{i+p}))'$ is given by the membership degree to the latter, *i.e.*

$$(\forall s \in S_1)(\forall y \in \mathbf{Y})(I_T(1, \mathrm{ATL}(A_{i_s})(y)) \geq I_T(1, \mathrm{ATL}(A_i)(y)))\,. \tag{7.28}$$

As the rules of the second group are not fired, their contributions to the global fuzzy output are identical to the universal set, *i.e.*

$$(\forall s \in S_2)(\forall y \in \mathbf{Y})((\mathrm{ATL}(A_{i_s}))'(y) = 1)\,. \tag{7.29}$$

From Eqs. (7.28–7.29) and the fact that for the three considered implicators I_T it holds that

$$(\forall\, x \in [0,1])(I_T(1,x) = x)\,, \tag{7.30}$$

a property known as the neutrality principle, it follows that the fuzzy output of the ATL model is given by the intersection of the original linguistic value $\mathrm{ATL}(A_i)$ and the adapted linguistic value $(\mathrm{ATL}(A_{i+p}))'$, *i.e.*

$$\begin{aligned} A_{\mathrm{ATL}}(y) &= \min(\min_{s\in S_1}(\mathrm{ATL}(A_{i_s}))'(y), (\mathrm{ATL}(A_i))'(y), (\mathrm{ATL}(A_{i+p}))'(y), \\ &\qquad \min_{s\in S_2}(\mathrm{ATL}(A_{i_s}))'(y)) \\ &= \min(I_T(1, \mathrm{ATL}(A_i)(y)), I_T(\gamma_1, \mathrm{ATL}(A_{i+p})(y))) \\ &= \min(\mathrm{ATL}(A_i)(y), I_T(\gamma_1, \mathrm{ATL}(A_{i+p})(y)))\,. \end{aligned} \tag{7.31}$$

Analogously, one can show that the fuzzy output of the ATM model is given by the intersection of the adapted linguistic value $(\mathrm{ATM}(A_i))'$ and the original linguistic value $\mathrm{ATM}(A_{i+p})$, *i.e.*

$$\begin{aligned} A_{\mathrm{ATM}}(y) &= \min_{s=1}^{r}(\mathrm{ATM}(A_{i_s}))'(y) \\ &= \min(I_T(1-\gamma_1, \mathrm{ATM}(A_i)(y)), \mathrm{ATM}(A_{i+p})(y))\,. \end{aligned} \tag{7.32}$$

Thus, the rules that should be considered when determining the fuzzy output of an ATL–ATM model with a single input value are the rules derived from

IF	X_1 IS $B^1_{j_1}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	THEN	Y IS A_{i+p}

i.e. the same rules that have to be considered when using Mamdani–Assilian or plain implicator-based inference to obtain the output of the linguistic fuzzy model from which the ATL and ATM models are derived.

7.3.2.2 Models with two input variables

For an input vector $\mathbf{x} = (x_1, x_2)$ there always exist j_1 and j_2 such that

$$\gamma_1 = 1 - B^1_{j_1}(x_1) = B^1_{j_1+1}(x_1)\,, \tag{7.33}$$

$$\gamma_2 = 1 - B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2)\,. \tag{7.34}$$

When applying Mamdani–Assilian or plain implicator-based inference to the linguistic fuzzy model from which the ATL and ATM models are derived, an input vector $\mathbf{x}$ fires at most four rules

IF X_1 IS $B^1_{j_1}$ AND X_2 IS $B^2_{j_2}$ THEN Y IS A_i
IF X_1 IS $B^1_{j_1}$ AND X_2 IS $B^2_{j_2+1}$ THEN Y IS $A_{i+p_1+p_2}$
IF X_1 IS $B^1_{j_1+1}$ AND X_2 IS $B^2_{j_2}$ THEN Y IS A_{i+p_1}
IF X_1 IS $B^1_{j_1+1}$ AND X_2 IS $B^2_{j_2+1}$ THEN Y IS $A_{i+p_1+p_2+p_3}$

As is shown below only the rules derived from these four rules need to be considered in order to determine the fuzzy output of the corresponding ATL and ATM models. The fuzzy outputs A_{ATL} and A_{ATM} of respectively the ATL and ATM models are obtained by

$$\begin{aligned} A_{\mathrm{ATL}}(y) = \min(&\mathrm{ATL}(A_i)(y), I_T(\gamma_2, \mathrm{ATL}(A_{i+p_1})(y)),\\ &I_T(\gamma_1, \mathrm{ATL}(A_{i+p_1+p_2})(y)), I_T(T(\gamma_1,\gamma_2), \mathrm{ATL}(A_{i+p_1+p_2+p_3})(y)))\,, \end{aligned} \tag{7.35}$$

$$\begin{aligned} A_{\mathrm{ATM}}(y) = \min(&I_T(T(1-\gamma_1, 1-\gamma_2), \mathrm{ATM}(A_i)(y)),\\ &I_T(1-\gamma_1, \mathrm{ATM}(A_{i+p_1})(y)), I_T(1-\gamma_2, \mathrm{ATM}(A_{i+p_1+p_2})(y)),\\ &\mathrm{ATM}(A_{i+p_1+p_2+p_3})(y))\,. \end{aligned} \tag{7.36}$$

In the rule base of the ATL model, apart from the rules derived from the four rules above

IF X_1 IS $\mathrm{ATL}(B^1_{j_1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2})$ THEN Y IS $\mathrm{ATL}(A_i)$
IF X_1 IS $\mathrm{ATL}(B^1_{j_1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2+1})$ THEN Y IS $\mathrm{ATL}(A_{i+p_1+p_2})$
IF X_1 IS $\mathrm{ATL}(B^1_{j_1+1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2})$ THEN Y IS $\mathrm{ATL}(A_{i+p_1})$
IF X_1 IS $\mathrm{ATL}(B^1_{j_1+1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2+1})$ THEN Y IS $\mathrm{ATL}(A_{i+p_1+p_2+p_3})$

four types of rules can be distinguished as illustrated in Fig. 7.5.

The first group of rules is derived from rules containing both for X_1 and X_2 linguistic values smaller than or equal to respectively $B^1_{j_1}$ and $B^2_{j_2}$, *i.e.*

$$S_1 = \{s \in S \mid j_{1,s} \le j_1 \wedge j_{2,s} \le j_2\} \setminus \{(j_1, j_2)\}\,. \tag{7.37}$$

These rules are fired to a fulfilment degree equal to 1 for any $s \in S_1$, *i.e.*

$$\beta_s = T(\mathrm{ATL}(B^1_{j_{1,s}})(x_1), \mathrm{ATL}(B^2_{j_{2,s}})(x_2)) = T(1,1) = 1\,. \tag{7.38}$$

As the rule

IF X_1 IS $\mathrm{ATL}(B^1_{j_1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2})$ THEN Y IS $\mathrm{ATL}(A_i)$

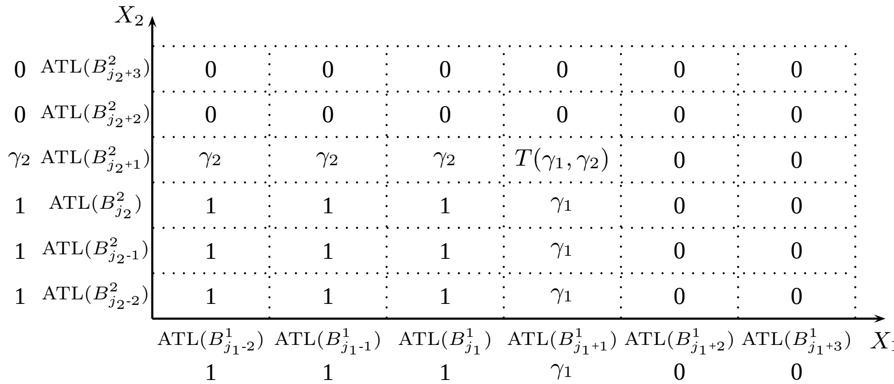


Figure 7.5: Fulfilment degrees obtained for the rules of an ATL model for an input vector $\mathbf{x}$ belonging to the indicated membership degrees to the linguistic values of X_1 and X_2.

is also fired to a fulfilment degree equal to 1, *i.e.*

$$\alpha_{\mathrm{ATL},i} = T(\mathrm{ATL}(B^1_{j_1})(x_1), \mathrm{ATL}(B^2_{j_2})(x_2)) = T(1,1) = 1\,, \tag{7.39}$$

and as the rule base from which the rule base of the ATL model is derived is monotone, *i.e.*

$$(\forall s \in S_1)(i_s \leq i)\,, \tag{7.40}$$

it follows from Eq. (7.25) that all output values have a smaller or equal membership degree to the adapted linguistic value $(\mathrm{ATL}(A_i))'$ than to the adapted linguistic output value of a rule of the first group of rules, *i.e.*

$$(\forall s \in S_1)(\forall y \in \mathbf{Y})((\mathrm{ATL}(A_{i_s}))'(y) \geq (\mathrm{ATL}(A_i))'(y))\,. \tag{7.41}$$

The second type of rules is derived from rules containing a linguistic value smaller than $B^1_{j_1}$ for X_1 and the linguistic value $B^2_{j_2+1}$ for X_2 in their antecedent, *i.e.*

$$S_2 = \{s \in S \mid j_{1,s} < j_1 \wedge j_{2,s} = j_2 + 1\}\,. \tag{7.42}$$

These rules are fired to the same fulfilment degree as the rule

IF X_1 IS $\mathrm{ATL}(B^1_{j_1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2+1})$ THEN Y IS $\mathrm{ATL}(A_{i+p_1+p_2})$

namely, to the degree γ_2

$$\alpha_{\mathrm{ATL},i+p_1+p_2} = T(\mathrm{ATL}(B^1_{j_1})(x_1), \mathrm{ATL}(B^2_{j_2+1})(x_2)) = T(1,\gamma_2) = \gamma_2\,. \tag{7.43}$$

As the rule base from which the rule base of the ATL model is derived is monotone, *i.e.*

$$(\forall s \in S_2)(i_s \leq i + p_1 + p_2)\,, \tag{7.44}$$

it follows from Eq. (7.25) that all output values have a smaller or equal membership degree to the adapted linguistic value $(\mathrm{ATL}(A_{i+p_1+p_2}))'$ than to the adapted linguistic output value of a rule of the second group of rules, *i.e.*

$$(\forall s \in S_2)(\forall y \in \mathbf{Y})((\mathrm{ATL}(A_{i_s}))'(y) \geq (\mathrm{ATL}(A_{i+p_1+p_2}))'(y)) \,. \tag{7.45}$$

The third group of rules is derived from rules containing either for X_1 or for X_2 (or for both) a linguistic value larger than respectively $B^1_{j_1+1}$ and $B^2_{j_2+1}$ in their antecedent, *i.e.*

$$S_3 = \{s \in S \mid j_{1,s} > j_1 + 1 \vee j_{2,s} > j_2 + 1\} \,. \tag{7.46}$$

These rules are not fired as the membership degree to at least one of the linguistic values in their antecedent is zero, *i.e.*

$$(\forall s \in S_3)(\beta_s = 0) \,, \tag{7.47}$$

and therefore do not determine the fuzzy output of the ATL model as the corresponding adapted output membership functions are identical to the universal set.

The fourth group of rules is derived from rules containing the linguistic value $B^1_{j_1+1}$ for X_1 and a linguistic value smaller than $B^2_{j_2}$ for X_2 in their antecedent, *i.e.*

$$S_4 = \{s \in S \mid j_{1,s} = j_1 + 1 \vee j_{2,s} < j_2\} \,. \tag{7.48}$$

These rules are fired to the same fulfilment degree as the rule

IF X_1 IS $\mathrm{ATL}(B^1_{j_1+1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2})$ THEN Y IS $\mathrm{ATL}(A_{i+p_1})$

namely, to the degree γ_1

$$\alpha_{\mathrm{ATL},i+p_1} = T(\mathrm{ATL}(B^1_{j_1+1})(x_1), \mathrm{ATL}(B^2_{j_2})(x_2)) = T(\gamma_1, 1) = \gamma_1 \,. \tag{7.49}$$

As the rule base from which the rule base of the ATL model is derived is monotone, *i.e.*

$$(\forall s \in S_4)(i_s \leq i + p_1) \,, \tag{7.50}$$

it follows from Eq. (7.25) that all output values have a smaller or equal membership degree to the adapted linguistic value $(\mathrm{ATL}(A_{i+p_1}))'$ than to the adapted linguistic output value of a rule of the fourth group of rules, *i.e.*

$$(\forall s \in S_4)(\forall y \in \mathbf{Y})((\mathrm{ATL}(A_{i_s}))'(y) \geq (\mathrm{ATL}(A_{i+p_1}))'(y)) \,. \tag{7.51}$$

Thus, the discussion above can be summarized in the following four equations

$$\min(\min_{s \in S_1}(\mathrm{ATL}(A_{i_s}))'(y), (\mathrm{ATL}(A_i))'(y)) = (\mathrm{ATL}(A_i))'(y) \,, \tag{7.52}$$

$$\min(\min_{s \in S_2}(\mathrm{ATL}(A_{i_s}))'(y), (\mathrm{ATL}(A_{i+p_1+p_2}))'(y)) = (\mathrm{ATL}(A_{i+p_1+p_2}))'(y) \,, \tag{7.53}$$

$$\min(\min_{s \in S_3}(\mathrm{ATL}(A_{i_s}))'(y), (\mathrm{ATL}(A_{i+p_1+p_2+p_3}))'(y)) = (\mathrm{ATL}(A_{i+p_1+p_2+p_3}))'(y) \,, \tag{7.54}$$

$$\min(\min_{s \in S_4}(\mathrm{ATL}(A_{i_s}))'(y), (\mathrm{ATL}(A_{i+p_1}))'(y)) = (\mathrm{ATL}(A_{i+p_1}))'(y) \,, \tag{7.55}$$

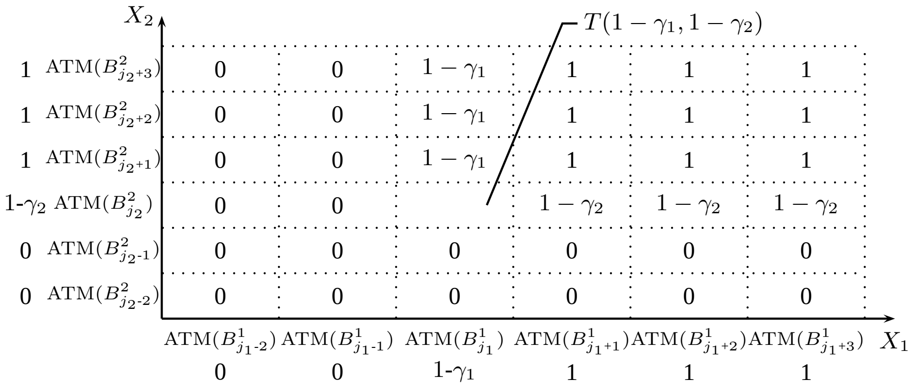


Figure 7.6: Fulfilment degrees obtained for the rules of an ATM model for an input vector $\mathbf{x}$ belonging to the indicated membership degrees to the linguistic values of X_1 and X_2.

and the general expression for the fuzzy output of the ATL model

$$A_{\mathrm{ATL}}(y) = \min_{s=1}^{r}(\mathrm{ATL}(A_{i_s}))'(y)\,, \tag{7.56}$$

can be simplified to the expression given in Eq. (7.35). The expression for the fuzzy output of the ATM model in Eq. (7.36) is obtained analogously. The fulfilment degrees of rules for an input vector $\mathbf{x}$ in an ATM model are schematically represented in Fig. 7.6.

7.3.2.3 Models with three or more input variables

The rule base of an ATL model, respectively ATM model, corresponding to a linguistic fuzzy model with three input variables contains r rules of the form

R_s: IF X_1 IS $\mathrm{ATL}(B^1_{j_{1,s}})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_{2,s}})$ AND X_3 IS $\mathrm{ATL}(B^3_{j_{3,s}})$
THEN Y IS $\mathrm{ATL}(A_{i_s})$

respectively,

R_s: IF X_1 IS $\mathrm{ATM}(B^1_{j_{1,s}})$ AND X_2 IS $\mathrm{ATM}(B^2_{j_{2,s}})$ AND X_3 IS $\mathrm{ATM}(B^3_{j_{3,s}})$
THEN Y IS $\mathrm{ATM}(A_{i_s})$

For an input vector $\mathbf{x} = (x_1, x_2, x_3)$ there always exist j_1, j_2 and j_3 such that

$$\gamma_1 = 1 - B^1_{j_1}(x_1) = B^1_{j_1+1}(x_1)\,, \tag{7.57}$$
$$\gamma_2 = 1 - B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2)\,, \tag{7.58}$$
$$\gamma_3 = 1 - B^3_{j_3}(x_3) = B^3_{j_3+1}(x_3)\,. \tag{7.59}$$

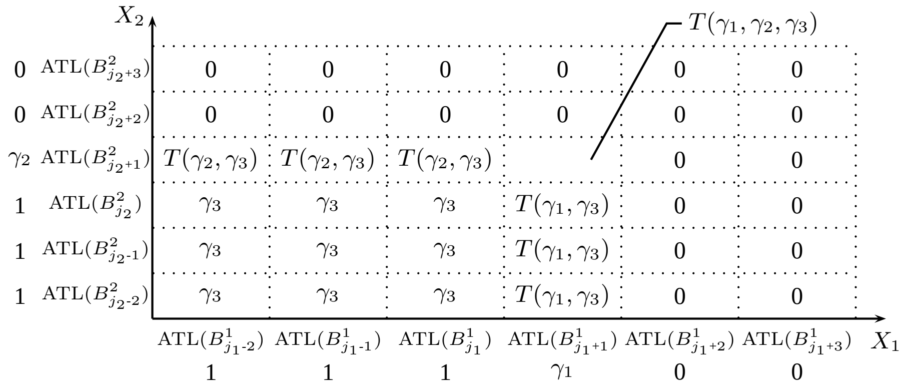


Figure 7.7: Fulfilment degrees obtained in an ATL model with three input variables for the rules containing the linguistic value $B^3_{j_3+1}$ and for an input vector $\mathbf{x}$ belonging to the indicated membership degrees to the linguistic values of X_1 and X_2 and to a degree γ_3 to $\mathrm{ATL}(B^3_{j_3+1})$.

The rules in the rule base of the ATL model can be divided in three groups. A first group of rules is derived from rules containing a linguistic value smaller than or equal to $B^3_{j_3}$ in their antecedent. Since

$$(\forall j \leq j_3)(\mathrm{ATL}(B^3_j)(x_3) = 1)\,, \tag{7.60}$$

for a given $\mathrm{ATL}(B^1_{j_1,s})$ and $\mathrm{ATL}(B^2_{j_2,s})$, the same fulfilment degree is obtained for these rules as the fulfilment degrees shown in Fig. 7.5 for the rules of a model with two input variables. Following a similar reasoning as in Section 7.3.2.2, one can show that the intersection of individual contributions of this first group of rules is given by the intersection of the individual contributions of the four rules

IF X_1 IS $\mathrm{ATL}(B^1_{j_1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2})$ AND X_3 IS $\mathrm{ATL}(B^3_{j_3})$
THEN Y IS $\mathrm{ATL}(A_i)$
IF X_1 IS $\mathrm{ATL}(B^1_{j_1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2+1})$ AND X_3 IS $\mathrm{ATL}(B^3_{j_3})$
THEN Y IS $\mathrm{ATL}(A_{i+p_1+p_2})$
IF X_1 IS $\mathrm{ATL}(B^1_{j_1+1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2})$ AND X_3 IS $\mathrm{ATL}(B^3_{j_3})$
THEN Y IS $\mathrm{ATL}(A_{i+p_1})$
IF X_1 IS $\mathrm{ATL}(B^1_{j_1+1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2+1})$ AND X_3 IS $\mathrm{ATL}(B^3_{j_3})$
THEN Y IS $\mathrm{ATL}(A_{i+p_1+p_2+p_4})$

A second group of rules contains the linguistic value $\mathrm{ATL}(B^3_{j_3+1})$ in their antecedent. The fulfilment degrees obtained for these rules as function of $\mathrm{ATL}(B^1_{j_1,s})$ and $\mathrm{ATL}(B^2_{j_2,s})$ are shown in Fig. 7.7. Following a similar reasoning as in Section 7.3.2.2, one can show that the intersection of individual contributions of this second group of rules is given by the intersection of the individual contributions of the four rules

IF X_1 IS $\mathrm{ATL}(B^1_{j_1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2})$ AND X_3 IS $\mathrm{ATL}(B^3_{j_3+1})$
THEN Y IS $\mathrm{ATL}(A_{i+p_1+p_2+p_3})$
IF X_1 IS $\mathrm{ATL}(B^1_{j_1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2+1})$ AND X_3 IS $\mathrm{ATL}(B^3_{j_3+1})$
THEN Y IS $\mathrm{ATL}(A_{i+p_1+p_2+p_3+p_5})$
IF X_1 IS $\mathrm{ATL}(B^1_{j_1+1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2})$ AND X_3 IS $\mathrm{ATL}(B^3_{j_3+1})$
THEN Y IS $\mathrm{ATL}(A_{i+p_1+p_2+p_3+p_6})$
IF X_1 IS $\mathrm{ATL}(B^1_{j_1+1})$ AND X_2 IS $\mathrm{ATL}(B^2_{j_2+1})$ AND X_3 IS $\mathrm{ATL}(B^3_{j_3+1})$
THEN Y IS $\mathrm{ATL}(A_{i+p'_7})$

with $p_1, p_2, p_3, p_4, p_5, p_6, p_7 \in \mathbb{N}$ and $p'_7 = p_1 + p_2 + \max(p_4, p_3 + p_5, p_3 + p_6) + p_7$.

A third, and last, group of rules is derived from rules containing a linguistic value larger than $B^3_{j_3+1}$ in their antecedent. Since

$$(\forall j > j_3 + 1)(\mathrm{ATL}(B^3_j)(x_3) = 0)\,, \tag{7.61}$$

these rules are not fired and their individual contributions to the global fuzzy output are identical to the universal set.

An analogous reasoning can be made for the ATM model. Summarizing, the fuzzy output obtained for an input vector $\mathbf{x}$ of an ATL and ATM model corresponding to a linguistic fuzzy model with three input variables is determined by the adapted membership functions in the consequents of the rules corresponding to the eight rules fired by the input vector $\mathbf{x}$ in the linguistic fuzzy model when Mamdani–Assilian or plain implicator-based inference is applied

$$\begin{aligned}
A_{\mathrm{ATL}}(y) = \min(&\mathrm{ATL}(A_i)(y), I_T(\gamma_3, \mathrm{ATL}(A_{i+p_1+p_2+p_3})(y)),\\
&I_T(\gamma_2, \mathrm{ATL}(A_{i+p_1+p_2})(y)),\\
&I_T(T(\gamma_2,\gamma_3), \mathrm{ATL}(A_{i+p_1+p_2+p_3+p_5})(y)), I_T(\gamma_1, \mathrm{ATL}(A_{i+p_1})(y)),\\
&I_T(T(\gamma_1,\gamma_3), \mathrm{ATL}(A_{i+p_1+p_2+p_3+p_6})(y)),\\
&I_T(T(\gamma_1,\gamma_2), \mathrm{ATL}(A_{i+p_1+p_2+p_4})(y)),\\
&I_T(T(\gamma_1,\gamma_2,\gamma_3), \mathrm{ATL}(A_{i+p'_7})(y)))\,, \qquad (7.62)\\
A_{\mathrm{ATM}}(y) = \min(&I_T(T(1-\gamma_1, 1-\gamma_2, 1-\gamma_3), \mathrm{ATM}(A_i)(y)),\\
&I_T(T(1-\gamma_1, 1-\gamma_2), \mathrm{ATM}(A_{i+p_1+p_2+p_3})(y)),\\
&I_T(T(1-\gamma_1, 1-\gamma_3), \mathrm{ATM}(A_{i+p_1+p_2})(y)),\\
&I_T(1-\gamma_1, \mathrm{ATM}(A_{i+p_1+p_2+p_3+p_5})(y)),\\
&I_T(T(1-\gamma_2, 1-\gamma_3), \mathrm{ATM}(A_{i+p_1})(y)),\\
&I_T(1-\gamma_2, \mathrm{ATM}(A_{i+p_1+p_2+p_3+p_6})(y)),\\
&I_T(1-\gamma_3, \mathrm{ATM}(A_{i+p_1+p_2+p_4})(y)), \mathrm{ATM}(A_{i+p'_7})(y))\,. \qquad (7.63)
\end{aligned}$$

Following a similar reasoning as above, one can easily verify that also for models with more than three input variables the fuzzy outputs of the corresponding ATL and ATM models obtained for an input vector $\mathbf{x}$ are determined by the rules corresponding to the rules fired by the given input vector $\mathbf{x}$ when applying Mamdani–Assilian or plain implicator-based inference.

7.4 Incomparable fuzzy model outputs

7.4.1 Circumventing incomparability by defuzzification

Investigating the monotonicity of a model requires the existence of an order on the (fuzzy) model outputs of, on the one hand, any input vector $\mathbf{x}_i$, and, on the other hand, all input vectors $\mathbf{x}_j$ differing in only one input value from $\mathbf{x}_i$. Since no order can be defined between the empty set or the universal set and any non-empty non-universal fuzzy set, nor can a defuzzification procedure be proposed to circumvent this incomparability, a prerequisite for a monotone model is to return a non-empty non-universal fuzzy output for any input vector $\mathbf{x}$.

The empty set could be consistently defuzzified by mapping it either to the expression *unknown* or to a certain crisp value. It is clear that the first procedure does not resolve the incomparability present on the level of the fuzzy model outputs. The second procedure only results in a global monotone input-output behaviour if a monotone input-output behaviour is obtained in those regions of the input space where only non-empty fuzzy sets are obtained as fuzzy model outputs and if the crisp value y^*_{emptyset} to which the empty set is mapped, satisfies

$$(\forall \mathbf{x} \in \mathbf{X}_{\mathrm{emptyset}})(\forall l \in L)(y^*_{lb,l}(\mathbf{x}) \le y^*_{\mathrm{emptyset}} \le y^*_{ub,l}(\mathbf{x}))\,, \tag{7.64}$$

with

$$\begin{aligned}
\mathbf{X}_{\mathrm{emptyset}} &= \{\mathbf{x} \mid (\forall y \in \mathbf{Y})(A(\mathbf{x})(y) = 0)\}\,,\\
y^*_{lb,l}(\mathbf{x}) &= \sup\{y^*(\mathbf{x}') \mid \mathbf{x}' \notin \mathbf{X}_{\mathrm{emptyset}} \wedge x_l > x'_l \wedge (\forall l' \in L \setminus \{l\})(x_{l'} = x'_{l'})\}\,,\\
y^*_{ub,l}(\mathbf{x}) &= \inf\{y^*(\mathbf{x}') \mid \mathbf{x}' \notin \mathbf{X}_{\mathrm{emptyset}} \wedge x_l < x'_l \wedge (\forall l' \in L \setminus \{l\})(x_{l'} = x'_{l'})\}\,.
\end{aligned}$$

Note that a slightly different notation is used in Eq. (7.64) for the fuzzy output $A(y)$ and the crisp output y^* in order to be able to indicate for which input vector $\mathbf{x}$ the fuzzy output $A(y)(\mathbf{x})$ and the crisp output $y^*(\mathbf{x})$ was obtained.

As shown in Fig. 7.8(a) a monotone input-output behaviour is obtained for a Mamdani–Assilian model applying the t-norm $T_{\mathbf{P}}$ combined with the COG defuzzification method, using the same membership functions for both input variables X_1 and X_2 and the output variable Y and containing 25 rules of the following form in its complete rule base

$$R_s\text{: IF } X_1 \text{ IS } B^1_{j_{1,s}} \text{ AND } X_2 \text{ IS } B^2_{j_{2,s}} \text{ THEN } Y \text{ IS } A_{i_s}$$

with $i_s = \min\{i \mid i \in \mathbb{N}_0, i \ge \frac{1}{5} j_{1,s} j_{2,s}\}$. When the rule

$$\text{IF } X_1 \text{ IS } B^1_3 \text{ AND } X_2 \text{ IS } B^2_2 \text{ THEN } Y \text{ IS } A_2$$

is removed from the rule base, the empty set is returned for inputs belonging to the kernels of the linguistic values B^1_3 and B^2_2 and a monotone input-output behaviour is obtained provided the empty set is mapped to a crisp value larger than 0.104 and smaller than 0.300. When the rule

$$\text{IF } X_1 \text{ IS } B^1_4 \text{ AND } X_2 \text{ IS } B^2_4 \text{ THEN } Y \text{ IS } A_4$$

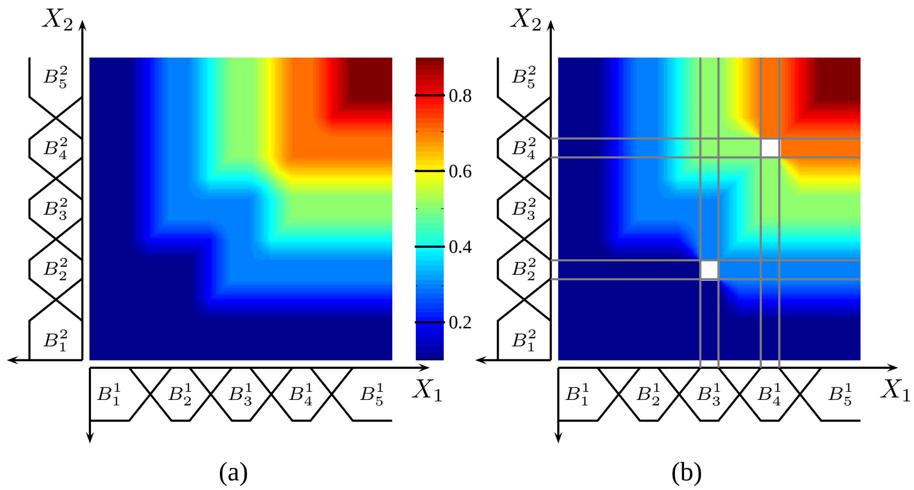


Figure 7.8: Model outputs of (a) a Mamdani–Assilian model returning a non-empty fuzzy set for any input vector and (b) a Mamdani–Assilian model returning the empty set in two regions of the input space.

is removed from the rule base, the empty set is returned for inputs belonging to the kernels of the linguistic values B_4^1 and B_4^2 and a monotone input-output behaviour is obtained provided the empty set is mapped to a crisp value larger than 0.500 and smaller than 0.700. When both rules are removed from the rule base (Fig. 7.8(b)), the empty set is returned as fuzzy output for inputs belonging to the kernels of the linguistic values B_3^1 and B_2^2 as well as for inputs belonging to the kernels of the linguistic values B_4^1 and B_4^2 and there exists no value y^*_{emptyset} satisfying Eq. (7.64) . The second defuzzification procedure might therefore seem valuable from a theoretical point of view, but it is hardly applicable in practice as y^*_{emptyset} has to be redefined every time a model property is altered, if, at all, a value satisfying Eq. (7.64) exists. Therefore, to return a non-empty fuzzy output for any input vector $\mathbf{x}$ remains a prerequisite for a monotone model.

Analogously it can be illustrated by considering a model applying plain implicator-based inference instead of Mamdani–Assilian inference in the above example that the incomparability between the universal set and any non-universal set cannot be circumvented by a defuzzification procedure.

7.4.2 Mamdani–Assilian models

Given the model properties assumed in this work (Section 7.2), the fuzzy output of a Mamdani–Assilian model is equal to the empty set if

$$(\forall y \in \mathbf{Y})(A(y) = \max_{i=1}^{n} A'_i(y) = 0)\,, \tag{7.65}$$

or, explicitly,

$$(\forall y \in \mathbf{Y})(\forall i \in I)(T(\alpha_i, A_i(y)) = 0)\,. \tag{7.66}$$

As all output membership functions have a non-empty kernel, it holds that

$$(\forall i \in I)(\exists y^i_{\mathrm{kern}} \in \mathbf{Y})(A_i(y^i_{\mathrm{kernel}}) = 1)\,. \tag{7.67}$$

It then follows in particular that

$$(\forall i \in I)(T(\alpha_i, A_i(y^i_{\mathrm{kern}})) = 0)\,, \tag{7.68}$$

or, in view of Eqs. (2.18–2.19), that

$$(\forall i \in I)(\alpha_i = 0)\,. \tag{7.69}$$

With Eq. (2.46) it then follows that

$$\beta_{\max} = \max_{s=1}^{r} \beta_s = 0\,. \tag{7.70}$$

In other words, a non-empty fuzzy set is obtained as model output for all input vectors $\mathbf{x}$ of a Mamdani–Assilian model if the maximum fulfilment degree $\beta_{\max}$ is strictly positive for all input vectors $\mathbf{x}$. In Mamdani–Assilian models with a single input variable X_1, the fulfilment degrees β_s are identical to the membership degrees of the input value x_1 to the linguistic values of X_1. Models with a single input variable, holding the properties defined in Section 7.2, never return the empty set as fuzzy output since the maximum fulfilment degree $\beta_{\max,1}$ is at least 0.5

$$\beta_{\max,1} = \max(1-\gamma_1, \gamma_1) \geq 0.5\,. \tag{7.71}$$

In Mamdani–Assilian models with two or more input variables, the fulfilment degrees β_s are calculated from the membership degrees to the linguistic values of the input variables by means of a t-norm. Let $\mathcal{B}_{T,m}$ be the set of all fulfilment degrees β_s corresponding to an input vector $\mathbf{x}$ defined as

$$\mathcal{B}_{T,m} = \{\beta_s(\mathbf{x}) = \mathop{T}_{l=1}^{m} B^l_{j_{l,s}}(x_l) \mid (j_{1,s}, \ldots, j_{m,s}) \in \prod_{l=1}^{m} \{j_l, j_l{+}1\}\}\,, \tag{7.72}$$

with the indices j_l determined by Eq. (7.8). Note that for the input vector $\mathbf{x}$ under consideration all fulfilment degrees not belonging to $\mathcal{B}_{T,m}$ are equal to zero.

When applying the t-norm $T_{\mathbf{M}}$ the maximum fulfilment degree $\beta_{\max,T_{\mathbf{M}},m}$ obtained for a model with m input variables is at least 0.5 as is shown below by induction, *i.e.*

$$\beta_{\max,T_{\mathbf{M}},m} = \max(\mathcal{B}_{T_{\mathbf{M}},m}) \geq 0.5\,. \tag{7.73}$$

For $m = 2$, Eq. (7.73) holds as

$$\begin{aligned}
\beta_{\max,T_{\mathbf{M}},2} &= \max(\mathcal{B}_{T_{\mathbf{M}},2}) \\
&= \max(\min(1-\gamma_1, 1-\gamma_2), \min(1-\gamma_1, \gamma_2), \min(\gamma_1, 1-\gamma_2), \\
&\qquad \min(\gamma_1, \gamma_2)) \\
&= \max(\max(\min(1-\gamma_1, 1-\gamma_2), \min(1-\gamma_1, \gamma_2)), \\
&\qquad \max(\min(\gamma_1, 1-\gamma_2), \min(\gamma_1, \gamma_2))) \\
&= \max(\min(1-\gamma_1, \max(1-\gamma_2, \gamma_2)), \min(\gamma_1, \max(1-\gamma_2, \gamma_2))) \\
&= \min(\max(1-\gamma_1, \gamma_1), \max(1-\gamma_2, \gamma_2)) \\
&\geq 0.5\,. \qquad (7.74)
\end{aligned}$$

Assuming that Eq. (7.73) holds for m^*,

$$\beta_{\max,T_{\mathbf{M}},m^*} = \max(\mathcal{B}_{T_{\mathbf{M}},m^*}) \geq 0.5\,, \qquad (7.75)$$

it also holds for $m^* + 1$ as

$$\begin{aligned}
\beta_{\max,T_{\mathbf{M}},m^*+1} &= \max(\mathcal{B}_{T_{\mathbf{M}},m^*+1}) \\
&= \max(\max_{\beta \in \mathcal{B}_{T_{\mathbf{M}},m^*}} \min(\beta, 1-\gamma_{m^*+1}), \max_{\beta \in \mathcal{B}_{T_{\mathbf{M}},m^*}} \min(\beta, \gamma_{m^*+1})) \\
&= \max(\min(\max(\mathcal{B}_{T_{\mathbf{M}},m^*}), 1-\gamma_{m^*+1}), \\
&\qquad \min(\max(\mathcal{B}_{T_{\mathbf{M}},m^*}), \gamma_{m^*+1})) \\
&= \min(\beta_{\max,T_{\mathbf{M}},m^*}, \max(1-\gamma_{m^*+1}, \gamma_{m^*+1})) \\
&\geq 0.5\,. \qquad (7.76)
\end{aligned}$$

When applying the t-norm $T_{\mathbf{P}}$ the maximum fulfilment degree $\beta_{\max,T_{\mathbf{P}},m}$ obtained for a model with m input variables is at least 2^{-m} as shown below by induction, *i.e.*

$$\beta_{\max,T_{\mathbf{P}},m} = \max(\mathcal{B}_{T_{\mathbf{P}},m}) \geq 2^{-m}\,. \qquad (7.77)$$

For $m = 2$, Eq. (7.77) holds as

$$\begin{aligned}
\beta_{\max,T_{\mathbf{P}},2} &= \max(\mathcal{B}_{T_{\mathbf{P}},2}) \\
&= \max((1-\gamma_1)(1-\gamma_2), (1-\gamma_1)\gamma_2, \gamma_1(1-\gamma_2), \gamma_1\gamma_2) \\
&= \max(\max((1-\gamma_1)(1-\gamma_2), (1-\gamma_1)\gamma_2, \max(\gamma_1(1-\gamma_2), \gamma_1\gamma_2)) \\
&= \max((1-\gamma_1)\max(1-\gamma_2, \gamma_2), \gamma_1 \max(1-\gamma_2, \gamma_2)) \\
&= (\max(1-\gamma_1, \gamma_1))(\max(1-\gamma_2, \gamma_2)) \\
&\geq 0.25\,. \qquad (7.78)
\end{aligned}$$

Assuming that Eq. (7.77) holds for m^*,

$$\beta_{\max,T_{\mathbf{P}},m^*} = \max(\mathcal{B}_{T_{\mathbf{P}},m^*}) \geq 2^{-m^*}\,, \qquad (7.79)$$

it also holds for $m^* + 1$ as

$$\begin{aligned}
\beta_{\max,T_{\mathbf{P}},m^*+1} &= \max(\mathcal{B}_{T_{\mathbf{P}},m^*+1}) \\
&= \max(\max_{\beta\in\mathcal{B}_{T_{\mathbf{P}},m^*}}(\beta\cdot(1-\gamma_{m^*+1})), \max_{\beta\in\mathcal{B}_{T_{\mathbf{P}},m^*}}(\beta\cdot\gamma_{m^*+1})) \\
&= \max((1-\gamma_{m^*+1})\cdot\max(\mathcal{B}_{T_{\mathbf{P}},m^*}), \gamma_{m^*+1}\cdot\max(\mathcal{B}_{T_{\mathbf{P}},m^*})) \\
&= \beta_{\max,T_{\mathbf{P}},m^*}\max(1-\gamma_{m^*+1}, \gamma_{m^*+1}) \\
&\geq 2^{-(m^*+1)}\,. \qquad (7.80)
\end{aligned}$$

For Mamdani–Assilian models with two or more input variables, the t-norm $T_{\mathbf{L}}$ is not appropriate as the empty set is obtained as fuzzy output when an input vector has two input values with equal membership degree to two linguistic values in their corresponding input domain ($\Gamma_m = (\gamma_1, \ldots, \gamma_m)$), *i.e.*

$$\min_{\Gamma_m\in[0,1]^m}\beta_{\max,T_{\mathbf{L}},m} = \min_{\Gamma_m\in[0,1]^m}\max(\mathcal{B}_{T_{\mathbf{L}},m}) = 0\,. \qquad (7.81)$$

For $m = 2$ the maximum fulfilment degree $\beta_{\max,T_{\mathbf{L}},2}$ is obtained by

$$\begin{aligned}
\beta_{\max,T_{\mathbf{L}},2} &= \max(\mathcal{B}_{T_{\mathbf{L}},2}) \\
&= \max(\max(1-\gamma_1+1-\gamma_2-1,0), \max(1-\gamma_1+\gamma_2-1,0), \\
&\qquad \max(\gamma_1+1-\gamma_2-1,0), \max(\gamma_1+\gamma_2-1,0)) \\
&= \max(1-\gamma_1+1-\gamma_2-1, 1-\gamma_1+\gamma_2-1, \\
&\qquad \gamma_1+1-\gamma_2-1, \gamma_1+\gamma_2-1, 0) \\
&= \max(1-\gamma_1+\max(1-\gamma_2,\gamma_2)-1, \gamma_1+\max(1-\gamma_2,\gamma_2)-1, 0) \\
&= \max(\max(1-\gamma_1,\gamma_1)+\max(1-\gamma_2,\gamma_2)-1, 0)\,. \qquad (7.82)
\end{aligned}$$

If γ_1 and γ_2 are both equal to 0.5, $\beta_{\max,T_{\mathbf{L}},2}$ is equal to zero

$$\beta_{\max,T_{\mathbf{L}},2} = \max(\max(1-0.5,0.5)+\max(1-0.5,0.5)-1,0) = 0\,. \qquad (7.83)$$

Therefore Eq. (7.81) holds for $m = 2$

$$\min_{\Gamma_2\in[0,1]^2}\beta_{\max,T_{\mathbf{L}},2} = 0\,. \qquad (7.84)$$

Assuming that Eq. (7.81) holds for m^*,

$$\min_{\Gamma_{m^*}\in[0,1]^{m^*}}\beta_{\max,T_{\mathbf{L}},m^*} = \max(\mathcal{B}_{T_{\mathbf{L}},m^*}) = 0\,, \qquad (7.85)$$

it also holds for $m^* + 1$

$$
\begin{aligned}
&\min_{\Gamma_{m^*+1}\in[0,1]^{m^*+1}} \beta_{\max,T_{\mathbf{L}},m^*+1} \\
&= \min_{\Gamma_{m^*+1}\in[0,1]^{m^*+1}} \max(\mathcal{B}_{T_{\mathbf{L}},m^*+1}) \\
&= \min_{\Gamma_{m^*+1}\in[0,1]^{m^*+1}} \max(\max_{\beta\in\mathcal{B}_{T_{\mathbf{L}},m^*}} \max(\beta + 1 - \gamma_{m^*+1} - 1, 0), \\
&\qquad\qquad\qquad\qquad \max_{\beta\in\mathcal{B}_{T_{\mathbf{L}},m^*}} \max(\beta + \gamma_{m^*+1} - 1, 0)) \\
&= \min_{\Gamma_{m^*+1}\in[0,1]^{m^*+1}} \max(\max_{\beta\in\mathcal{B}_{T_{\mathbf{L}},m^*}} (\beta + 1 - \gamma_{m^*+1} - 1), \\
&\qquad\qquad\qquad\qquad \max_{\beta\in\mathcal{B}_{T_{\mathbf{L}},m^*}} (\beta + \gamma_{m^*+1} - 1), 0) \\
&= \min_{\Gamma_{m^*+1}\in[0,1]^{m^*+1}} \max(\max_{\beta\in\mathcal{B}_{T_{\mathbf{L}},m^*}} \max(\beta + 1 - \gamma_{m^*+1} - 1, \beta + \gamma_{m^*+1} - 1), 0) \\
&= \min_{\Gamma_{m^*+1}\in[0,1]^{m^*+1}} \max(\max_{\beta\in\mathcal{B}_{T_{\mathbf{L}},m^*}} (\beta + \max(1 - \gamma_{m^*+1}, \gamma_{m^*+1}) - 1), 0) \\
&= \min_{\Gamma_{m^*+1}\in[0,1]^{m^*+1}} \max(\beta_{\max,T_{\mathbf{L}},m^*} + \max(1 - \gamma_{m^*+1}, \gamma_{m^*+1}) - 1, 0) \\
&= 0\,,
\end{aligned}
\tag{7.86}
$$

as $0.5 \leq \max(1 - \gamma_l, \gamma_l) \leq 1$.

In Chapters 8–9 Mamdani–Assilian models with one or more input variables will be considered for $T_{\mathbf{M}}$ and $T_{\mathbf{P}}$, while for $T_{\mathbf{L}}$, only models with a single input variable will be investigated.

7.4.3 Models applying implicator-based inference

For the three considered implicators I_T it holds that

$$(\forall\, x \in [0,1])(I_T(0, x) = 1)\,. \tag{7.87}$$

Thus, if for a given input vector $\mathbf{x}$ none of the rules of a model applying implicator-based inference is fired, *i.e.*

$$\beta_{\max} = \max_{s=1}^{r} \beta_s = 0\,, \tag{7.88}$$

the model returns the universal set for this input vector $\mathbf{x}$, *i.e.*

$$(\forall y \in \mathbf{Y})(A(y) = \min_{s=1}^{r} I(\beta_s, A_{i_s}(y)) = 1)\,. \tag{7.89}$$

Since for a given input vector $\mathbf{x}$ the same fulfilment degrees β_s are obtained when applying Mamdani–Assilian or plain implicator-based inference, it follows from Section 7.4.2 that a model with two or more input variables applying plain implicator-based inference and using $T_{\mathbf{L}}$ as t-norm will always return the universal set for some input vectors $\mathbf{x}$. Therefore, models with two or more variables applying $T_{\mathbf{L}}$ will not be considered in Section 10.2 discussing the monotonicity of models applying plain implicator-based inference.

From the discussion regarding the representation of rules in ATL–ATM models in Section 7.3.2 it follows that the fuzzy output of an ATL (resp. ATM model) is never identical to the empty set or the universal set since at least one linguistic output value $\mathrm{ATL}(A_i)$ (resp. $\mathrm{ATM}(A_i)$) has a fulfilment degree $\alpha_{\mathrm{ATL},i}$ (resp. $\alpha_{\mathrm{ATM},i}$) equal to 1. The fuzzy output of the ATL–ATM model, which is the intersection, based on the minimum, of the fuzzy outputs of a corresponding ATL and ATM model might, however, be the empty set. The issue of incomparable model outputs of ATL–ATM models is discussed in more detail in Chapter 10.

7.5 Monotone models in bioscience engineering

7.5.1 Land management

Soil erosion is one of the leading environmental problems of the world. In many areas, loss of this valuable natural resource takes place almost imperceptibly, and slowly affects the long-term productivity of the land. Soil erosion also contributes to the degradation of the quality of surface and ground waters by adding transported sediments, nutrients, pesticides and increased turbidity. Areas of erosion therefore need to be identified and appropriate conservation measures implemented (Mitra et al., 1998). Two linguistic fuzzy models describing the relationship between the soil erosion potential, *i.e.* the susceptibility of an area to erosion, and soil properties and landscape elements were developed by Mitra et al. (1998). The models were used to generate maps showing the location and distribution of soil erosion potential, which are very useful tools for policymakers.

The first model has two input variables: land use and slope angle class. For land use 11 classes are defined as described in Table 7.2, whereas slope angles were reorganized in 15 classes. To the variables land use and slope angle class respectively two and five ordered linguistic values were assigned, defined by the membership functions shown in Fig. 7.9(a–b). The values of both input variables can be derived from (hard copy) topographic maps. To the output variable, soil erosion potential, five linguistic values were assigned: low, moderately low, moderate, moderately high and high (Fig. 7.9(c)). The rule base is complete, *i.e.* it contains one rule for each combination of a linguistic value of land use class and a linguistic value of slope angle class. The rule base is represented in Fig. 7.9(d). The bottom left cell of the matrix corresponds to the rule 'IF land use class IS forest AND slope angle class IS very small THEN soil erosion potential IS low'. The rule base is monotone: the soil erosion potential is increasing in the land use class and the slope angle class. However, the rule base is not smooth since the following rules contain non-consecutive output values in their consequents

Table 7.2: Classes assigned to the variables land use and slope angle.

class	land use	slope angle (°)
1	deciduous forest	1–5
2	mixed forest	6–10
3	evergreen forest	11–15
4	good pasture	16–20
5	fair pasture	21–25
6	poor pasture	26–30
7	woodland pasture	31–35
8	over grazed	36–41
9	double cropped	42–47
10	row cropped	48–54
11		55–61
12		62–68
13		69–75
14	bare soil	76–81
15		82–87

IF land use class IS forest AND slope angle class IS small
THEN soil erosion potential IS low,
IF land use class IS pasture AND slope angle class IS small
THEN soil erosion potential IS moderate.

The second model has three input variables: soil erodibility factor K, cover factor and slope length. To these input variables respectively three, two and three ordered linguistic values are assigned. To the output value, soil erosion potential, the same linguistic values and membership functions are assigned as in the first model. This second model also has a monotone non-smooth rule base. The membership functions and rule base of the second model can be found in the work by Mitra et al. (1998).

7.5.2 Food technology

With consumers' demand for high-quality products, quality assurance has become a major concern in all manufacturing environments, including the food industry. In food manufacturing, a substantial amount of product grading and quality assurance is performed by human inspectors. However, manual inspection tends to be laborious, tedious, and prone to inconsistency. To solve these difficulties, food manufacturers are interested in automated visual inspection for quality assessment. At a low level of information processing, there are many advantages to automated inspection. Feature extraction (*e.g.* physical aspects such as size, area and colour) is consistent, unbiased, and quantitative. However, many food inspection operations also require a higher level of information processing. It is often necessary to integrate a number of physical features to make an inference about overall quality that is consistent with expert graders'

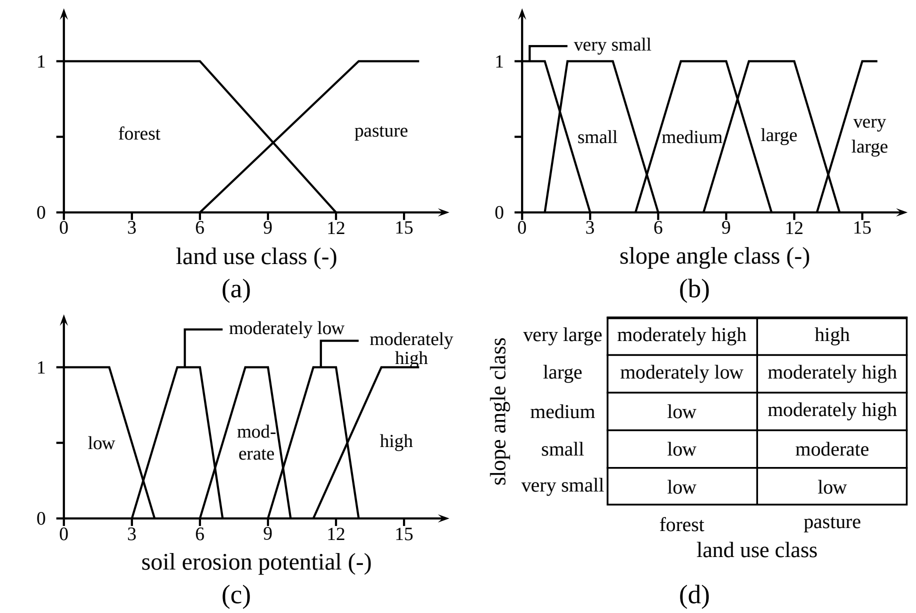


Figure 7.9: Membership functions defining the linguistic values assigned to (a) land use class , (b) slope angle class and (c) soil erosion potential, as well as (d) the rule base of the first model developed by Mitra et al. (1998).

or consumers' judgements (Davidson et al., 2001).

The work by Davidson et al. (2001) discusses the development of fuzzy models applied in an automated inspection system for chocolate chip cookies. The models assign a global quality score to a biscuit based on its size, baked dough colour and fraction of the top surface area that is chocolate chips. In this section one of the four models with a similar model structure discussed in the article is described. The model has three input variables, *i.e.* lightness, size and chips, and one output variable, *i.e.* consumer rating. The linguistic values assigned to the input variables are defined by trapezial membership functions forming a fuzzy partition as shown in Fig. 7.10(a–c), while the linguistic values assigned to the output variable are defined by the singletons in Fig. 7.10(d). The monotone non-smooth rule base contains 12 rules. In Fig. 7.11 the bottom left cell of the left matrix represents the rule 'IF size IS small AND lightness IS dark AND chips IS few THEN consumer rating IS unacceptable'. The crisp model output y^* is given by

$$y^* = \frac{\sum_{s=1}^{r} \beta_s y_s}{\sum_{s=1}^{r} \beta_s}\,, \tag{7.90}$$

with β_s the fulfilment degree and y_s the value of the singleton in the consequent of rule

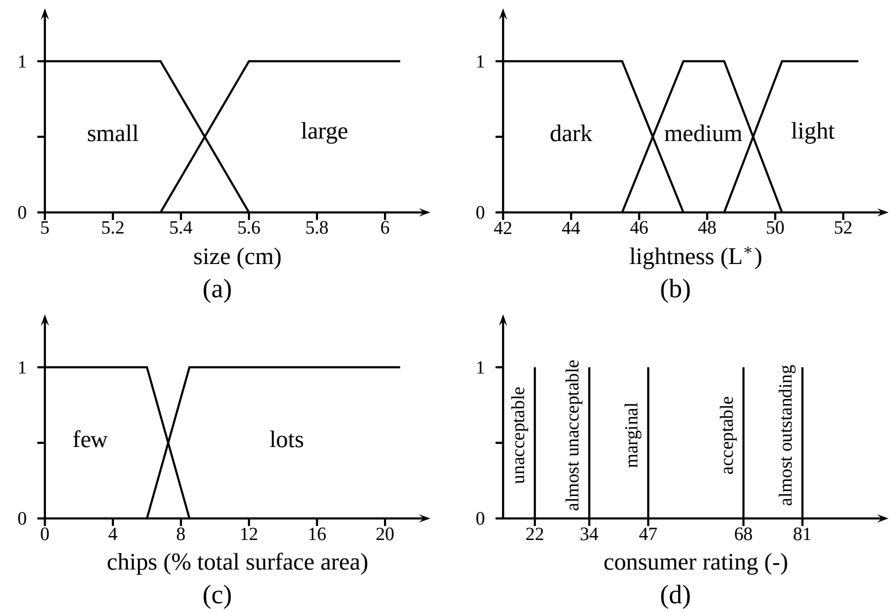


Figure 7.10: Membership functions defining the classes assigned to (a) size, (b) lightness, (c) chips and (d) consumer rating.

chips IS few

lightness \ size	small	large
light	marginal	marginal
medium	almost unacceptable	marginal
dark	unacceptable	unacceptable

chips IS lots

lightness \ size	small	large
light	acceptable	almost outstanding
medium	acceptable	almost outstanding
dark	unacceptable	unacceptable

Figure 7.11: Rule base of the model developed by Davidson et al. (2001).

R_s ($s \in \{1, \ldots, r\}$). So this model can also be regarded as a zero-order Takagi–Sugeno model.

7.5.3 Dairy farming

Replacing a conventional milking machine by an automatic milking system (milking robot) leads to more flexible working hours and a considerable time gain of 30 to 40 % for the herd manager. Furthermore, the cows can decide themselves when and how

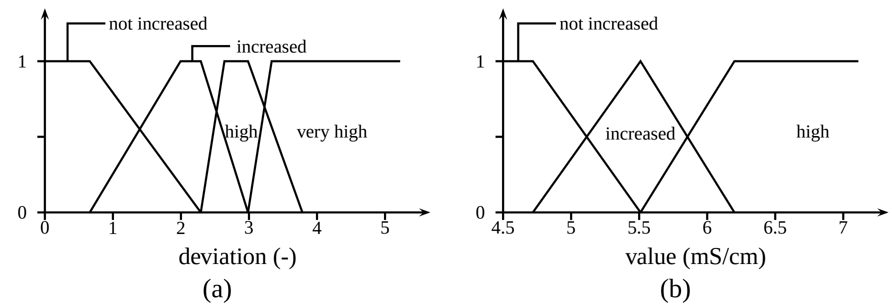


Figure 7.12: Membership functions defining the linguistic values assigned to (a) deviation and (b) value.

often they are milked, which — as is observed in practice — increases the milk production. However, with a conventional milking machine the herd manager can check twice a day the general condition of each cow, and more specific the udder (temperature, hardness, sensitivity) and the milk (flocks, viscosity). These quick examinations allow the herd manager to identify clinical udder infections, such as clinical mastitis, at an early stage and hereby restrict economical loss. With the introduction of an automatic milking system, these daily visual examinations of udder and milk are not longer carried out and it would be desirable if this task is also taken over by the milking robot. Nowadays sensors can be integrated in milking robots and most robots are equipped with a mastitis detection system based on the variation of the milk parameters accompanying udder infections. However, the currently commercialized mastitis detection systems suffer from a high rate of false negatives and false positives. In case of a false negative an infected cow is not registered by the system, resulting in economical loss due to inferior milk quality, while in case of a false positive the herd manager is urged by the system to conduct needless bacteriological examinations or to treat an uninfected cow (Piepers, 2005).

de Mol and Woldt (2001) developed a fuzzy model to reclassify the mastitis alerts generated by a statistical model developed in an earlier research (de Mol and Ouweltjes, 2000). The statistical model is based on sensor measurements of the electrical conductivity of milk and returns a high number of false positives. The fuzzy model has a hierarchical structure. For each quarter a fuzzy model was developed with deviation and value of the conductivity as input variables and adjusted deviation as output variable. Deviation is the difference between the expected and the measured conductivity divided by the variance of these differences. To deviation the four linguistic values were assigned, *i.e.* not increased, increased, high and very high, whereas value is granulated in three linguistic values, *i.e.* not increased, increased and high. The membership functions used for the right hind quarter are shown in Fig. 7.12. The membership functions applied in the models of the other quarters are similar.

For each quarter the fulfilment degrees of the four linguistic values of adjusted deviation were obtained with Mamdani–Assilian inference applying $T_{\mathbf{M}}$ and the rules

value \ deviation	not increased	increased	high	very high
high	not increased	increased	high	very high
increased	not increased	not increased	not increased	increased
not increased	not increased	not increased	not increased	not increased

Figure 7.13: Rule base of the model developed by de Mol and Woldt (2001).

represented in Fig. 7.13. The bottom left cell corresponds to the rule ‘IF deviation IS not increased AND value IS not increased THEN adjusted deviation IS not increased’. Note that this rule base is monotone but non-smooth as it contains following two rules with non-consecutive linguistic output values in their consequents

IF deviation IS high AND value IS increased THEN adj. deviation IS not increased,
IF deviation IS high AND value IS high THEN adj. deviation IS high.

Next, the fulfilment degrees of each linguistic value of adjusted deviation are given by the maximum fulfilment degree obtained for the linguistic output value under consideration in the four models. These fulfilment degrees are used to obtained the fulfilment degrees of the linguistic values of alert, *i.e.* false and true, using the following rules

IF adjusted deviation IS not increased THEN alert IS false,
IF adjusted deviation IS increased THEN alert IS false,
IF adjusted deviation IS high THEN alert IS true,
IF adjusted deviation IS very high THEN alert IS true.

Finally, the model output is defuzzified by selecting the linguistic value of alert with the highest fulfilment degree.

The fuzzy model developed by Piepers (2005) can be used directly as mastitis detection system. The model has three input variables, *i.e.* somatic cell number, milk production decrease and electrical conductivity increase and one output variable, *i.e.* need of a bacteriological examination. For all variables fuzzy partitions of trapezial membership functions were applied to define the linguistic values. The linguistic values assigned to the variables are given in the representation of the rule base in Fig. 7.14. The bottom left cell corresponds to the rule ‘IF somatic cell number IS low AND production decrease IS low AND electrical conductivity IS low THEN need of a bacteriological examination IS low’. The rule base is monotone and smooth.

electrical conductivity IS high

production decrease \ somatic cell number	low	moderate	moderately high	high	very high
high	moderate	moderately high	high	high	high
moderate	moderate	moderately high	moderately high	high	high
low	low	moderate	moderately high	moderately high	high

electrical conductivity IS moderate

production decrease \ somatic cell number	low	moderate	moderately high	high	very high
high	low	moderate	moderately high	high	high
moderate	low	moderate	moderately high	moderately high	high
low	low	moderate	moderate	moderate	moderately high

electrical conductivity IS low

production decrease \ somatic cell number	low	moderate	moderately high	high	very high
high	low	moderate	moderate	moderately high	high
moderate	low	moderate	moderate	moderately high	moderately high
low	low	low	moderate	moderate	moderate

Figure 7.14: Rule base of the model developed by Piepers (2005).

7.5.4 Ecological quality assessment

For years, the quality of stream sediments in Flanders (Belgium) has been influenced in a negative way by the poor quality of the surface water. Due to a decrease in the amount of waste water discharged untreated in water bodies, the water quality steadily improves. At some locations where the water quality has improved, a reversed problem arises. The contaminated sediment makes a further improvement of the water quality and the ecological recovery of the stream impossible, as pollutants migrate back from the sediment to the surface water. In order to further improve the quality of the surface waters, not only should actions be taken to reduce the effect of discharges, but also should efforts be made in the field of sediment sanitation. Since the sediment is an important component of the aquatic ecosystem and dredging and cleaning operations result in mud that should be disposed, it is important to monitor the quality of sediments. The Department of Environment, Nature and Energy of the Flemish government uses the TRIADE method to assess the ecological equality of sediments in Flanders (De Cooman and Detemmerman, 2004; Ministerie van de Vlaamse Gemeenschap, 2000). No elements from the fuzzy modelling field are incorporated in the TRIADE method, but the current procedure can easily be fuzzified as will be illustrated below. First, however, the current procedure is described.

The TRIADE method classifies a sediment into one of four ecological quality classes based on the outcome of three specific classifications reflecting respectively the physical-chemical, ecotoxicological and biological quality of the sediment.

Physical-chemical quality The physical-chemical quality is derived from the concentrations of 13 (groups of) components in the sediment: arsenic, cadmium, chromium, copper, mercury, lead, nickel, zinc, apolar hydrocarbons, extractable organohalogens, the sum of a group of pesticides, the sum of 7 polychlorobifenyls, the sum of the six polyaromatic hydrocarbons of Borneff. For each component the ratio of the measured concentration $C_{\text{measured},i}$ to the concentration $C_{\text{reference},i}$ in a reference sediment, *i.e.* a sediment that is (almost) unaffected by human activity, is calculated ($i \in \{1, \ldots, 13\}$)

$$\text{VTR}_i = \frac{C_{\text{measured},i}}{C_{\text{reference},i}}\,, \tag{7.91}$$

and converted in a variable logindex_i given by

$$\text{logindex}_i = \min(2, \log_{10}(\max(1, \text{VTR}_i)))\,, \tag{7.92}$$

with $0 \leq \text{logindex}_i \leq 2$.

Next, each component is assigned to one of four classes based on logindex_i

$$C_{\text{chem},i} = \begin{cases} C_1 & \text{, if } 0 \leq \text{logindex}_i < 0.4\,, \\ C_2 & \text{, if } 0.4 \leq \text{logindex}_i < 0.8\,, \\ C_3 & \text{, if } 0.8 \leq \text{logindex}_i < 1.2\,, \\ C_4 & \text{, if } 1.2 \leq \text{logindex}_i \leq 2\,. \end{cases} \tag{7.93}$$

The smaller the assigned class is, the better is the physical-chemical quality of the sediment.

Finally, the sediment is assigned to one of four physical-chemical quality classes. This class C_{chem} is equal to the highest class obtained for the 13 components, except if the number of components assigned to this highest class is smaller than or equal to two and the logindex_i values obtained for these components are furthermore smaller than the midpoint of the interval defining this highest class. In the latter case, the rank of the global physical-chemical class assigned to the sediment is equal to the rank of the highest class obtained for the 13 components reduced by 1.

Ecotoxicological quality An ecotoxicological assessment gives an indication of the potential effects on organisms. Lab-bred organisms are exposed to pore water or sediment for a certain time (hours or days). The three test organisms used in the TRIADE method, the algae *Raphidocelis subcapitata* (pore water test), the fairy shrimp *Thamnocephalus platyurus* (pore water test) and the amphipod *Hyalella azteca* (sediment test), strongly differ in susceptibility to specific toxic components. Furthermore, the biological availability of components in the sediment can vary strongly among the organisms.

The obtained results are again compared to those obtained for a reference sediment. The results obtained for *Raphidocelis subcapitata* and *Thamnocephalus platyurus* are expressed by the variable VTR, while results obtained for *Hyalella azteca* are represented by the variable mortality. The sediment is once classified in one of four classes based on VTR

$$C_{\text{toxi},1} = \begin{cases} C_1 & \text{, if VTR} = 1\,, \\ C_2 & \text{, if } 1 < \text{VTR} \leq 150\,, \\ C_3 & \text{, if } 150 < \text{VTR} \leq 300\,, \\ C_4 & \text{, if } 300 < \text{VTR}\,, \end{cases} \tag{7.94}$$

and once based on mortality

$$C_{\text{toxi},2} = \begin{cases} C_1 & \text{, if } 0 \leq \text{mortality} < 20\,, \\ C_2 & \text{, if } 20 \leq \text{mortality} < 50\,, \\ C_3 & \text{, if } 50 \leq \text{mortality} < 75\,, \\ C_4 & \text{, if } 75 \leq \text{mortality} \leq 100\,. \end{cases} \tag{7.95}$$

The global ecotoxicological quality class C_{toxi} is the highest of the classes obtained for $C_{\text{toxi},1}$ and $C_{\text{toxi},2}$.

Biological quality Benthic macroinvertebrates are used as indicator species for the biological quality of sediments. The Biotic Sediment Index (BWI) gives an indication of the biological quality based on the occurrence of certain indicator species and the taxonomic diversity of the (epi)benthic macroinvertebrate community. Sediments are assigned to one of four biological quality classes based on BWI

$$C_{\text{biol}} = \begin{cases} C_1 & \text{, if BWI} \in \{7, 8, 9, 10\}\,, \\ C_2 & \text{, if BWI} \in \{5, 6\}\,, \\ C_3 & \text{, if BWI} \in \{3, 4\}\,, \\ C_4 & \text{, if BWI} \in \{0, 1, 2\}\,. \end{cases} \tag{7.96}$$

Ecological quality In a last step of the TRIADE method, the sediment is assigned to one of four ecological quality classes following the procedure described by the eight rules below. Hereby, C_{1-2}, C_{2-4} and C_{3-4} respectively represent 'C_1 to C_2', 'C_2 to C_4' and 'C_3 to C_4'.

R_1: IF C_{chem} IS C_{1-2} AND C_{toxi} IS C_1 AND C_{biol} IS C_1 THEN C_{ecol} IS C_1
R_2: IF C_{chem} IS C_{1-2} AND C_{toxi} IS C_1 AND C_{biol} IS C_{2-4} THEN C_{ecol} IS C_2
R_3: IF C_{chem} IS C_{1-2} AND C_{toxi} IS C_{2-4} AND C_{biol} IS C_1 THEN C_{ecol} IS C_2
R_4: IF C_{chem} IS C_{1-2} AND C_{toxi} IS C_{2-4} AND C_{biol} IS C_{2-4} THEN C_{ecol} IS C_3
R_5: IF C_{chem} IS C_{3-4} AND C_{toxi} IS C_1 AND C_{biol} IS C_1 THEN C_{ecol} IS C_2
R_6: IF C_{chem} IS C_{3-4} AND C_{toxi} IS C_1 AND C_{biol} IS C_{2-4} THEN C_{ecol} IS C_3
R_7: IF C_{chem} IS C_{3-4} AND C_{toxi} IS C_{2-4} AND C_{biol} IS C_1 THEN C_{ecol} IS C_3
R_8: IF C_{chem} IS C_{3-4} AND C_{toxi} IS C_{2-4} AND C_{biol} IS C_{2-4} THEN C_{ecol} IS C_4

Fuzzified TRIADE method The definition of $C_{\text{chem},i}$, $C_{\text{toxi},1}$, $C_{\text{toxi},2}$ and C_{biol} can easily be fuzzified by replacing the crisp sets defined in Eqs. (7.93–7.96) by fuzzy sets as shown in Fig. 7.15. The fuzzified TRIADE method is illustrated on an example in Table 7.3. The original TRIADE method classifies a sediment characterized by the values in the second column in the ecological quality class C_3. In the fuzzified TRIADE method, the variables logindex$_i$, VTR, mortality and BWI are first classified in the four corresponding fuzzy classes described by the membership functions in Fig. 7.15, which results in a vector with four values between zero and one, summing up to one. The 13 fuzzy classifications $\mathbf{C}_{\text{chem},i}$ for the physical-chemical quality and the two fuzzy classifications $\mathbf{C}_{\text{toxi},1}$ and $\mathbf{C}_{\text{toxi},2}$ for the ecotoxicological quality are aggregated in the example by taking the classification corresponding to the highest class. There exists a wide range of aggregation operators, such as Ordered Weighted Average operators, where $\mathbf{C}_{\text{chem}}$ is given by a weighted sum of the 13 $\mathbf{C}_{\text{chem},i}$ with the weights being a function of the order of the 13 $\mathbf{C}_{\text{chem},i}$ (Calvo et al., 2002). In the example the fuzzy classifications obtained for the sediment are $\mathbf{C}_{\text{chem}} = (0, 0.1, 0.9, 0)$, $\mathbf{C}_{\text{toxi}} = (0, 0.6, 0.4, 0)$ and $\mathbf{C}_{\text{biol}} = (0.25, 0.75, 0, 0)$. When applying the t-norm $T_{\mathbf{P}}$ the fulfilment degrees of the eight if-then rules above are given by

$$\begin{aligned}
\beta_1 &= 0.1 \times 0 \times 0.25 = 0\,, && (7.97)\\
\beta_2 &= 0.1 \times 0 \times 0.75 = 0\,, && (7.98)\\
\beta_3 &= 0.1 \times 1 \times 0.25 = 0.025\,, && (7.99)\\
\beta_4 &= 0.1 \times 1 \times 0.75 = 0.075\,, && (7.100)\\
\beta_5 &= 0.9 \times 0 \times 0.25 = 0\,, && (7.101)\\
\beta_6 &= 0.9 \times 0 \times 0.75 = 0\,, && (7.102)\\
\beta_7 &= 0.9 \times 1 \times 0.25 = 0.225\,, && (7.103)\\
\beta_8 &= 0.9 \times 1 \times 0.75 = 0.675\,. && (7.104)
\end{aligned}$$

The fulfilment degrees of the output classes C_1, C_2, C_3 and C_4 are obtained by

$$\begin{aligned}
\alpha_1 &= \beta_1 && = 0\,, && (7.105)\\
\alpha_2 &= \max(\beta_2, \beta_3, \beta_5) = \max(0, 0.025, 0) && = 0.025\,, && (7.106)\\
\alpha_3 &= \max(\beta_4, \beta_6, \beta_7) = \max(0.075, 0, 0.225) && = 0.225\,, && (7.107)\\
\alpha_4 &= \beta_8 && = 0.675\,, && (7.108)
\end{aligned}$$

or, after normalization, by

$$\mathbf{C}_{\text{ecol}} = (0, 0.027, 0.243, 0.730)\,. \tag{7.109}$$

If needed, the obtained fuzzy output can be defuzzified, for instance by assigning the sediment to the ecological quality class with the highest membership degree, *i.e.* C_4 in the example. Note that the membership functions were chosen arbitrarily. The behaviour of the fuzzified TRIADE method can be tuned by adapting the membership functions defining the classes assigned to the variables logindex$_i$, VTR, mortality and BWI or by changing the procedure to obtain $\mathbf{C}_{\text{chem}}$ and $\mathbf{C}_{\text{toxi}}$.

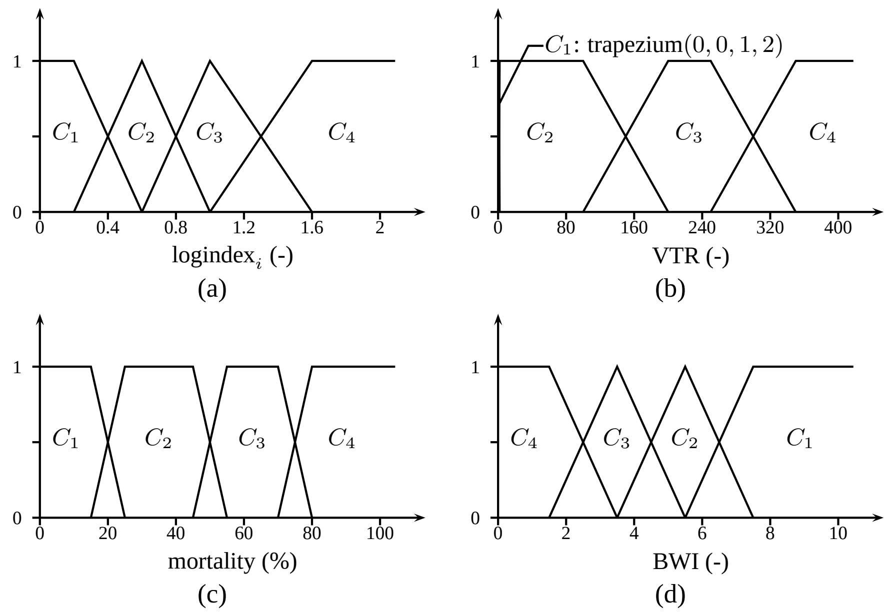


Figure 7.15: Membership functions defining the classes assigned to (a) logindex$_i$, (b) VTR, (c) mortality and (d) BWI.

Table 7.3: Classification with the original and fuzzified TRIADE method.

		classification	
variable	value	original	fuzzy
logindex$_1$	0.00	$C_{\mathrm{chem},1} = 1$	$\mathbf{C}_{\mathrm{chem},1} = (1, 0, 0, 0)$
logindex$_2$	0.00	$C_{\mathrm{chem},2} = 1$	$\mathbf{C}_{\mathrm{chem},2} = (1, 0, 0, 0)$
logindex$_3$	0.20	$C_{\mathrm{chem},3} = 1$	$\mathbf{C}_{\mathrm{chem},3} = (1, 0, 0, 0)$
logindex$_4$	0.24	$C_{\mathrm{chem},4} = 1$	$\mathbf{C}_{\mathrm{chem},4} = (0.9, 0.1, 0, 0)$
logindex$_5$	0.00	$C_{\mathrm{chem},5} = 1$	$\mathbf{C}_{\mathrm{chem},5} = (1, 0, 0, 0)$
logindex$_6$	0.40	$C_{\mathrm{chem},6} = 2$	$\mathbf{C}_{\mathrm{chem},6} = (0.5, 0.5, 0, 0)$
logindex$_7$	0.96	$C_{\mathrm{chem},7} = 3$	$\mathbf{C}_{\mathrm{chem},7} = (0, 0.1, 0.9, 0)$
logindex$_8$	0.84	$C_{\mathrm{chem},8} = 3$	$\mathbf{C}_{\mathrm{chem},8} = (0, 0.4, 0.6, 0)$
logindex$_9$	0.36	$C_{\mathrm{chem},9} = 1$	$\mathbf{C}_{\mathrm{chem},9} = (0.6, 0.4, 0, 0)$
logindex$_{10}$	0.00	$C_{\mathrm{chem},10} = 1$	$\mathbf{C}_{\mathrm{chem},10} = (1, 0, 0, 0)$
logindex$_{11}$	0.52	$C_{\mathrm{chem},11} = 2$	$\mathbf{C}_{\mathrm{chem},11} = (0.2, 0.8, 0, 0)$
logindex$_{12}$	0.00	$C_{\mathrm{chem},12} = 1$	$\mathbf{C}_{\mathrm{chem},12} = (1, 0, 0, 0)$
logindex$_{13}$	0.10	$C_{\mathrm{chem},13} = 1$	$\mathbf{C}_{\mathrm{chem},13} = (1, 0, 0, 0)$
		$C_{\mathrm{chem}} = 2$	$\mathbf{C}_{\mathrm{chem}} = (0, 0.1, 0.9, 0)$
VTR	140	$C_{\mathrm{toxi},1} = 2$	$\mathbf{C}_{\mathrm{toxi},1} = (0, 0.6, 0.4, 0)$
mortality	30	$C_{\mathrm{toxi},2} = 2$	$\mathbf{C}_{\mathrm{toxi},2} = (0, 1, 0, 0)$
		$C_{\mathrm{toxi}} = 2$	$\mathbf{C}_{\mathrm{toxi}} = (0, 0.6, 0.4, 0)$
BWI	6	$C_{\mathrm{biol}} = 2$	$\mathbf{C}_{\mathrm{biol}} = (0.25, 0.75, 0, 0)$

7.6 Conclusion

Part III of this dissertation, consisting of Chapters 7–10, is dedicated to my work on the monotonicity of linguistic fuzzy models. In this first chapter, some general aspects were discussed, such as the model properties assumed in this work, the applied representation of if-then rules, the issue of incomparable fuzzy model outputs and the applicability of monotone linguistic fuzzy models in bioscience engineering.

In Chapters 8–10 the monotonicity of linguistic fuzzy models under different inference procedures is discussed. Chapters 8–9 deal with Mamdani–Assilian models applying the t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ combined with respectively the COG and MOM defuzzification method. Chapter 10 focusses on models applying either plain implicator-based inference or ATL–ATM inference, one of the three basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$, one of the three R-implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ and the MOM defuzzification method. The objective of this study was to select, for each inference procedure, combinations of t-norm, implicator or defuzzification method resulting in a monotone input-output behaviour for any monotone rule base, or at least for any monotone smooth rule base.

Assuming the model properties in Section 7.2, the input-output behaviour of models with m input variables reduces to the input-output behaviour of models with m^* ($m^* < m$) input variables in those regions of the input space where the inputs belong to the kernel of the same linguistic value in all but m^* input domains. Thus, if certain model properties are necessary to guarantee monotonicity for models with m^* input variables, these model properties are also required to guarantee a monotone input-output behaviour for models with more than m^* input variables. Therefore, in Chapters 8–10 the monotonicity of models with a single input variable is studied first and throughout the discussion, the number of input variables considered, is gradually increased. For Mamdani–Assilian models applying the COG defuzzification method, models with up to three input variables are considered. For Mamdani–Assilian models applying the MOM defuzzification method, models with up to two input variables are considered in case of a monotone rule base, whereas the number of input variables is not restricted for models with a monotone smooth rule base. In Section 10.2 it will be shown that monotonicity cannot be guaranteed for models with two input variables applying plain implicator-based inference for the nine considered combinations of the t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$ and the three R-implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$, no models with a higher number of input variables are considered for plain implicator-based inference. Finally, for ATL–ATM models, only models with a single or two input variables are considered. The purpose of Chapter 10 is to illustrate the new inference method, rather than to give an extensive description of the monotonicity of these models as is done for Mamdani–Assilian models in Chapters 8–9.

CHAPTER 8

Mamdani–Assilian models: COG defuzzification

Turning points always seem so sudden and absolute, as if they have come bolt out of the blue. That is not true, of course. A whole slow process goes into their making.
(*Reading Lolita in Teheran*, Azar Hafisi, 2003)

8.1 Introduction

In this chapter the monotonicity is investigated of Mamdani–Assilian models holding the properties described in Section 7.2 and applying the Center of Gravity defuzzification method. It is verified for the three t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ whether a monotone input-output behaviour is obtained for any monotone rule base, or at least for any monotone smooth rule base.

First, in Section 8.2, the general definition of the crisp output y^*_{COG} (Eq. (2.44)) is reformulated for models holding the properties described in Section 7.2, using the variables introduced in the same section to characterize the output membership functions. In Section 8.3 the monotonicity of models with a single input variable is studied for the t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$. As discussed in Section 7.4, obtaining the empty set as fuzzy output cannot be avoided when using of the t-norm $T_{\mathbf{L}}$ in models with two or more input variables and holding the assumed properties, which makes $T_{\mathbf{L}}$ an inappropriate t-norm for these models. Therefore, Section 8.4 deals with the monotonicity of models with two input variables for the t-norms $T_{\mathbf{M}}$ and $T_{\mathbf{P}}$ only. As the research pointed out that a monotone input-output behaviour cannot be guaranteed for models with two input variables and any monotone (smooth) rule base when applying the t-norm $T_{\mathbf{M}}$, only the t-norm $T_{\mathbf{P}}$ is considered in Section 8.5 when investigating the monotonicity of models with three input variables. The chapter concludes with a summary of the obtained results in Section 8.6.

8.2 Tailoring the definition of y^*_{COG}

The general definition of the crisp output y^*_{COG} has been reformulated for models applying $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ and trapezial membership functions forming a fuzzy partition in Chapter 3. To facilitate the reading, the formulae are recapitulated in this section using the parameters introduced in Chapter 7 (Eq. (7.1)) and used throughout Part III.

The center of gravity y^*_{COG} of the surface defined by the global fuzzy output can be computed from the centers of gravity y^*_i and $y^*_{\mathrm{op},i}$ and areas S_i and $S_{\mathrm{op},i}$ of the n adapted membership functions and the n-1 overlapping parts,

$$y^*_{\mathrm{COG}} = \frac{\sum_{i=1}^{n}(y^*_i \cdot S_i) - \sum_{i=1}^{n\text{-}1}(y^*_{\mathrm{op},i} \cdot S_{\mathrm{op},i})}{\sum_{i=1}^{n} S_i - \sum_{i=1}^{n\text{-}1} S_{\mathrm{op},i}} \,. \tag{8.1}$$

The formulae listed in Table 8.1 for the terms y^*_i, $y^*_{\mathrm{op},i}$, S_i and $S_{\mathrm{op},i}$ in Eq. (8.1) are applicable to models using trapezial membership functions forming a fuzzy partition as shown in Fig. 7.1. When the linguistic output values used in the consequents of the rules are all described by membership functions with intervals of changing membership degree of equal length

$$(\forall s \in \{1,\ldots,r\})(i_s \notin \{1,n\})\,,(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)\,,$$

the formulae in Table 8.2 can be used. If, furthermore, the kernels of the output membership functions are of equal length

$$(\exists k \geq 0)(\forall i \in I)(k_i = k)\,,$$

the formula for S_i can even further be simplified.

8.3 Models with a single input variable

In a model with a single input variable at most two rules are fired: the rule corresponding to some linguistic value B^1_j is fired to a degree $(1-\gamma_1)$ and the rule corresponding to the linguistic input value B^1_{j+1} to a degree γ_1 (Eq. (7.8)). In case of a monotone rule base, B^1_j and B^1_{j+1} can either be mapped to

1. the same linguistic output value A_i: the *constant* case,
2. two consecutive output values A_i and A_{i+1}: the *smooth* case, or
3. two non-consecutive output values A_i and A_{i+p} ($p \in \mathbb{N}$, $p > 1$, $i + p \leq n$): the *non-smooth* case.

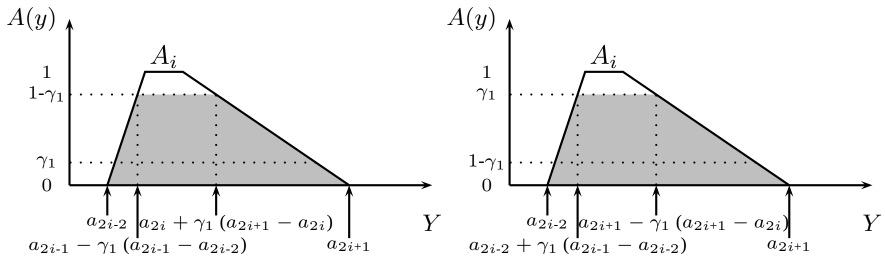
The *constant* case is meaningless for models with a single input variable, as it indicates the presence of redundant linguistic input values. If two adjacent linguistic input values of a model with a single input variable are mapped to the same linguistic output value they should be merged into a single linguistic input value defined by their convex hull in order to reduce the complexity and improve the interpretability of the

Table 8.1: Formulae for the centers of gravity y_i^* and $y_{\mathrm{op},i}^*$ and areas S_i and $S_{\mathrm{op},i}$ of the adapted membership functions and overlapping parts in Eq. (8.1)

	$T_{\mathbf{M}}$	$T_{\mathbf{P}}$	$T_{\mathbf{L}}$
y_i^*	$c_i + \frac{(l_i - l_{i-1})(3(2-\alpha_i)k_i + 2(3-3\alpha_i+\alpha_i^2)(l_{i-1}+l_i))}{6(2k_i+(2-\alpha_i)(l_{i-1}+l_i))}$	$c_i + \frac{(l_i-l_{i-1})(3k_i+2l_{i-1}+2l_i)}{6(2k_i+l_{i-1}+l_i)}$	$c_i + \frac{(l_i-l_{i-1})(3k_i+2\alpha_i(l_{i-1}+l_i))\alpha_i}{6(2k_i+\alpha_i(l_{i-1}+l_i))}$
S_i	$\frac{1}{2}\alpha_i(2k_i + (2-\alpha_i)(l_{i-1}+l_i))$	$\frac{1}{2}\alpha_i(2k_i + l_{i-1} + l_i)$	$\frac{1}{2}\alpha_i(2k_i + \alpha_i(l_{i-1}+l_i))$
$y_{\mathrm{op},i}^*$	o_i	$o_i + \frac{\alpha_i-\alpha_{i+1}}{6(\alpha_i+\alpha_{i+1})}l_i$	$o_i + \frac{1}{2}(\alpha_i-\alpha_{i+1})l_i$
$S_{\mathrm{op},i}$	$(1-\min(\alpha_i,\alpha_{i+1},0.5))\min(\alpha_i,\alpha_{i+1},0.5)l_i$	$\frac{\alpha_i\alpha_{i+1}l_i}{2(\alpha_i+\alpha_{i+1})}$	$\frac{1}{4}l_i(\max(\alpha_i+\alpha_{i+1}-1,0))^2$

Table 8.2: Formulae for the centers of gravity y_i^* and $y_{\mathrm{op},i}^*$ and areas S_i and $S_{\mathrm{op},i}$ of the adapted membership functions and overlapping parts in Eq. (8.1) for models holding specific properties

	$T_{\mathbf{M}}$	$T_{\mathbf{P}}$	$T_{\mathbf{L}}$
	$(\forall s \in \{1,\ldots,r\})(i_s \notin \{1,n\})\,,(\exists l>0)(\forall i \in I \setminus \{n\})(l_i = l)$		
y_i^*	c_i	c_i	c_i
S_i	$\alpha_i(k_i + (2-\alpha_i)l)$	$\alpha_i(k_i + l)$	$\alpha_i(k_i + \alpha_i l)$
$y_{\mathrm{op},i}^*$	o_i	$o_i + \frac{\alpha_i-\alpha_{i+1}}{6(\alpha_i+\alpha_{i+1})}l$	$o_i + \frac{1}{2}(\alpha_i-\alpha_{i+1})l$
$S_{\mathrm{op},i}$	$(1-\min(\alpha_i,\alpha_{i+1},0.5))\min(\alpha_i,\alpha_{i+1},0.5)l$	$\frac{\alpha_i\alpha_{i+1}l}{2(\alpha_i+\alpha_{i+1})}$	$\frac{1}{4}l(\max(\alpha_i+\alpha_{i+1}-1,0))^2$
	$(\forall s \in \{1,\ldots,r\})(i_s \notin \{1,n\})\,,(\exists l>0)(\forall i \in I \setminus \{n\})(l_i = l)\,,(\exists k \geq 0)(\forall i \in I)(k_i = k)$		
S_i	$\alpha_i(k + (2-\alpha_i)l)$	$\alpha_i(k + l)$	$\alpha_i(k + \alpha_i l)$



(a) $\gamma_1 \leq 0.5$ (b) $\gamma_1 \geq 0.5$

Figure 8.1: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to the same linguistic output value A_i for $T = T_{\mathbf{M}}$.

model. A rule base of a model with more than one input variable, however, might contain a pair of non-redundant rules R_{s_1} and R_{s_2} with the same linguistic values for all but one input variable X_{l_1} in their antecedents, adjacent linguistic values $B^{l_1}_{j_{l_1}}$ and $B^{l_1}_{j_{l_1}+1}$ for the input variable X_{l_1}, and the same linguistic output value in their consequents. For input values belonging to the kernels of the linguistic values $B^{l}_{j_l}$ ($l \neq l_1$) and partially belonging to $B^{l_1}_{j_{l_1}}$ and $B^{l_1}_{j_{l_1}+1}$, the model behaviour then corresponds to the above *constant* case.

When monotonicity should be guaranteed for any monotone smooth rule base, the first two cases should be considered, while for any monotone rule base, all three cases should be considered.

8.3.1 Models applying $T_{\mathbf{M}}$

When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are mapped to a same linguistic output value A_i (Fig. 8.1), the crisp output y^*_{COG} is computed with Eq. (8.1) using the formulae in Table 8.1. Since α_i is equal to $(1-\gamma_1)$ for $\gamma_1 \in [0, 0.5]$ and equal to γ_1 for $\gamma_1 \in [0.5, 1]$, monotonicity holds if

$$\left(\forall \gamma_1 \in [0, 0.5]\right)\left(\frac{dy^*_{\mathrm{COG},\mathbf{1M},11}}{d\gamma_1} \geq 0\right) \wedge \left(\forall \gamma_1 \in [0.5, 1]\right)\left(\frac{dy^*_{\mathrm{COG},\mathbf{1M},12}}{d\gamma_1} \geq 0\right), \tag{8.2}$$

with

$$y^*_{\mathrm{COG},\mathbf{1M},11} = c_i + \frac{(l_i - l_{i\text{-}1})(3(1+\gamma_1)k_i + 2(1+\gamma_1+\gamma_1^2)(l_{i\text{-}1}+l_i))}{6(2k_i + (1+\gamma_1)(l_{i\text{-}1}+l_i))}\,, \tag{8.3}$$

$$y^*_{\mathrm{COG},\mathbf{1M},12} = c_i + \frac{(l_i - l_{i\text{-}1})(3(2-\gamma_1)k_i + 2(3-3\gamma_1+\gamma_1^2)(l_{i\text{-}1}+l_i))}{6(2k_i + (2-\gamma_1)(l_{i\text{-}1}+l_i))}\,. \tag{8.4}$$

One easily verifies by substituting $\gamma_1 = 0.5$ in Eqs. (8.3–8.4) that $y^*_{\mathrm{COG},\mathbf{1M},11}(\gamma_1 = 0.5) = y^*_{\mathrm{COG},\mathbf{1M},12}(\gamma_1 = 0.5)$, and as the derivatives of

$y^*_{\mathrm{COG},\mathbf{1M},11}$ and $y^*_{\mathrm{COG},\mathbf{1M},12}$ are given by

$$\frac{dy^*_{\mathrm{COG},\mathbf{1M},11}}{d\gamma_1} = \frac{(l_i - l_{i\text{-}1})(k_i + (l_{i\text{-}1} + l_i)\gamma_1)(3k_i + (l_{i\text{-}1} + l_i)(2+\gamma_1))}{3(2k_i + (1+\gamma_1)(l_{i\text{-}1} + l_i))^2}\,, \tag{8.5}$$

$$\frac{dy^*_{\mathrm{COG},\mathbf{1M},12}}{d\gamma_1} = \frac{(l_{i\text{-}1} - l_i)(k_i + (l_{i\text{-}1} + l_i)(1-\gamma_1))(3k_i + (l_{i\text{-}1} + l_i)(3-\gamma_1))}{3(2k_i + (2-\gamma_1)(l_{i\text{-}1} + l_i))^2}\,, \tag{8.6}$$

that Eq. (8.2) is satisfied if and only if

$$l_{i\text{-}1} = l_i\,. \tag{8.7}$$

As the extreme linguistic output values A_1 and A_n are both described by a trapezium with one vertical side, monotonicity can only be guaranteed for a model with a single input variable applying $T_{\mathbf{M}}$ if the following constraints are satisfied

$$(\forall s \in \{1,\ldots,r\})(i_s \notin \{1,n\})\,, \tag{8.8}$$

$$(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)\,. \tag{8.9}$$

From here on, Eqs. (8.8–8.9) are assumed to hold and the formulae in Table 8.2 can be used for the terms y^*_i, $y^*_{\mathrm{op},i}$, S_i and $S_{\mathrm{op},i}$ in Eq. (8.1) when proving the monotonicity in the *smooth* and *non-smooth* case.

When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are mapped to the linguistic output values A_i and A_{i+1} respectively (Fig. 8.2), monotonicity holds if

$$\left(\forall\gamma_1 \in [0,1]\right)\left(\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = \frac{d}{d\gamma_1}\left(\frac{y^*_i S_i + y^*_{i+1}S_{i+1} - y^*_{\mathrm{op},i}S_{\mathrm{op},i}}{S_i + S_{i+1} - S_{\mathrm{op},i}}\right) \geq 0\right). \tag{8.10}$$

Although, the value of the term $\min(\alpha_i, \alpha_{i+1}, 0.5)$ in the formula of $S^*_{\mathrm{op},i}$ differs for $\gamma_1 \in [0, 0.5]$ and $\gamma_1 \in [0.5, 1]$,

$$\min(\alpha_i, \alpha_{i+1}, 0.5) = \begin{cases} \alpha_{i+1} = \gamma_1 & \text{, if } \gamma_1 \in [0,0.5]\,, \\ \alpha_i = 1-\gamma_1 & \text{, if } \gamma_1 \in [0.5,1]\,, \end{cases} \tag{8.11}$$

the same equation can be used for the area $S_{\mathrm{op},i}$ for $\gamma_1 \in [0,1]$:

$$S_{\mathrm{op},i} = (1 - \min(\alpha_i, \alpha_{i+1}, 0.5))\min(\alpha_i, \alpha_{i+1}, 0.5)l = (1-\gamma_1)\gamma_1 l\,. \tag{8.12}$$

Thus for $\gamma_1 \in [0,1]$, using

$$c_i = o_i - \frac{1}{2}k_i - \frac{1}{2}l\,, \tag{8.13}$$

$$c_{i+1} = o_i + \frac{1}{2}k_{i+1} + \frac{1}{2}l\,, \tag{8.14}$$

the crisp output y^*_{COG} is given by

$$y^*_{\mathrm{COG}} = o_i + \frac{l(k_i - k_{i+1})\gamma_1^2 + ((l+k_i)k_i + (l+k_{i+1})(2l+k_{i+1}))\gamma_1 - (l+k_i)^2}{2(-l\gamma_1^2 + (l - k_i + k_{i+1})\gamma_1 + l + k_i)}\,, \tag{8.15}$$

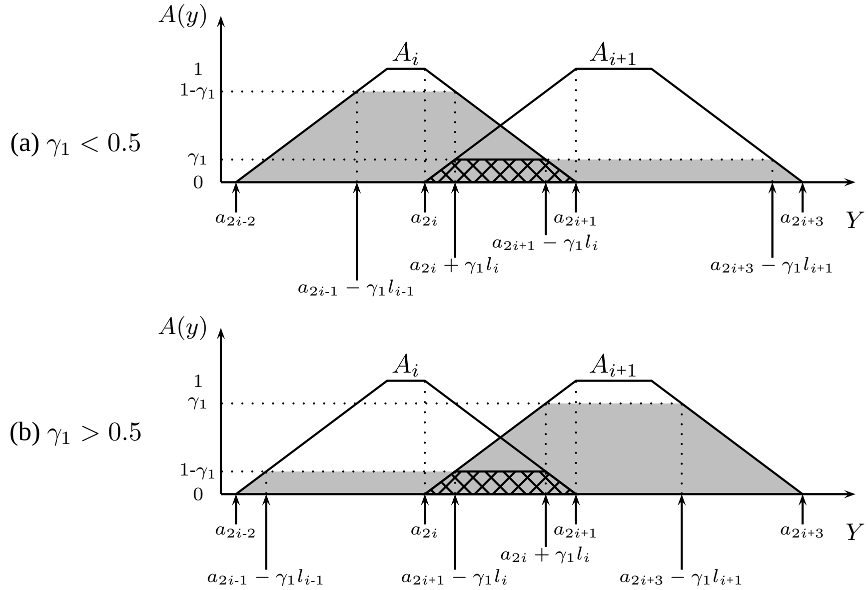


Figure 8.2: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to two consecutive linguistic output values A_i and A_{i+1} for $T = T_{\mathbf{M}}$.

and monotonicity is guaranteed as

$$\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = \frac{(l + k_i)(l + k_{i+1})(2\gamma_1(1-\gamma_1)l + 3l + k_i + k_{i+1})}{2((\gamma_1(1-\gamma_1)+1)l + (1-\gamma_1)k_i + k_{i+1}\gamma_1)^2} \geq 0\,. \tag{8.16}$$

When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are mapped to two non-consecutive output values A_i and A_{i+p} ($p \in \mathbb{N}$, $p > 1$, $i + p \leq n$) respectively (Fig. 8.3), monotonicity holds if

$$\Big(\forall \gamma_1 \in [0,1]\Big)\left(\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = \frac{d}{d\gamma_1}\left(\frac{y^*_i S_i + y^*_{i+p} S_{i+p}}{S_i + S_{i+p}}\right) \geq 0\right). \tag{8.17}$$

The crisp output y^*_{COG} is given by

$$y^*_{\mathrm{COG}} = \frac{(1-\gamma_1)((1+\gamma_1)l + k_i)c_i + \gamma_1((2-\gamma_1)l + k_{i+p})c_{i+p}}{(1-\gamma_1)((1+\gamma_1)l + k_i) + \gamma_1((2-\gamma_1)l + k_{i+p})} \tag{8.18}$$

or, with $c_{i+p} = c_i + p\,l + \frac{1}{2}k_i + \sum\limits_{j=i+1}^{i+p-1} k_j + \frac{1}{2}k_{i+p}$,

$$y^*_{\mathrm{COG}} = c_i + \frac{\Big(p\,l + \frac{1}{2}k_i + \sum\limits_{j=i+1}^{i+p-1} k_j + \frac{1}{2}k_{i+p}\Big)\gamma_1\Big((2-\gamma_1)l + k_{i+p}\Big)}{(2(1-\gamma_1)\gamma_1 + 1)l + (1-\gamma_1)k_i + \gamma_1 k_{i+p}}\,, \tag{8.19}$$

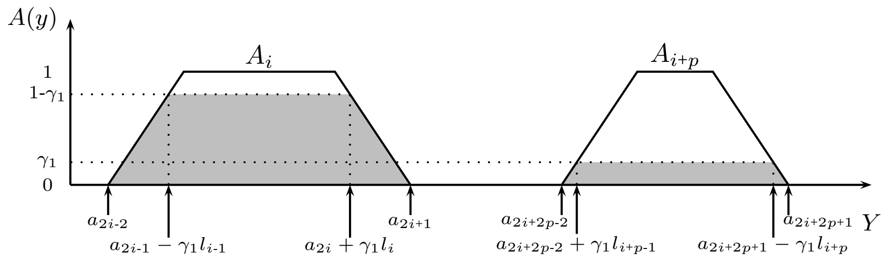


Figure 8.3: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to two non-consecutive linguistic output values A_i and A_{i+p} for $T = T_{\mathbf{M}}$.

and monotonicity is guaranteed as its derivative is positive for all $l \in \mathbb{R}_0^+$, $k_i, k_{i+p} \in \mathbb{R}^+$,

$$\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = \Big(p\,l + \frac{1}{2}k_i + \sum_{j=i+1}^{i+p-1} k_j + \frac{1}{2}k_{i+p}\Big)\times \\ \frac{(2l + k_i + k_{i+p})l\gamma_1^2 + (l + k_i)(2(1-\gamma_1)l + k_{i+p})}{2((2(1-\gamma_1)\gamma_1 + 1)l + (1-\gamma_1)k_i + \gamma_1 k_{i+p})^2} \geq 0\,. \tag{8.20}$$

8.3.2 Models applying $T_{\mathbf{P}}$

When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are mapped to a same linguistic output value A_i, the crisp output y^*_{COG} is constant for all inputs larger than the lower bound of the kernel of B^1_j and smaller than the upper bound of the kernel of B^1_{j+1} as the abscissa of the vertices of the adapted membership function coincide with the abscissa of the original output membership function as shown in Fig. 8.4. Thus, as the crisp output y^*_{COG} is independent of γ_1 (Table 8.1)

$$y^*_{\mathrm{COG}} = c_i + \frac{(l_i - l_{i\text{-}1})(3k_i + 2l_{i\text{-}1} + 2l_i)}{6(2k_i + l_{i\text{-}1} + l_i)}\,, \tag{8.21}$$

monotonicity is guaranteed for any fuzzy output partition

$$\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = 0\,. \tag{8.22}$$

When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are mapped to the linguistic output values A_i and A_{i+1} respectively (Fig. 8.5), monotonicity holds if

$$\Big(\forall \gamma_1 \in [0,1]\Big)\left(\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = \frac{d}{d\gamma_1}\left(\frac{y^*_i S_i + y^*_{i+1} S_{i+1} - y^*_{\mathrm{op},i} S_{\mathrm{op},i}}{S_i + S_{i+1} - S_{\mathrm{op},i}}\right) \geq 0\right). \tag{8.23}$$

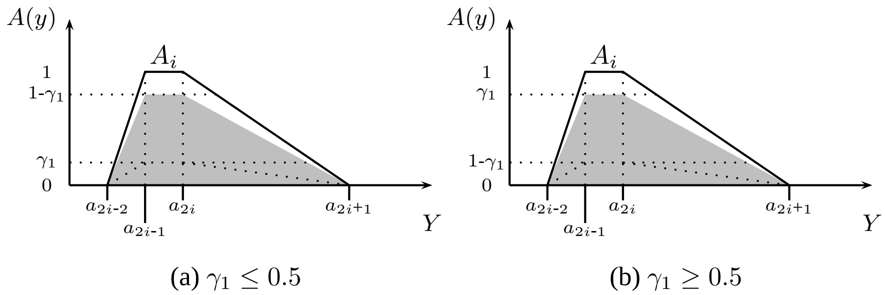


Figure 8.4: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to a same linguistic output value A_i for $T = T_{\mathbf{P}}$.

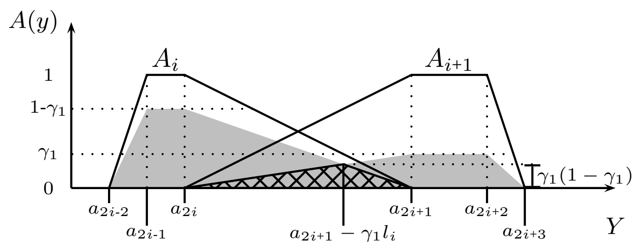


Figure 8.5: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to two consecutive linguistic output values A_i and A_{i+1} for $T = T_{\mathbf{P}}$.

By expressing the midpoints c_i and c_{i+1} of the kernel of the membership functions A_i and A_{i+1} as a function of o_i

$$c_i = o_i - \frac{1}{2}k_i - \frac{1}{2}l_i\,, \tag{8.24}$$

$$c_{i+1} = o_i + \frac{1}{2}k_{i+1} + \frac{1}{2}l_i\,, \tag{8.25}$$

the following expression is obtained for the crisp output y^*_{COG}

$$\begin{aligned} y^*_{\mathrm{COG}} = o_i + \Big[& l_i^2(3-2\gamma_1)\gamma_1^2 + \big(3l_il_{i+1} + 6l_ik_{i+1} + 2l_{i+1}^2 + 6l_{i+1}k_{i+1} + 6k_{i+1}^2\big)\gamma_1 \\ & - (2l_{i-1}^2 + 3l_{i-1}l_i + 6l_{i-1}k_i + l_i^2 + 6l_ik_i + 6k_i^2)(1-\gamma_1)\Big] \times \\ & \Big[6(l_i\gamma_1^2 + (l_{i+1} + 2k_{i+1})\gamma_1 + (l_{i-1} + l_i + 2k_i)(1-\gamma_1))\Big]^{-1}\,, \end{aligned} \tag{8.26}$$

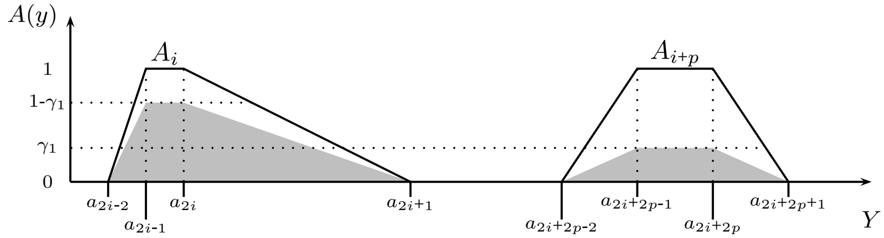


Figure 8.6: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to two non-consecutive linguistic output values A_i and A_{i+p} for $T = T_{\mathbf{P}}$.

and monotonicity is guaranteed as the derivative of y^*_{COG} is positive for all $l \in \mathbb{R}_0^+$, $k_i, k_{i+1} \in \mathbb{R}^+$,

$$\begin{aligned}\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = \Big[&(2-\gamma_1)\gamma_1 l_{i-1}^2 l_i + l_{i-1}^2 l_{i+1} + 2l_{i-1}^2 k_{i+1} + 2(\gamma_1^2 + 3(1-\gamma_1))\gamma_1 l_{i-1} l_i^2 \\ &+ 3l_{i-1}l_i l_{i+1} + 3(2-\gamma_1)\gamma_1 l_{i-1} l_i k_i + 6l_{i-1} l_i k_{i+1} + l_{i-1} l_{i+1}^2 \\ &+ 3l_{i-1} l_{i+1} k_i + 3l_{i-1} l_{i+1} k_{i+1} + 6l_{i-1} k_i k_{i+1} + 3l_{i-1} k_{i+1}^2 \\ &+ (\gamma_1^2 - \gamma_1 + 4)(1-\gamma_1)\gamma_1 l_i^3 + 2(1-\gamma_1^3) l_i^2 l_{i+1} \\ &+ 4(\gamma_1^2 + 3(1-\gamma_1))\gamma_1 l_i^2 k_i + 4(1-\gamma_1^3) l_i^2 k_{i+1} + (1-\gamma_1^2) l_i l_{i+1}^2 \\ &+ 6l_i l_{i+1} k_i + 3(1-\gamma_1^2) l_i l_{i+1} k_{i+1} + 3(2-\gamma_1)\gamma_1 l_i k_i^2 + 12 l_i k_i k_{i+1} \\ &+ 3(1-\gamma_1^2) l_i k_{i+1}^2 + 2l_{i+1}^2 k_i + 3l_{i+1} k_i^2 + 6l_{i+1} k_i k_{i+1} + 6k_i^2 k_{i+1} \\ &+ 6k_i k_{i+1}^2\Big] \times \\ &\Big[6(l_i\gamma_1^2 + (l_{i+1} + 2k_{i+1})\gamma_1 + (l_{i-1} + l_i + 2k_i)(1-\gamma_1))^2\Big]^{-1} \\ \geq 0\,. \end{aligned} \tag{8.27}$$

When B_j^1 and B_{j+1}^1 ($j \in J_1 \setminus \{n_1\}$) are mapped to two non-consecutive output values A_i and A_{i+p} ($p \in \mathbb{N}$, $p > 1$, $i + p \leq n$) respectively (Fig. 8.6), monotonicity holds if

$$\left(\forall \gamma_1 \in [0,1]\right)\left(\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = \frac{d}{d\gamma_1}\left(\frac{y_i^* S_i + y_{i+p}^* S_{i+p}}{S_i + S_{i+p}}\right) \geq 0\right). \tag{8.28}$$

The centers of gravity y_i^* and y_{i+p}^* of the adapted membership functions A_i' and A_{i+p}' are respectively equal to the abscissa of the center of gravity of the trapezia defining A_i and A_{i+p}, *i.e.* the centers of gravity y_i^* and y_{i+p}^* are independent of γ_1. The abscissa of the center of gravity of a trapezium is always an element of its base, thus the center of gravity y_i^* of the adapted membership function A_i' is smaller than the center of gravity y^*_{i+p} of the adapted membership function A'_{i+p}

$$a_{2i\text{-}2} < y^*_i < a_{2i+1} \leq a_{2i+2p\text{-}2} < y^*_{i+p} < a_{2i+2p+1}\,, \tag{8.29}$$

and y^*_{i+p} in the equation of the crisp output y^*_{COG} can be substituted by a function of y^*_i

$$y^*_{i+p} = y^*_i + C \;\; (C \in \mathbb{R}^+_0)\,. \tag{8.30}$$

The crisp model output y^*_{COG} is given by

$$\begin{aligned} y^*_{\mathrm{COG}} &= y^*_i + C\frac{S_{i+p}}{S_i + S_{i+p}} \\ &= y^*_i + C\frac{(l_{i+p\text{-}1} + l_{i+p} + 2k_{i+p})\gamma_1}{(l_{i\text{-}1} + l_i + 2k_i)(1-\gamma_1) + (l_{i+p\text{-}1} + l_{i+p} + 2k_{i+p})\gamma_1}\,, \end{aligned} \tag{8.31}$$

and monotonicity is guaranteed as its derivative is positive for all $l \in \mathbb{R}^+_0$, $k_i, k_{i+p} \in \mathbb{R}^+$,

$$\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = \frac{C(l_{i\text{-}1} + l_i + 2k_i)(l_{i+p\text{-}1} + l_{i+p} + 2k_{i+p})}{((l_{i\text{-}1} + l_i + 2k_i)(1-\gamma_1) + (l_{i+p\text{-}1} + l_{i+p} + 2k_{i+p})\gamma_1)^2} \geq 0\,. \tag{8.32}$$

8.3.3 Models applying $T_{\mathbf{L}}$

When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are mapped to a same linguistic output value A_i (Fig. 8.7), the crisp output y^*_{COG} is computed with Eq. (8.1) using the formulae in Table 8.1. Since α_i is equal to $(1-\gamma_1)$ for $\gamma_1 \in [0, 0.5]$ and equal to γ_1 for $\gamma_1 \in [0.5, 1]$, monotonicity holds if

$$\left(\forall \gamma_1 \in [0, 0.5]\right)\left(\frac{dy^*_{\mathrm{COG},\mathbf{1L},11}}{d\gamma_1} \geq 0\right) \wedge \left(\forall \gamma_1 \in [0.5, 1]\right)\left(\frac{dy^*_{\mathrm{COG},\mathbf{1L},12}}{d\gamma_1} \geq 0\right), \tag{8.33}$$

with

$$y^*_{\mathrm{COG},\mathbf{1L},11} = c_i + \frac{(l_i - l_{i\text{-}1})(3k_i + 2(1-\gamma_1)(l_{i\text{-}1} + l_i))(1-\gamma_1)}{6(2k_i + (1-\gamma_1)(l_{i\text{-}1} + l_i))}\,, \tag{8.34}$$

$$y^*_{\mathrm{COG},\mathbf{1L},12} = c_i + \frac{(l_i - l_{i\text{-}1})(3k_i + 2\gamma_1(l_{i\text{-}1} + l_i))\gamma_1}{6(2k_i + \gamma_1(l_{i\text{-}1} + l_i))}\,. \tag{8.35}$$

One easily verifies by substituting $\gamma_1 = 0.5$ in Eqs. (8.34–8.35) that $y^*_{\mathrm{COG},\mathbf{1L},11}(\gamma_1 = 0.5) = y^*_{\mathrm{COG},\mathbf{1L},12}(\gamma_1 = 0.5)$, and as the derivatives of $y^*_{\mathrm{COG},\mathbf{1L},11}$ and $y^*_{\mathrm{COG},\mathbf{1L},12}$ are given by

$$\frac{d(y^*_{\mathrm{COG},\mathbf{1L},11})}{d\gamma_1} = \frac{(l_{i\text{-}1} - l_i)((l_{i\text{-}1} + l_i)(1-\gamma_1) + k_i)((l_{i\text{-}1} + l_i)(1-\gamma_1) + 3k_i)}{3((l_{i\text{-}1} + l_i)(1-\gamma_1) + 2k_i)^2}\,, \tag{8.36}$$

$$\frac{d(y^*_{\mathrm{COG},\mathbf{1L},12})}{d\gamma_1} = \frac{(l_i - l_{i\text{-}1})((l_{i\text{-}1} + l_i)\gamma_1 + k_i)((l_{i\text{-}1} + l_i)\gamma_1 + 3k_i)}{3((l_{i\text{-}1} + l_i)\gamma_1 + 2k_i)^2}\,, \tag{8.37}$$

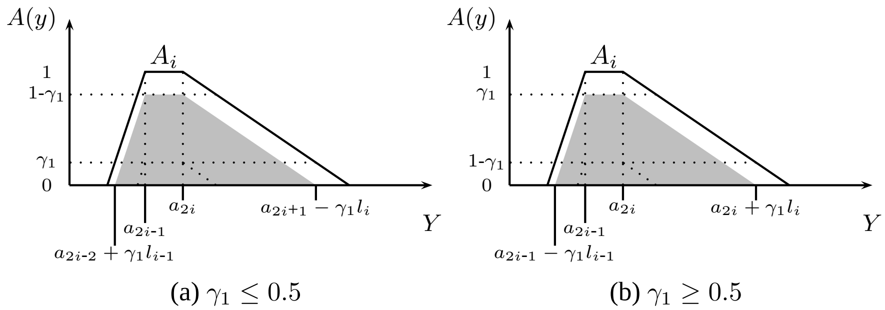


Figure 8.7: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to a same linguistic output value A_i for $T = T_{\mathbf{L}}$.

that Eq. (8.33) is satisfied if and only if

$$l_{i\text{-}1} = l_i \,. \tag{8.38}$$

As the extreme linguistic output values A_1 and A_n are both described by a trapezium with one vertical side, monotonicity can only be guaranteed for a model with a single input variable applying $T_{\mathbf{L}}$ if the following constraints are satisfied

$$(\forall s \in \{1, \ldots, r\})(i_s \notin \{1, n\}) \,, \tag{8.39}$$
$$(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l) \,. \tag{8.40}$$

From here on, Eqs. (8.39–8.40) are assumed to hold and the formulae in Table 8.2 can be used for the terms y_i^*, $y_{\text{op},i}^*$, S_i and $S_{\text{op},i}$ in Eq. (8.1) when proving the monotonicity in the *smooth* and *non-smooth* case.

When B_j^1 and B_{j+1}^1 ($j \in J_1 \setminus \{n_1\}$) are mapped to the linguistic output values A_i and A_{i+1} respectively, monotonicity holds if

$$\left(\forall \gamma_1 \in [0,1]\right)\left(\frac{dy_{\text{COG}}^*}{d\gamma_1} = \frac{d}{d\gamma_1}\left(\frac{y_i^* S_i + y_{i+1}^* S_{i+1} - y_{\text{op},i}^* S_{\text{op},i}}{S_i + S_{i+1} - S_{\text{op},i}}\right) \geq 0\right). \tag{8.41}$$

As illustrated by Fig. 8.8, the adapted membership functions A_i' and A_{i+1}' do not overlap

$$S_{\text{op},i} = \frac{1}{4} l(\max(\alpha_i + \alpha_{i+1} - 1, 0))^2 = \frac{1}{4} l(\max((1-\gamma_1) + \gamma_1 - 1, 0))^2 = 0 \,. \tag{8.42}$$

and the *smooth* case can be treated as a special case of the *non-smooth* case.

When B_j^1 and B_{j+1}^1 ($j \in J_1 \setminus \{n_1\}$) are mapped to two non-consecutive output values A_i and A_{i+p} ($p \in \mathbb{N}_0$, $i + p \leq n$) respectively (Fig. 8.9), monotonicity holds if

$$\left(\forall \gamma_1 \in [0,1]\right)\left(\frac{dy_{\text{COG}}^*}{d\gamma_1} = \frac{d}{d\gamma_1}\left(\frac{y_i^* S_i + y_{i+p}^* S_{i+p}}{S_i + S_{i+p}}\right) \geq 0\right). \tag{8.43}$$

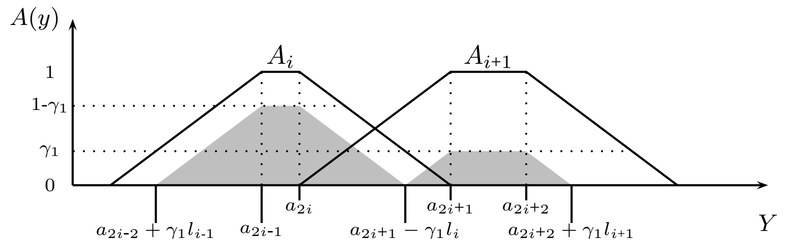


Figure 8.8: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to two consecutive linguistic output values A_i and A_{i+1} for $T = T_{\mathbf{L}}$.

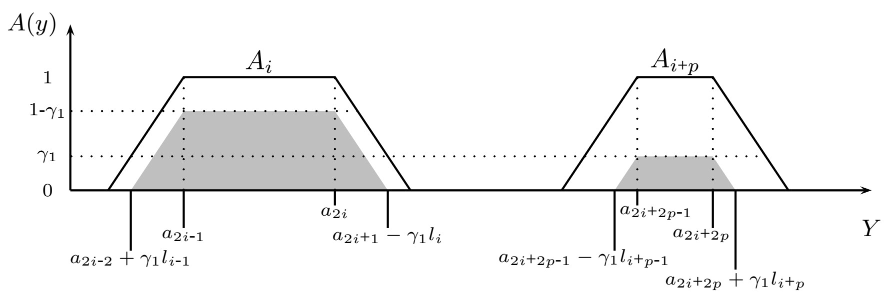


Figure 8.9: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to two non-consecutive linguistic output values A_i and A_{i+p} for $T = T_{\mathbf{L}}$.

The crisp output y^*_{COG} is given by

$$y^*_{\mathrm{COG}} = \frac{(1-\gamma_1)((1-\gamma_1)l + k_i)c_i + \gamma_1(\gamma_1 l + k_{i+p})c_{i+p}}{(1-\gamma_1)((1-\gamma_1)l + k_i) + \gamma_1(\gamma_1 l + k_{i+p})}\,, \tag{8.44}$$

or, with $c_{i+p} = c_i + p\,l - \frac{1}{2}k_i + \sum\limits_{j=i}^{i+p} k_j - \frac{1}{2}k_{i+p}$

$$y^*_{\mathrm{COG}} = c_i + \frac{\Big(p\,l - \frac{1}{2}k_i + \sum\limits_{j=i}^{i+p} k_j - \frac{1}{2}k_{i+p}\Big)\gamma_1\Big(\gamma_1 l + k_{i+p}\Big)}{(2\gamma_1^2 - 2\gamma_1 + 1)l + (1-\gamma_1)k_i + \gamma_1 k_{i+p}}\,, \tag{8.45}$$

and monotonicity is guaranteed as its derivative is positive for all $l \in \mathbb{R}_0^+$, $k_i, k_{i+p} \in$

$\mathbb{R}^+$,

$$\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = \Big(p\,l - \frac{1}{2}k_i + \sum_{j=i}^{i+p} k_j - \frac{1}{2}k_{i+p}\Big)$$
$$\frac{2l^2(1-\gamma_1)\gamma_1 + lk_i(2-\gamma_1)\gamma_1 + lk_{i+p}(1-\gamma_1^2) + k_i k_{i+p}}{((2\gamma_1^2 - 2\gamma_1 + 1)l + (1-\gamma_1)k_i + \gamma_1 k_{i+p})^2} \geq 0\,. \quad (8.46)$$

8.4 Models with two input variables

In this section the monotonicity of models with two input variables applying either $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$ is discussed. Models with more than one input variable applying $T_{\mathbf{L}}$ are not considered since they return the empty set as fuzzy model output for some inputs as discussed in detail in Section 7.4. The results obtained for models with a single input variable also apply to models with two input variables, as the latter behave as a 'single input model' in parts of their input space. Therefore, for models applying $T_{\mathbf{M}}$ the output membership functions used in the consequents of the rules are assumed to have intervals of changing membership degree of equal length. For models applying $T_{\mathbf{P}}$ no additional model properties were required to guarantee the monotonicity of models with a single input variable.

8.4.1 Models applying $T_{\mathbf{M}}$

As shown by the counterexample below, monotonicity cannot be guaranteed for any monotone rule base, nor for any monotone smooth rule base, if combining the t-norm $T_{\mathbf{M}}$ with the COG defuzzification method in models with two input variables.

The set of four rules represented in Fig. 8.10

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}

can occur in a monotone smooth rule base as well as in a monotone rule base. For inputs $\mathbf{x} = (x_1, x_2)$ not firing any other rule than the four rules above

$$\gamma_1 = 1 - B^1_{j_1}(x_1) = B^1_{j_1+1}(x_1)\,, \quad (8.47)$$
$$\gamma_2 = 1 - B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2)\,, \quad (8.48)$$

the crisp output y^*_{COG} (Eq. (8.1) with Table 8.1) is given by

$$y^*_{\mathrm{COG}} = \frac{c_i((2-\alpha_i)l + k_i)\alpha_i + c_{i+1}((2-\alpha_{i+1})l + k_{i+1})\alpha_{i+1} - o_i(1-\alpha')\alpha' l}{((2-\alpha_i)l + k_i)\alpha_i + ((2-\alpha_{i+1})l + k_{i+1})\alpha_{i+1} - (1-\alpha')\alpha' l}\,,$$

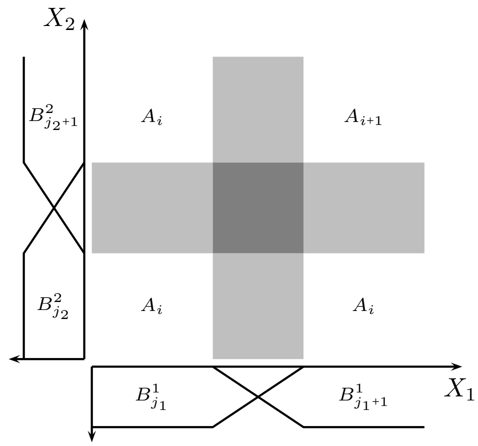
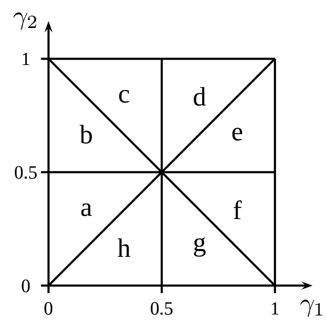


Figure 8.10: Schematic representation of the rules for which a non-monotone input-output behaviour is obtained when applying $T_{\mathbf{M}}$ combined with the COG defuzzification method.

Table 8.3: Values taken by the fulfilment degrees α_i and α_{i+1} and $\min(\alpha_i, \alpha_{i+1}, 0.5)$ in different regions of the input space

	α_i	α_{i+1}	$\min(\alpha_i, \alpha_{i+1}, 0.5)$
a	$1-\gamma_2$	γ_1	γ_1
b	γ_2	γ_1	γ_1
c	$1-\gamma_1$	γ_1	γ_1
d	$1-\gamma_1$	γ_1	$1-\gamma_1$
e	$1-\gamma_2$	γ_2	$1-\gamma_2$
f	$1-\gamma_2$	γ_2	γ_2
g	γ_1	γ_2	γ_2
h	$1-\gamma_1$	γ_2	γ_2



with

$$\alpha_i = \max(\min(1-\gamma_1, 1-\gamma_2), \min(1-\gamma_1, \gamma_2), \min(\gamma_1, 1-\gamma_2)) \,, \tag{8.49}$$

$$\alpha_{i+1} = \min(\gamma_1, \gamma_2) \,, \tag{8.50}$$

$$\alpha' = \min(\alpha_i, \alpha_{i+1}, 0.5) \,. \tag{8.51}$$

In Table 8.3 an overview is given of the values taken by the fulfilment degrees α_i and α_{i+1} and the term $\min(\alpha_i, \alpha_{i+1}, 0.5)$ in different regions of the input space.

For inputs $\mathbf{x}$ having the following membership degrees to the linguistic values in the antecedents of the rules (Case b in Table 8.3)

$$\gamma_1 < 1 - \gamma_2 \qquad \wedge \qquad \gamma_2 > 0.5\,,$$

the fulfilment degrees α_i and α_{i+1} of the linguistic output values A_i and A_{i+1} are

$$\alpha_i = \gamma_2\,, \quad \alpha_{i+1} = \gamma_1\,, \quad \text{and} \quad \min(\alpha_i, \alpha_{i+1}, 0.5) = \gamma_1\,.$$

In this case, the crisp output y^*_{COG} is given by

$$y^*_{\mathrm{COG}} = \frac{c_i((2-\gamma_2)l + k_i)\gamma_2 + c_{i+1}((2-\gamma_1)l + k_{i+1})\gamma_1 - o_i l(1-\gamma_1)\gamma_1}{((2-\gamma_2)l + k_i)\gamma_2 + ((2-\gamma_1)l + k_{i+1})\gamma_1 - l(1-\gamma_1)\gamma_1}\,, \tag{8.52}$$

or, after substituting c_i and c_{i+1}

$$c_i = o_i - \frac{1}{2}k_i - \frac{1}{2}l\,, \tag{8.53}$$

$$c_{i+1} = o_i + \frac{1}{2}k_{i+1} + \frac{1}{2}l\,, \tag{8.54}$$

by

$$y^*_{\mathrm{COG}} = o_i + \frac{(l + k_{i+1})((2-\gamma_1)l + k_{i+1})\gamma_1 - (l + k_i)((2-\gamma_2)l + k_i)\gamma_2}{2(((2-\gamma_2)l + k_i)\gamma_2 + (l + k_{i+1})\gamma_1)}\,. \tag{8.55}$$

A non-monotone input-output behaviour is obtained for any fuzzy output partition as the derivative of y^*_{COG} to γ_2 is negative for all $l \in \mathbb{R}^+_0$, $k_i, k_{i+1} \in \mathbb{R}^+$,

$$\begin{aligned}
\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} &= \frac{l + k_{i+1}}{2(((2-\gamma_2)l + k_i)\gamma_2 + (l + k_{i+1})\gamma_1)^2} \\
&\quad \big[l^2((\gamma_2 - \gamma_1^2) + 2\gamma_2(1-\gamma_1)(2-\gamma_2) + \gamma_2(1-\gamma_2)) \\
&\quad + l k_i(5 - 2\gamma_1 - \gamma_2) + l k_{i+1}(\gamma_2(1-\gamma_2) + (\gamma_2 - \gamma_1^2)) + k_i^2\gamma_2\big] \\
&> 0\,, \qquad (8.56) \\
\frac{dy^*_{\mathrm{COG}}}{d\gamma_2} &= -\frac{(l + k_{i+1})(2l(1-\gamma_2) + k_i)(l(3-\gamma_1) + k_i + k_{i+1})\gamma_1}{2(((2-\gamma_2)l + k_i)\gamma_2 + (l + k_{i+1})\gamma_1)^2} \\
&< 0\,. \qquad (8.57)
\end{aligned}$$

In Fig. 8.11 one can see that non-monotone input-output behaviour is also obtained for inputs $\mathbf{x}$ having the following membership degrees to the linguistic values in the antecedents of the rules (Case g in Table 8.3)

$$\gamma_1 < 1 - \gamma_2 \qquad \wedge \qquad \gamma_1 > 0.5\,.$$

One can easily verify that in this case the derivative of y^*_{COG} to γ_1 is negative for all $l \in \mathbb{R}^+_0$, $k_i, k_{i+1} \in \mathbb{R}^+$.

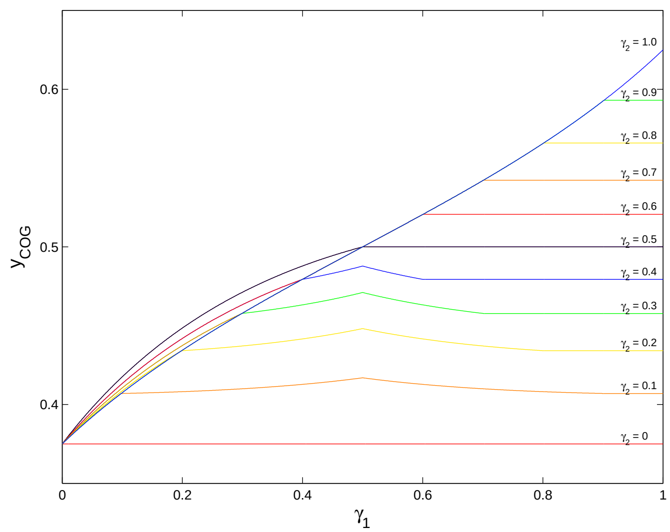


Figure 8.11: Crisp output y^*_{COG} as a function of γ_1 for the rules used in the counterexample when discussing the monotonicity of models with two input variables applying $T_{\mathbf{M}}$ combined with the COG defuzzification method.

8.4.2 Models with a monotone smooth rule base applying $T_{\mathbf{P}}$

It is shown in this section that one will always obtain a monotone input-output behaviour for a model with two input variables and a monotone smooth rule base when applying $T_{\mathbf{P}}$ combined with the COG defuzzification method.

The general representation of a set of four rules that can be fired simultaneously in a model with two input variables X_1 and X_2 is

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS $A_{i+p_1+p_2}$
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_{i+p_1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS $A_{i+p_1+p_2+p_3}$

When the rule base of a model is smooth, the values of p_1, p_2 and p_3 in the rules above are restricted to

$$(p_1, p_2, p_3) \in \{(0,0,0), (0,0,1), (0,1,0), (1,0,0), (1,0,1)\}\,. \tag{8.58}$$

Case I If $(p_1, p_2, p_3) = (0,0,0)$, the four rules

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_i

contain a same linguistic output value A_i in their consequent. As a result, for all inputs $\mathbf{x}$ not firing any other rule than the four rules above (Eqs. (8.47)–(8.48)), only the linguistic output value A_i is fired

$$(\alpha_i > 0)\,, (\forall j \in I \setminus \{i\})(\alpha_j = 0) \tag{8.59}$$

and the crisp output y^*_{COG} (Eq. (8.1) with Table 8.1) is equal to the abscissa of the center of gravity of A_i

$$y^*_{\mathrm{COG}} = c_i + \frac{(l_i - l_{i\text{-}1})(3k_i + 2l_{i\text{-}1} + 2l_i)}{6(2k_i + l_{i\text{-}1} + l_i)}\,. \tag{8.60}$$

As the crisp output y^*_{COG} is independent of α_i, it holds that

$$\frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = 0 \qquad \text{and} \qquad \frac{dy^*_{\mathrm{COG}}}{d\gamma_2} = 0\,, \tag{8.61}$$

and monotonicity is guaranteed for any fuzzy output partition.

Case II The four rules obtained if $(p_1, p_2, p_3) = (0, 0, 1)$

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}

are represented schematically in Fig. 8.12. For all inputs $\mathbf{x}$ not firing any other rule than these four rules (Eqs. (8.47)–(8.48)), the fulfilment degrees of the linguistic output values A_i and A_{i+1} are obtained by

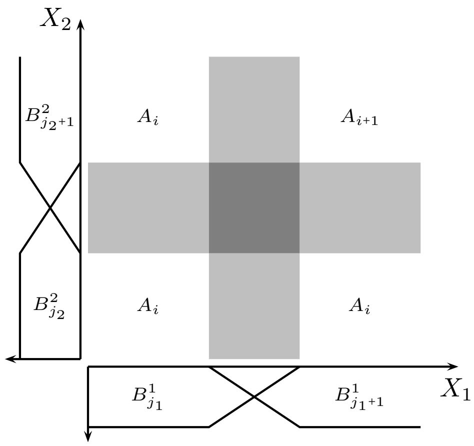
$$\alpha_i = \max((1-\gamma_1)(1-\gamma_2), (1-\gamma_1)\gamma_2, \gamma_1(1-\gamma_2))\,, \tag{8.62}$$

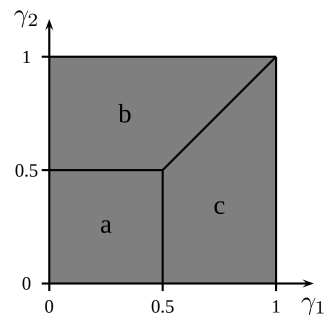
$$\alpha_{i+1} = \gamma_1\gamma_2\,. \tag{8.63}$$

In Fig. 8.12 the different regions of the input space are indicated in which the fulfilment degree α_i is computed by a different function of γ_1 and γ_2.

For Case IIa, the crisp output y^*_{COG} is given by

$$\begin{aligned} y^*_{\mathrm{COG}} = o_i - \Big[&\big[(1-\gamma_1)(1-\gamma_2)(\gamma_1\gamma_2 + (1-\gamma_1)(1-\gamma_2))(2l^2_{i\text{-}1} + 3l_{i\text{-}1}l_i + 6l_{i\text{-}1}k_i \\ &+ 6l_ik_i + 6k_i^2)\big] + \big[(1-\gamma_1-\gamma_2)(5\gamma_1\gamma_2(1-\gamma_1)(1-\gamma_2) + \gamma_1^2 \\ &+ \gamma_2^2 + 2(1-\gamma_1)(1-\gamma_2) - 1)l_i^2\big] - \big[(\gamma_1\gamma_2 + (1-\gamma_1)(1-\gamma_2)) \\ &\gamma_1\gamma_2(3l_il_{i+1} + 6l_ik_{i+1} + 2l^2_{i+1} + 6l_{i+1}k_{i+1} + 6k^2_{i+1})\big]\Big] \times \end{aligned}$$





	α_i	α_{i+1}
a	$(1\text{-}\gamma_1)(1\text{-}\gamma_2)$	$\gamma_1\gamma_2$
b	$(1\text{-}\gamma_1)\gamma_2$	$\gamma_1\gamma_2$
c	$\gamma_1(1\text{-}\gamma_2)$	$\gamma_1\gamma_2$

Figure 8.12: Schematic representation of the rules considered in Case II of the discussion about models with two input variables applying $T_{\mathbf{P}}$ combined with the COG defuzzification method.

$$\begin{aligned}&\Big[6\big[\gamma_1\gamma_2 + (1-\gamma_1)(1-\gamma_2)\big]\big[(1-\gamma_1)(1-\gamma_2)(\gamma_1\gamma_2 + (1-\gamma_1)\\ &\quad (1-\gamma_2))(l_{i\text{-}1} + 2k_i) + (\gamma_1\gamma_2 + (1-\gamma_1)(1-\gamma_2))\gamma_1\gamma_2\\ &\quad (l_{i+1} + 2k_{i+1}) + (3\gamma_1\gamma_2(1-\gamma_1)(1-\gamma_2) + \gamma_1^2 + \gamma_2^2 + 2(1-\gamma_1)\\ &\quad (1-\gamma_2) - 1)l_i\big]\Big]^{-1} .\end{aligned} \tag{8.64}$$

As the crisp output y^*_{COG} is a rather complex function of γ_1 and γ_2, the chain rule will be used to prove that the derivative of y^*_{COG} to γ_1 and γ_2 is positive

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma} = \frac{\partial y^*_{\mathrm{COG}}}{\partial \alpha_i}\frac{\partial \alpha_i}{\partial \gamma} + \frac{\partial y^*_{\mathrm{COG}}}{\partial \alpha_{i+1}}\frac{\partial \alpha_{i+1}}{\partial \gamma} . \tag{8.65}$$

Expressed as a function of α_i and α_{i+1}, the crisp output y^*_{COG} is given by

$$\begin{aligned}y^*_{\mathrm{COG}} = o_i - &\Big[(C_1\alpha_i - C_2\alpha_{i+1})(\alpha_i + \alpha_{i+1})^2 + (\alpha_i - \alpha_{i+1})(\alpha_i^2 + 3\alpha_i\alpha_{i+1} + \alpha_{i+1}^2)\\ &\quad l_i^2\Big] \times \Big[6(\alpha_i + \alpha_{i+1})((\alpha_i + \alpha_{i+1})(C_3\alpha_i + C_4\alpha_{i+1})\\ &\quad + (\alpha_i^2 + \alpha_i\alpha_{i+1} + \alpha_{i+1}^2)l_i)\Big]^{-1} ,\end{aligned} \tag{8.66}$$

with

$$\begin{aligned}
C_1 &= 2l_{i\text{-}1}^2 + 3l_{i\text{-}1}l_i + 6l_{i\text{-}1}k_i + 6l_ik_i + 6k_i^2\,,\\
C_2 &= 3l_il_{i+1} + 6l_ik_{i+1} + 2l_{i+1}^2 + 6l_{i+1}k_{i+1} + 6k_{i+1}^2\,,\\
C_3 &= l_{i\text{-}1} + 2k_i\,,\\
C_4 &= l_{i+1} + 2k_{i+1}\,,
\end{aligned}$$

and its derivatives to α_i and α_{i+1} are

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \alpha_i} = -\alpha_{i+1}\frac{C_5}{C_6}\,, \tag{8.67}$$

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \alpha_{i+1}} = \alpha_i\frac{C_5}{C_6}\,, \tag{8.68}$$

with $C_5, C_6 \in \mathbb{R}_0^+$

$$\begin{aligned}
C_5 = {} & (\alpha_i+\alpha_{i+1})^4(l_{i\text{-}1}^2l_{i+1} + 2l_{i\text{-}1}^2k_{i+1} + 3l_{i\text{-}1}l_il_{i+1} + 6l_{i\text{-}1}l_ik_{i+1} + l_{i\text{-}1}l_{i+1}^2\\
& + 3l_{i\text{-}1}l_{i+1}k_i + 3l_{i\text{-}1}l_{i+1}k_{i+1} + 6l_{i\text{-}1}k_ik_{i+1} + 3l_{i\text{-}1}k_{i+1}^2 + 6l_il_{i+1}k_i\\
& + 12l_ik_ik_{i+1} + 2l_{i+1}^2k_i + 3l_{i+1}k_i^2 + 6l_{i+1}k_ik_{i+1} + 6k_i^2k_{i+1} + 6k_ik_{i+1}^2)\\
& + (\alpha_i+\alpha_{i+1})^2((\alpha_i+2\alpha_{i+1})(l_{i+1}^2 + 3l_{i+1}k_{i+1} + 3k_{i+1}^2)\alpha_i + (2\alpha_i+\alpha_{i+1})\\
& (l_{i\text{-}1}^2 + 3l_{i\text{-}1}k_i + 3k_i^2)\alpha_{i+1})l_i + (\alpha_i+\alpha_{i+1})(2(\alpha_i^2 + 3\alpha_i\alpha_{i+1} + 3\alpha_{i+1}^2)\\
& (l_{i+1}+2k_{i+1})\alpha_i + 2(3\alpha_i^2 + 3\alpha_i\alpha_{i+1} + \alpha_{i+1}^2)(l_{i\text{-}1}+2k_i)\alpha_{i+1})l_i^2\\
& + (4\alpha_i^2 + 7\alpha_i\alpha_{i+1} + 4\alpha_{i+1}^2)\alpha_i\alpha_{i+1}l_i^3\,,\\
C_6 = {} & (\alpha_i+\alpha_{i+1})(\alpha_i(l_{i\text{-}1}+2k_i) + \alpha_{i+1}(l_{i+1}+2k_{i+1}))\\
& + (\alpha_i^2 + \alpha_i\alpha_{i+1} + \alpha_{i+1}^2)l_i\,.
\end{aligned}$$

Thus, the positivity of the derivatives of y^*_{COG} to γ_1 and γ_2 can be restated as

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} \geq 0 \qquad \Leftrightarrow \qquad -\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_1} + \alpha_i\frac{\partial \alpha_{i+1}}{\partial \gamma_1} \geq 0\,, \tag{8.69}$$

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_2} \geq 0 \qquad \Leftrightarrow \qquad -\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_2} + \alpha_i\frac{\partial \alpha_{i+1}}{\partial \gamma_2} \geq 0\,. \tag{8.70}$$

For Case IIa (Fig. 8.12) monotonicity is obtained since

$$-\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_1} + \alpha_i\frac{\partial \alpha_{i+1}}{\partial \gamma_1} = \gamma_1\gamma_2(1-\gamma_2) + (1-\gamma_1)(1-\gamma_2)\gamma_2 = \gamma_2(1-\gamma_2) \geq 0\,, \tag{8.71}$$

$$-\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_2} + \alpha_i\frac{\partial \alpha_{i+1}}{\partial \gamma_2} = \gamma_1\gamma_2(1-\gamma_1) + (1-\gamma_1)(1-\gamma_2)\gamma_1 = \gamma_1(1-\gamma_1) \geq 0\,. \tag{8.72}$$

The equations Eqs. (8.67)–(8.68) of the derivatives of y^*_{COG} to α_i and α_{i+1} hold for all inputs $\mathbf{x}$ only firing rules containing A_i and A_{i+1} in their consequent, if of course, the t-norm $T_{\mathbf{P}}$ is applied. Monotonicity is guaranteed for Case IIb and Case IIc since, if $(\alpha_i, \alpha_{i+1}) = ((1-\gamma_1)\gamma_2, \gamma_1\gamma_2)$

$$-\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_1} + \alpha_i \frac{\partial \alpha_{i+1}}{\partial \gamma_1} = \gamma_1\gamma_2\gamma_2 + (1-\gamma_1)\gamma_2\gamma_2 = \gamma_2^2 \geq 0\,, \tag{8.73}$$

$$-\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_2} + \alpha_i \frac{\partial \alpha_{i+1}}{\partial \gamma_2} = -\gamma_1\gamma_2(1-\gamma_1) + (1-\gamma_1)\gamma_2\gamma_1 = 0\,, \tag{8.74}$$

and, if $(\alpha_i, \alpha_{i+1}) = (\gamma_1(1-\gamma_2), \gamma_1\gamma_2)$

$$-\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_1} + \alpha_i \frac{\partial \alpha_{i+1}}{\partial \gamma_1} = -\gamma_1\gamma_2(1-\gamma_2) + \gamma_1(1-\gamma_2)\gamma_2 = 0\,, \tag{8.75}$$

$$-\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_2} + \alpha_i \frac{\partial \alpha_{i+1}}{\partial \gamma_2} = \gamma_1\gamma_2\gamma_1 + \gamma_1(1-\gamma_2)\gamma_1 = \gamma_1^2 \geq 0\,. \tag{8.76}$$

Note that if the fulfilment degrees α_i and α_{i+1} can be written as

$$\alpha_i = (1-a)b \qquad \alpha_{i+1} = a\,b\,, \tag{8.77}$$

as for instance in Case IIb and Case IIc, the crisp output y^*_{COG} is independent of b ($C_1, C_2, C_3, C_4 \in \mathbb{R}_0^+$ are defined in Eq. (8.66))

$$\begin{aligned} y^*_{\mathrm{COG}} &= o_i + \frac{(C_1(1-a)b - C_2ab)b^2 + (1-2a)b(1+a-a^2)b^2l_i^2}{6b(b(C_3(1-a)b + C_4ab) + (1-a+a^2)b^2l_i)} \\ &= o_i + \frac{C_1(1-a) - C_2a + (1-2a)(1+a-a^2)l_i^2}{6(C_3(1-a) + C_4a + (1-a+a^2)l_i)}\,. \end{aligned} \tag{8.78}$$

Case III The four rules obtained if $(p_1, p_2, p_3) = (0, 1, 0)$

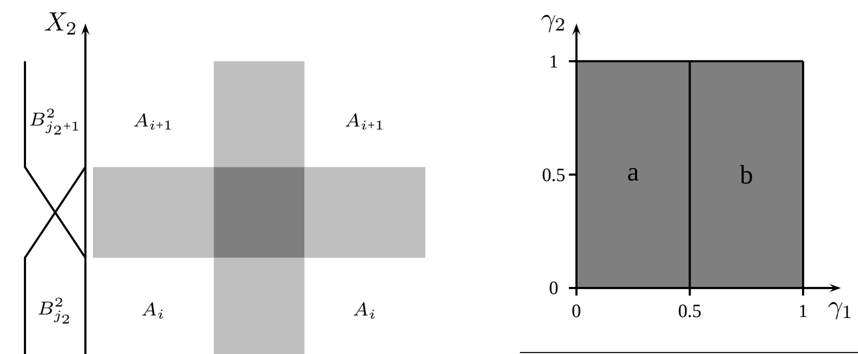
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}

are represented schematically in Fig. 8.13. For all inputs $\mathbf{x}$ not firing any other rule than these four rules (Eqs. (8.47)–(8.48)), the fulfilment degrees of the linguistic output values A_i and A_{i+1} are obtained by

$$\alpha_i = \max((1-\gamma_1)(1-\gamma_2), \gamma_1(1-\gamma_2))\,, \tag{8.79}$$

$$\alpha_{i+1} = \max((1-\gamma_1)\gamma_2, \gamma_1\gamma_2)\,. \tag{8.80}$$

In both regions of the input space indicated in Fig. 8.13 monotonicity is guaranteed. As the linguistic output values A_i and A_{i+1} are the only linguistic output values with a non-zero fulfilment degree, Eqs. (8.69)–(8.70) can be applied. If $(\alpha_i, \alpha_{i+1}) = ((1-\gamma_1)(1-\gamma_2), (1-\gamma_1)\gamma_2)$ (Case IIIa) the derivatives of the crisp output y^*_{COG} to



	α_i	α_{i+1}
a	(1-γ_1)(1-γ_2)	(1-γ_1)γ_2
b	γ_1(1-γ_2)	$\gamma_1\gamma_2$

Figure 8.13: Schematic representation of the rules considered in Case III of the discussion about models with two input variables applying $T_{\mathbf{P}}$ combined with the COG defuzzification method.

γ_1 and γ_2 are positive, since

$$-\alpha_{i+1}\frac{\partial\alpha_i}{\partial\gamma_1}+\alpha_i\frac{\partial\alpha_{i+1}}{\partial\gamma_1}=(1-\gamma_1)\gamma_2(1-\gamma_2)-(1-\gamma_1)(1-\gamma_2)\gamma_2=0\,, \tag{8.81}$$

$$-\alpha_{i+1}\frac{\partial\alpha_i}{\partial\gamma_2}+\alpha_i\frac{\partial\alpha_{i+1}}{\partial\gamma_2}=(1-\gamma_1)^2\gamma_2+(1-\gamma_1)^2(1-\gamma_2)=(1-\gamma_1)^2\geq 0\,. \tag{8.82}$$

For Case IIIb monotonicity is also guaranteed, as in this case the fulfilment degrees α_i and α_{i+1} are described by the same functions of γ_1 and γ_2 as in Case IIc discussed earlier in this section.

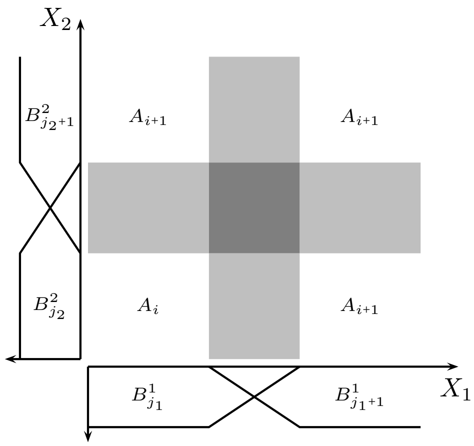
Case IV The four rules obtained if $(p_1,p_2,p_3)=(1,0,0)$

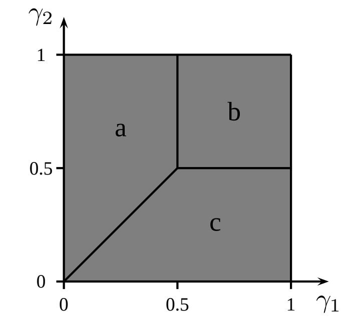
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_{i+1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}

are represented schematically in Fig. 8.14. For all inputs $\mathbf{x}$ not firing any other rule than these four rules (Eqs. (8.47)–(8.48)), the fulfilment degrees of the linguistic output values A_i and A_{i+1} are obtained by

$$\alpha_i=(1-\gamma_1)(1-\gamma_2)\,, \tag{8.83}$$

$$\alpha_{i+1}=\max(\gamma_1(1-\gamma_2),(1-\gamma_1)\gamma_2,\gamma_1\gamma_2)\,. \tag{8.84}$$





	α_i	α_{i+1}
a	$(1\text{-}\gamma_1)(1\text{-}\gamma_2)$	$(1\text{-}\gamma_1)\gamma_2$
b	$(1\text{-}\gamma_1)(1\text{-}\gamma_2)$	$\gamma_1\gamma_2$
c	$(1\text{-}\gamma_1)(1\text{-}\gamma_2)$	$\gamma_1(1\text{-}\gamma_2)$

Figure 8.14: Schematic representation of the rules considered in Case IV of the discussion about models with two input variables applying $T_{\mathbf{P}}$ combined with the COG defuzzification method.

As Case IVa and Case IVb correspond to Case IIIa and Case IIa respectively, only Case IVc still needs to be discussed. As all fulfilment degrees, other than α_i and α_{i+1} are equal to zero, Eqs. (8.69)–(8.70) can be used to prove that if $(\alpha_i, \alpha_{i+1}) = ((1-\gamma_1)(1-\gamma_2), \gamma_1(1-\gamma_2))$ the derivatives of the crisp output y^*_{COG} to γ_1 and γ_2 are positive, since

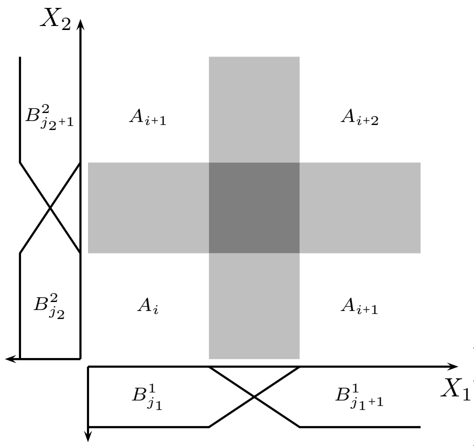
$$-\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_1} + \alpha_i \frac{\partial \alpha_{i+1}}{\partial \gamma_1} = \gamma_1(1-\gamma_2)^2 + (1-\gamma_1)(1-\gamma_2)^2 = (1-\gamma_2)^2 \geq 0\,, \tag{8.85}$$

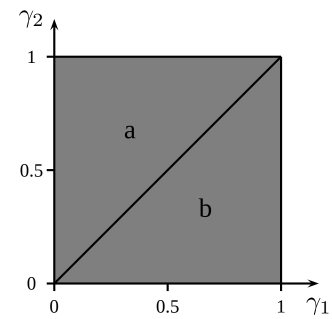
$$-\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_2} + \alpha_i \frac{\partial \alpha_{i+1}}{\partial \gamma_2} = \gamma_1(1-\gamma_2)(1-\gamma_1) - (1-\gamma_1)(1-\gamma_2)\gamma_1 = 0\,. \tag{8.86}$$

Case V The four rules obtained if $(p_1, p_2, p_3) = (1, 0, 1)$

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_{i+1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+2}

are represented schematically in Fig. 8.15. For all inputs $\mathbf{x}$ not firing any other rule than these four rules (Eqs. (8.47)–(8.48)), the fulfilment degrees of the linguistic output





	α_i	α_{i+1}	α_{i+2}
a	$(1\text{-}\gamma_1)(1\text{-}\gamma_2)$	$(1\text{-}\gamma_1)\gamma_2$	$\gamma_1\gamma_2$
b	$(1\text{-}\gamma_1)(1\text{-}\gamma_2)$	$\gamma_1(1\text{-}\gamma_2)$	$\gamma_1\gamma_2$

Figure 8.15: Schematic representation of the rules considered in Case V of the discussion about models with two input variables applying $T_{\mathbf{P}}$ combined with the COG defuzzification method.

values A_i, A_{i+1} and A_{i+2} are obtained by

$$\alpha_i = (1-\gamma_1)(1-\gamma_2)\,, \tag{8.87}$$

$$\alpha_{i+1} = \max(\gamma_1(1-\gamma_2), (1-\gamma_1)\gamma_2)\,, \tag{8.88}$$

$$\alpha_{i+2} = \gamma_1\gamma_2\,. \tag{8.89}$$

For Case Va, the crisp output y^*_{COG}, as a function of γ_1 and γ_2, is given by

$$\begin{aligned} y^*_{\mathrm{COG}} = c_{i+1} - \Big[&(1-\gamma_1)(1-\gamma_2)(2l_{i\text{-}1}^2 + 6l_{i\text{-}1}l_i + 6l_{i\text{-}1}k_i + 3l_{i\text{-}1}k_{i+1} + 12l_ik_i \\ &+ 6k_i^2 + 6k_ik_{i+1}) + 2(1-\gamma_1)(\gamma_2^3 - 2\gamma_2 + 2)l_i^2 + 3(\gamma_2^2 - \gamma_2 + 1) \\ &(1-\gamma_1)l_ik_{i+1} - 2(-\gamma_1^3 + 3\gamma_1^2 - \gamma_1 + 1)\gamma_2 l_{i+1}^2 \\ &- 3\gamma_2(\gamma_1^2 - \gamma_1 + 1)l_{i+1}k_{i+1} - (6l_{i+1}l_{i+2} + 12l_{i+1}k_{i+2} + 2l_{i+2}^2 \\ &+ 3l_{i+2}k_{i+1} + 6l_{i+2}k_{i+2} + 6k_{i+1}k_{i+2} + 6k_{i+2}^2)\gamma_1\gamma_2\Big] \times \\ \Big[&6((1-\gamma_1)(1-\gamma_2)(l_{i\text{-}1} + 2k_i) + \gamma_1\gamma_2(l_{i+2} + 2k_{i+2}) + (1-\gamma_1) \\ &(\gamma_2^2 - \gamma_2 + 1)l_i + \gamma_2(\gamma_1^2 - \gamma_1 + 1)l_{i+1} + 2(1-\gamma_1)\gamma_2k_{i+1})\Big]^{-1}\,. \end{aligned} \tag{8.90}$$

As the equation obtained if y^*_{COG} is written as a function of α_i, α_{i+1} and α_{i+2} is more complex, the chain rule will not be applied. The derivatives of y^*_{COG} to γ_1 and γ_2

are both positive for all $l_{i\text{-}1}, l_i, l_{i+1}, l_{i+2} \in \mathbb{R}_0^+$, $k_i k_{i+1}, k_{i+2} \in \mathbb{R}^+$ and $(\gamma_1, \gamma_2) \in \,]0,1[^2$

$$\frac{\partial y^*_{\text{COG}}}{\partial \gamma_1} = \frac{C_1}{C_2}\,, \tag{8.91}$$

$$\frac{\partial y^*_{\text{COG}}}{\partial \gamma_2} = \frac{C_3}{C_2}\,, \tag{8.92}$$

with

$$\begin{aligned}
C_1 = {} & ((6l_{i+1}l_{i+2} + 12l_{i+1}k_{i+2} + 2l_{i+2}^2 + 3l_{i+2}k_{i+1} + 6l_{i+2}k_{i+2} + 6k_{i+1}k_{i+2} + 6k_{i+2}^2) \\
& k_{i+1}\gamma_2 + (1-\gamma_2)(l_{i\text{-}1}^2 l_{i+2} + 2l_{i\text{-}1}^2 k_{i+2} + 3l_{i\text{-}1}l_i l_{i+2} + 6l_{i\text{-}1}l_i k_{i+2} + 3l_{i\text{-}1}l_{i+1}l_{i+2} \\
& + 6l_{i\text{-}1}l_{i+1}k_{i+2} + l_{i\text{-}1}l_{i+2}^2 + 3l_{i\text{-}1}l_{i+2}k_i + 3l_{i\text{-}1}l_{i+2}k_{i+1} + 3l_{i\text{-}1}l_{i+2}k_{i+2} \\
& + 6l_{i\text{-}1}k_i k_{i+2} + 6l_{i\text{-}1}k_{i+1}k_{i+2} + 3l_{i\text{-}1}k_{i+2}^2 + 6l_i l_{i+2}k_i + 12l_i k_i k_{i+2} + 6l_{i+1}l_{i+2}k_i \\
& + 12l_{i+1}k_i k_{i+2} + 2l_{i+2}^2 k_i + 3l_{i+2}k_i^2 + 6l_{i+2}k_i k_{i+1} + 6l_{i+2}k_i k_{i+2} + 6k_i^2 k_{i+2} \\
& + 12k_i k_{i+1}k_{i+2} + 6k_i k_{i+2}^2) + (2-\gamma_1)(1-\gamma_2)\gamma_1 l_{i+1}(l_{i\text{-}1}^2 + 3l_{i\text{-}1}l_i + 3l_{i\text{-}1}k_i \\
& + 3l_{i\text{-}1}k_{i+1} + 6l_i k_i + 3k_i^2 + 6k_i k_{i+1}) + 2\gamma_1(\gamma_1^2 - 3\gamma_1 + 3)(1-\gamma_2)l_{i+1}^2 \\
& (l_{i\text{-}1} + 2k_i) + \gamma_1(\gamma_2^3 - 2\gamma_2 + 2)(2-\gamma_1)(l_i^2 l_{i+1}) + (\gamma_2^2 - \gamma_2 + 1)l_i(3l_{i+1}l_{i+2} \\
& + 6l_{i+1}k_{i+2} + l_{i+2}^2 + 3l_{i+2}k_{i+1} + 3l_{i+2}k_{i+2} + 6k_{i+1}k_{i+2} + 3k_{i+2}^2) \\
& + 2\gamma_1(\gamma_1^2 - 3\gamma_1 + 3)((\gamma_2^2 - \gamma_2 + 1)l_i + 2\gamma_2 k_{i+1})l_{i+1}^2 + 3\gamma_1(2-\gamma_1) \\
& ((\gamma_2^2 - \gamma_2 + 1)l_i + \gamma_2 k_{i+1})l_{i+1}k_{i+1} + (\gamma_2^3 - 2\gamma_2 + 2)l_i^2(l_{i+2} + 2k_{i+2}) \\
& + \gamma_1\gamma_2(1-\gamma_1)(\gamma_1^2 - \gamma_1 + 4)l_{i+1}^3 + (1-\gamma_1)(1+\gamma_1)\gamma_2 l_{i+1}(l_{i+2}^2 + 3l_{i+2}k_{i+2} \\
& + 3k_{i+2}^2) + 2(1-\gamma_1)(1+\gamma_1+\gamma_1^2)\gamma_2(l_{i+2} + 2k_{i+2})l_{i+1}^2)\gamma_2\,, \\
C_2 = {} & 3((1-\gamma_1)(1-\gamma_2)(l_{i\text{-}1} + 2k_i) + \gamma_1\gamma_2(l_{i+2} + 2k_{i+2}) \\
& + (1-\gamma_1)(\gamma_2^2 - \gamma_2 + 1)l_i + (\gamma_1^2 - \gamma_1 + 1)\gamma_2 l_{i+1} + 2(1-\gamma_1)\gamma_2 k_{i+1})^2\,, \\
C_3 = {} & ((6l_i l_{i\text{-}1} + 12l_i k_i + 2l_{i\text{-}1}^2 + 3l_{i\text{-}1}k_{i+1} + 6l_{i\text{-}1}k_i + 6k_{i+1}k_i + 6k_i^2)k_{i+1}(1-\gamma_1) \\
& + \gamma_1(l_{i+2}^2 l_{i\text{-}1} + 2l_{i+2}^2 k_i + 3l_{i+2}l_{i+1}l_{i\text{-}1} + 6l_{i+2}l_{i+1}k_i + 3l_{i+2}l_i l_{i\text{-}1} + 6l_{i+2}l_i k_i \\
& + l_{i+2}l_{i\text{-}1}^2 + 3l_{i+2}l_{i\text{-}1}k_{i+2} + 3l_{i+2}l_{i\text{-}1}k_{i+1} + 3l_{i+2}l_{i\text{-}1}k_i + 6l_{i+2}k_{i+2}k_i \\
& + 6l_{i+2}k_{i+1}k_i + 3l_{i+2}k_i^2 + 6l_{i+1}l_{i\text{-}1}k_{i+2} + 12l_{i+1}k_{i+2}k_i + 6l_i l_{i\text{-}1}k_{i+2} \\
& + 12l_i k_{i+2}k_i + 2l_{i\text{-}1}^2 k_{i+2} + 3l_{i\text{-}1}k_{i+2}^2 + 6l_{i\text{-}1}k_{i+2}k_{i+1} + 6l_{i\text{-}1}k_{i+2}k_i + 6k_{i+2}^2 k_i \\
& + 12k_{i+2}k_{i+1}k_i + 6k_{i+2}k_i^2) + \gamma_1(1+\gamma_2)(1-\gamma_2)l_i(l_{i+2}^2 + 3l_{i+2}l_{i+1} + 3l_{i+2}k_{i+2} \\
& + 3l_{i+2}k_{i+1} + 6l_{i+1}k_{i+2} + 3k_{i+2}^2 + 6k_{i+2}k_{i+1}) + 2\gamma_1(\gamma_2^2 + \gamma_2 + 1)(1-\gamma_2) \\
& (l_{i+2} + 2k_{i+2})l_i^2 + (-\gamma_1^3 + 3\gamma_1^2 - \gamma_1 + 1)(1-\gamma_2)(1+\gamma_2)l_{i+1}^2 l_i \\
& + (\gamma_1^2 - \gamma_1 + 1)l_{i+1}(3l_i l_{i\text{-}1} + 6l_i k_i + l_{i\text{-}1}^2 + 3l_{i\text{-}1}k_{i+1} + 3l_{i\text{-}1}k_i + 6k_{i+1}k_i \\
& + 3k_i^2) + 2(1-\gamma_2)(\gamma_2^2 + \gamma_2 + 1)((\gamma_1^2 - \gamma_1 + 1)l_{i+1} + 2(1-\gamma_1)k_{i+1})l_i^2
\end{aligned}$$

$$+ 3(1-\gamma_2)(1+\gamma_2)((\gamma_1^2-\gamma_1+1)l_{i+1}+(1-\gamma_1)k_{i+1})l_ik_{i+1}+(-\gamma_1^3+3\gamma_1^2
- \gamma_1+1)l_{i+1}^2(l_{i-1}+2k_i)+(1-\gamma_2)\gamma_2(1-\gamma_1)(\gamma_2^2-\gamma_2+4)l_i^3+(1-\gamma_1)
(2-\gamma_2)\gamma_2 l_i(l_{i-1}^2+3l_{i-1}k_i+3k_i^2)+2(1-\gamma_1)(\gamma_2^2-3\gamma_2+3)\gamma_2(l_{i-1}+2k_i)l_i^2)
(1-\gamma_1)\,.$$

As by interchanging γ_1 and γ_2 in the equations of the fulfilment degrees α_i, α_{i+1} and α_{i+2} for Case Va, the equations for Case Vb are obtained, one only needs to interchange γ_1 and γ_2 in Eqs. (8.90)–(8.92) to prove that monotonicity is also guaranteed for Case Vb.

8.4.3 Models with a monotone rule base applying $T_{\mathbf{P}}$

As shown by the counterexample below, monotonicity cannot be guaranteed for any monotone rule base, if combining the t-norm $T_{\mathbf{P}}$ with the COG defuzzification method in models with two input variables.

The set of four rules represented in Fig. 8.16

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+2}

can occur in a monotone, but non-smooth rule base. For all inputs $\mathbf{x}$ not firing any other rule than these four rules (Eqs. (8.47)–(8.48)), the fulfilment degrees of the linguistic output values A_i and A_{i+1} are obtained by

$$\alpha_i = \max((1-\gamma_1)(1-\gamma_2),\gamma_1(1-\gamma_2))\,, \tag{8.93}$$

$$\alpha_{i+1} = (1-\gamma_1)\gamma_2\,, \tag{8.94}$$

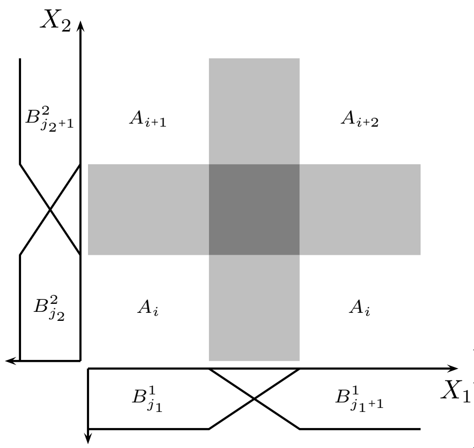
$$\alpha_{i+2} = \gamma_1\gamma_2\,. \tag{8.95}$$

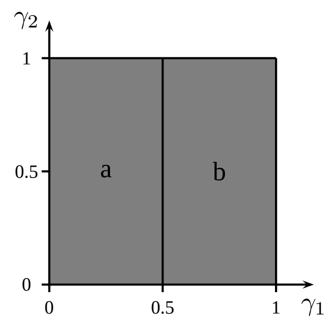
If γ_1 is smaller than 0.5 (Case a) the fulfilment degrees α_i, α_{i+1} and α_{i+2} are described by the same functions of γ_1 and γ_2 as in Case Va in Section 8.4.2. As the proof given in Section 8.4.2 holds for all $(\gamma_1,\gamma_2) \in\ ]0,1[^2$, it also proves that monotonicity is guaranteed in Case a.

Numerical experiments revealed that when γ_1 is larger than 0.5 (Case b), the crisp output y^*_{COG} decreases with increasing γ_1 when γ_2 is larger than 0 but smaller than a critical value $\gamma_{2,\mathrm{zero}}$ and increases when γ_2 is larger than that critical value and smaller than 1. The critical γ_2-value is a complex function of all parameters defining the three membership functions A_i, A_{i+1} and A_{i+2}, as illustrated in Fig. 8.17 for a fuzzy output partition with intervals of changing membership degree of equal length l, and is equal to 0.5 if the three membership functions are identical trapezia as in Fig. 8.18.

For Case b, the crisp output y^*_{COG} is given by

$$y^*_{\mathrm{COG}} = \frac{y^*_iS_i + y^*_{i+1}S_{i+1} + y^*_{i+2}S_{i+2} - y^*_{\mathrm{op},i}S_{\mathrm{op},i} - y^*_{\mathrm{op},i+1}S_{\mathrm{op},i+1}}{S_i + S_{i+1} + S_{i+2} - S_{\mathrm{op},i} - S_{\mathrm{op},i+1}}$$





	α_i	α_{i+1}	α_{i+2}
a	$(1\text{-}\gamma_1)(1\text{-}\gamma_2)$	$(1\text{-}\gamma_1)\gamma_2$	$\gamma_1\gamma_2$
b	$\gamma_1(1\text{-}\gamma_2)$	$(1\text{-}\gamma_1)\gamma_2$	$\gamma_1\gamma_2$

Figure 8.16: Schematic representation of the rules for which a non-monotone input-output behaviour is obtained when applying $T_{\mathbf{P}}$ combined with the COG defuzzification method.

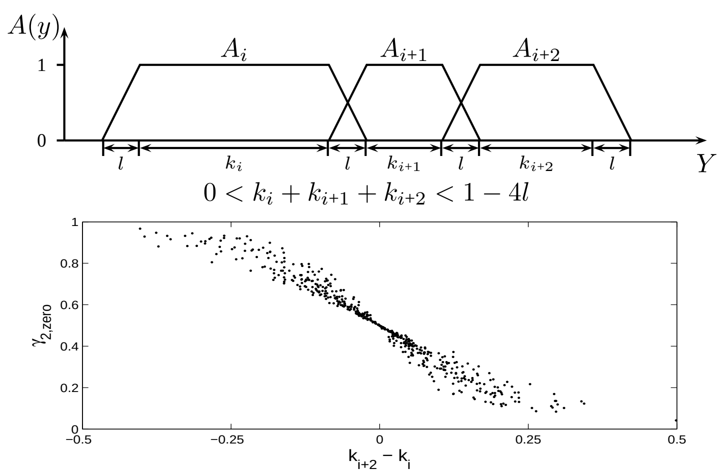


Figure 8.17: The upper bound $\gamma_{2,\text{zero}}$ of the interval of γ_2-values for which a non-monotone input-output behaviour is obtained in Case b as a function of $k_{i+2} - k_i$.

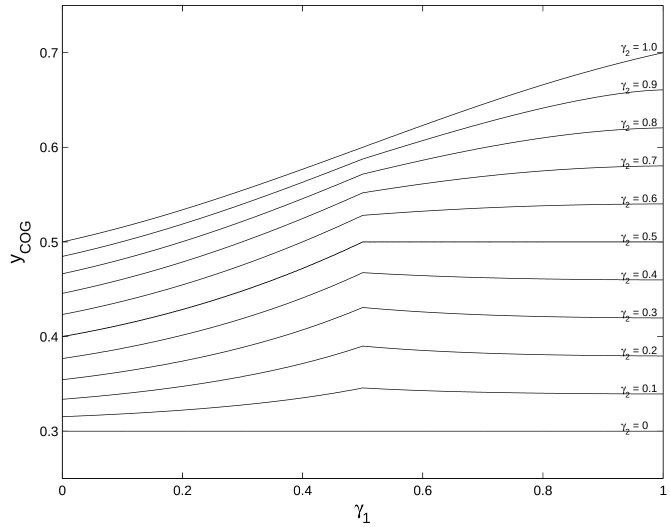


Figure 8.18: Crisp output y^*_{COG} as a function of γ_1 for the rules used in the counterexample when discussing the monotonicity of models with two input variables and a non-smooth rule base applying $T_{\mathbf{P}}$ combined with the COG defuzzification method.

$$
\begin{aligned}
= c_{i+1} + \Big[&\gamma_1(1-\gamma_2)(2\gamma_1\gamma_2-\gamma_1-\gamma_2)^2(2l_{i\text{-}1}^2 + 6l_{i\text{-}1}l_i + 6l_{i\text{-}1}k_i + 3l_{i\text{-}1}k_{i+1} \\
&+ 12l_ik_i + 6k_i^2 + 6k_ik_{i+1}) - \gamma_1\gamma_2(2\gamma_1\gamma_2-\gamma_1-\gamma_2)^2(6l_{i+1}l_{i+2} \\
&+ 12l_{i+1}k_{i+2} + 2l_{i+2}^2 + 3l_{i+2}k_{i+1} + 6l_{i+2}k_{i+2} + 6k_{i+1}k_{i+2} + 6k_{i+2}^2) \\
&+ (-18\gamma_1^3\gamma_2^3 + 32\gamma_1^3\gamma_2^2 - 20\gamma_1^3\gamma_2 + 4\gamma_1^3 + 22\gamma_1^2\gamma_2^3 - 24\gamma_1^2\gamma_2^2 \\
&+ 8\gamma_1^2\gamma_2 - 10\gamma_1\gamma_2^3 + 4\gamma_1\gamma_2^2 + 2\gamma_2^3)l_i^2 - 3(2\gamma_1\gamma_2-\gamma_1-\gamma_2) \\
&(3\gamma_1^2\gamma_2^2 - 3\gamma_1^2\gamma_2 + \gamma_1^2 - 3\gamma_1\gamma_2^2 + \gamma_1\gamma_2 + \gamma_2^2)l_ik_{i+1} \\
&+ (2\gamma_1\gamma_2-\gamma_1-\gamma_2)^2\gamma_2 l_{i+1}(2(\gamma_1^3 - 3\gamma_1^2 + \gamma_1 - 1)l_{i+1} \\
&- 3(\gamma_1^2 - \gamma_1 + 1)k_{i+1})\Big] \times \\
\Big[&6(2\gamma_1\gamma_2-\gamma_1-\gamma_2)(-(1-\gamma_2)(2\gamma_1\gamma_2-\gamma_1-\gamma_2)\gamma_1(l_{i\text{-}1} + 2k_i) \\
&+ (3\gamma_1^2\gamma_2^2 - 3\gamma_1^2\gamma_2 + \gamma_1^2 - 3\gamma_1\gamma_2^2 + \gamma_1\gamma_2 + \gamma_2^2)l_i \\
&- (2\gamma_1\gamma_2-\gamma_1-\gamma_2)\gamma_2((\gamma_1^2 - \gamma_1 + 1)l_{i+1} + 2(1-\gamma_1)k_{i+1}) \\
&- (2\gamma_1\gamma_2-\gamma_1-\gamma_2)\gamma_1\gamma_2(l_{i+2} + 2k_{i+2}))\Big]^{-1}, \qquad (8.96)
\end{aligned}
$$

and its derivative to γ_1 is

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} = \frac{\gamma_2(C_1\gamma_2^5 + C_2\gamma_2^4 + C_3\gamma_2^3 + C_4\gamma_2^2 + C_5\gamma_2 + C_6)}{(C_7)^2}\,. \tag{8.97}$$

The coefficients $C_1, C_2, C_3, C_4, C_5, C_6$ of the polynomial function of γ_2 and the term C_7 in the denominator are functions of $\gamma_1 \in [0.5, 1]$, $l_{i\text{-}1}, l_i, l_{i+1}, l_{i+2} \in \mathbb{R}_0^+$ and $k_i k_{i+1}$, $k_{i+2} \in \mathbb{R}^+$ and are given in Eqs. (C.1–C.7) in Appendix C. Eq. (8.97) shows that the derivative of y^*_{COG} to γ_1 is equal to zero for $\gamma_2 = 0$. Instead of proving that the derivative of y^*_{COG} to γ_1 is negative for some values $\gamma_2 \in [0, 1]$ regardless of the fuzzy output partition applied by determining the roots of the polynomial function of degree five in γ_2, it is shown that the derivative of the crisp output y^*_{COG} to γ_1 is strictly negative for any A_i, A_{i+1} and A_{i+2} when γ_2 approaches 0

$$\begin{aligned}
\lim_{\gamma_2 \overset{>}{\to} 0} \frac{dy^*_{\mathrm{COG}}}{d\gamma_1} = & -\gamma_1^4 \lim_{\gamma_2 \overset{>}{\to} 0} \gamma_2 \Big[3(l_{i\text{-}1} + l_i + 2k_i)^2\gamma_1^6\Big]^{-1} \Big[\big(l_{i\text{-}1} + l_i + 2k_i\big) \\
& \big((1-\gamma_1)^2(2\gamma_1+1)l_{i+1}^2 + 3(1-\gamma_1^2)l_{i+1}k_{i+1} + 3k_{i+1}^2\big) \\
& + \big(3k_i(l_{i\text{-}1} + 2l_i + k_i) + (l_{i\text{-}1} + l_i)(l_{i\text{-}1} + 2l_i)\big) \\
& \big((1-\gamma_1^2)l_{i+1} + 2k_{i+1}\big)\Big] \\
< {} & 0
\end{aligned} \tag{8.98}$$

8.5 Models with three input variables

Only models with a monotone smooth rule base applying the t-norm $T_{\mathbf{P}}$ are considered in this section, since models with two input variables applying $T_{\mathbf{M}}$ show a non-monotone input-output behaviour for some monotone smooth rule bases (Section 8.4.1), models with two input variables applying $T_{\mathbf{P}}$ show a non-monotone input-output behaviour for some monotone, but non-smooth rule bases (Section 8.4.3) and models applying $T_{\mathbf{L}}$ return the empty set as fuzzy model output for some inputs if the number of input variables is larger than two (Section 7.4).

8.5.1 Numerical experiments

To get more insight in the behaviour of models with three input variables applying $T_{\mathbf{P}}$ and the COG defuzzification method, numerical experiments were carried out. In all models used during the experiments two linguistic values B_1^l and B_2^l, defined by the membership functions shown in Fig. 8.19, were assigned to the three input variables X_1, X_2 and X_3. By combining each of the 18 monotone smooth rule bases obtained by applying the 18 combinations of the parameters p_i listed in Table 7.1 to the following set of eight rules

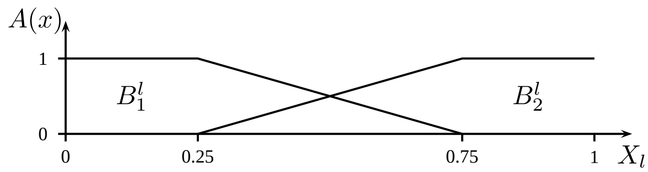


Figure 8.19: Membership functions assigned to the three input variables X_1, X_2 and X_3 during the numerical experiments.

Table 8.4: Output membership functions used in the numerical experiments, characterized by lengths l of the intervals of changing membership degree and lengths k of the kernels, as well as the parameter d defining the size of the region of the input space where a non-monotone input-output behaviour is obtained for models with rule base 'Case XI' and rule base 'Case XVI'.

	$l_{i\text{-}1}$	k_i	l_i	k_{i+1}	l_{i+1}	k_{i+2}	l_{i+2}	k_{i+3}	l_{i+3}	d - Case XI $\pm$ 0.005	d - Case XVI $\pm$ 0.005
1	0.150	0.017	0.150	0.017	0.150	0.017	0.150	0.017	0.150	0.005	0.005
2	0.118	0.177	0.053	0.005	0.328	0.075	0.081	0.023	0.019	0.005	0.005
3	0.183	0.216	0.047	0.059	0.147	0.009	0.157	0.008	0.086	0.255	0.005
4	0.228	0.118	0.045	0.052	0.104	0.124	0.032	0.006	0.040	0.115	0.005
5	0.134	0.085	0.014	0.125	0.046	0.046	0.067	0.079	0.327	0.155	0.005
6	0.050	0.126	0.050	0.239	0.050	0.257	0.050	0.073	0.050	0.005	0.175
7	0.050	0.042	0.050	0.114	0.050	0.170	0.050	0.117	0.050	0.005	0.275
8	0.050	0.034	0.050	0.203	0.050	0.107	0.050	0.221	0.050	0.005	0.185
9	0.050	0.132	0.050	0.063	0.050	0.059	0.050	0.073	0.050	0.185	0.005
10	0.050	0.104	0.050	0.173	0.050	0.098	0.050	0.097	0.050	0.015	0.005

$$
\begin{array}{llllllll}
\text{IF} & X_1 \text{ IS } B_1^1 & \text{AND} & X_2 \text{ IS } B_1^2 & \text{AND} & X_3 \text{ IS } B_1^3 & \text{THEN} & Y \text{ IS } A_i \\
\text{IF} & X_1 \text{ IS } B_1^1 & \text{AND} & X_2 \text{ IS } B_1^2 & \text{AND} & X_3 \text{ IS } B_2^3 & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_3} \\
\text{IF} & X_1 \text{ IS } B_1^1 & \text{AND} & X_2 \text{ IS } B_2^2 & \text{AND} & X_3 \text{ IS } B_1^3 & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2} \\
\text{IF} & X_1 \text{ IS } B_1^1 & \text{AND} & X_2 \text{ IS } B_2^2 & \text{AND} & X_3 \text{ IS } B_2^3 & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_3+p_5} \\
\text{IF} & X_1 \text{ IS } B_2^1 & \text{AND} & X_2 \text{ IS } B_1^2 & \text{AND} & X_3 \text{ IS } B_1^3 & \text{THEN} & Y \text{ IS } A_{i+p_1} \\
\text{IF} & X_1 \text{ IS } B_2^1 & \text{AND} & X_2 \text{ IS } B_1^2 & \text{AND} & X_3 \text{ IS } B_2^3 & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_3+p_6} \\
\text{IF} & X_1 \text{ IS } B_2^1 & \text{AND} & X_2 \text{ IS } B_2^2 & \text{AND} & X_3 \text{ IS } B_1^3 & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_4} \\
\text{IF} & X_1 \text{ IS } B_2^1 & \text{AND} & X_2 \text{ IS } B_2^2 & \text{AND} & X_3 \text{ IS } B_2^3 & \text{THEN} & Y \text{ IS } A_{i+p_7'}
\end{array}
$$

with $p_1, p_2, p_3, p_4, p_5, p_6, p_7 \in \mathbb{N}$ and $p_7' = p_1 + p_2 + \max(p_4, p_3 + p_5, p_3 + p_6) + p_7$, with each of the ten fuzzy output partitions characterized in Table 8.4, 180 models were obtained. Note that in partition 1 the membership functions A_i, A_{i+1}, A_{i+2} and A_{i+3} have an identical shape and that in partitions 6-10 the intervals of changing membership degree are of equal length.

The model output y^*_{COG} of all models was determined for 103^3 inputs $(X_1, X_2, X_3) \in [0.245 : 0.005 : 0.755]^3$. A monotone input-output behaviour was recorded for almost all models. Only the models applying fuzzy output partitions 3, 4, 5, 9 and 10 combined with rule base 'Case XI'

IF	X_1 IS B_1^1	AND	X_2 IS B_1^2	AND	X_3 IS B_1^3	THEN	Y IS A_i
IF	X_1 IS B_1^1	AND	X_2 IS B_1^2	AND	X_3 IS B_2^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_1^1	AND	X_2 IS B_2^2	AND	X_3 IS B_1^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_1^1	AND	X_2 IS B_2^2	AND	X_3 IS B_2^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_2^1	AND	X_2 IS B_1^2	AND	X_3 IS B_1^3	THEN	Y IS A_i
IF	X_1 IS B_2^1	AND	X_2 IS B_1^2	AND	X_3 IS B_2^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_2^1	AND	X_2 IS B_2^2	AND	X_3 IS B_1^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_2^1	AND	X_2 IS B_2^2	AND	X_3 IS B_2^3	THEN	Y IS A_{i+2}

and the models applying fuzzy output partitions 6, 7, and 8 combined with rule base ‘Case XVI’

IF	X_1 IS B_1^1	AND	X_2 IS B_1^2	AND	X_3 IS B_1^3	THEN	Y IS A_i
IF	X_1 IS B_1^1	AND	X_2 IS B_1^2	AND	X_3 IS B_2^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_1^1	AND	X_2 IS B_2^2	AND	X_3 IS B_1^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_1^1	AND	X_2 IS B_2^2	AND	X_3 IS B_2^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_2^1	AND	X_2 IS B_1^2	AND	X_3 IS B_1^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_2^1	AND	X_2 IS B_1^2	AND	X_3 IS B_2^3	THEN	Y IS A_{i+1}
IF	X_1 IS B_2^1	AND	X_2 IS B_2^2	AND	X_3 IS B_1^3	THEN	Y IS A_{i+2}
IF	X_1 IS B_2^1	AND	X_2 IS B_2^2	AND	X_3 IS B_2^3	THEN	Y IS A_{i+2}

show a non-monotone input-output behaviour in a region of the 3-dimensional input space.

For models with rule base ‘Case XI’ the derivative of $y^*_{\rm COG}$ to γ_1 is negative for inputs $\mathbf{x}$ within the region of the input space defined by the projections onto the (γ_1,γ_2)-, (γ_1,γ_3)- and (γ_2,γ_3)-planes coloured gray in Fig. 8.20. Additional numerical experiments with models using rule base ‘Case XI’ and 10000 randomly generated fuzzy output partitions, showed that the size of the region of the input space, characterized by the parameter d, is a complex function of the parameters $l_{i\text{-}1}$, k_i, l_i, k_{i+1}, l_{i+1} , k_{i+2} and l_{i+2} defining the output membership functions A_i, A_{i+1} and A_{i+2}. The values of d obtained for the ten partitions used in the numerical experiments are given in Table 8.4. If $d = 0.005 \pm 0.005$ is mentioned, d is an element of the interval $[0, 0.01]$, *i.e.* either no non-monotone behaviour occurs for the given fuzzy partition or non-monotone behaviour is obtained in a very small region of the input space characterized by a d-value smaller than the discretization step 0.01 used when determining d.

For models with rule base ‘Case XI’ the crisp model output $y^*_{\rm COG}$

$$y^*_{\rm COG} = \frac{y^*_i S_i + y^*_{i+1} S_{i+1} + y^*_{i+2} S_{i+2} - y^*_{{\rm op},i} S_{{\rm op},i} - y^*_{{\rm op},i+1} S_{{\rm op},i+1}}{S_i + S_{i+1} + S_{i+2} - S_{{\rm op},i} - S_{{\rm op},i+1}}\,, \tag{8.99}$$

is obtained using the formulae in Table 8.1 with the fulfilment degrees α_i, α_{i+1} and α_{i+2} being given by

$$\alpha_i = \max((1-\gamma_1)(1-\gamma_2)(1-\gamma_3), \gamma_1(1-\gamma_2)(1-\gamma_3))\,, \tag{8.100}$$

$$\begin{aligned}\alpha_{i+1} = \max(&(1-\gamma_1)\gamma_2(1-\gamma_3), \gamma_1\gamma_2(1-\gamma_3), (1-\gamma_1)(1-\gamma_2)\gamma_3, \gamma_1(1-\gamma_2)\gamma_3,\\ &(1-\gamma_1)\gamma_2\gamma_3)\,,\end{aligned} \tag{8.101}$$

$$\alpha_{i+2} = \gamma_1\gamma_2\gamma_3\,. \tag{8.102}$$

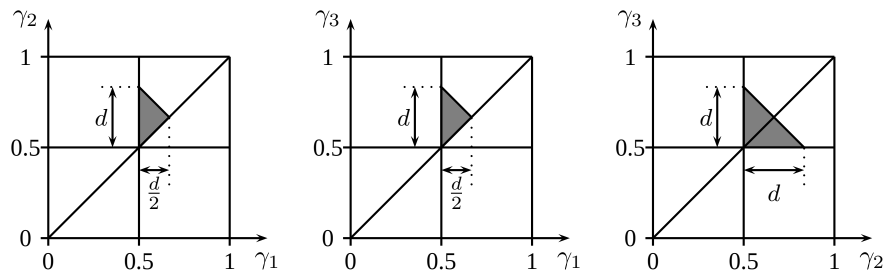


Figure 8.20: Region of the input space where a non-monotone input-output behaviour is recorded for some fuzzy output partitions in Case XI.

In the region of the input space where non-monotonicity is recorded for some fuzzy output partitions, the membership degrees γ_1, γ_2 and γ_3 satisfy

$$\gamma_2 > \gamma_1 > 0.5 \qquad \wedge \qquad \gamma_3 > \gamma_1 > 0.5\,, \tag{8.103}$$

and the fulfilment degrees α_i, α_{i+1} and α_{i+2} are given by

$$\alpha_i = \gamma_1(1-\gamma_2)(1-\gamma_3)\,, \tag{8.104}$$
$$\alpha_{i+1} = (1-\gamma_1)\gamma_2\gamma_3\,, \tag{8.105}$$
$$\alpha_{i+2} = \gamma_1\gamma_2\gamma_3\,, \tag{8.106}$$

The derivative of y^*_{COG} to γ_1 is an even more complex function of γ_1, γ_2, γ_3, $l_{i\text{-}1}$, l_i, l_{i+1}, l_{i+2}, k_i, k_{i+1} and k_{i+2}, than Eq. (8.97). However, it is known from the experiments that if the derivative of y^*_{COG} to γ_1 is negative for some inputs $\mathbf{x}$, it is negative for γ_1, γ_2 and γ_3 approaching 0.5 (from the right). When γ_1, γ_2 and γ_3

Table 8.5: Derivative of y^*_{COG} to γ_1 in models with rule base 'Case XI' for γ_1, γ_2 and γ_3 approaching 0.5 (Eq. (8.107)) for the fuzzy output partitions in Table 8.4.

	1	2	3	4	5	6	7	8	9	10
$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1}$	0.000	0.006	-0.081	-0.029	-0.045	0.100	0.091	0.079	-0.041	-0.006

approach 0.5 the derivative of y^*_{COG} to γ_1 is given by

$$\lim_{\substack{\text{Case XI}\\ \gamma_1\to 0.5\\ \gamma_2\to 0.5\\ \gamma_3\to 0.5}} \frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} = \Big[4(l_{i+2} - l_{i\text{-}1})((l_{i\text{-}1} + l_{i+2})(3l_i + 3l_{i+1} + 8k_{i+1}) + 9l_i l_{i+1} \\ + 12k_{i+1}^2) + 12(k_{i+2} - k_i)((k_i + k_{i+2})(3l_i + 3l_{i+1} + 8k_{i+1}) \\ + 6l_i l_{i+1} + 8k_{i+1}^2) + (l_{i+1} - l_i)((l_i + l_{i+1})(15l_i + 15l_{i+1} \\ + 56k_{i+1}) + 36k_{i+1}^2) + 8(l_i^2 l_{i+2} - l_{i\text{-}1} l_{i+1}^2) + 36k_{i+1}(l_i l_{i+2} \\ - l_{i\text{-}1} l_{i+1}) + 16(l_i^2 k_{i+2} - l_{i+1}^2 k_i) + 72k_{i+1}(l_i k_{i+2} - l_{i+1} k_i) \\ + 28(l_{i+1}^2 l_{i+2} - l_{i\text{-}1} l_i^2) + 96k_{i+1}(l_{i+1} l_{i+2} - l_{i\text{-}1} l_i) + 56(l_{i+1}^2 k_{i+2} \\ - l_i^2 k_i) + 192k_{i+1}(l_{i+1} k_{i+2} - l_i k_i) + 12(l_{i+2} k_{i+2} - l_{i\text{-}1} k_i) \\ (3l_i + 3l_{i+1} + 8k_{i+1})\Big] \times \Big[3(2l_{i\text{-}1} + 3l_i + 3l_{i+1} + 2l_{i+2} + 4k_i \\ + 4k_{i+1} + 4k_{i+2})^2\Big]^{-1} . \tag{8.107}$$

The values obtained for $\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1}$ in Eq. (8.107) for the ten fuzzy partitions used in the experiments are given in Table 8.5. One sees that, on the one hand, if a negative derivative is recorded for some inputs $(X_1, X_2, X_3) \in [0.245 : 0.005 : 0.755]^3$, a negative value is obtained (partitions 3, 4, 5, 9 and 10) and on the other hand, if a positive value is obtained, no negative derivatives are recorded for any input $(X_1, X_2, X_3) \in [0.245 : 0.005 : 0.755]^3$. Eq. (8.107) also shows that the membership functions A_i, A_{i+1} and A_{i+2} for which a positive value is obtained for $\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1}$ in Eq. (8.107), *i.e.* for which a monotone input-output behaviour is obtained in case the rule base contains a set of rules corresponding to Case XI, cannot be defined in a straightforward way. However, one can easily verify that if the membership functions A_i, A_{i+1} and A_{i+2} have an identical shape,

$$(\exists l > 0)(\forall j \in \{i-1, \ldots, i+2\})(l_j = l)\,, \tag{8.108}$$

$$(\exists k \geq 0)(\forall j \in \{i, \ldots, i+2\})(k_j = k)\,, \tag{8.109}$$

the derivative $\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1}$ in Eq. (8.107) is equal to zero. This analytical observation is supported by the results obtained for partition 1.

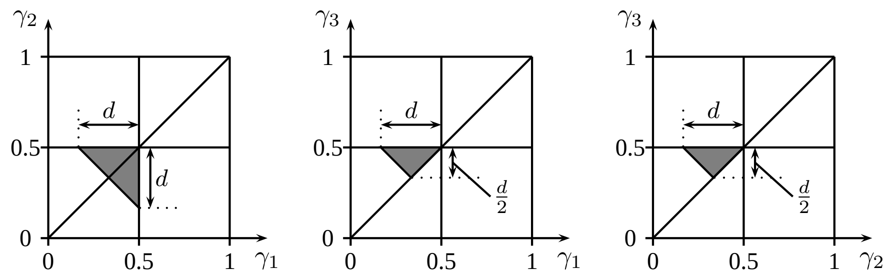


Figure 8.21: Region of the input space where a non-monotone input-output behaviour is recorded for some fuzzy output partitions in Case XVI.

For models with rule base 'Case XVI' the derivative of y^*_{COG} to γ_3 is negative for inputs $\mathbf{x}$ within the region of the input space defined by the projections onto the (γ_1,γ_2)-, (γ_1,γ_3)- and (γ_2,γ_3)-planes coloured gray in Fig. 8.21. The values of d obtained for the ten fuzzy partitions used in the numerical experiments are given in Table 8.4. If $d = 0.005 \pm 0.005$ is mentioned, d is either smaller than the discretization step 0.01 used when determining d or no non-monotone behaviour occurs for the given fuzzy partition. For a certain fuzzy partition, non-monotonicity is never observed for rule base 'Case XI' and for rule base 'Case XVI'. Note furthermore the analogy between the regions shown in Figs. 8.20–8.21. If in Fig. 8.20 γ_1 is substituted by $1-\gamma_3$, γ_2 by $1-\gamma_2$ and γ_3 by $1-\gamma_1$, Fig. 8.21 is obtained.

For models with rule base 'Case XVI' the crisp model output y^*_{COG}

$$y^*_{\mathrm{COG}} = \frac{y^*_i S_i + y^*_{i+1} S_{i+1} + y^*_{i+2} S_{i+2} - y^*_{\mathrm{op},i} S_{\mathrm{op},i} - y^*_{\mathrm{op},i+1} S_{\mathrm{op},i+1}}{S_i + S_{i+1} + S_{i+2} - S_{\mathrm{op},i} - S_{\mathrm{op},i+1}}\,, \tag{8.110}$$

is obtained using the formulae in Table 8.1 with the fulfilment degrees α_i, α_{i+1} and α_{i+2} being given by

$$\alpha_i = (1-\gamma_1)(1-\gamma_2)(1-\gamma_3)\,, \tag{8.111}$$

$$\begin{aligned}\alpha_{i+1} = \max(&\gamma_1(1-\gamma_2)(1-\gamma_3), (1-\gamma_1)\gamma_2(1-\gamma_3), (1-\gamma_1)(1-\gamma_2)\gamma_3,\\ &\gamma_1(1-\gamma_2)\gamma_3, (1-\gamma_1)\gamma_2\gamma_3)\,,\end{aligned} \tag{8.112}$$

$$\alpha_{i+2} = \max(\gamma_1\gamma_2(1-\gamma_3), \gamma_1\gamma_2\gamma_3)\,. \tag{8.113}$$

In the region of the input space where non-monotonicity is recorded for some fuzzy output partitions, the membership degrees γ_1, γ_2 and γ_3 satisfy

$$0.5 > \gamma_3 > \gamma_1 \qquad \wedge \qquad 0.5 > \gamma_3 > \gamma_2\,, \tag{8.114}$$

and the fulfilment degrees α_i, α_{i+1} and α_{i+2} are given by

$$\alpha_i = (1-\gamma_1)(1-\gamma_2)(1-\gamma_3)\,, \tag{8.115}$$

$$\alpha_{i+1} = (1-\gamma_1)(1-\gamma_2)\gamma_3\,, \tag{8.116}$$

$$\alpha_{i+2} = \gamma_1\gamma_2(1-\gamma_3)\,. \tag{8.117}$$

The derivative of y^*_{COG} to γ_3 is a complex function of γ_1, γ_2, γ_3, $l_{i\text{-}1}$, l_i, l_{i+1}, l_{i+2}, k_i, k_{i+1} and k_{i+2}. Again, the characterization of the class of fuzzy output partitions resulting in a positive derivative of y^*_{COG} to γ_3 for all inputs $\mathbf{x}$ will not be derived from this complex function, but from the equation obtained for the derivative if γ_1, γ_2 and γ_3 approach 0.5. We learned from the experiments that if the derivative of y^*_{COG} to γ_3 is negative for some inputs $\mathbf{x}$, it is negative for γ_1, γ_2 and γ_3 approaching 0.5 (from the left). When γ_1, γ_2 and γ_3 approach 0.5 the derivative of y^*_{COG} to γ_3 is given by

$$\lim_{\substack{\text{Case XVI}\\ \gamma_1\to 0.5\\ \gamma_2\to 0.5\\ \gamma_3\to 0.5}} \frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_3} = - \lim_{\substack{\text{Case XI}\\ \gamma_1\to 0.5\\ \gamma_2\to 0.5\\ \gamma_3\to 0.5}} \frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1}\,. \tag{8.118}$$

Thus, models with fuzzy partitions for which a strictly positive value was obtained for $\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1}$ in Eq. (8.107), being partitions 2, 6, 7 and 8 (Table 8.5), are not monotone if their rule base contains rules corresponding to Case XVI. For models applying partitions 6, 7 and 8 non-monotonicity was indeed reported for some inputs $(X_1, X_2, X_3) \in [0.245 : 0.005 : 0.755]^3$. Experiments with a smaller discretization step revealed that also for models applying partition 2 non-monotonicity is obtained (inputs $(X_1, X_2, X_3) \in [0.45 : 0.001 : 0.55]^3$ and $d = 0.009 \pm 0.002$).

In order to guarantee monotonicity for models with a rule base containing rules corresponding to Case XI, as well as for models with a rule base containing rules corresponding to Case XVI, a fuzzy partition should be used satisfying

$$\lim_{\substack{\text{Case XI}\\ \gamma_1\to 0.5\\ \gamma_2\to 0.5\\ \gamma_3\to 0.5}} \frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} = \lim_{\substack{\text{Case XVI}\\ \gamma_1\to 0.5\\ \gamma_2\to 0.5\\ \gamma_3\to 0.5}} \frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_3} = 0\,. \tag{8.119}$$

As the two derivatives are equal to zero if the membership functions A_i, A_{i+1} and A_{i+2} have an identical shape,

$$(\exists l > 0)(\forall j \in \{i-1, \ldots, i+2\})(l_j = l)\,, \tag{8.120}$$

$$(\exists k \geq 0)(\forall j \in \{i, \ldots, i+2\})(k_j = k)\,, \tag{8.121}$$

and Eq. (8.107) does not allow a straightforward, user friendly formulation of the class of membership functions A_i, A_{i+1} and A_{i+2} for which both derivatives are zero, only fuzzy models with linguistic output values described by identically shaped membership functions in the consequents of their rules (The linguistic output values A_1 and A_n are excluded as they are described by a trapezium with a vertical side ($l_0 = l_n = 0$).)

$$(\forall s \in \{1, \ldots, r\})(i_s \notin \{1, n\})\,, \tag{8.122}$$

$$(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)\,, \tag{8.123}$$

$$(\exists k \leq 0)(\forall i \in I)(k_i = k)\,, \tag{8.124}$$

are considered in the theoretical analysis of the monotonicity of models with three input variables applying $T_{\mathbf{P}}$ combined with the COG defuzzification method.

8.5.2 Theoretical analysis

In this section it is shown that when applying $T_{\mathbf{P}}$ combined with the COG defuzzification method, one will always obtain a monotone input-output behaviour for a model with three input variables, a monotone smooth rule base and linguistic values in its rule consequents satisfying Eqs. (8.122)–(8.124).

The general representation of the set of eight rules that can be fired simultaneously in a model with three input variables X_1, X_2 and X_3 is

IF	X_1 IS B_1^1	AND	X_2 IS B_1^2	AND	X_3 IS B_1^3	THEN Y IS A_i
IF	X_1 IS B_1^1	AND	X_2 IS B_1^2	AND	X_3 IS B_2^3	THEN Y IS $A_{i+p_1+p_2+p_3}$
IF	X_1 IS B_1^1	AND	X_2 IS B_2^2	AND	X_3 IS B_1^3	THEN Y IS $A_{i+p_1+p_2}$
IF	X_1 IS B_1^1	AND	X_2 IS B_2^2	AND	X_3 IS B_2^3	THEN Y IS $A_{i+p_1+p_2+p_3+p_5}$
IF	X_1 IS B_2^1	AND	X_2 IS B_1^2	AND	X_3 IS B_1^3	THEN Y IS A_{i+p_1}
IF	X_1 IS B_2^1	AND	X_2 IS B_1^2	AND	X_3 IS B_2^3	THEN Y IS $A_{i+p_1+p_2+p_3+p_6}$
IF	X_1 IS B_2^1	AND	X_2 IS B_2^2	AND	X_3 IS B_1^3	THEN Y IS $A_{i+p_1+p_2+p_4}$
IF	X_1 IS B_2^1	AND	X_2 IS B_2^2	AND	X_3 IS B_2^3	THEN Y IS $A_{i+p_7'}$

with $p_7' = p_1 + p_2 + \max(p_4, p_3 + p_5, p_3 + p_6) + p_7$ and $p_1, p_2, p_3, p_4, p_5, p_6, p_7 \in \mathbb{N}$.

When the rule base of a model is smooth, one of the 18 combinations listed in Table 7.1 should be used for the parameters p_i. In the following paragraphs the monotonicity for Cases I to XVIII will be investigated for inputs $\mathbf{x}$ for which membership degrees γ_1, γ_2 and γ_3 can be defined as follows

$$\gamma_1 = 1 - B_{j_1}^1(x_1) = B_{j_1+1}^1(x_1)\,, \tag{8.125}$$
$$\gamma_2 = 1 - B_{j_2}^2(x_2) = B_{j_2+1}^2(x_2)\,, \tag{8.126}$$
$$\gamma_3 = 1 - B_{j_3}^3(x_3) = B_{j_3+1}^3(x_3)\,, \tag{8.127}$$

or in other words, for inputs $\mathbf{x}$ not firing any other rule than the eight rules above.

Non-zero α_i If $(p_1, p_2, p_3, p_4, p_5, p_6, p_7) = (0, 0, 0, 0, 0, 0, 0)$ (Case I), the eight rules contain a same linguistic output value A_i in their consequent. As a result, for all inputs $\mathbf{x}$ not firing any other rule than these eight rules (Eqs. (8.125)–(8.127)), only the linguistic output value A_i is fired

$$(\alpha_i > 0)\,, (\forall j \in I \setminus \{i\})(\alpha_j = 0)\,, \tag{8.128}$$

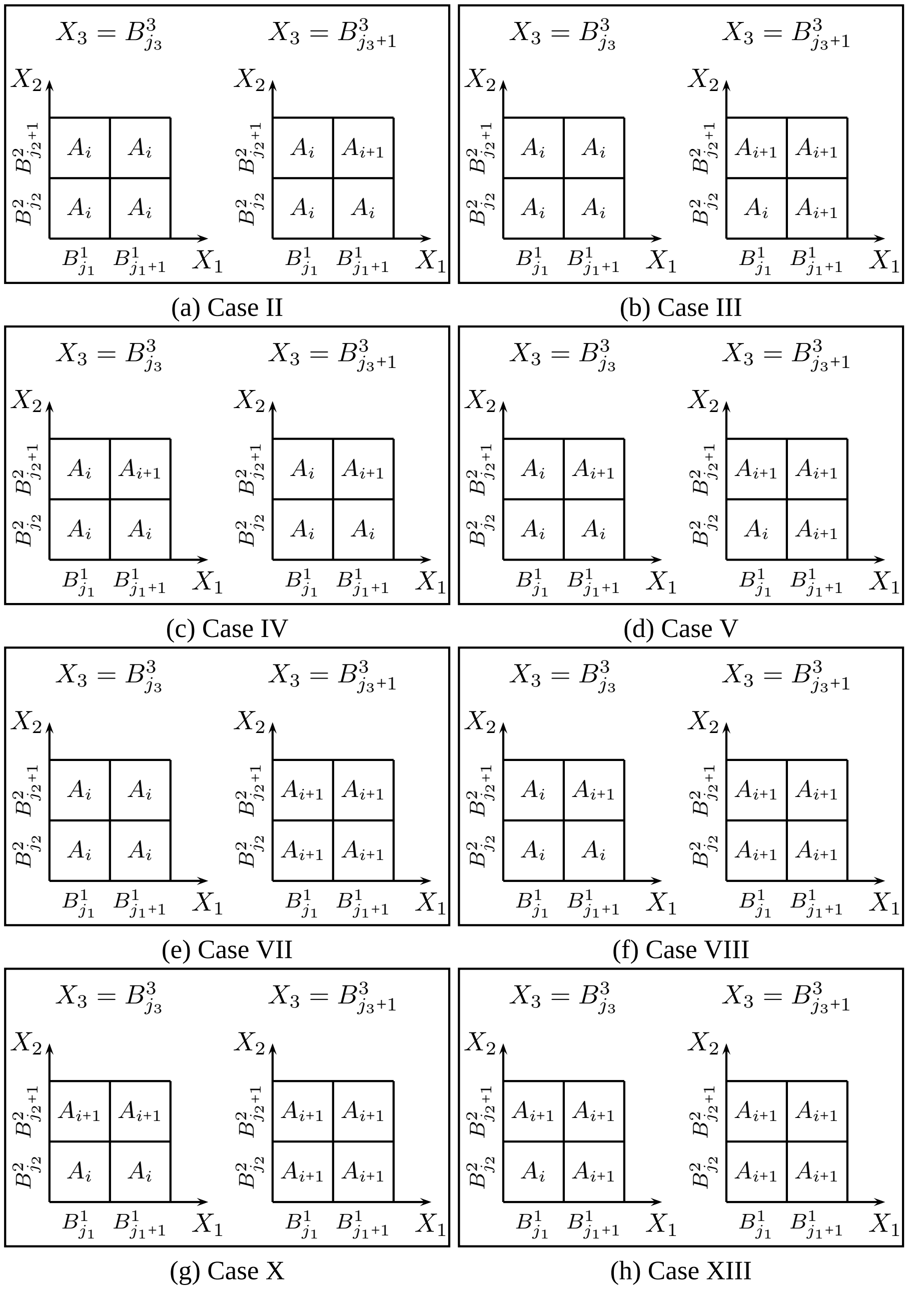
and the crisp output y^*_{COG} (Eq. (8.1) with Table 8.2) is equal to the midpoint of the kernel of A_i

$$y^*_{\text{COG}} = c_i\,. \tag{8.129}$$

As the crisp output y^*_{COG} is independent of α_i, it holds that

$$\frac{dy^*_{\text{COG}}}{d\gamma_1} = 0 \qquad \frac{dy^*_{\text{COG}}}{d\gamma_2} = 0 \qquad \frac{dy^*_{\text{COG}}}{d\gamma_3} = 0\,, \tag{8.130}$$

and monotonicity is guaranteed.



(a) Case II (b) Case III

(c) Case IV (d) Case V

(e) Case VII (f) Case VIII

(g) Case X (h) Case XIII

Figure 8.22: Cases considered in the discussion about models with three input variables and a monotone smooth rule base with A_i and A_{i+1} in the rule consequents.

Non-zero α_i and α_{i+1} The eight rules obtained for Cases II-V, VII-VIII, X and XIII are represented in Fig. 8.22. In these cases the linguistic values A_i and A_{i+1} appear in the consequents. For all inputs $\mathbf{x}$ not firing any other rule than these eight rules (Eqs. (8.125)–(8.127)), the crisp output y^*_{COG} , expressed as a function of α_i and α_{i+1}, is given by

$$y^*_{\mathrm{COG}} = o_i + \frac{(\alpha_{i+1} - \alpha_i)(6(l+k)^2(\alpha_i^2 + \alpha_{i+1}^2) + (13l^2 + 24lk + 12k^2)\alpha_i\alpha_{i+1})}{6(\alpha_i + \alpha_{i+1})(2(l+k)(\alpha_i^2 + \alpha_{i+1}^2) + (3l+4k)\alpha_i\alpha_{i+1})}\,, \tag{8.131}$$

and its derivatives to α_i and α_{i+1} are

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \alpha_i} = -\alpha_{i+1}\frac{C_1}{C_2}\,, \tag{8.132}$$

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \alpha_{i+1}} = \alpha_i\frac{C_1}{C_2}\,, \tag{8.133}$$

with $C_1, C_2 \in \mathbb{R}_0^+$

$$\begin{aligned} C_1 &= (8l^3 + 29l^2k + 33lk^2 + 12k^3)(\alpha_i + \alpha_{i+1})^4 \\ &\quad + (11l^2 + 18lk + 6k^2)l\alpha_i\alpha_{i+1}(\alpha_i + \alpha_{i+1})^2 \\ &\quad + l^3\alpha_{i+1}\alpha_i(\alpha_i^2 + \alpha_{i+1}\alpha_i + \alpha_{i+1}^2)\,, \\ C_2 &= 3(\alpha_i + \alpha_{i+1})^2(2(l+k)(\alpha_i^2 + \alpha_{i+1}^2) + (3l+4k)\alpha_i\alpha_{i+1})^2\,. \end{aligned}$$

Thus, the positivity of the derivatives of y^*_{COG} to γ_1, γ_2 and γ_3 can be restated as

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} \geq 0 \qquad \Leftrightarrow \qquad -\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_1} + \alpha_i\frac{\partial \alpha_{i+1}}{\partial \gamma_1} \geq 0\,, \tag{8.134}$$

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_2} \geq 0 \qquad \Leftrightarrow \qquad -\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_2} + \alpha_i\frac{\partial \alpha_{i+1}}{\partial \gamma_2} \geq 0\,, \tag{8.135}$$

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_3} \geq 0 \qquad \Leftrightarrow \qquad -\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_3} + \alpha_i\frac{\partial \alpha_{i+1}}{\partial \gamma_3} \geq 0\,. \tag{8.136}$$

For models with three input variables, it is hard to graphically represent the regions of the input space where the fulfilment degrees are described by a different function of γ_1, γ_2 and γ_3. Therefore the regions are defined by the equations in Table 8.6. One easily verifies that for all $(\gamma_1, \gamma_2, \gamma_3)$ belonging to a boundary plane between different regions of the input space and thus satisfying equations defining different subcases of a case, the functions for the fulfilment degrees holding in the corresponding subcases coincide. Monotonicity is guaranteed in Cases II-V, VII-VIII, X and XIII as for all (α_i, α_{i+1})-pairs the expressions on the right hand side of the arrows in Eqs. (8.134)–(8.136) are satisfied for $(\gamma_1, \gamma_2, \gamma_3) \in\ ]0,1[^3$. The values obtained for $-\alpha_{i+1}\frac{\partial \alpha_i}{\partial \gamma_l} + \alpha_i\frac{\partial \alpha_{i+1}}{\partial \gamma_l}$ are given in Table 8.7 for all (α_i, α_{i+1})-pairs.

Table 8.6: Definition of the regions of the input space where α_i and α_{i+1} are described by different functions of γ_1, γ_2 and γ_3 for Cases II-V, VII-VIII, X and XIII. The functions are given in Table 8.7.

	conditions on $(\gamma_1, \gamma_2, \gamma_3)$	(α_i, α_{i+1})
II a	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $0.5 \geq \gamma_3$	1
b	$\gamma_1 \geq 0.5$, $0.5 \geq \gamma_2$, $0.5 \geq \gamma_3$	2
c	$0.5 \geq \gamma_1$, $\gamma_2 \geq 0.5$, $0.5 \geq \gamma_3$	3
d	$\gamma_1 \geq 0.5$, $\gamma_2 \geq 0.5$, $\gamma_1 \geq \gamma_3$, $\gamma_2 \geq \gamma_3$	4
e	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $\gamma_3 \geq 0.5$	5
f	$\gamma_1 \geq 0.5$, $\gamma_1 \geq \gamma_2$, $\gamma_3 \geq \gamma_2$, $\gamma_3 \geq 0.5$	6
g	$\gamma_2 \geq \gamma_1$, $\gamma_3 \geq \gamma_1$, $\gamma_2 \geq 0.5$, $\gamma_3 \geq 0.5$	7
III a	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $0.5 \geq \gamma_3$, $\gamma_1 \geq \gamma_2$	8
b	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $0.5 \geq \gamma_3$, $\gamma_2 \geq \gamma_1$	9
c	$\gamma_1 \geq 0.5$, $0.5 \geq \gamma_2$, $\gamma_1 \geq \gamma_3$	10
d	$\gamma_3 \geq 0.5 \geq \gamma_2$, $\gamma_3 \geq \gamma_1 \geq \gamma_2$	11
e	$\gamma_3 \geq 0.5 \geq \gamma_1$, $\gamma_3 \geq \gamma_2 \geq \gamma_1$	12
f	$0.5 \geq \gamma_1$, $\gamma_2 \geq 0.5$, $\gamma_2 \geq \gamma_3$	13
g	$\gamma_1 \geq 0.5$, $\gamma_2 \geq 0.5$, $\gamma_1\gamma_2(1-\gamma_3) \geq (1-\gamma_1)(1-\gamma_2)\gamma_3$	4
h	$\gamma_1 \geq 0.5$, $\gamma_2 \geq 0.5$, $(1-\gamma_1)(1-\gamma_2)\gamma_3 \geq \gamma_1\gamma_2(1-\gamma_3)$	5
IV a	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $0.5 \geq \gamma_3$	14
b	$\gamma_1 \geq 0.5$, $\gamma_1 \geq \gamma_2$, $0.5 \geq \gamma_3$	15
c	$\gamma_2 \geq \gamma_1$, $\gamma_2 \geq 0.5$, $0.5 \geq \gamma_3$	16
d	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $\gamma_3 \geq 0.5$	5
e	$\gamma_1 \geq 0.5$, $\gamma_1 \geq \gamma_2$, $\gamma_3 \geq 0.5$	6
f	$\gamma_2 \geq \gamma_1$, $\gamma_2 \geq 0.5$, $\gamma_3 \geq 0.5$	7
V a	$0.5 \geq \gamma_2 \geq \gamma_1$, $0.5 \geq \gamma_3 \geq \gamma_1$	9
b	$0.5 \geq \gamma_1 \geq \gamma_2$, $0.5 \geq \gamma_3 \geq \gamma_2$	8
c	$0.5 \geq \gamma_1 \geq \gamma_3$, $0.5 \geq \gamma_2 \geq \gamma_3$	14
d	$\gamma_1 \geq \gamma_2 \geq 0.5$, $\gamma_1 \geq \gamma_3 \geq 0.5$	2
e	$\gamma_2 \geq \gamma_1 \geq 0.5$, $\gamma_2 \geq \gamma_3 \geq 0.5$	3
f	$\gamma_3 \geq \gamma_2 \geq 0.5$, $\gamma_3 \geq \gamma_1 \geq 0.5$	5
g	$\gamma_1 \geq \gamma_3 \geq \gamma_2$, $\gamma_1 \geq 0.5 \geq \gamma_2$	10
h	$\gamma_1 \geq \gamma_2 \geq \gamma_3$, $\gamma_1 \geq 0.5 \geq \gamma_3$	15
i	$\gamma_2 \geq \gamma_3 \geq \gamma_1$, $\gamma_2 \geq 0.5 \geq \gamma_1$	13
j	$\gamma_2 \geq \gamma_1 \geq \gamma_3$, $\gamma_2 \geq 0.5 \geq \gamma_3$	16
k	$\gamma_3 \geq \gamma_1 \geq \gamma_2$, $\gamma_3 \geq 0.5 \geq \gamma_2$	11
l	$\gamma_3 \geq \gamma_2 \geq \gamma_1$, $\gamma_3 \geq 0.5 \geq \gamma_1$	12
VII a	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$	17
b	$\gamma_1 \geq 0.5$, $0.5 \geq \gamma_2$	10
c	$0.5 \geq \gamma_1$, $\gamma_2 \geq 0.5$	13
d	$\gamma_1 \geq 0.5$, $\gamma_2 \geq 0.5$	4
VIII a	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $\gamma_1\gamma_2(1-\gamma_3) \geq (1-\gamma_1)(1-\gamma_2)\gamma_3$	14

continued on next page

continued from previous page

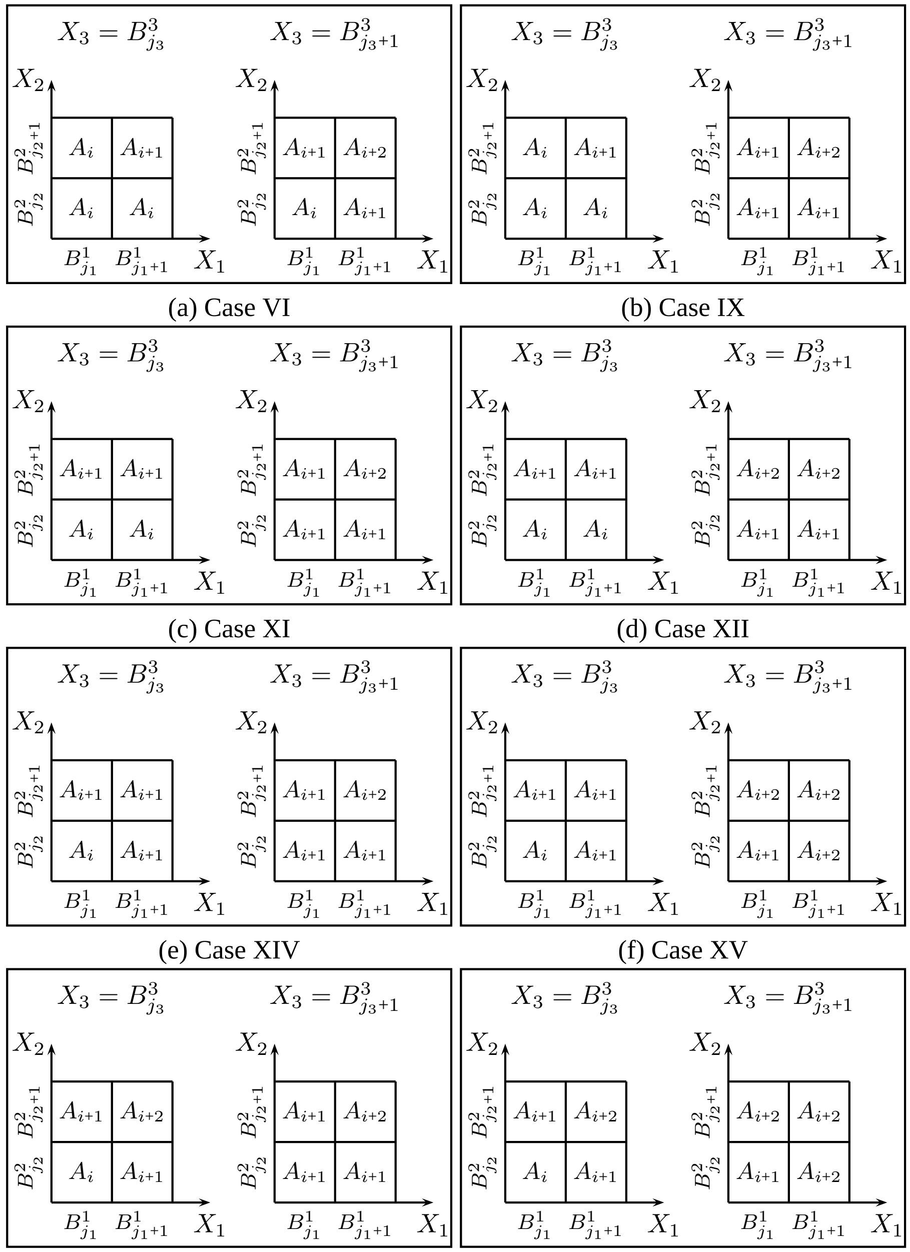
	conditions on $(\gamma_1, \gamma_2, \gamma_3)$	(α_i, α_{i+1})
b	$0.5 \geq \gamma_1, 0.5 \geq \gamma_2, (1-\gamma_1)(1-\gamma_2)\gamma_3 \geq \gamma_1\gamma_2(1-\gamma_3)$	17
c	$\gamma_1 \geq 0.5 \geq \gamma_3, \gamma_1 \geq \gamma_2 \geq \gamma_3$	15
d	$\gamma_1 \geq 0.5 \geq \gamma_2, \gamma_1 \geq \gamma_2, \gamma_3 \geq \gamma_2$	10
e	$\gamma_1 \geq \gamma_2 \geq 0.5, \gamma_3 \geq 0.5$	2
f	$\gamma_2 \geq \gamma_1 \geq \gamma_3, \gamma_2 \geq 0.5 \geq \gamma_3$	16
g	$\gamma_2 \geq 0.5 \geq \gamma_1, \gamma_3 \geq \gamma_1$	13
h	$\gamma_2 \geq \gamma_1 \geq 0.5, \gamma_3 \geq 0.5$	3
X a	$0.5 \geq \gamma_1, \gamma_2 \geq 0.5, \gamma_3 \geq 0.5$	9
b	$0.5 \geq \gamma_1, 0.5 \geq \gamma_3, \gamma_2 \geq \gamma_3$	18
c	$0.5 \geq \gamma_1, 0.5 \geq \gamma_2, \gamma_3 \geq \gamma_2$	17
d	$\gamma_1 \geq 0.5, \gamma_2 \geq 0.5, \gamma_3 \geq 0.5$	2
e	$\gamma_1 \geq 0.5, 0.5 \geq \gamma_3, \gamma_2 \geq \gamma_3$	15
f	$\gamma_1 \geq 0.5, 0.5 \geq \gamma_2, \gamma_3 \geq \gamma_2$	10
XIII a	$\gamma_1 \geq \gamma_2, \gamma_1 \geq \gamma_3, 0.5 \geq \gamma_2, 0.5 \geq \gamma_3$	19
b	$\gamma_2 \geq \gamma_1, \gamma_2 \geq \gamma_3, 0.5 \geq \gamma_1, 0.5 \geq \gamma_3$	18
c	$\gamma_1 \geq 0.5, \gamma_2 \geq 0.5, 0.5 \geq \gamma_3$	14
d	$\gamma_3 \geq \gamma_1, \gamma_3 \geq \gamma_2, 0.5 \geq \gamma_1, 0.5 \geq \gamma_2$	17
e	$\gamma_1 \geq 0.5, 0.5 \geq \gamma_2, \gamma_3 \geq 0.5$	8
f	$0.5 \geq \gamma_1, \gamma_2 \geq 0.5, \gamma_3 \geq 0.5$	9
g	$\gamma_1 \geq 0.5, \gamma_2 \geq 0.5, \gamma_3 \geq 0.5$	1

Non-zero α_i, α_{i+1} and α_{i+2} The eight rules obtained for Cases VI, IX, XI-XII and XIV-XVII are represented in Fig. 8.23. In these cases the linguistic values A_i, A_{i+1} and A_{i+2} appear in the consequents. For all inputs $\mathbf{x}$ not firing any other rule than these eight rules (Eqs. (8.125)–(8.127)), the crisp output y^*_{COG} , expressed as a function of α_i, α_{i+1} and α_{i+2}, is given by

$$\begin{aligned} y^*_{\mathrm{COG}} = c_{i+1} + \Big[&(\alpha_{i+2}-\alpha_i)(12(k+l)^2(\alpha_i\alpha_{i+2}+\alpha_{i+1}(\alpha_i+\alpha_{i+2}))^2 + 3(8(l+k)^2 \\ &- lk)\alpha_i\alpha_{i+1}^2\alpha_{i+2} + (22(l+k)^2 + (l+2k)k)(\alpha_i+\alpha_{i+2})\alpha_{i+1}^3 \\ &+ (8(l+k)^2 + (5l+4k)k)\alpha_{i+1}^4)\Big] \times \Big[6(\alpha_i+\alpha_{i+1})(\alpha_{i+1}+\alpha_{i+2}) \\ &((l+k)(2\alpha_i\alpha_{i+2}(\alpha_i+\alpha_{i+2}) + 2(\alpha_i+\alpha_{i+2})^2\alpha_{i+1} + 3(\alpha_i+\alpha_{i+2}) \\ &\alpha_{i+1}^2 + 2\alpha_{i+1}^3) + k\alpha_{i+1}(2\alpha_i\alpha_{i+2} + (\alpha_i+\alpha_{i+2})\alpha_{i+1}))\Big]^{-1}. \quad (8.137) \end{aligned}$$

Table 8.7: Values obtained for $-\alpha_{i+1}\frac{\partial\alpha_i}{\partial\gamma_l}+\alpha_i\frac{\partial\alpha_{i+1}}{\partial\gamma_l}$ in Eqs. (8.134)–(8.136) for (α_i,α_{i+1}) occurring in Cases II-V, VII-VIII, X and XIII.

	α_i	α_{i+1}	$-\alpha_{i+1}\frac{\partial\alpha_i}{\partial\gamma_1}+\alpha_i\frac{\partial\alpha_{i+1}}{\partial\gamma_1}$	$-\alpha_{i+1}\frac{\partial\alpha_i}{\partial\gamma_2}+\alpha_i\frac{\partial\alpha_{i+1}}{\partial\gamma_2}$	$-\alpha_{i+1}\frac{\partial\alpha_i}{\partial\gamma_3}+\alpha_i\frac{\partial\alpha_{i+1}}{\partial\gamma_3}$
1	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$\gamma_1\gamma_2\gamma_3$	$\gamma_2(1-\gamma_2)\gamma_3(1-\gamma_3)$	$\gamma_1(1-\gamma_1)\gamma_3(1-\gamma_3)$	$\gamma_1(1-\gamma_1)\gamma_2(1-\gamma_2)$
2	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1\gamma_2\gamma_3$	0	$\gamma_1^2\gamma_3(1-\gamma_3)$	$\gamma_1^2\gamma_2(1-\gamma_2)$
3	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2\gamma_3$	$\gamma_2^2\gamma_3(1-\gamma_3)$	0	$\gamma_1(1-\gamma_1)\gamma_2^2$
4	$\gamma_1\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2\gamma_3$	0	0	$\gamma_1^2\gamma_2^2$
5	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$\gamma_1\gamma_2\gamma_3$	$\gamma_2(1-\gamma_2)\gamma_3^2$	$\gamma_1(1-\gamma_1)\gamma_3^2$	0
6	$\gamma_1(1-\gamma_2)\gamma_3$	$\gamma_1\gamma_2\gamma_3$	0	$\gamma_1^2\gamma_3^2$	0
7	$(1-\gamma_1)\gamma_2\gamma_3$	$\gamma_1\gamma_2\gamma_3$	$\gamma_2^2\gamma_3^2$	0	0
8	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)\gamma_3$	$(1-\gamma_2)^2\gamma_3(1-\gamma_3)$	0	$\gamma_1(1-\gamma_1)(1-\gamma_2)^2$
9	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)\gamma_2\gamma_3$	0	$(1-\gamma_1)^2\gamma_3(1-\gamma_3)$	$(1-\gamma_1)^2\gamma_2(1-\gamma_2)$
10	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)\gamma_3$	0	0	$\gamma_1^2(1-\gamma_2)^2$
11	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$\gamma_1(1-\gamma_2)\gamma_3$	$(1-\gamma_2)^2\gamma_3^2$	0	0
12	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$(1-\gamma_1)\gamma_2\gamma_3$	0	$(1-\gamma_1)^2\gamma_3^2$	0
13	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$(1-\gamma_1)\gamma_2\gamma_3$	0	0	$(1-\gamma_1)^2\gamma_2^2$
14	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$	$\gamma_2(1-\gamma_2)(1-\gamma_3)^2$	$\gamma_1(1-\gamma_1)(1-\gamma_3)^2$	0
15	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$	0	$\gamma_1^2(1-\gamma_3)^2$	0
16	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$	$\gamma_2^2(1-\gamma_3)^2$	0	0
17	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	0	0	$(1-\gamma_1)^2(1-\gamma_2)^2$
18	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	0	$(1-\gamma_1)^2(1-\gamma_3)^2$	0
19	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_2)^2(1-\gamma_3)^2$	0	0



(a) Case VI (b) Case IX
(c) Case XI (d) Case XII
(e) Case XIV (f) Case XV
(g) Case XVI (h) Case XVII

Figure 8.23: Cases considered in the discussion about models with three input variables and a monotone smooth rule base with A_i, A_{i+1} and A_{i+2} in the rule consequents.

and its derivatives to α_i, α_{i+1} and α_{i+2} are

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \alpha_i} = -C_1(\alpha_i, \alpha_{i+1}, \alpha_{i+2}) \,, \tag{8.138}$$

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \alpha_{i+1}} = (\alpha_{i+2} - \alpha_i) C_2(\alpha_i, \alpha_{i+1}, \alpha_{i+2}) \,, \tag{8.139}$$

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \alpha_{i+2}} = C_1(\alpha_{i+2}, \alpha_{i+1}, \alpha_i) \,. \tag{8.140}$$

The functions C_1 and C_2 of α_i, α_{i+1} and α_{i+2} are given in Eqs. (D.1)–(D.2) in Appendix D and satisfy following properties

$$(\forall l \in \mathbb{R}_0^+)(\forall k \in \mathbb{R}^+)(\forall (\alpha_i, \alpha_{i+1}, \alpha_{i+2}) \in \,]0,1[^3)(C_1(\alpha_i, \alpha_{i+1}, \alpha_{i+2}) \geq 0) \,, \tag{8.141}$$

$$(\forall l \in \mathbb{R}_0^+)(\forall k \in \mathbb{R}^+)(\forall (\alpha_i, \alpha_{i+1}, \alpha_{i+2}) \in \,]0,1[^3)(C_2(\alpha_i, \alpha_{i+1}, \alpha_{i+2}) \geq 0) \,, \tag{8.142}$$

$$C_2(\alpha_i, \alpha_{i+1}, \alpha_{i+2}) = C_2(\alpha_{i+2}, \alpha_{i+1}, \alpha_i) \,. \tag{8.143}$$

Thus, the derivative of y^*_{COG} to γ_1 (resp. γ_2 and γ_3) is given by

$$\begin{aligned} \frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} = &-C_1(\alpha_i, \alpha_{i+1}, \alpha_{i+2}) \frac{\partial \alpha_i}{\partial \gamma_1} + (\alpha_{i+2} - \alpha_i) C_2(\alpha_i, \alpha_{i+1}, \alpha_{i+2}) \frac{\partial \alpha_{i+1}}{\partial \gamma_1} \\ &+ C_1(\alpha_{i+2}, \alpha_{i+1}, \alpha_i) \frac{\partial \alpha_{i+2}}{\partial \gamma_1} \,. \end{aligned} \tag{8.144}$$

One easily verifies that if monotonicity is guaranteed for fulfilment degrees α_i, α_{i+1} and α_{i+2} described by certain functions of γ_1, γ_2 and γ_3, monotonicity is also guaranteed for fulfilment degrees α_i, α_{i+1} and α_{i+2} obtained by permutation of the membership degrees γ_1, γ_2 and γ_3 in these functions. Furthermore it is shown in the following paragraphs that if monotonicity is obtained for $(\gamma_1, \gamma_2, \gamma_3) \in \,]\underline{\gamma_1}, \overline{\gamma_1}[\, \times \,]\underline{\gamma_2}, \overline{\gamma_2}[\, \times \,]\underline{\gamma_3}, \overline{\gamma_3}[$ and a certain set of fulfilment degrees α_i, α_{i+1} and α_{i+2}

$$\alpha_i = C_3(\gamma_1, \gamma_2, \gamma_3) \,, \quad \alpha_{i+1} = C_4(\gamma_1, \gamma_2, \gamma_3) \,, \quad \alpha_{i+2} = C_5(\gamma_1, \gamma_2, \gamma_3) \,, \tag{8.145}$$

monotonicity is also obtained for $(\gamma_1, \gamma_2, \gamma_3) \in \,]1 - \overline{\gamma_1}, 1 - \underline{\gamma_1}[\, \times \,]1 - \overline{\gamma_2}, 1 - \underline{\gamma_2}[\, \times \,]1 - \overline{\gamma_3}, 1 - \underline{\gamma_3}[$ and the fulfilment degrees α_i, α_{i+1}, and α_{i+2}

$$\begin{aligned} &\alpha_i = C_5(1-\gamma_1, 1-\gamma_2, 1-\gamma_3) \,, \qquad \alpha_{i+1} = C_4(1-\gamma_1, 1-\gamma_2, 1-\gamma_3) \,, \\ &\alpha_{i+2} = C_3(1-\gamma_1, 1-\gamma_2, 1-\gamma_3) \,. \end{aligned} \tag{8.146}$$

If monotonicity is guaranteed for the fulfilment degrees in Eq. (8.145), the following inequality holds for all $(\gamma_1,\gamma_2,\gamma_3) \in \left]\underline{\gamma_1},\overline{\gamma_1}\right[\times \left]\underline{\gamma_2},\overline{\gamma_2}\right[\times \left]\underline{\gamma_3},\overline{\gamma_3}\right[$:

$$\begin{aligned}
&- C_1(C_3(\gamma_1,\gamma_2,\gamma_3),C_4(\gamma_1,\gamma_2,\gamma_3),C_5(\gamma_1,\gamma_2,\gamma_3))\frac{\partial C_3(\gamma_1,\gamma_2,\gamma_3)}{\partial\gamma_1}\\
&+ (C_5(\gamma_1,\gamma_2,\gamma_3) - C_3(\gamma_1,\gamma_2,\gamma_3))\\
&C_2(C_3(\gamma_1,\gamma_2,\gamma_3),C_4(\gamma_1,\gamma_2,\gamma_3),C_5(\gamma_1,\gamma_2,\gamma_3))\frac{\partial C_4(\gamma_1,\gamma_2,\gamma_3)}{\partial\gamma_1}\\
&+ C_1(C_5(\gamma_1,\gamma_2,\gamma_3),C_4(\gamma_1,\gamma_2,\gamma_3),C_3(\gamma_1,\gamma_2,\gamma_3))\frac{\partial C_5(\gamma_1,\gamma_2,\gamma_3)}{\partial\gamma_1} \qquad \geq 0\,,
\end{aligned} \tag{8.147}$$

as well as for all $(1-\gamma_1, 1-\gamma_2, 1-\gamma_3)$ with $(\gamma_1,\gamma_2,\,\gamma_3) \in \left]1-\overline{\gamma_1},1-\underline{\gamma_1}\right[\times \left]1-\overline{\gamma_2},1-\underline{\gamma_2}\right[\times \left]1-\overline{\gamma_3},1-\underline{\gamma_3}\right[$:

$$\begin{aligned}
&- C_1(C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_4(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3))\\
&\quad\frac{\partial C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3)}{\partial(1-\gamma_1)}\\
&+ (C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3) - C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3))\\
&\quad C_2(C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_4(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3))\\
&\quad\frac{\partial C_4(1-\gamma_1,1-\gamma_2,1-\gamma_3)}{\partial(1-\gamma_1)}\\
&+ C_1(C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_4(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3))\\
&\quad\frac{\partial C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3)}{\partial(1-\gamma_1)}\\
&\geq 0\,.
\end{aligned} \tag{8.148}$$

Applying

$$\frac{\partial f(x)}{\partial x} = \frac{\partial f(x)}{\partial(1-x)}\frac{\partial(1-x)}{\partial x} = -\frac{\partial f(x)}{\partial(1-x)}\,, \tag{8.149}$$

and Eq. (8.143) the expression Eq. (8.148) converts to the derivative of y^*_{COG} to γ_1 (resp. γ_2 and γ_3) for the fulfilment degrees in Eq. (8.146)

$$\begin{aligned}
&- C_1(C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_4(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3))\\
&\quad\frac{\partial C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3)}{\partial\gamma_1}\\
&+ (C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3) - C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3))\\
&\quad C_2(C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_4(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3))\\
&\quad\frac{\partial C_4(1-\gamma_1,1-\gamma_2,1-\gamma_3)}{\partial\gamma_1}\\
&+ C_1(C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_4(1-\gamma_1,1-\gamma_2,1-\gamma_3),C_5(1-\gamma_1,1-\gamma_2,1-\gamma_3))\\
&\quad\frac{\partial C_3(1-\gamma_1,1-\gamma_2,1-\gamma_3)}{\partial\gamma_1}\\
&\geq 0\,,
\end{aligned} \tag{8.150}$$

which proves that monotonicity is also obtained for the fulfilment degrees in Eq. (8.146).

In Table 8.8 the regions are defined where the fulfilment degrees are described by a different function of γ_1, γ_2 and γ_3 for Cases VI, IX, XI-XII and XIV-XVII. In Table 8.9 an overview is given of the 20 types of $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplets that occur in these eight cases. Note that for all $(\gamma_1, \gamma_2, \gamma_3)$ belonging to a boundary plane between different regions of the input space and thus satisfying equalities defining different subcases of a case, the functions for the fulfilment degrees in the corresponding subcases coincide.

Table 8.8: Definition of the regions of the input space where α_i, α_{i+1} and α_{i+2} are described by different functions of γ_1, γ_2 and γ_3 for Cases VI, IX, XI-XII and XIV-XVII. The functions are given in Table 8.9.

	conditions on $(\gamma_1, \gamma_2, \gamma_3)$	$(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$
VI a	$0.5 \geq \gamma_2 \geq \gamma_1$, $0.5 \geq \gamma_3 \geq \gamma_1$	1
b	$0.5 \geq \gamma_1 \geq \gamma_2$, $0.5 \geq \gamma_3 \geq \gamma_2$	2
c	$0.5 \geq \gamma_1 \geq \gamma_3$, $0.5 \geq \gamma_2 \geq \gamma_3$	3
d	$\gamma_1 \geq \gamma_3 \geq \gamma_2$, $\gamma_1 \geq 0.5$	4
e	$\gamma_1 \geq \gamma_2 \geq \gamma_3$, $\gamma_1 \geq 0.5$	5
f	$\gamma_2 \geq \gamma_3 \geq \gamma_1$, $\gamma_2 \geq 0.5$	6
g	$\gamma_2 \geq \gamma_1 \geq \gamma_3$, $\gamma_2 \geq 0.5$	7
h	$\gamma_3 \geq \gamma_2 \geq \gamma_1$, $\gamma_3 \geq 0.5$	8
i	$\gamma_3 \geq \gamma_1 \geq \gamma_2$, $\gamma_3 \geq 0.5$	9
IX a	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $\gamma_1\gamma_2(1-\gamma_3) \geq (1-\gamma_1)(1-\gamma_2)\gamma_3$	3
b	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $(1-\gamma_1)(1-\gamma_2)\gamma_3 \geq \gamma_1\gamma_2(1-\gamma_3)$	10
c	$\gamma_1 \geq 0.5$, $\gamma_1 \geq \gamma_2 \geq \gamma_3$	5
d	$\gamma_1 \geq 0.5$, $\gamma_1 \geq \gamma_2$, $\gamma_3 \geq \gamma_2$	4
e	$\gamma_2 \geq 0.5$, $\gamma_2 \geq \gamma_1 \geq \gamma_3$	7
f	$\gamma_2 \geq 0.5$, $\gamma_2 \geq \gamma_1$, $\gamma_3 \geq \gamma_1$	6
XI a	$0.5 \geq \gamma_1$, $\gamma_2 \geq 0.5$, $\gamma_3 \geq 0.5$	1
b	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_3$, $\gamma_2 \geq \gamma_3$	11
c	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $\gamma_3 \geq \gamma_2$	10
d	$\gamma_1 \geq 0.5$, $\gamma_2 \geq \gamma_1$, $\gamma_3 \geq \gamma_1$	12
e	$\gamma_1 \geq 0.5$, $\gamma_1 \geq \gamma_3$, $\gamma_2 \geq \gamma_3$	5
f	$\gamma_1 \geq 0.5$, $\gamma_1 \geq \gamma_2$, $\gamma_3 \geq \gamma_2$	4
XII a	$0.5 \geq \gamma_1$, $\gamma_3 \geq \gamma_2$	13
b	$0.5 \geq \gamma_1$, $\gamma_2 \geq \gamma_3$	14
c	$\gamma_1 \geq 0.5$, $\gamma_3 \geq \gamma_2$	4
d	$\gamma_1 \geq 0.5$, $\gamma_2 \geq \gamma_3$	5
XIV a	$\gamma_1 \geq \gamma_2$, $\gamma_1 \geq \gamma_3$, $0.5 \geq \gamma_2$, $0.5 \geq \gamma_3$	15
b	$0.5 \geq \gamma_1$, $\gamma_2 \geq \gamma_1$, $\gamma_2 \geq \gamma_3$, $0.5 \geq \gamma_3$	11
c	$\gamma_1 \geq 0.5$, $\gamma_2 \geq 0.5$, $\gamma_1 \geq \gamma_3$, $\gamma_2 \geq \gamma_3$	3
d	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $\gamma_3 \geq \gamma_1$, $\gamma_3 \geq \gamma_2$	10

continued on next page

continued from previous page

	conditions on $(\gamma_1, \gamma_2, \gamma_3)$	$(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$
e	$\gamma_1 \geq 0.5$, $\gamma_3 \geq 0.5$, $\gamma_1 \geq \gamma_2$, $\gamma_3 \geq \gamma_2$	2
f	$\gamma_2 \geq 0.5$, $\gamma_3 \geq 0.5$, $\gamma_2 \geq \gamma_1$, $\gamma_3 \geq \gamma_1$	1
XV a	$\gamma_1 \geq 0.5$, $\gamma_2 \geq 0.5$, $\gamma_1\gamma_2(1-\gamma_3) \geq (1-\gamma_1)(1-\gamma_2)\gamma_3$	3
b	$\gamma_1 \geq 0.5$, $\gamma_2 \geq 0.5$, $(1-\gamma_1)(1-\gamma_2)\gamma_3 \geq \gamma_1\gamma_2(1-\gamma_3)$	10
c	$\gamma_1 \geq \gamma_2$, $0.5 \geq \gamma_2$, $\gamma_1 \geq \gamma_3$	16
d	$\gamma_3 \geq \gamma_1 \geq \gamma_2$, $0.5 \geq \gamma_2$	17
e	$\gamma_2 \geq \gamma_1$, $0.5 \geq \gamma_1$, $\gamma_2 \geq \gamma_3$	14
f	$\gamma_3 \geq \gamma_2 \geq \gamma_1$, $0.5 \geq \gamma_1$	13
XVI a	$\gamma_1 \geq \gamma_2$, $\gamma_1 \geq \gamma_3$, $0.5 \geq \gamma_3$	18
b	$\gamma_2 \geq \gamma_1$, $\gamma_2 \geq \gamma_3$, $0.5 \geq \gamma_3$	19
c	$0.5 \geq \gamma_3 \geq \gamma_1$, $0.5 \geq \gamma_3 \geq \gamma_2$	20
d	$0.5 \geq \gamma_1$, $0.5 \geq \gamma_2$, $\gamma_3 \geq 0.5$	10
e	$\gamma_1 \geq \gamma_2$, $\gamma_1 \geq 0.5$, $\gamma_3 \geq 0.5$	2
f	$\gamma_2 \geq \gamma_1$, $\gamma_2 \geq 0.5$, $\gamma_3 \geq 0.5$	1
XVII a	$\gamma_1 \geq \gamma_2 \geq \gamma_3$, $0.5 \geq \gamma_3$	18
b	$\gamma_1 \geq \gamma_3 \geq \gamma_2$, $0.5 \geq \gamma_2$	16
c	$\gamma_1 \geq \gamma_2 \geq 0.5$, $\gamma_1 \geq \gamma_3 \geq 0.5$	15
d	$\gamma_2 \geq \gamma_1 \geq \gamma_3$, $0.5 \geq \gamma_3$	19
e	$\gamma_2 \geq \gamma_3 \geq \gamma_1$, $0.5 \geq \gamma_1$	14
f	$\gamma_2 \geq \gamma_1 \geq 0.5$, $\gamma_2 \geq \gamma_3 \geq 0.5$	11
g	$\gamma_3 \geq \gamma_1 \geq \gamma_2$, $0.5 \geq \gamma_2$	17
h	$\gamma_3 \geq \gamma_2 \geq \gamma_1$, $0.5 \geq \gamma_1$	13
i	$\gamma_3 \geq \gamma_1 \geq 0.5$, $\gamma_3 \geq \gamma_2 \geq 0.5$	10

The expressions of the derivatives of y^*_{COG} to γ_1 and γ_2 for $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet 1 in Table 8.9, with

$$\begin{aligned} \alpha_i &= (1-\gamma_1)(1-\gamma_2)(1-\gamma_3)\,, \\ \alpha_{i+1} &= (1-\gamma_1)\gamma_2\gamma_3\,, \\ \alpha_{i+2} &= \gamma_1\gamma_2\gamma_3\,, \end{aligned}$$

are given in Eqs. (D.3)–(D.4) in Appendix D. Both derivatives are positive for all $l \in \mathbb{R}_0^+$, $k \in \mathbb{R}^+$ and $\gamma_1, \gamma_2, \gamma_3 \in\ ]0,1[$. The derivative of y^*_{COG} to γ_3 is obtained by substituting γ_2 by γ_3 in Eq. (D.4) and is therefore also positive for all $l \in \mathbb{R}_0^+$, $k \in \mathbb{R}^+$ and $\gamma_1, \gamma_2, \gamma_3 \in\ ]0,1[$. This not only proves that monotonicity is guaranteed for $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet 1 in Table 8.9, but also for the $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplets corresponding to the first triplet, *i.e.* triplets 2-3, 10-11 and 15.

Also for $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet 4, with

$$\begin{aligned} \alpha_i &= \gamma_1(1-\gamma_2)(1-\gamma_3)\,, \\ \alpha_{i+1} &= \gamma_1(1-\gamma_2)\gamma_3\,, \\ \alpha_{i+2} &= \gamma_1\gamma_2\gamma_3\,, \end{aligned}$$

Table 8.9: Triplets of fulfilment degrees α_i, α_{i+1} and α_{i+2} occurring in Cases VI, IX, XI-XII and XIV-XVII with their relationship to either triplet 1, 4 or 12.

	α_i	α_{i+1}	α_{i+2}	equal to	after substitution of			and inter-changing α_i and α_{i+2}
					γ_1	γ_2 by	γ_3	
1	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)\gamma_2\gamma_3$	$\gamma_1\gamma_2\gamma_3$	–				
2	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)\gamma_3$	$\gamma_1\gamma_2\gamma_3$	1	γ_2	γ_1	–	
3	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2\gamma_3$	1	γ_3	–	γ_1	
4	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)\gamma_3$	$\gamma_1\gamma_2\gamma_3$	–				
5	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2\gamma_3$	4	–	γ_3	γ_2	
6	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$(1-\gamma_1)\gamma_2\gamma_3$	$\gamma_1\gamma_2\gamma_3$	4	γ_2	γ_1	–	
7	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2\gamma_3$	4	γ_2	γ_3	γ_1	
8	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$(1-\gamma_1)\gamma_2\gamma_3$	$\gamma_1\gamma_2\gamma_3$	4	γ_3	γ_1	γ_2	
9	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$\gamma_1(1-\gamma_2)\gamma_3$	$\gamma_1\gamma_2\gamma_3$	4	γ_3	–	γ_1	
10	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$\gamma_1\gamma_2\gamma_3$	1	$1-\gamma_3$	$1-\gamma_2$	$1-\gamma_1$	yes
11	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2\gamma_3$	1	$1-\gamma_2$	$1-\gamma_1$	$1-\gamma_3$	yes
12	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)\gamma_2\gamma_3$	$\gamma_1\gamma_2\gamma_3$	–				
13	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$(1-\gamma_1)\gamma_2\gamma_3$	4	$1-\gamma_1$	$1-\gamma_3$	$1-\gamma_2$	yes
14	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$(1-\gamma_1)\gamma_2\gamma_3$	4	$1-\gamma_1$	$1-\gamma_2$	$1-\gamma_3$	yes
15	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1\gamma_2\gamma_3$	1	$1-\gamma_1$	$1-\gamma_2$	$1-\gamma_3$	yes
16	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)\gamma_3$	4	$1-\gamma_2$	$1-\gamma_1$	$1-\gamma_3$	yes
17	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$\gamma_1(1-\gamma_2)\gamma_3$	4	$1-\gamma_2$	$1-\gamma_3$	$1-\gamma_1$	yes
18	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$	4	$1-\gamma_3$	$1-\gamma_1$	$1-\gamma_2$	yes
19	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$	4	$1-\gamma_3$	$1-\gamma_2$	$1-\gamma_1$	yes
20	$(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$\gamma_1\gamma_2(1-\gamma_3)$	12	$1-\gamma_3$	$1-\gamma_2$	$1-\gamma_1$	yes

the derivatives of y^*_{COG} to γ_1, γ_2 and γ_3, given in Eqs. (D.5)–(D.7) in Appendix D, are positive for all $l \in \mathbb{R}_0^+$, $k \in \mathbb{R}^+$ and $\gamma_1, \gamma_2, \gamma_3 \in \,]0,1[$. Thus, monotonicity is guaranteed for $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplets 4-9, 13-14 and 16-19.

Finally, the derivatives of y^*_{COG} to γ_1, γ_2 and γ_3 show to be positive for $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet 12, with

$$\begin{aligned}
\alpha_i &= \gamma_1(1-\gamma_2)(1-\gamma_3)\,,\\
\alpha_{i+1} &= (1-\gamma_1)\gamma_2\gamma_3\,,\\
\alpha_{i+2} &= \gamma_1\gamma_2\gamma_3\,,
\end{aligned}$$

for all $l \in \mathbb{R}_0^+$, $k \in \mathbb{R}^+$ and $\gamma_1, \gamma_2, \gamma_3 \in \,]0.5,1[$. The derivatives to γ_1 and γ_2 are given in Eqs. (D.8)–(D.9) in Appendix D, and the derivative to γ_3 is obtained by exchanging γ_2 and γ_3 in Eq. (D.9). As $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet 12 only occurs for Case XId

$$\gamma_2 \geq \gamma_1 \geq 0.5 \qquad \wedge \qquad \gamma_3 \geq \gamma_1 \geq 0.5\,, \tag{8.151}$$

monotonicity is always guaranteed even if the derivatives are positive for $\gamma_1, \gamma_2, \gamma_3 \in \,]0.5,1[$ only. If monotonicity is guaranteed for $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet 12, it is also guaranteed for $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet 20 only occurring for Case XVIc

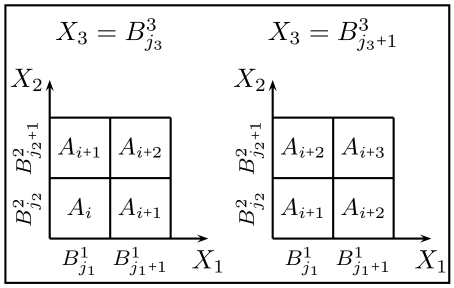
$$0.5 \geq \gamma_3 \geq \gamma_2 \qquad \wedge \qquad 0.5 \geq \gamma_3 \geq \gamma_1\,, \tag{8.152}$$

since in this case the derivatives of y^*_{COG} to γ_1, γ_2 and γ_3 are positive for $\gamma_1, \gamma_2, \gamma_3 \in \,]0,0.5[$.

Non-zero α_i, α_{i+1}, α_{i+2} and α_{i+3} Only for Case XVIII, represented in Fig. 8.24, the linguistic values A_i, A_{i+1}, A_{i+2} and A_{i+3} appear in the consequents of the eight rules. The functions of γ_1, γ_2 and γ_3 describing the fulfilment degrees α_i, α_{i+1}, α_{i+2} and α_{i+3} in the different regions of the input space are given in Table 8.10. Only Case XVIIIa will be discussed, as both the functions describing the region of the input space as the functions describing the fulfilment degrees for Cases XVIIIb-f can be obtained from the corresponding functions for Case XVIIIa by permuting γ_1, γ_2 and γ_3. This also implies that for $(\gamma_1, \gamma_2, \gamma_3)$ satisfying the inequalities describing the regions of the input space for different subcases of Case XVIII, the functions for the fulfilment degrees for the corresponding subcases coincide.

For all inputs $\mathbf{x}$ with

$$\begin{aligned}
\alpha_i &= (1-\gamma_1)(1-\gamma_2)(1-\gamma_3)\,,\\
\alpha_{i+1} &= \gamma_1(1-\gamma_2)(1-\gamma_3)\,,\\
\alpha_{i+2} &= \gamma_1\gamma_2(1-\gamma_3)\,,\\
\alpha_{i+3} &= \gamma_1\gamma_2\gamma_3\,,
\end{aligned}$$



Case XVIII

Figure 8.24: Case considered in the discussion about models with three input variables and a monotone smooth rule base with A_i, A_{i+1}, A_{i+2} and A_{i+3} in the rule consequents.

Table 8.10: Fulfilment degrees α_{i+1} and α_{i+2} in different parts of the input space for Case XVIII (with $\alpha_i = (1-\gamma_1)(1-\gamma_2)(1-\gamma_3)$ and $\alpha_{i+3} = \gamma_1\gamma_2\gamma_3$).

	conditions on $(\gamma_1, \gamma_2, \gamma_3)$	α_{i+1}	α_{i+2}
a	$\gamma_1 \geq \gamma_2 \geq \gamma_3$	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$
b	$\gamma_1 \geq \gamma_3 \geq \gamma_2$	$\gamma_1(1-\gamma_2)(1-\gamma_3)$	$\gamma_1(1-\gamma_2)\gamma_3$
c	$\gamma_2 \geq \gamma_1 \geq \gamma_3$	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$\gamma_1\gamma_2(1-\gamma_3)$
d	$\gamma_2 \geq \gamma_3 \geq \gamma_1$	$(1-\gamma_1)\gamma_2(1-\gamma_3)$	$(1-\gamma_1)\gamma_2\gamma_3$
e	$\gamma_3 \geq \gamma_1 \geq \gamma_2$	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$\gamma_1(1-\gamma_2)\gamma_3$
f	$\gamma_3 \geq \gamma_2 \geq \gamma_1$	$(1-\gamma_1)(1-\gamma_2)\gamma_3$	$(1-\gamma_1)\gamma_2\gamma_3$

the crisp output y^*_{COG} is given by

$$\begin{aligned} y^*_{\mathrm{COG}} = o_{i+1} + \Big[& l^2((\gamma_2^2 - 12\gamma_2 + 17)\gamma_1(1-\gamma_3) - (2\gamma_1^3 + 3\gamma_1^2 + 18)(1-\gamma_2) \\ & (1-\gamma_3) + 2\gamma_1\gamma_2^2(1-\gamma_2)(1-\gamma_3) + \gamma_1\gamma_2\gamma_3(\gamma_3+1)(11-2\gamma_3)) \\ & + 6lk(-(\gamma_1^2+6)(1-\gamma_2)(1-\gamma_3) + 5\gamma_1(1-\gamma_3) + (\gamma_3^2 + 8\gamma_3 \\ & - 3)\gamma_1\gamma_2) + 6k^2(-3(1-\gamma_2)(1-\gamma_3) + 2\gamma_1(1-\gamma_3) + \gamma_1\gamma_2 \\ & (4\gamma_3 - 1))\Big] \times \Big[6(l((\gamma_1^2 - \gamma_1 + 2)(1-\gamma_2)(1-\gamma_3) + \gamma_1\gamma_2(\gamma_2(1 \\ & - \gamma_3) + \gamma_3^2 + 1)) + 2k((1-\gamma_2)(1-\gamma_3) + \gamma_1\gamma_2))\Big]^{-1}, \end{aligned} \tag{8.153}$$

and its derivatives, given in Eqs. (D.10)–(D.12) in Appendix D, are positive for all $l \in \mathbb{R}_0^+$, $k \in \mathbb{R}^+$ and $\gamma_1, \gamma_2, \gamma_3$ satisfying $1 > \gamma_1 \geq \gamma_2 \geq \gamma_3 > 0$, *i.e.* for the $(\gamma_1, \gamma_2, \gamma_3)$ defining Case XVIIIa.

Table 8.11: Mamdani–Assilian models for which monotonicity is guaranteed when applying the COG defuzzification method characterized by a number of input variables m, a t-norm T, an either monotone or monotone smooth rule base and additional properties of the membership functions appearing in the rule consequents.

	m	T	rule base	additional properties A_{i_s}
1	1	$T_{\mathbf{M}}$	monotone	$(\forall s \in \{1,\ldots,r\})(i_s \notin \{1,n\})$ $(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)$
2	1	$T_{\mathbf{P}}$	monotone	
3	1	$T_{\mathbf{L}}$	monotone	$(\forall s \in \{1,\ldots,r\})(i_s \notin \{1,n\})$ $(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)$
4	2	$T_{\mathbf{P}}$	monotone and smooth	
5	3	$T_{\mathbf{P}}$	monotone and smooth	$(\forall s \in \{1,\ldots,r\})(i_s \notin \{1,n\})$ $(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)$ $(\exists k \geq 0)(\forall i \in I \setminus \{1,n\})(k_i = k)$

8.6 Conclusion

In this chapter, it was proved that a Mamdani–Assilian model applying the COG defuzzification method is monotone if it corresponds to one of the five model types listed in Table 8.11, characterized by a number of input variables m, a t-norm T, an either monotone or monotone smooth rule base and additional properties of the membership functions appearing in the rule consequents. For the t-norms $T_{\mathbf{M}}$ and $T_{\mathbf{L}}$, models with a single input variable show a monotone input-output behaviour for any monotone rule base when the linguistic output values in the consequents of the rules are defined by trapezial or triangular membership functions with intervals of changing membership degree of equal length, whereas for the t-norm $T_{\mathbf{P}}$, models with a single input variable show a monotone input-output behaviour for any monotone rule base and any fuzzy output partition. When designing a monotone model with more than one input variable, one should opt for the t-norm $T_{\mathbf{P}}$ and use a monotone smooth rule base. It was shown that monotonicity of models with two input variables applying $T_{\mathbf{P}}$ is guaranteed for any monotone smooth rule base and any fuzzy partition. Finally, it was proved that a monotone input-output behaviour is always obtained for models with three input variables applying $T_{\mathbf{P}}$ and a monotone smooth rule base when the linguistic output values in the consequents of the rules are defined by trapezial or triangular membership functions of identical shape.

For models with three input variables and applying $T_{\mathbf{P}}$, apart from an analytic investigation revealing that monotonicity is guaranteed for all models with a monotone smooth rule base with linguistic output values defined by membership functions of identical shape in the rule consequents, numerical experiments were carried out. These numerical experiments leads one to suspect that monotonicity is not only guaranteed for models with membership functions of identical shape in the rule consequents, but more generally for all models with a fuzzy output partition satisfying Eq. (8.119), in other words for all models with a fuzzy output partition for which the numerator of the expression in Eq. (8.107) is equal to zero for all $i \in I \setminus \{n\text{-}1, n\}$

$$\begin{aligned}
&4(l_{i+2} - l_{i-1})((l_{i-1} + l_{i+2})(3l_i + 3l_{i+1} + 8k_{i+1}) + 9l_i l_{i+1} + 12k_{i+1}^2) \\
&\quad + 12(k_{i+2} - k_i)((k_i + k_{i+2})(3l_i + 3l_{i+1} + 8k_{i+1}) + 6l_i l_{i+1} + 8k_{i+1}^2) \\
&\quad + (l_{i+1} - l_i)((l_i + l_{i+1})(15l_i + 15l_{i+1} + 56k_{i+1}) + 36k_{i+1}^2) \\
&\quad + 8(l_i^2 l_{i+2} - l_{i-1} l_{i+1}^2) + 36k_{i+1}(l_i l_{i+2} - l_{i-1} l_{i+1}) + 16(l_i^2 k_{i+2} - l_{i+1}^2 k_i) \\
&\quad + 72k_{i+1}(l_i k_{i+2} - l_{i+1} k_i) + 28(l_{i+1}^2 l_{i+2} - l_{i-1} l_i^2) + 96k_{i+1}(l_{i+1} l_{i+2} - l_{i-1} l_i) \\
&\quad + 56(l_{i+1}^2 k_{i+2} - l_i^2 k_i) + 192k_{i+1}(l_{i+1} k_{i+2} - l_i k_i) + 12(l_{i+2} k_{i+2} - l_{i-1} k_i) \\
&\quad (3l_i + 3l_{i+1} + 8k_{i+1}) = 0\,.
\end{aligned} \tag{8.154}$$

However, the fact that monotonicity might be guaranteed for a larger class of fuzzy output partitions than those for which monotonicity was proved, is in practice of minor relevance, since a straightforward interpolation procedure allows the use of any fuzzy output partition for all five combinations of m, T and monotone (and smooth) rule base mentioned in Table 8.11, while guaranteeing a monotone input-output behaviour. In this interpolation procedure the crisp model output y'^* of a second model is mapped to a value y^* in the output domain of the model defined by the user. The second model applies the same fuzzy partitions as the user-defined model in the input domain(s). Instead of the user-defined fuzzy output partition of n membership functions, however, the second model uses a fuzzy output partition of $2n$+2 trapezial membership functions as illustrated in Fig. 8.25, satisfying the additional model properties needed to guarantee monotonicity for the applied number of input variables, t-norm and type of rule base

$$(\exists l > 0)(\forall i \in \{1, \ldots n\text{-}1\})(l_i = l)\,, \tag{8.155}$$

and, if required,

$$(\exists k > 0)(\forall i \in \{2, \ldots n\text{-}1\})(k_i = k)\,. \tag{8.156}$$

Note that k should be strictly positive, *i.e.* the identically shaped membership functions in the second fuzzy partition should not be triangular, in order to allow for an interpolation to all elements belonging to kernels of trapezial membership functions in the user-defined fuzzy partition. The user-defined fuzzy output partition is characterized by $(a_1, \ldots, a_n)$ and the second fuzzy output partition by $(a'_1, \ldots, a'_{2n+4})$. Furthermore, the rule base applied in the second model is slightly different from the user-defined rule base. The rule base of the second model is obtained by augmenting the indices of the output membership functions in the consequents of the user-defined rules by 1

$$i'_s = i_s + 1\,, \tag{8.157}$$

while keeping the antecedents of the rules unaltered.

The crisp model output y'^* obtained for the second model is never smaller than the midpoint of the kernel of the second membership function nor larger than the midpoint of the kernel of the next-to-last membership function of the second fuzzy output

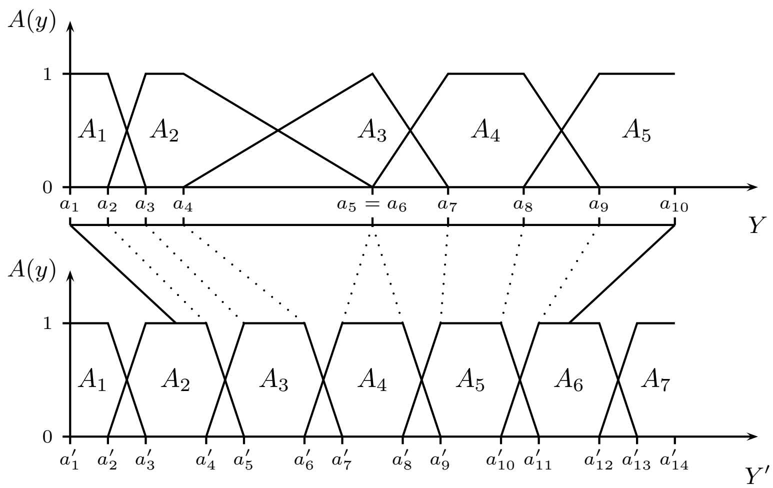


Figure 8.25: Interpolation procedure between the user defined fuzzy output partition (top) and the fuzzy output partition used in the second model (bottom).

partition, *i.e.*

$$\frac{1}{2}(a'_3 + a'_4) \leq y'^* \leq \frac{1}{2}(a'_{2n+1} + a'_{2n+2})\,. \tag{8.158}$$

The minimum and maximum values of y'^* are respectively mapped to the lower and upper bound of the output domain defined by the user. Crisp outputs y'^* belonging to the kernel of the second membership function of the second fuzzy output partition, are mapped to a value y^* belonging to the kernel of the first membership function of the user-defined fuzzy partition, explicitly

$$y^* = \frac{1}{2}(a_1 + a_2) + (a_2 - a_1)\frac{y'^* - \frac{1}{4}(a'_3 + 3a'_4)}{\frac{1}{2}(a'_4 - a'_3)}\,. \tag{8.159}$$

Crisp outputs y'^* belonging to the kernel of the next-to-last membership function of the second fuzzy output partition, are mapped to a value y^* belonging to the kernel of the last membership function of the user-defined fuzzy partition, explicitly

$$y^* = \frac{1}{2}(a_{2n\text{-}1} + a_{2n}) + (a_{2n} - a_{2n\text{-}1})\frac{y'^* - \frac{1}{4}(3a'_{2n+1} + a'_{2n+2})}{\frac{1}{2}(a'_{2n+2} - a'_{2n+1})}\,. \tag{8.160}$$

Intermediate values $y'^* \leq a'_{i_\text{right}}$ with $i_\text{left} = \max\{i \mid a'_i < y'^*\}$ and $i_\text{right} = \min\{i \mid a'_i \geq y'^*\}$, are mapped to a value y^* in the corresponding interval $\left[a_{i_\text{left}\text{-}2}, a_{i_\text{right}\text{-}2}\right]$ using the expression

$$y^* = \frac{1}{2}(a_{i_\text{left}\text{-}2} + a_{i_\text{right}\text{-}2}) + (a_{i_\text{right}\text{-}2} - a_{i_\text{left}\text{-}2})\frac{y'^* - \frac{1}{2}(a'_{i_\text{left}} + a'_{i_\text{right}})}{a'_{i_\text{right}} - a'_{i_\text{left}}}\,. \tag{8.161}$$

The three equations Eqs. (8.159–8.161), can be written in a more compact way. To map any y'^* to a value y^* in the user-defined output domain the following general expression can be used:

$$\begin{aligned} y^* = &\frac{1}{2}(a_{i_{\text{left}}\text{-}2} + a_{i_{\text{right}}\text{-}2}) + (a_{i_{\text{right}}\text{-}2} - a_{i_{\text{left}}\text{-}2})\times \\ &\frac{y'^* - \min(\max(\frac{1}{2}(a'_{i_{\text{left}}} + a'_{i_{\text{right}}}), \frac{1}{4}(a'_3 + 3a'_4)), \frac{1}{4}(3a'_{2n+1} + a'_{2n+2}))}{\min(a'_{i_{\text{right}}}, \frac{1}{2}(a'_{2n+1} + a'_{2n+2})) - \max(a'_{i_{\text{left}}}, \frac{1}{2}(a'_3 + a'_4))}\,. \end{aligned} \tag{8.162}$$

Monotonicity of models with more than three input variables was not investigated in this study, but the obtained results show that for models with more than three input variables only models should be considered with a monotone smooth rule base with membership functions of identical shape in the rule consequents applying $T_{\mathbf{P}}$.

CHAPTER 9

Mamdani–Assilian models: MOM defuzzification

Jamais je n'ai tant pensé, tant existé, tant vécu, tant été moi, si j'ose ainsi dire, que dans les voyages que j'ai faits seul et à pied.
(*Confessions*, Jean-Jacques Rousseau, 1782)

9.1 Introduction

In this chapter the monotonicity is investigated of Mamdani–Assilian models holding the properties described in Section 7.2 and applying the Mean of Maxima defuzzification method. It is verified for the three t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ whether a monotone input-output behaviour is obtained for any monotone rule base, or at least for any monotone smooth rule base.

First, in Section 9.2, the general definition of the crisp output y^*_{MOM} (Eq. (2.45)) is reformulated for models holding the properties described in Section 7.2, using the variables introduced in the same section to characterize the output membership functions. In Section 9.3 the monotonicity of models with a single input variable is studied for the t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$. As discussed in Section 7.4, obtaining the empty set as fuzzy output cannot be avoided when using the t-norm $T_{\mathbf{L}}$ in models with two or more input variables and holding the assumed properties, which makes $T_{\mathbf{L}}$ an inappropriate t-norm for these models. Therefore, Sections 9.4–9.5 deal with the monotonicity of models with two (or more) input variables for the t-norms $T_{\mathbf{M}}$ and $T_{\mathbf{P}}$ only. In Section 9.4 the monotonicity of models with a monotone smooth rule base and two or more input variables is discussed. In Section 9.5 it is shown that monotonicity cannot be guaranteed for models with two input variables and any monotone rule base when applying the t-norm $T_{\mathbf{M}}$, nor for models with two input variables and any monotone rule base using six or more linguistic output values when applying the t-norm $T_{\mathbf{P}}$. The chapter concludes with a summary of the obtained results in Section 9.6.

9.2 Tailoring the definition of y^*_{MOM}

In this section, the general definition of the crisp output y^*_{MOM} is reformulated to facilitate the investigation of the monotonicity of Mamdani–Assilian models holding the properties described in Section 7.2. The crisp output y^*_{MOM} only depends on the endpoints of the intervals forming the core of the fuzzy output A. As in Mamdani–Assilian models the membership degree of any output value y to the fuzzy output A is equal to the maximum membership degree obtained for the n adapted output membership functions A'_i, an output value y can only be an element of the core of the global fuzzy output A if it belongs to the core of at least one adapted output membership functions A'_i fired to the maximum fulfilment degree $\alpha_{\max}$

$$\text{core}(A) = \bigcup_{i \in I_{\max}} \text{core}(A'_i)\,, \tag{9.1}$$

with

$$\alpha_{\max} = \max_{i=1}^{n} \alpha_i = \beta_{\max} = \max_{s=1}^{r} \beta_s\,, \tag{9.2}$$

$$I_{\max} = \{i \in I \mid \alpha_i = \alpha_{\max}\}\,. \tag{9.3}$$

When applying the t-norm $T_{\mathbf{M}}$, the core of the adapted membership function coincides with the α-cut of the original membership function

$$(\forall \alpha \in [0,1])(\text{core}(T_{\mathbf{M}}(\alpha, A)) = A_\alpha)\,, \tag{9.4}$$

whereas, when applying the t-norms $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$, the core of the adapted membership function is nothing else but the kernel of the original membership function

$$(\forall \alpha \in \,]0,1])(\text{core}(T_{\mathbf{P}}(\alpha, A)) = \text{core}(T_{\mathbf{L}}(\alpha, A)) = \text{kern}(A))\,. \tag{9.5}$$

9.2.1 Linguistic output values fired to the maximum fulfilment degree

In the following paragraphs it is shown that if none of the components x_l of the input vector $\mathbf{x}$ has a membership degree 0.5 to a linguistic value of the variable X_l, only one rule is fired to the maximum fulfilment degree $\alpha_{\max}$ and, as a result, the core of the fuzzy output coincides with the core of one of the n adapted membership functions

$$(\forall l \in L)(\gamma_l \neq 0.5) \Rightarrow |I_{\max}| = 1\,. \tag{9.6}$$

Given the expressions for the maximum fulfilment degree $\beta_{\max,T_{\mathbf{M}},m}$ for models with m input variables applying $T_{\mathbf{M}}$ in Eqs. (7.71–7.76) and considering that for any $\gamma \in [0,1]$

$$\gamma \neq 0.5 \Leftrightarrow \max(1-\gamma, \gamma) > 0.5\,, \tag{9.7}$$

it follows that the maximum fulfilment degree $\beta_{\max,T_{\mathbf{M}},m}$ is greater than 0.5 if and only if none of the components x_l of the input vector $\mathbf{x}$ has a membership degree 0.5 to a linguistic value of the variable X_l

$$(l \in L)(\gamma_l \neq 0.5) \Leftrightarrow \beta_{\max,T_{\mathbf{M}},m} > 0.5\,. \tag{9.8}$$

Thus, if and only if none of the components x_l of the input vector $\mathbf{x}$ has a membership degree 0.5 to a linguistic value of the variable X_l there exists a rule $R_{s_{\max}}$ which is fired to a degree greater than 0.5, where the fulfilment degree is obtained using $T_{\mathbf{M}}$

$$(l \in L)(\gamma_l \neq 0.5) \Leftrightarrow (\exists s_{\max} \in S)(\min_{l=1}^{m} B^l_{s_{\max}}(x_l) > 0.5)\,, \tag{9.9}$$

or, in other words, there exists a rule $R_{s_{\max}}$ for which each component x_l of the input vector $\mathbf{x}$ has a membership degree greater than 0.5 to the corresponding linguistic value in its antecedent

$$(l \in L)(\gamma_l \neq 0.5) \Leftrightarrow (\exists s_{\max} \in S)((\forall l \in L)(B^l_{s_{\max}}(x_l) > 0.5))\,. \tag{9.10}$$

Note that the equivalence in Eq. (9.10) also holds for models applying a t-norm different from $T_{\mathbf{M}}$, in this models the index $s_{\max}$ however does not necessarily correspond to the index of a rule fired to the maximum fulfilment degree.

Since fuzzy partitions as described in Section 7.2 are used in all input domains and as the rule base is complete and consistent, it holds that if the membership degree to all linguistic values in the antecedent of the rule R_{s_1} are greater than 0.5, the other rules contain at least one linguistic value in their antecedent to which the input vector $\mathbf{x}$ has a membership degree smaller than 0.5, or, expressed mathematically, for any $s_1 \neq s_2$

$$\begin{aligned}&(\forall l \in L)(B^l_{s_1}(x_l) > 0.5)\\ \Rightarrow&(\forall l \in L)(B^l_{s_1}(x_l) \geq B^l_{s_2}(x_l)) \wedge (\exists l^* \in L)(0.5 > B^{l^*}_{s_2}(x_l))\,.\end{aligned} \tag{9.11}$$

Thus, there can only exist one rule to which all components x_l of the input vector $\mathbf{x}$ have a membership degree greater than 0.5 to the corresponding linguistic value in its antecedent

$$\begin{aligned}&(\exists s_{\max} \in S)((\forall l \in L)(B^l_{s_{\max}}(x_l) > 0.5))\\ \Rightarrow&(\exists! s_{\max} \in S)((\forall l \in L)(B^l_{s_{\max}}(x_l) > 0.5))\,,\end{aligned} \tag{9.12}$$

and this rule is furthermore the only rule fired to the maximum fulfilment degree since for any $s \neq s_{\max}$

$$((\forall l \in L)(B^l_{s_{\max}}(x_l) \geq B^l_s(x_l)) \wedge (\exists l^* \in L)(B^{l^*}_{s_{\max}}(x_l) > 0.5 > B^{l^*}_s(x_l))) \tag{9.13}$$

$$\Rightarrow (\forall T \in \{T_{\mathbf{M}}, T_{\mathbf{P}}\})(\mathop{T}_{l=1}^{m} B^l_{s_{\max}}(x_l) > \mathop{T}_{l=1}^{m} B^l_{s_{\max}}(x_l)) \tag{9.14}$$

$$\Rightarrow (\forall T \in \{T_{\mathbf{M}}, T_{\mathbf{P}}\})(\beta_{s_{\max}} > \beta_s)\,. \tag{9.15}$$

When only one rule is fired to the maximum fulfilment degree, only one linguistic output value can be fired to the maximum fulfilment degree, or

$$|I_{\max}| = 1 \tag{9.16}$$

In models with a single input variable the fulfilment degrees are equal to the membership degrees to the linguistic values of the input variable and no t-norm is applied to determine them. Therefore, Eq. (9.6) is also satisfied for models with a single input variable applying $T_{\mathbf{L}}$, since it was shown above that Eq. (9.6) is satisfied for models with one or more input variables applying $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$.

Note that $I_{\max}$ is always a singleton if all γ_l differ from 0.5, but might also be a singleton if not all γ_l differ from 0.5 when all rules fired to the maximum fulfilment degree contain the same linguistic output value in their consequent. By contraposition, it then follows that if two or more linguistic output values are fired to the maximum fulfilment degree, at least one γ_l is equal to 0.5

$$|I_{\max}| \geq 2 \Rightarrow (\exists l \in L)(\gamma_l = 0.5)\,. \tag{9.17}$$

9.2.2 Generally applicable expressions for y^*_{MOM}

For models applying $T_{\mathbf{M}}$ the maximum fulfilment degree $\alpha_{\max}$ ($= \beta_{\max,T_{\mathbf{M}},m}$) (Eqs. (7.71–7.76)) is given by

$$\alpha_{\max} = \min_{l=1}^{m} \max(1-\gamma_l, \gamma_l) \geq 0.5\,. \tag{9.18}$$

By formulating Eq. (9.17) slightly differently,

$$|I_{\max}| \geq 2 \Rightarrow (\exists l \in L)(\max(1-\gamma_l,\gamma_l) = 0.5)\,, \tag{9.19}$$

one can easily see that in models applying $T_{\mathbf{M}}$ the maximum fulfilment degree $\alpha_{\max}$ is always equal to 0.5 if two or more linguistic output values are fired to the maximum fulfilment degree, *i.e.*

$$(|I_{\max}| \geq 2 \wedge T = T_{\mathbf{M}}) \Rightarrow\ \alpha_{\max} = 0.5\,. \tag{9.20}$$

For $T = T_{\mathbf{M}}$ the core of an adapted membership function coincides with an α-cut of the original membership function (Eq. (9.4)). Given the properties of the output membership functions (Section 7.2) and the fact that $\alpha_{\max}$ is equal to 0.5 for $|I_{\max}| \geq 2$ and greater than 0.5 for $|I_{\max}| = 1$, the cores of two adapted output membership functions fired to the maximum fulfilment degree $\alpha_{\max}$ share at most a boundary point. When applying the t-norms $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$, the core of the adapted membership function coincides with the kernel of the original membership function (Eq. (9.5)). As the output membership functions form a fuzzy partition, the intersection of the cores of the two adapted output membership functions fired to the maximum fulfilment degree $\alpha_{\max}$ is always empty. Summarizing, it holds for any $i, j \in I_{\max}$, $i < j$, that

$$\begin{aligned} &\mathrm{core}(T(\alpha_{\max}, A_i)) \cap \mathrm{core}(T(\alpha_{\max}, A_j)) \\ &= \begin{cases} \{\frac{1}{2}(a_{2i} + a_{2i+1})\} & \text{, if } T = T_{\mathbf{M}} \text{ and } j = i+1\,, \\ \emptyset & \text{, otherwise}\,. \end{cases} \end{aligned} \tag{9.21}$$

For models applying $T_{\mathbf{M}}$, the crisp output y^*_{MOM} is a function of the maximum fulfilment degree $\alpha_{\max}$ if only one linguistic output value is fired to the degree $\alpha_{\max}$. Explicitly, if $I_{\max} = \{i_{\max}\}$, then

$$y^*_{\mathrm{MOM}} = c_{i_{\max}} + \frac{1}{2}(l_{i_{\max}} - l_{i_{\max}\text{-}1})(1 - \alpha_{\max})\,. \tag{9.22}$$

If more than one linguistic output value is fired to this maximum fulfilment degree $\alpha_{\max}$, *i.e.* if $|I_{\max}| \geq 2$, then $\alpha_{\max} = 0.5$ and

$$y^*_{\mathrm{MOM}} = \frac{\sum\limits_{i \in I_{\max}} \left((2a_{2i} + l_i)^2 - (2a_{2i\text{-}1} - l_{i\text{-}1})^2\right)}{4 \sum\limits_{i \in I_{\max}} (l_{i\text{-}1} + l_i + 2k_i)}\,. \tag{9.23}$$

For models applying $T_{\mathbf{P}}$, the expression of y^*_{MOM} can be reformulated as

$$y^*_{\mathrm{MOM}} = \frac{\sum\limits_{i \in I_{\max}} w_i c_i}{\sum\limits_{i \in I_{\max}} w_i}\,, \tag{9.24}$$

with

$$w_i = \begin{cases} k_i & , \text{if } \sum\limits_{j \in I_{\max}} k_j > 0\,, \\ 1 & , \text{if } \sum\limits_{j \in I_{\max}} k_j = 0\,. \end{cases}$$

As the crisp output y^*_{MOM} only depends on the endpoints of the intervals forming the core of the fuzzy output A, which is the union of the cores of the adapted membership functions fired to the maximum fulfilment degree $\alpha_{\max}$, and as for a given membership function A the core of the adapted membership function obtained with $T_{\mathbf{P}}$ is equal to the core of the adapted membership function obtained with $T_{\mathbf{L}}$, for models with a single input variable applying $T_{\mathbf{L}}$ as t-norm, Eq. (9.24) is also applied to calculate the crisp output y^*_{MOM}.

9.2.3 Expressions for y^*_{MOM} valid in special cases for models applying $T_{\mathbf{M}}$

First, if all linguistic output values between the smallest linguistic output value $A_{i_{\text{left}}}$ ($i_{\text{left}} = \min I_{\max}$) and the largest linguistic output value $A_{i_{\text{right}}}$ ($i_{\text{right}} = \max I_{\max}$) fired to the degree $\alpha_{\max}$ are also fired to this degree, *i.e.*

$$(\forall i_1, i_2 \in I_{\max})(\forall j \in I)(i_1 < j < i_2 \Rightarrow j \in I_{\max})\,, \tag{9.25}$$

Eq. (9.23) can be further simplified. In this case for models applying $T_{\mathbf{M}}$ and $|I_{\max}| \geq 2$, y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = \frac{1}{4}(2a_{2i_{\text{left}}\text{-}1} - l_{i_{\text{left}}\text{-}1} + 2a_{2i_{\text{right}}} + l_{i_{\text{right}}})\,. \tag{9.26}$$

Second, if $|I_{\max}| \geq 2$ but none of the extreme linguistic values A_1 and A_n is fired to the degree $\alpha_{\max}$, *i.e.* $I_{\max} \cap \{1, n\} = \emptyset$, and if the intervals where the membership functions overlap are of equal length, *i.e.* $(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)$, then y^*_{MOM} can be computed using Eq. (9.24), but with

$$w_i = l + k_i \,. \tag{9.27}$$

Third, when combining the above two cases, Eq. (9.26) can be further simplified to

$$y^*_{\mathrm{MOM}} = \frac{1}{2}(a_{2i_{\mathrm{left}}\text{-}1} + a_{2i_{\mathrm{right}}}) \,. \tag{9.28}$$

9.3 Models with a single input variable

In a model with a single input variable at most two rules are fired: the rule corresponding to some linguistic input value B^1_j is fired to a degree $(1 - \gamma_1)$ and the rule corresponding to the linguistic value B^1_{j+1} to a degree γ_1. In case of a monotone rule base, B^1_j and B^1_{j+1} can either be mapped to

1. the same linguistic output value A_i: the *constant* case,
2. two consecutive output values A_i and A_{i+1}: the *smooth* case, or
3. two non-consecutive output values A_i and A_{i+p} ($p \in \mathbb{N}$, $p > 1$, $i + p \leq n$): the *non-smooth* case.

As discussed in Section 8.3, considering the *constant* case for a model with a single input variable might seem in disaccord with the aim to safeguard the model interpretability, but is nevertheless meaningful as interpretable models with more than one input variable might behave as a model with a single input variable in the *constant* case in some parts of the input space.

9.3.1 Models applying $T_{\mathbf{M}}$

When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are both mapped to a same linguistic output value A_i (Fig. 8.1), the crisp output y^*_{MOM} is computed with Eq. (9.22). Since $\alpha_{\max}$ is equal to $(1 - \gamma_1)$ for $\gamma_1 \in [0, 0.5]$ and equal to γ_1 for $\gamma_1 \in [0.5, 1]$, monotonicity holds if

$$\Bigl(\forall \gamma_1 \in [0, 0.5]\Bigr)\left(\frac{dy^*_{\mathrm{MOM},\mathbf{1M},11}}{d\gamma_1} \geq 0\right) \wedge \Bigl(\forall \gamma_1 \in [0.5, 1]\Bigr)\left(\frac{dy^*_{\mathrm{MOM},\mathbf{1M},12}}{d\gamma_1} \geq 0\right), \tag{9.29}$$

with

with

$$y^*_{\mathrm{MOM},\mathbf{1M},11} = c_i + \frac{1}{2}(l_i - l_{i\text{-}1})\gamma_1 \,, \tag{9.30}$$

$$y^*_{\mathrm{MOM},\mathbf{1M},12} = c_i + \frac{1}{2}(l_i - l_{i\text{-}1})(1 - \gamma_1) \,. \tag{9.31}$$

One easily verifies that $y^*_{\mathrm{MOM},\mathbf{1M},11}(\gamma_1 = 0.5) = y^*_{\mathrm{MOM},\mathbf{1M},12}(\gamma_1 = 0.5)$ and that Eq. (9.29) is satisfied if and only if

$$l_{i\text{-}1} = l_i \,. \tag{9.32}$$

As the extreme linguistic output values A_1 and A_n are both described by a trapezium with one vertical side, monotonicity can only be guaranteed for a model with a single input variable applying $T_{\mathbf{M}}$ if the following constraints are satisfied

$$(\forall s \in \{1, \ldots, r\})(i_s \notin \{1, n\}) \,, \tag{9.33}$$

$$(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l) \,. \tag{9.34}$$

From here on, Eqs. (9.33–9.34) are assumed to hold. When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are mapped to two consecutive output values A_i and A_{i+1}, Eq. (9.28) is used to compute the crisp output y^*_{MOM}. The crisp output coincides with the midpoint of the interval with as lower bound, the lower bound of the kernel of the smallest linguistic output value $A_{i_{\mathrm{left}}}$ fired to the maximum fulfilment degree $\alpha_{\max}$, and as upper bound, the upper bound of the kernel of the largest linguistic output value $A_{i_{\mathrm{right}}}$ fired to the maximum fulfilment degree $\alpha_{\max}$. The set $I_{\max}$ and the corresponding indices i_{left} and i_{right} are given by

1. if $\gamma_1 \in \,]0, 0.5[$, then $I_{\max} = \{i\}$, hence $i_{\mathrm{left}} = i$ and $i_{\mathrm{right}} = i$,
2. if $\gamma_1 = 0.5$, then $I_{\max} = \{i, i{+}1\}$, hence $i_{\mathrm{left}} = i$ and $i_{\mathrm{right}} = i{+}1$,
3. if $\gamma_1 \in \,]0.5, 1[$, then $I_{\max} = \{i{+}1\}$, hence $i_{\mathrm{left}} = i{+}1$ and $i_{\mathrm{right}} = i{+}1$.

The desired monotonicity trivially holds since

$$\text{midpoint}([a_{2i\text{-}1}, a_{2i}]) \leq \text{midpoint}([a_{2i\text{-}1}, a_{2i+2}]) \leq \text{midpoint}([a_{2i+1}, a_{2i+2}]) \,, \tag{9.35}$$

and it is know from interval calculus that

$$(lb_1 \leq lb_2) \wedge (ub_1 \leq ub_2) \Rightarrow \text{midpoint}([lb_1, ub_1]) \leq \text{midpoint}([lb_2, ub_2]) \,. \tag{9.36}$$

When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are mapped to two non-consecutive output values A_i and A_{i+p} ($p \in \mathbb{N}$, $p > 1$, $i + p \leq n$), monotonicity holds if

$$y^*_{\mathrm{MOM},\mathbf{1M},31} \leq y^*_{\mathrm{MOM},\mathbf{1M},32} \leq y^*_{\mathrm{MOM},\mathbf{1M},33} \,, \tag{9.37}$$

with

$$y^*_{\mathrm{MOM},\mathbf{1M},31} = c_i \,, \tag{9.38}$$

$$y^*_{\mathrm{MOM},\mathbf{1M},32} = \frac{(l + k_i)c_i + (l + k_{i+p})c_{i+p}}{2l + k_i + k_{i+p}} \,, \tag{9.39}$$

$$y^*_{\mathrm{MOM},\mathbf{1M},33} = c_{i+p} \,. \tag{9.40}$$

The above chain of inequalities always holds since

$$y^*_{\mathrm{MOM},\mathbf{1M},31} = y^*_{\mathrm{MOM},\mathbf{1M},32} - \frac{(l+k_i)(c_{i+p}-c_i)}{2l+k_i+k_{i+p}} \qquad < y^*_{\mathrm{MOM},\mathbf{1M},32}\,, \tag{9.41}$$

$$y^*_{\mathrm{MOM},\mathbf{1M},33} = y^*_{\mathrm{MOM},\mathbf{1M},32} + \frac{(l+k_{i+p})(c_{i+p}-c_i)}{2l+k_i+k_{i+p}} \qquad > y^*_{\mathrm{MOM},\mathbf{1M},32}\,, \tag{9.42}$$

and thus monotonicity is guaranteed.

From this section, it can be concluded that models with a single input variable applying the t-norm $T_{\mathbf{M}}$ show a monotone input-output behaviour for any monotone rule base when the linguistic output values in the consequents of the rules are defined by trapezial or triangular membership functions with intervals of changing membership degree of equal length.

9.3.2 Models applying $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$

In models applying $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$ the crisp model output y^*_{MOM} depends on which linguistic output values are fired to the maximum fulfilment degree $\alpha_{\max}$, but does not depend on the value as such, of the maximum fulfilment degree as shown by Eq. (9.24). When B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are both mapped to a same linguistic output value A_i, y^*_{MOM} is constant and equal to c_i for all $\gamma_1 \in [0,1]$. The *smooth* and *non-smooth* cases, when B^1_j and B^1_{j+1} ($j \in J_1 \setminus \{n_1\}$) are mapped to different output values A_i and A_{i+p} ($p \in \mathbb{N}_0$, $i+p \leq n$), can be considered simultaneously. In these cases monotonicity holds if

$$y^*_{\mathrm{MOM},\mathbf{1P},31} \leq y^*_{\mathrm{MOM},\mathbf{1P},32} \leq y^*_{\mathrm{MOM},\mathbf{1P},33}\,, \tag{9.43}$$

with

$$y^*_{\mathrm{MOM},\mathbf{1P},31} = c_i\,, \tag{9.44}$$

$$y^*_{\mathrm{MOM},\mathbf{1P},32} = \begin{cases} \frac{k_i c_i + k_{i+p} c_{i+p}}{k_i + k_{i+p}} & \text{, if } k_i > 0 \text{ and } k_{i+p} > 0\,,\\ c_{i+p} & \text{, if } k_i = 0 \text{ and } k_{i+p} > 0\,,\\ c_i & \text{, if } k_i > 0 \text{ and } k_{i+p} = 0\,,\\ \frac{1}{2}(c_i + c_{i+p}) & \text{, if } k_i = 0 \text{ and } k_{i+p} = 0\,, \end{cases} \tag{9.45}$$

$$y^*_{\mathrm{MOM},\mathbf{1P},33} = c_{i+p}\,, \tag{9.46}$$

which is always satisfied.

From this section, it can be concluded that models with a single input variable applying the t-norm $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$ show a monotone input-output behaviour for any monotone rule base.

9.4 Models with a monotone smooth rule base and two or more input variables

In this section it is shown that for models with two or more input variables and a monotone smooth rule base monotonicity is guaranteed when applying $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$.

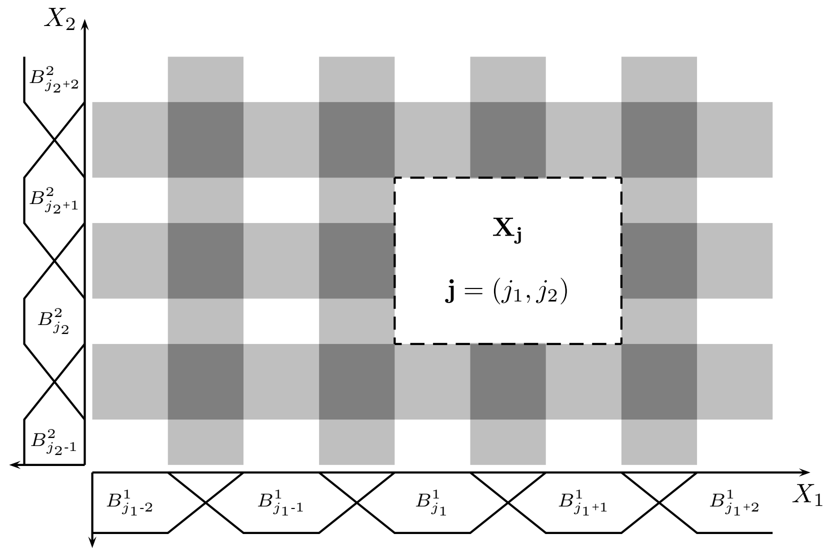


Figure 9.1: Example of a subspace $\mathbf{X_j}$ for a model with two input variables

Models with more than one input variable applying $T_{\mathbf{L}}$ are not considered since they return the empty set as fuzzy model output for some input vectors as discussed in Section 7.4. The results obtained for models with a single input variable also apply to models with two input variables, as the latter behave as a 'single input model' in parts of their input space. Therefore, for models applying $T_{\mathbf{M}}$ the output membership functions used in the consequents of the rules are assumed to have intervals of changing membership degree of equal length (Eqs. (9.33–9.34)). For models applying $T_{\mathbf{P}}$ no additional model properties were required to guarantee the monotonicity of models with a single input variable.

The input space of a model with m input variables can be seen as the union of several m-dimensional subspaces whose projections onto the m one-dimensional input domains coincide with the interval bounded by the lower bound of the kernel of a linguistic value $B^l_{j_l}$ and the upper bound of the kernel of the linguistic value $B^l_{j_l+1}$ ($j_l \in J_l \setminus \{n_l\}$), as illustrated in Fig. 9.1. In such a subspace $\mathbf{X_j}$ all input vectors $\mathbf{x}$ have a non-zero membership degree to at least one of the two linguistic values $B^l_{j_l}$ and $B^l_{j_l+1}$

$$\mathbf{X_j} = \{\mathbf{x} \mid (\forall l \in L)(B^l_{j_l}(x_l) = 1 - B^l_{j_l+1}(x_l))\}\,, \tag{9.47}$$

with $\mathbf{j} = (j_1, \ldots, j_m) \in (J_1 \setminus \{n_1\}) \times \ldots \times (J_m \setminus \{n_m\})$. An input vector $\mathbf{x}$ with an input value x_l belonging to the kernel of some $B^l_{j_l}$ always belongs to two or more 'adjacent' subspaces. This observation allows us to extend the results, shown in Sections 9.4.1–9.4.2 for a single subspace, to the whole input space.

For any input vector $\mathbf{x} \in \mathbf{X_j}$ three groups of input domains can be distinguished:

those for which x_l is greater than or equal to the lower bound of the kernel of $B^l_{j_l}$ and smaller than the crisp value corresponding to the intersection of the membership functions of $B^l_{j_l}$ and $B^l_{j_l+1}$, *i.e.*

$$L_1(\mathbf{x}) = \{l \in L \mid B^l_{j_l}(x_l) > 0.5\}\,, \tag{9.48}$$

those for which x_l corresponds to the intersection of the membership functions $B^l_{j_l}$ and $B^l_{j_l+1}$, *i.e.*

$$L_2(\mathbf{x}) = \{l \in L \mid B^l_{j_l}(x_l) = 0.5\}\,, \tag{9.49}$$

and finally, those for which x_l is larger than the crisp value corresponding to the intersection of the membership functions $B^l_{j_l}$ and $B^l_{j_l+1}$ and smaller than or equal to the upper bound of the kernel of $B^l_{j_l+1}$, *i.e.*

$$L_3(\mathbf{x}) = \{l \in L \mid B^l_{j_l}(x_l) < 0.5\}\,. \tag{9.50}$$

9.4.1 Models applying $T_\mathbf{M}$

When none of the input values x_l of an input vector $\mathbf{x} \in \mathbf{X_j}$ coincides with the intersection of the membership functions $B^l_{j_l}$ and $B^l_{j_l+1}$, the set $L_2(\mathbf{x})$ is empty and only one rule R_s is fired to the maximum fulfilment degree $\alpha_{\max}$, with

$$j_{l,s} = \begin{cases} j_l & \text{, if } l \in L_1(\mathbf{x})\,, \\ j_l + 1 & \text{, if } l \in L_3(\mathbf{x})\,. \end{cases} \tag{9.51}$$

Furthermore, for these input vectors, $\alpha_{\max}$ is larger than 0.5

$$\beta_s = \min\left(\min_{l \in L_1(\mathbf{x})} B^l_{j_l}(x_l), \min_{l \in L_3(\mathbf{x})} B^l_{j_l+1}(x_l)\right) > 0.5\,. \tag{9.52}$$

For an input vector $\mathbf{x} \in \mathbf{X_j}$ with a single input value x_{l_1} coinciding with the intersection of the corresponding membership functions $B^{l_1}_{j_{l_1}}$ and $B^{l_1}_{j_{l_1}+1}$, two rules R_{s_1} and R_{s_2} are fired to the degree $\alpha_{\max}$, with

$$j_{l,s_1} = \begin{cases} j_l & \text{, if } l \in L_1(\mathbf{x}) \cup \{l_1\}\,, \\ j_l + 1 & \text{, if } l \in L_3(\mathbf{x})\,, \end{cases} \tag{9.53}$$

$$j_{l,s_2} = \begin{cases} j_l & \text{, if } l \in L_1(\mathbf{x})\;\;, \\ j_l + 1 & \text{, if } l \in L_3(\mathbf{x}) \cup \{l_1\}\,, \end{cases} \tag{9.54}$$

and $\alpha_{\max}$ is equal to 0.5 since

$$\beta_{s_1} = \min(\underbrace{\min(\min_{l \in L_1(\mathbf{x})} B^l_{j_l}(x_l), \min_{l \in L_3(\mathbf{x})} B^l_{j_l+1}(x_l))}_{>0.5}, \underbrace{B^{l_1}_{j_{l_1}}(x_{l_1})}_{=0.5}) = 0.5\,, \tag{9.55}$$

$$\beta_{s_2} = \min(\overbrace{\min(\min_{l \in L_1(\mathbf{x})} B^l_{j_l}(x_l), \min_{l \in L_3(\mathbf{x})} B^l_{j_l+1}(x_l))}^{>0.5}, \overbrace{B^{l_1}_{j_{l_1}+1}(x_{l_1})}^{=0.5}) = 0.5\,. \tag{9.56}$$

In general, for an input vector $\mathbf{x} \in \mathbf{X_j}$, $2^{|L_2(\mathbf{x})|}$ rules R_s are fired to the maximum fulfilment degree $\alpha_{\max}$, with

$$j_{l,s} = \begin{cases} j_l & \text{, if } l \in L_1(\mathbf{x}) \,, \\ \in \{j_l, j_l+1\} & \text{, if } l \in L_2(\mathbf{x}) \,, \\ j_l + 1 & \text{, if } l \in L_3(\mathbf{x}) \,. \end{cases} \tag{9.57}$$

To investigate the monotonicity of a model in a variable X_{l_1}, two input vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ are considered, such that $x_{1,l} = x_{2,l}$ for any $l \in L \setminus \{l_1\}$ and $x_{1,l_1} < x_{2,l_1}$, with

$$L_{1\setminus l_1}(\mathbf{x}_1) = \{l \in L \setminus \{l_1\} \mid B^l_{j_l}(x_{1,l}) = B^l_{j_l}(x_{2,l}) > 0.5\} \,, \tag{9.58}$$

$$L_{2\setminus l_1}(\mathbf{x}_1) = \{l \in L \setminus \{l_1\} \mid B^l_{j_l}(x_{1,l}) = B^l_{j_l}(x_{2,l}) = 0.5\} \,, \tag{9.59}$$

$$L_{3\setminus l_1}(\mathbf{x}_1) = \{l \in L \setminus \{l_1\} \mid B^l_{j_l}(x_{1,l}) = B^l_{j_l}(x_{2,l}) < 0.5\} \,. \tag{9.60}$$

Case a If $B^{l_1}_{j_{l_1}}(x_{1,l_1}) > 0.5$ and $B^{l_1}_{j_{l_1}}(x_{2,l_1}) > 0.5$, the same rules are fired to the maximum fulfilment degree $\alpha_{\max}(\mathbf{x}_1)$ for $\mathbf{x}_1$ and to the maximum fulfilment degree $\alpha_{\max}(\mathbf{x}_2)$ for $\mathbf{x}_2$, since

$$L_1(\mathbf{x}_1) = L_1(\mathbf{x}_2) = L_{1\setminus l_1}(\mathbf{x}_1) \cup \{l_1\} \,, \tag{9.61}$$

$$L_2(\mathbf{x}_1) = L_2(\mathbf{x}_2) = L_{2\setminus l_1}(\mathbf{x}_1) \,, \tag{9.62}$$

$$L_3(\mathbf{x}_1) = L_3(\mathbf{x}_2) = L_{3\setminus l_1}(\mathbf{x}_1) \,, \tag{9.63}$$

thus

$$y^*_{\mathrm{MOM}}(\mathbf{x}_1) = y^*_{\mathrm{MOM}}(\mathbf{x}_2) \,, \tag{9.64}$$

and monotonicity is guaranteed.

Case b If $B^{l_1}_{j_{l_1}}(x_{1,l_1}) > 0.5$ and $B^{l_1}_{j_{l_1}}(x_{2,l_1}) = 0.5$, the set $\mathcal{R}_1$ of rules R_{s_1} fired to the maximum fulfilment degree $\alpha_{\max}(\mathbf{x}_1)$ for $\mathbf{x}_1$, with

$$j_{l,s_1} = \begin{cases} j_l & \text{, if } l \in L_{1\setminus l_1}(\mathbf{x}_1) \cup \{l_1\} \,, \\ \in \{j_l, j_l+1\} & \text{, if } l \in L_{2\setminus l_1}(\mathbf{x}_1) \,, \\ j_l + 1 & \text{, if } l \in L_{3\setminus l_1}(\mathbf{x}_1) \,, \end{cases} \tag{9.65}$$

is a subset of the set $\mathcal{R}_2$ of rules R_{s_2} fired to the maximum fulfilment degree $\alpha_{\max}(\mathbf{x}_2)$ for $\mathbf{x}_2$, with

$$j_{l,s_2} = \begin{cases} j_l & \text{, if } l \in L_{1\setminus l_1}(\mathbf{x}_1) \,, \\ \in \{j_l, j_l+1\} & \text{, if } l \in L_{2\setminus l_1}(\mathbf{x}_1) \cup \{l_1\} \,, \\ j_l + 1 & \text{, if } l \in L_{3\setminus l_1}(\mathbf{x}_1) \,. \end{cases} \tag{9.66}$$

Let $\mathcal{R}_3 = \mathcal{R}_2 \setminus \mathcal{R}_1$, then the indices in the antecedent of a rule $R_{s_3} \in \mathcal{R}_3$ are given by

$$j_{l,s_3} = \begin{cases} j_l & \text{, if } l \in L_{1\setminus l_1}(\mathbf{x}_1) \,, \\ \in \{j_l, j_l+1\} & \text{, if } l \in L_{2\setminus l_1}(\mathbf{x}_1) \,, \\ j_l + 1 & \text{, if } l \in L_{3\setminus l_1}(\mathbf{x}_1) \cup \{l_1\} \,, \end{cases} \tag{9.67}$$

and there exists a bijection between $\mathcal{R}_1$ and $\mathcal{R}_3$ as for any rule in $\mathcal{R}_1$ containing the linguistic value $B^{l_1}_{j_{l_1}}$ in its antecedent, there exists a rule in $\mathcal{R}_3$ containing the same linguistic values for all input variables different from X_{l_1} and the linguistic value $B^{l_1}_{j_{l_1}+1}$ in its antecedent and, as the rule base is monotone and smooth (Corollary 7.1), the same or next linguistic output value in its consequent. Thus, with S_1 (resp. S_3) the set of indices of the rules in $\mathcal{R}_1$ (resp. $\mathcal{R}_3$), it follows that

$$\min_{s\in S_1} i_s \leq \min_{s\in S_3} i_s \,, \tag{9.68}$$

$$\max_{s\in S_1} i_s \leq \max_{s\in S_3} i_s \,, \tag{9.69}$$

and as the set S_2 of indices of the rules in $\mathcal{R}_2$ is the union of the sets S_1 and S_3 ($S_2 = S_1 \cup S_3$), it follows that

$$\min_{s\in S_1} i_s = \min_{s\in S_2} i_s \leq \min_{s\in S_3} i_s \,, \tag{9.70}$$

$$\max_{s\in S_1} i_s \leq \max_{s\in S_2} i_s = \max_{s\in S_3} i_s \,. \tag{9.71}$$

Since the rule base is assumed to be monotone and smooth, it follows from the above reasoning that all linguistic values between $A_{i_{\text{left}}}$ and $A_{i_{\text{right}}}$ are fired to the maximum fulfilment degree. Moreover, as the output membership functions are assumed to satisfy (Eqs. (9.33–9.34)), Eq. (9.28) may be used to calculate the crisp outputs $y^*_{\text{MOM}}(\mathbf{x}_1)$ and $y^*_{\text{MOM}}(\mathbf{x}_2)$, given by

$$y^*_{\text{MOM}}(\mathbf{x}_1) = \frac{1}{2}(a_{2i_{\text{left}}(\mathbf{x}_1)\text{-}1} + a_{2i_{\text{right}}(\mathbf{x}_1)}) \tag{9.72}$$

$$\text{with} \qquad i_{\text{left}}(\mathbf{x}_1) = \min_{s\in S_1} i_s \qquad i_{\text{right}}(\mathbf{x}_1) = \max_{s\in S_1} i_s \,,$$

$$y^*_{\text{MOM}}(\mathbf{x}_2) = \frac{1}{2}(a_{2i_{\text{left}}(\mathbf{x}_2)\text{-}1} + a_{2i_{\text{right}}(\mathbf{x}_2)}) \tag{9.73}$$

$$\text{with} \qquad i_{\text{left}}(\mathbf{x}_2) = \min_{s\in S_2} i_s \qquad i_{\text{right}}(\mathbf{x}_2) = \max_{s\in S_2} i_s \,.$$

Obviously, $y^*_{\text{MOM}}(\mathbf{x}_1) \leq y^*_{\text{MOM}}(\mathbf{x}_2)$ and monotonicity is guaranteed.

Case c If $B^{l_1}_{j_{l_1}}(x_{1,l_1}) > 0.5$ and $B^{l_1}_{j_{l_1}}(x_{2,l_1}) < 0.5$, all rules R_{s_1} belonging to the set $\mathcal{R}_1$ defined by Eq. (9.65) are fired to the degree $\alpha_{\max}(\mathbf{x}_1)$ and all rules R_{s_3} belonging to the set $\mathcal{R}_3$ defined by Eq. (9.67) are fired to the degree $\alpha_{\max}(\mathbf{x}_2)$. Given Eqs. (9.70–9.71), the crisp outputs $y^*_{\text{MOM}}(\mathbf{x}_1)$ and $y^*_{\text{MOM}}(\mathbf{x}_2)$ are obtained by substituting $i_{\text{left}}(\mathbf{x}_1)$, $i_{\text{right}}(\mathbf{x}_1)$, $i_{\text{left}}(\mathbf{x}_2)$ and $i_{\text{right}}(\mathbf{x}_2)$ in Eqs. (9.72–9.73) by

$$i_{\text{left}}(\mathbf{x}_1) = \min_{s\in S_1} i_s \qquad i_{\text{right}}(\mathbf{x}_1) = \max_{s\in S_1} i_s \,, \tag{9.74}$$

$$i_{\text{left}}(\mathbf{x}_2) = \min_{s\in S_3} i_s \qquad i_{\text{right}}(\mathbf{x}_2) = \max_{s\in S_3} i_s \,. \tag{9.75}$$

Thus, $y^*_{\text{MOM}}(\mathbf{x}_1) \leq y^*_{\text{MOM}}(\mathbf{x}_2)$ and monotonicity is guaranteed.

Case d If $B^{l_1}_{j_{l_1}}(x_{1,l_1}) = 0.5$ and $B^{l_1}_{j_{l_1}}(x_{2,l_1}) < 0.5$, all rules R_{s_2} belonging to the set $\mathcal{R}_2$ defined by Eq. (9.66) are fired to the degree $\alpha_{\max}(\mathbf{x}_1)$ and all rules R_{s_3} belonging to the set $\mathcal{R}_3$ defined by Eq. (9.67) are fired to the degree $\alpha_{\max}(\mathbf{x}_2)$. Given Eqs. (9.70–9.71), the crisp outputs $y^*_{\mathrm{MOM}}(\mathbf{x}_1)$ and $y^*_{\mathrm{MOM}}(\mathbf{x}_2)$ are obtained by substituting $i_{\mathrm{left}}(\mathbf{x}_1)$, $i_{\mathrm{right}}(\mathbf{x}_1)$, $i_{\mathrm{left}}(\mathbf{x}_2)$ and $i_{\mathrm{right}}(\mathbf{x}_2)$ in Eqs. (9.72–9.73) by

$$i_{\mathrm{left}}(\mathbf{x}_1) = \min_{s\in S_2} i_s \qquad i_{\mathrm{right}}(\mathbf{x}_1) = \max_{s\in S_2} i_s \tag{9.76}$$

$$i_{\mathrm{left}}(\mathbf{x}_2) = \min_{s\in S_3} i_s \qquad i_{\mathrm{right}}(\mathbf{x}_2) = \max_{s\in S_3} i_s\,. \tag{9.77}$$

Thus, $y^*_{\mathrm{MOM}}(\mathbf{x}_1) \le y^*_{\mathrm{MOM}}(\mathbf{x}_2)$ and monotonicity is guaranteed.

Case e If $B^{l_1}_{j_{l_1}}(x_{1,l_1}) < 0.5$ and $B^{l_1}_{j_{l_1}}(x_{2,l_1}) < 0.5$, the same set $\mathcal{R}_3$ (Eq. (9.67)) of rules is fired to the degree $\alpha_{\max}(\mathbf{x}_1)$ and the degree $\alpha_{\max}(\mathbf{x}_2)$, since

$$L_1(\mathbf{x}_1) = L_1(\mathbf{x}_2) = L_{1\setminus l_1}(\mathbf{x}_1)\,, \tag{9.78}$$

$$L_2(\mathbf{x}_1) = L_2(\mathbf{x}_2) = L_{2\setminus l_1}(\mathbf{x}_1)\,, \tag{9.79}$$

$$L_3(\mathbf{x}_1) = L_3(\mathbf{x}_2) = L_{3\setminus l_1}(\mathbf{x}_1) \cup \{l_1\}\,, \tag{9.80}$$

thus $y^*_{\mathrm{MOM}}(\mathbf{x}_1) = y^*_{\mathrm{MOM}}(\mathbf{x}_2)$ and monotonicity is guaranteed.

Hence, a monotone input-output behaviour is obtained for each subspace $\mathbf{X_j}$, and, by construction, for the whole input space of a Mamdani–Assilian model with a smooth rule base, output membership functions satisfying Eqs. (9.33–9.34) and applying $T_{\mathbf{M}}$.

9.4.2 Models applying $T_{\mathbf{P}}$

When none of the input values x_l of an input vector $\mathbf{x} \in \mathbf{X_j}$ coincides with the intersection of the membership functions $B^l_{j_l}$ and $B^l_{j_l+1}$, the set $L_2(\mathbf{x})$ is empty and only one rule R_s is fired to the degree $\alpha_{\max}$, with

$$j_{l,s} = \begin{cases} j_l & \text{, if } l \in L_1(\mathbf{x})\,, \\ j_l + 1 & \text{, if } l \in L_3(\mathbf{x})\,. \end{cases} \tag{9.81}$$

For an input vector $\mathbf{x} \in \mathbf{X_j}$ with one single input value x_{l_1} coinciding with the intersection of the corresponding membership functions $B^{l_1}_{j_{l_1}}$ and $B^{l_1}_{j_{l_1}+1}$, two rules R_{s_1} and R_{s_2}, with

$$j_{l,s_1} = \begin{cases} j_l & \text{, if } l \in L_1(\mathbf{x}) \cup \{l_1\}\,, \\ j_l + 1 & \text{, if } l \in L_3(\mathbf{x})\,, \end{cases}$$

$$j_{l,s_2} = \begin{cases} j_l & \text{, if } l \in L_1(\mathbf{x})\,, \\ j_l + 1 & \text{, if } l \in L_3(\mathbf{x}) \cup \{l_1\}\,, \end{cases}$$

are fired to the maximum fulfilment degree $\alpha_{\max}$

$$\beta_{s_1} = \underbrace{\prod_{l\in L_1(\mathbf{x})} B^l_{j_l}(x_l) \times \prod_{l\in L_3(\mathbf{x})} B^l_{j_l+1}(x_l)} \times \underbrace{B^{l_1}_{j_{l_1}}(x_{l_1})} = \alpha_{\max}\,,$$

$$= \qquad\qquad = 0.5$$

$$\beta_{s_2} = \overbrace{\prod_{l\in L_1(\mathbf{x})} B^l_{j_l}(x_l) \times \prod_{l\in L_3(\mathbf{x})} B^l_{j_l+1}(x_l)} \times \overbrace{B^{l_1}_{j_{l_1}+1}(x_{l_1})} = \alpha_{\max}\,.$$

All other rules $R_s \in \mathcal{R}\setminus\{R_{s_1}, R_{s_2}\}$ contain at least one linguistic input value to which $\mathbf{x}$ has a membership degree smaller than 0.5, with $L_1(\mathbf{x}) = L_{1,1}(\mathbf{x}) \cup L_{1,2}(\mathbf{x})$, $L_3(\mathbf{x}) = L_{3,1}(\mathbf{x}) \cup L_{3,2}(\mathbf{x})$ and $L_{1,2}(\mathbf{x}) \cup L_{3,2}(\mathbf{x}) \neq \emptyset$, with

$$j_{l,s} = \begin{cases} j_l & \text{, if } l \in L_{1,1}(\mathbf{x}) \cup L_{3,2}(\mathbf{x})\,,\\ \in \{j_l, j_l+1\} & \text{, if } l = l_1,\\ j_l + 1 & \text{, if } l \in L_{3,1}(\mathbf{x}) \cup L_{1,2}(\mathbf{x})\,, \end{cases} \tag{9.82}$$

and have a fulfilment degree β_s, given by

$$\beta_s = \prod_{\substack{l\in L_{1,1}(\mathbf{x})\\ \cup L_{3,2}(\mathbf{x})}} B^l_{j_l}(x_l) \times \prod_{\substack{l\in L_{3,1}(\mathbf{x})\\ \cup L_{1,2}(\mathbf{x})}} B^l_{j_l+1}(x_l) \times 0.5\,, \tag{9.83}$$

smaller than $\alpha_{\max}$, since

$$(\forall l \in L_1(\mathbf{x}))(B^l_{j_l}(x_l) > B^l_{j_l+1}(x_l))\,, \tag{9.84}$$

$$(\forall l \in L_3(\mathbf{x}))(B^l_{j_l}(x_l) < B^l_{j_l+1}(x_l))\,. \tag{9.85}$$

In general, when applying $T_{\mathbf{P}}$, for an input vector $\mathbf{x} \in \mathbf{X_j}$, $2^{|L_2(\mathbf{x})|}$ rules R_s are fired to the degree $\alpha_{\max}$, with

$$j_{l,s} = \begin{cases} j_l & \text{, if } l \in L_1(\mathbf{x})\,,\\ \in \{j_l, j_l+1\} & \text{, if } l \in L_2(\mathbf{x})\,,\\ j_l + 1 & \text{, if } l \in L_3(\mathbf{x})\,. \end{cases} \tag{9.86}$$

The expression above is identical to Eq. (9.57) obtained for models applying $T_{\mathbf{M}}$. Thus, for a given model with a monotone smooth rule base and for a given input vector $\mathbf{x}$, the set $I_{\max}$ of indices of linguistic output values fired to the degree $\alpha_{\max}$ is the same for a model applying $T_{\mathbf{M}}$ as for a model applying $T_{\mathbf{P}}$. Thus, for two input vectors $\mathbf{x}_1, \mathbf{x}_2 \in \mathbf{X_j}$, such that $x_{1,l} = x_{2,l}$ for any $l \in L \setminus \{l_1\}$ and $x_{1,l_1} \leq x_{2,l_1}$,

$$i_{\text{left}}(\mathbf{x}_1) \leq i_{\text{left}}(\mathbf{x}_2) \qquad i_{\text{left}} = \min_{i\in I_{\max}} i\,, \tag{9.87}$$

$$i_{\text{right}}(\mathbf{x}_1) \leq i_{\text{right}}(\mathbf{x}_2) \qquad i_{\text{right}} = \max_{i\in I_{\max}} i\,, \tag{9.88}$$

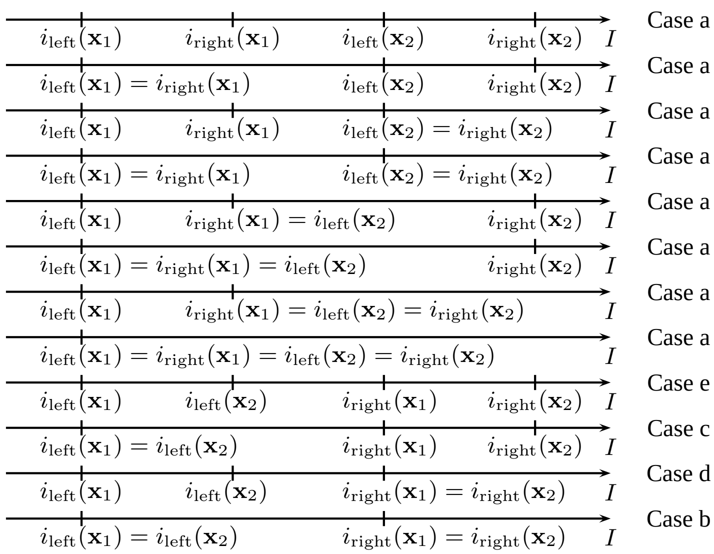


Figure 9.2: Cases to be considered for $i_{\text{left}}(\mathbf{x}_1)$, $i_{\text{right}}(\mathbf{x}_1)$, $i_{\text{left}}(\mathbf{x}_2)$ and $i_{\text{right}}(\mathbf{x}_2)$ when investigating models with a monotone smooth rule base applying $T_{\mathbf{P}}$.

also hold when the t-norm $T_{\mathbf{P}}$ is used (Section 9.4.1 and in particular Eqs. (9.70–9.71)). Furthermore, also for models with a monotone smooth rule base applying $T_{\mathbf{P}}$, all linguistic output values between $A_{i_{\text{left}}}$ and $A_{i_{\text{right}}}$ are fired to the degree $\alpha_{\max}$. Therefore, $I_{\max}$ in Eq. (9.24) can be replaced by $[i_{\text{left}}, i_{\text{right}}]$ when calculating $y^*_{\text{MOM}}(\mathbf{x})$ for all $\mathbf{x}$ in the subspace $\mathbf{X_j}$.

To investigate the monotonicity of a model in a variable X_{l_1}, two input vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ are considered, such that $x_{1,l} = x_{2,l}$ for any $l \in L \setminus \{l_1\}$ and $x_{1,l_1} \leq x_{2,l_1}$. In Fig. 9.2 the 12 cases are represented that should be considered for the four integers $i_{\text{left}}(\mathbf{x}_1)$, $i_{\text{right}}(\mathbf{x}_1)$, $i_{\text{left}}(\mathbf{x}_2)$ and $i_{\text{right}}(\mathbf{x}_2)$. In the discussion below, the first eight cases are combined in Case a. Before focussing on the model behaviour in Cases a–e, a useful property is stated:

$$c_{i'_{\text{left}}} \leq \frac{\sum\limits_{j=i_{\text{left}}}^{i_{\text{right}}} k_j c_j}{\sum\limits_{j=i_{\text{left}}}^{i_{\text{right}}} k_j} \leq c_{i'_{\text{right}}}\,, \tag{9.89}$$

with

$$(\forall j \in [i_{\text{left}}, i'_{\text{left}}\text{-}1] \cup [i'_{\text{right}}\text{+}1, i_{\text{right}}])(k_j = 0)\,, \tag{9.90}$$

$$(\exists j \in [i'_{\text{left}}, i'_{\text{right}}])(k_j > 0)\,. \tag{9.91}$$

From Eqs. (9.90–9.91) it follows that

$$\frac{\sum\limits_{j=i_{\text{left}}}^{i_{\text{right}}} k_j c_j}{\sum\limits_{j=i_{\text{left}}}^{i_{\text{right}}} k_j} = \frac{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j c_j}{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j} \,. \tag{9.92}$$

As furthermore, for $i'_{\text{left}} < i'_{\text{right}}$, it holds that

$$\begin{aligned}
\frac{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j c_j}{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j} - c_{i'_{\text{left}}} &= \frac{k_{i'_{\text{left}}} c_{i'_{\text{left}}} + \sum\limits_{j=i'_{\text{left}}+1}^{i'_{\text{right}}} k_j c_j - k_{i'_{\text{left}}} c_{i'_{\text{left}}} - \sum\limits_{j=i'_{\text{left}}+1}^{i'_{\text{right}}} k_j c_{i'_{\text{left}}}}{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j} \\
&= \left(\sum_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j\right)^{-1} \left(\sum_{j=i'_{\text{left}}+1}^{i'_{\text{right}}} k_j (c_j - c_{i'_{\text{left}}})\right) \\
&\geq 0 \,,
\end{aligned} \tag{9.93}$$

$$\begin{aligned}
c_{i'_{\text{right}}} - \frac{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j c_j}{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j} &= \frac{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}\text{-}1} k_j c_{i'_{\text{right}}} + k_{i'_{\text{right}}} c_{i'_{\text{right}}} - \sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}\text{-}1} k_j c_j - k_{i'_{\text{right}}} c_{i'_{\text{right}}}}{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j} \\
&= \left(\sum_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j\right)^{-1} \left(\sum_{j=i'_{\text{left}}}^{i'_{\text{right}}\text{-}1} k_j (c_{i'_{\text{left}}} - c_j)\right) \\
&\geq 0 \,,
\end{aligned} \tag{9.94}$$

and for $i'_{\text{left}} = i'_{\text{right}}$, it holds that

$$\frac{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j c_j}{\sum\limits_{j=i'_{\text{left}}}^{i'_{\text{right}}} k_j} = c_{i'_{\text{left}}} = c_{i'_{\text{right}}} \,, \tag{9.95}$$

it follows that Eq. (9.89) holds.

Case a When the largest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_1$ is smaller than or equal to the smallest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_2$, *i.e.*

$$i_{\text{left}}(\mathbf{x}_1) \leq i_{\text{right}}(\mathbf{x}_1) \leq i_{\text{left}}(\mathbf{x}_2) \leq i_{\text{right}}(\mathbf{x}_2) \,, \tag{9.96}$$

the following chain of inequalities holds

$$c_{i_{\mathrm{left}}(\mathbf{x}_1)} \leq y^*_{\mathrm{MOM}}(\mathbf{x}_1) \leq c_{i_{\mathrm{right}}(\mathbf{x}_1)} \leq c_{i_{\mathrm{left}}(\mathbf{x}_2)} \leq y^*_{\mathrm{MOM}}(\mathbf{x}_2) \leq c_{i_{\mathrm{right}}(\mathbf{x}_2)}\,, \tag{9.97}$$

and it follows that the crisp output $y^*_{\mathrm{MOM}}(\mathbf{x}_1)$ obtained for $\mathbf{x}_1$ is smaller than or equal to the crisp output $y^*_{\mathrm{MOM}}(\mathbf{x}_2)$ obtained for $\mathbf{x}_2$

$$y^*_{\mathrm{MOM}}(\mathbf{x}_1) \leq y^*_{\mathrm{MOM}}(\mathbf{x}_2)\,. \tag{9.98}$$

Case b When the smallest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_1$ is equal to the smallest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_2$, the largest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_1$ is equal to the largest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_2$ and this smallest linguistic output value differs from this largest linguistic output value, *i.e.*

$$i_{\mathrm{left}}(\mathbf{x}_1) = i_{\mathrm{left}}(\mathbf{x}_2) < i_{\mathrm{right}}(\mathbf{x}_1) = i_{\mathrm{right}}(\mathbf{x}_2)\,, \tag{9.99}$$

the crisp output $y^*_{\mathrm{MOM}}(\mathbf{x}_1)$ obtained for $\mathbf{x}_1$ is equal to the crisp output $y^*_{\mathrm{MOM}}(\mathbf{x}_2)$ obtained for $\mathbf{x}_2$

$$y^*_{\mathrm{MOM}}(\mathbf{x}_1) = y^*_{\mathrm{MOM}}(\mathbf{x}_2)\,. \tag{9.100}$$

This can easily be verified by substituting $i_{\mathrm{left}}(\mathbf{x}_2)$ by $i_{\mathrm{left}}(\mathbf{x}_1)$ and $i_{\mathrm{right}}(\mathbf{x}_2)$ by $i_{\mathrm{right}}(\mathbf{x}_1)$ in the expression for $y^*_{\mathrm{MOM}}(\mathbf{x}_2)$.

Case c When the smallest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_1$ is equal to the smallest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_2$, the largest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_1$ is smaller than the largest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_2$ and the smallest linguistic output value differs from the two 'largest linguistic output values', *i.e.*

$$i_{\mathrm{left}}(\mathbf{x}_1) = i_{\mathrm{left}}(\mathbf{x}_2) < i_{\mathrm{right}}(\mathbf{x}_1) < i_{\mathrm{right}}(\mathbf{x}_2)\,, \tag{9.101}$$

three subcases can be distinguished.

Case c1 If $\sum\limits_{j=i_{\mathrm{left}}(\mathbf{x}_2)}^{i_{\mathrm{right}}(\mathbf{x}_1)} k_j > 0$, then the difference between the crisp outputs $y^*_{\mathrm{MOM}}(\mathbf{x}_2)$ and $y^*_{\mathrm{MOM}}(\mathbf{x}_1)$ is given by

$$y^*_{\mathrm{MOM}}(\mathbf{x}_2) - y^*_{\mathrm{MOM}}(\mathbf{x}_1) = \frac{\sum\limits_{j_1=i_{\mathrm{left}}(\mathbf{x}_2)}^{i_{\mathrm{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1}}{\sum\limits_{j_2=i_{\mathrm{left}}(\mathbf{x}_2)}^{i_{\mathrm{right}}(\mathbf{x}_2)} k_{j_2}} - \frac{\sum\limits_{j_3=i_{\mathrm{left}}(\mathbf{x}_1)}^{i_{\mathrm{right}}(\mathbf{x}_1)} k_{j_3} c_{j_3}}{\sum\limits_{j_4=i_{\mathrm{left}}(\mathbf{x}_1)}^{i_{\mathrm{right}}(\mathbf{x}_1)} k_{j_4}}\,, \tag{9.102}$$

or, after substitution of $i_{\text{left}}(\mathbf{x}_1)$ by $i_{\text{left}}(\mathbf{x}_2)$, by

$$
\begin{aligned}
& y^*_{\text{MOM}}(\mathbf{x}_2) - y^*_{\text{MOM}}(\mathbf{x}_1) \\
&= \frac{\sum\limits_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1}}{\sum\limits_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2}} - \frac{\sum\limits_{j_3=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_3} c_{j_3}}{\sum\limits_{j_4=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4}} \\
&= \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum_{j_4=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \right)^{-1} \\
& \quad \times \left(\sum_{j_4=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \left(\sum_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_1} c_{j_1} + \sum_{j_1=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1} \right) \right. \\
& \qquad \left. - \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_2} + \sum_{j_2=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \right) \sum_{j_3=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_3} c_{j_3} \right) \\
&= \frac{\sum\limits_{j_4=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \sum\limits_{j_1=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1} - \sum\limits_{j_2=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum\limits_{j_3=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_3} c_{j_3}}{\sum\limits_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum\limits_{j_4=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4}}
\end{aligned}
\tag{9.103}
$$

If $\sum\limits_{j=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_j = 0$, then the difference between $y^*_{\text{MOM}}(\mathbf{x}_2)$ and $y^*_{\text{MOM}}(\mathbf{x}_1)$ is zero. If $\sum\limits_{j=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_j > 0$, then the difference between $y^*_{\text{MOM}}(\mathbf{x}_2)$ and $y^*_{\text{MOM}}(\mathbf{x}_1)$ is strictly positive since for the assumed fuzzy output partitions $c_{i_{\text{right}}(\mathbf{x}_1)} < c_{i_{\text{right}}(\mathbf{x}_1)+1}$ and it follows with Eq. (9.89) that

$$
c_{i_{\text{right}}(\mathbf{x}_1)+1} \leq \frac{\sum\limits_{j=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_j c_j}{\sum\limits_{j=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_j} \leq c_{i_{\text{right}}(\mathbf{x}_2)} \,, \tag{9.104}
$$

$$
c_{i_{\text{left}}(\mathbf{x}_2)} \leq \frac{\sum\limits_{j=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_j c_j}{\sum\limits_{j=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_j} \leq c_{i_{\text{right}}(\mathbf{x}_1)} \,. \tag{9.105}
$$

Case c2 If $(\forall j \in [i_{\text{left}}(\mathbf{x}_2), i_{\text{right}}(\mathbf{x}_1)])(k_j = 0)$ and $\sum_{j=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_j > 0$, then the following chain of inequalities holds

$$c_{i_{\text{left}}(\mathbf{x}_1)} \leq y^*_{\text{MOM}}(\mathbf{x}_1) \leq c_{i_{\text{right}}(\mathbf{x}_1)} < c_{i_{\text{right}}(\mathbf{x}_1)+1} \leq y^*_{\text{MOM}}(\mathbf{x}_2) \leq c_{i_{\text{right}}(\mathbf{x}_2)}\,, \tag{9.106}$$

and it follows that the crisp output $y^*_{\text{MOM}}(\mathbf{x}_1)$ is smaller than or equal to the crisp output $y^*_{\text{MOM}}(\mathbf{x}_2)$

$$y^*_{\text{MOM}}(\mathbf{x}_1) \leq y^*_{\text{MOM}}(\mathbf{x}_2)\,. \tag{9.107}$$

Case c3 Finally, when all linguistic output values fired to the maximum fulfilment degree are described by triangular membership functions, $(\forall j \in [i_{\text{left}}(\mathbf{x}_2), i_{\text{right}}(\mathbf{x}_2)])(k_j = 0)$, the difference between $y^*_{\text{MOM}}(\mathbf{x}_2)$ and $y^*_{\text{MOM}}(\mathbf{x}_1)$ is given by a special case of Eq. (9.103)

$$\begin{aligned} &y^*_{\text{MOM}}(\mathbf{x}_2) - y^*_{\text{MOM}}(\mathbf{x}_1) \\ &= \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} 1\right)^{-1} \left(\sum_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} c_{j_1}\right) - \left(\sum_{j_4=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} 1\right)^{-1} \left(\sum_{j_3=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} c_{j_3}\right), \end{aligned} \tag{9.108}$$

which was shown to be positive.

Case d When the smallest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_1$ is smaller than the smallest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_2$, the largest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_1$ is equal to the largest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_2$ and two 'smallest linguistic output values' differ from the largest linguistic output value, *i.e.*

$$i_{\text{left}}(\mathbf{x}_1) < i_{\text{left}}(\mathbf{x}_2) < i_{\text{right}}(\mathbf{x}_1) = i_{\text{right}}(\mathbf{x}_2)\,, \tag{9.109}$$

three subcases can be distinguished.

Case d1 If $\sum_{j=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_j > 0$, the difference between the crisp outputs $y^*_{\text{MOM}}(\mathbf{x}_2)$ and $y^*_{\text{MOM}}(\mathbf{x}_1)$ is given by Eq. (9.102), or, after substitution of $i_{\text{right}}(\mathbf{x}_1)$ by $i_{\text{right}}(\mathbf{x}_2)$, by

$$\begin{aligned} &y^*_{\text{MOM}}(\mathbf{x}_2) - y^*_{\text{MOM}}(\mathbf{x}_1) \\ &= \frac{\sum_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1}}{\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2}} - \frac{\sum_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_3} c_{j_3}}{\sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_4}} \end{aligned}$$

$$
\begin{aligned}
&= \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_4}\right)^{-1} \\
&\quad \times \left(\left(\sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_{j_4} + \sum_{j_4=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_4}\right) \sum_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1} \right. \\
&\quad \left. - \sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \left(\sum_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_{j_3} c_{j_3} + \sum_{j_3=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_3} c_{j_3}\right)\right) \\
&= \frac{\sum\limits_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_{j_4} \sum\limits_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1} - \sum\limits_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum\limits_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_{j_3} c_{j_3}}{\sum\limits_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum\limits_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_4}}
\end{aligned}
\tag{9.110}
$$

One can easily verify following a similar procedure as described in Case c1 that this difference between $y^*_{\text{MOM}}(\mathbf{x}_2)$ and $y^*_{\text{MOM}}(\mathbf{x}_1)$ is positive.

Case d2 If $(\forall j \in [i_{\text{left}}(\mathbf{x}_2), i_{\text{right}}(\mathbf{x}_2)])(k_j = 0)$ and $\sum\limits_{j=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_j > 0$, the following chain of inequalities holds

$$
c_{i_{\text{left}}(\mathbf{x}_1)} \leq y^*_{\text{MOM}}(\mathbf{x}_1) \leq c_{i_{\text{left}}(\mathbf{x}_2)\text{-}1} < c_{i_{\text{left}}(\mathbf{x}_2)} \leq y^*_{\text{MOM}}(\mathbf{x}_2) \leq c_{i_{\text{right}}(\mathbf{x}_2)}\,, \tag{9.111}
$$

and it follows that the crisp output $y^*_{\text{MOM}}(\mathbf{x}_1)$ is smaller than or equal to the crisp output $y^*_{\text{MOM}}(\mathbf{x}_2)$

$$
y^*_{\text{MOM}}(\mathbf{x}_1) \leq y^*_{\text{MOM}}(\mathbf{x}_2)\,. \tag{9.112}
$$

Case d3 Finally, when all linguistic output values fired to the maximum fulfilment degree are described by triangular membership functions, $(\forall j \in [i_{\text{left}}(\mathbf{x}_1), i_{\text{right}}(\mathbf{x}_2)])(k_j = 0)$, the difference between $y^*_{\text{MOM}}(\mathbf{x}_2)$ and $y^*_{\text{MOM}}(\mathbf{x}_1)$ is given by a special case of Eq. (9.110)

$$
\begin{aligned}
&y^*_{\text{MOM}}(\mathbf{x}_2) - y^*_{\text{MOM}}(\mathbf{x}_1) \\
&= \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} 1\right)^{-1} \left(\sum_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} c_{j_1}\right) - \left(\sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_2)} 1\right)^{-1} \left(\sum_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_2)} c_{j_3}\right),
\end{aligned}
\tag{9.113}
$$

which was shown to be positive.

Case e When the smallest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_1$ is smaller than the smallest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_2$, which on its turn is smaller than the largest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_1$, which on its turn is smaller than the largest linguistic output value fired to the maximum fulfilment degree for $\mathbf{x}_2$, *i.e.*

$$i_{\text{left}}(\mathbf{x}_1) < i_{\text{left}}(\mathbf{x}_2) < i_{\text{right}}(\mathbf{x}_1) < i_{\text{right}}(\mathbf{x}_2)\,, \tag{9.114}$$

four subcases can be distinguished.

Case e1 If $\sum_{j=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_j > 0$ and $\sum_{j=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_j > 0$, then the difference between the crisp outputs $y^*_{\text{MOM}}(\mathbf{x}_2)$ and $y^*_{\text{MOM}}(\mathbf{x}_1)$ is given by Eq. (9.102)

$$\begin{aligned}
&y^*_{\text{MOM}}(\mathbf{x}_2) - y^*_{\text{MOM}}(\mathbf{x}_1) \\
&= \frac{\sum\limits_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1}}{\sum\limits_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2}} - \frac{\sum\limits_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_3} c_{j_3}}{\sum\limits_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4}} \\
&= \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \right)^{-1} \\
&\quad \times \left(\sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \left(\sum_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_1} c_{j_1} + \sum_{j_1=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1} \right) \right. \\
&\qquad \left. - \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_2} + \sum_{j_2=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \right) \sum_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_3} c_{j_3} \right) \\
&= \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \right)^{-1} \\
&\quad \times \left(\left(\sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_{j_4} + \sum_{j_4=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \right) \sum_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_1} c_{j_1} \right. \\
&\qquad - \sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_2} \left(\sum_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_{j_3} c_{j_3} + \sum_{j_3=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_3} c_{j_3} \right) \\
&\qquad \left. + \sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \sum_{j_1=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1} - \sum_{j_2=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_3} c_{j_3} \right)
\end{aligned}$$

$$
\begin{aligned}
&= \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \right)^{-1} \\
&\quad \times \left(\sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_{j_4} \sum_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_1} c_{j_1} - \sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_2} \sum_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_{j_3} c_{j_3} \right. \\
&\quad\quad \left. + \sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_4} \sum_{j_1=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_1} c_{j_1} - \sum_{j_2=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_{j_2} \sum_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} k_{j_3} c_{j_3} \right)
\end{aligned}
\tag{9.115}
$$

One can easily verify following a similar procedure as described in Case c1 that this difference between $y^*_{\text{MOM}}(\mathbf{x}_2)$ and $y^*_{\text{MOM}}(\mathbf{x}_1)$ is positive.

Case e2 If $(\forall j \in [i_{\text{left}}(\mathbf{x}_1), i_{\text{right}}(\mathbf{x}_1)])(k_j = 0)$ and $\sum_{j=i_{\text{right}}(\mathbf{x}_1)+1}^{i_{\text{right}}(\mathbf{x}_2)} k_j > 0$, then the following chain of inequalities holds

$$
c_{i_{\text{left}}(\mathbf{x}_1)} \leq y^*_{\text{MOM}}(\mathbf{x}_1) \leq c_{i_{\text{right}}(\mathbf{x}_1)} < c_{i_{\text{right}}(\mathbf{x}_1)+1} \leq y^*_{\text{MOM}}(\mathbf{x}_2) \leq c_{i_{\text{right}}(\mathbf{x}_2)} \,, \tag{9.116}
$$

and it follows that the crisp output $y^*_{\text{MOM}}(\mathbf{x}_1)$ is smaller than the crisp output $y^*_{\text{MOM}}(\mathbf{x}_2)$

$$
y^*_{\text{MOM}}(\mathbf{x}_1) < y^*_{\text{MOM}}(\mathbf{x}_2) \,. \tag{9.117}
$$

Case e3 If $(\forall j \in [i_{\text{left}}(\mathbf{x}_2), i_{\text{right}}(\mathbf{x}_2)])(k_j = 0)$ and $\sum_{j=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{left}}(\mathbf{x}_2)\text{-}1} k_j > 0$, then the following chain of inequalities holds

$$
c_{i_{\text{left}}(\mathbf{x}_1)} \leq y^*_{\text{MOM}}(\mathbf{x}_1) \leq c_{i_{\text{left}}(\mathbf{x}_2)\text{-}1} < c_{i_{\text{left}}(\mathbf{x}_2)} \leq y^*_{\text{MOM}}(\mathbf{x}_2) \leq c_{i_{\text{right}}(\mathbf{x}_2)} \tag{9.118}
$$

and it follows that the crisp output $y^*_{\text{MOM}}(\mathbf{x}_1)$ is smaller than the crisp output $y^*_{\text{MOM}}(\mathbf{x}_2)$

$$
y^*_{\text{MOM}}(\mathbf{x}_1) < y^*_{\text{MOM}}(\mathbf{x}_2) \,. \tag{9.119}
$$

Case e4 Finally, when all linguistic output values fired to the maximum fulfilment degree are described by triangular membership functions, $(\forall j \in [i_{\text{left}}(\mathbf{x}_1), i_{\text{right}}(\mathbf{x}_2)])(k_j = 0)$, the difference between $y^*_{\text{MOM}}(\mathbf{x}_2)$ and $y^*_{\text{MOM}}(\mathbf{x}_1)$ is given by a special case of Eq. (9.115)

$$
\begin{aligned}
&y^*_{\text{MOM}}(\mathbf{x}_2) - y^*_{\text{MOM}}(\mathbf{x}_1) \\
&= \left(\sum_{j_2=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} 1 \right)^{-1} \left(\sum_{j_1=i_{\text{left}}(\mathbf{x}_2)}^{i_{\text{right}}(\mathbf{x}_2)} c_{j_1} \right) - \left(\sum_{j_4=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} 1 \right)^{-1} \left(\sum_{j_3=i_{\text{left}}(\mathbf{x}_1)}^{i_{\text{right}}(\mathbf{x}_1)} c_{j_3} \right) ,
\end{aligned}
\tag{9.120}
$$

which was shown to be positive.

Based on the results obtained for Cases a–e it can be concluded that a monotone input-output behaviour is obtained for each subspace $\mathbf{X_j}$, and, by construction, for the whole input space of a Mamdani–Assilian model with a smooth rule base and applying $T_{\mathbf{P}}$.

9.5 Models with a monotone rule base and two input variables

In this section the monotonicity of models with a monotone rule base and two input variables is investigated for models applying $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$. Models with more than one input variable applying $T_{\mathbf{L}}$ are again not considered. For models applying $T_{\mathbf{M}}$ the output membership functions are again assumed to satisfy Eqs. (9.33–9.34).

9.5.1 Models applying $T_{\mathbf{M}}$

In this section it is shown that there exist models with two input variables and a monotone non-smooth rule base, for which a non-monotone input-output behaviour is obtained for any fuzzy output partition as described in Section 7.2. As the goal of this study was to select combinations of t-norm and defuzzification method resulting in a monotone input-output behaviour for any monotone rule base or any monotone smooth rule base, the combination of $T_{\mathbf{M}}$ and MOM defuzzification is hereby abandoned as appropriate combination in case of a monotone non-smooth rule bases.

When a model with two input variables X_1 and X_2 contains the following rules, represented in Fig. 9.3, in its monotone but non-smooth rule base ($q_1, q_2 \in \mathbb{N}_0, q_1 < q_2$)

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+q_2}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS $A_{i+q_1+q_2}$
IF	X_1 IS $B^1_{j_1+2}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_{i+q_1}
IF	X_1 IS $B^1_{j_1+2}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS $A_{i+q_1+q_2}$

the following chain of inequalities should be satisfied when the model has a monotone input-output behaviour

$$y^*_{\mathrm{MOM},\mathbf{2M},1} \leq y^*_{\mathrm{MOM},\mathbf{2M},2} \leq y^*_{\mathrm{MOM},\mathbf{2M},3}\,, \tag{9.121}$$

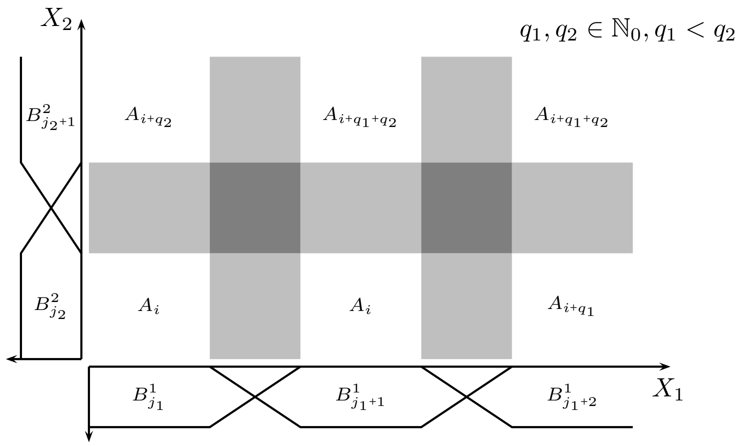


Figure 9.3: Example of a monotone non-smooth rule base for which no monotone input-output behaviour can be obtained when applying $T_{\mathbf{M}}$ combined with the MOM defuzzification method.

with

$$y^*_{\text{MOM}} = \begin{cases} y^*_{\text{MOM},\mathbf{2M},1} & \text{, if } B^1_{j_1}(x_1) = B^1_{j_1+1}(x_1) = 0.5, \\ & B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2) = 0.5\,, \\ y^*_{\text{MOM},\mathbf{2M},2} & \text{, if } B^1_{j_1}(x_1) < 0.5, B^1_{j_1+1}(x_1) > 0.5, \\ & B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2) = 0.5\,, \\ y^*_{\text{MOM},\mathbf{2M},3} & \text{, if } B^1_{j_1+1}(x_1) = B^1_{j_1+2}(x_1) = 0.5, \\ & B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2) = 0.5\,. \end{cases}$$

The crisp outputs $y^*_{\text{MOM},\mathbf{2M},i}$ are derived from fuzzy outputs with corresponding sets $I_{\max,i}$,

- $I_{\max,1} = \{i, i+q_2, i+q_1+q_2\}$,
- $I_{\max,2} = \{i, i+q_1+q_2\}$, and
- $I_{\max,3} = \{i, i+q_1, i+q_1+q_2\}$.

Since the weights w_i, w_{i+q_1}, w_{i+q_2}, $w_{i+q_1+q_2}$ in Eq. (9.27) are strictly positive, the difference between the outputs is given by

$$y^*_{\text{MOM},\mathbf{2M},2} - y^*_{\text{MOM},\mathbf{2M},1} = \frac{w_{i+q_2}(w_{i+q_1+q_2}(c_{i+q_1+q_2} - c_{i+q_2}) - w_i(c_{i+q_2} - c_i))}{(w_i + w_{i+q_2} + w_{i+q_1+q_2})(w_i + w_{i+q_1+q_2})} \tag{9.122}$$

$$y^*_{\text{MOM},\mathbf{2M},3} - y^*_{\text{MOM},\mathbf{2M},2} = \frac{w_{i+q_1}(w_i(c_{i+q_1} - c_i) - w_{i+q_1+q_2}(c_{i+q_1+q_2} - c_{i+q_1}))}{(w_i + w_{i+q_1} + w_{i+q_1+q_2})(w_i + w_{i+q_1+q_2})} \tag{9.123}$$

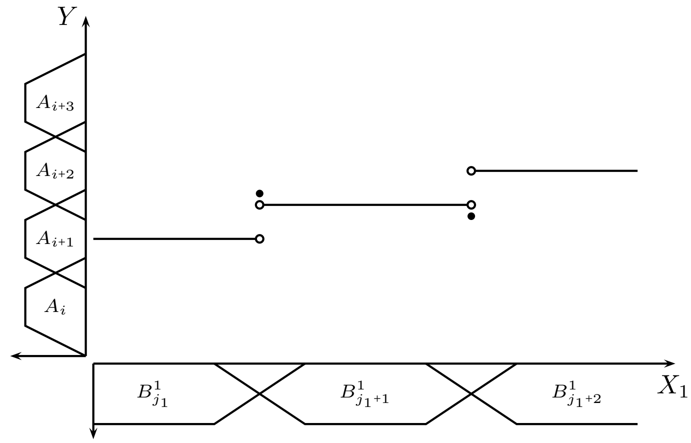


Figure 9.4: Crisp output y^*_{MOM} obtained for the rules represented in Fig. 9.3 for $T = T_{\mathbf{M}}$, $q_1 = 1$, $q_2 = 2$ and $B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2) = 0.5$.

and, since $c_i < c_{i+q_1} < c_{i+q_2} < c_{i+q_1+q_2}$, it follows that

$$\begin{aligned}
& y^*_{\mathrm{MOM},\mathbf{2M},1} \leq y^*_{\mathrm{MOM},\mathbf{2M},2} \leq y^*_{\mathrm{MOM},\mathbf{2M},3} \\
& \Leftrightarrow (w_i(c_{i+q_2} - c_i) \leq w_{i+q_1+q_2}(c_{i+q_1+q_2} - c_{i+q_2})) \\
& \quad \wedge (w_{i+q_1+q_2}(c_{i+q_1+q_2} - c_{i+q_1}) \leq w_i(c_{i+q_1} - c_i)) \\
& \Leftrightarrow \frac{c_{i+q_1+q_2} - c_{i+q_1}}{c_{i+q_1} - c_i} \leq \frac{w_i}{w_{i+q_1+q_2}} \leq \frac{c_{i+q_1+q_2} - c_{i+q_2}}{c_{i+q_2} - c_i} \\
& \Rightarrow (c_{i+q_2} - c_i)(c_{i+q_1+q_2} - c_{i+q_1}) \leq (c_{i+q_1} - c_i)(c_{i+q_1+q_2} - c_{i+q_2}) \\
& \Leftrightarrow 0 \leq (c_{i+q_1+q_2} - c_i)(c_{i+q_1} - c_{i+q_2})\,.
\end{aligned} \tag{9.124}$$

However, as $c_{i+q_1+q_2} > c_i$ and $c_{i+q_1} < c_{i+q_2}$, the latter expression can never be positive, and therefore Eq. (9.121) does not hold. This is illustrated in Fig. 9.4 for the rule base in Fig. 9.3.

9.5.2 Models applying $T_{\mathbf{P}}$

We show in this section that also when applying $T_{\mathbf{P}}$, opting for a monotone smooth rule base is recommended when designing a monotone model, since monotonicity is not obtained for any monotone, but non-smooth rule base if the model uses more than five linguistic output values.

A monotone input-output behaviour should be obtained for a monotone, but non-smooth rule base containing the following rules, represented in Fig. 9.5, with the three linguistic output values A_i, A_{i+q_1} and $A_{i+q_1+q_2}$ in their consequents ($q_1, q_2 \in \mathbb{N}_0$)

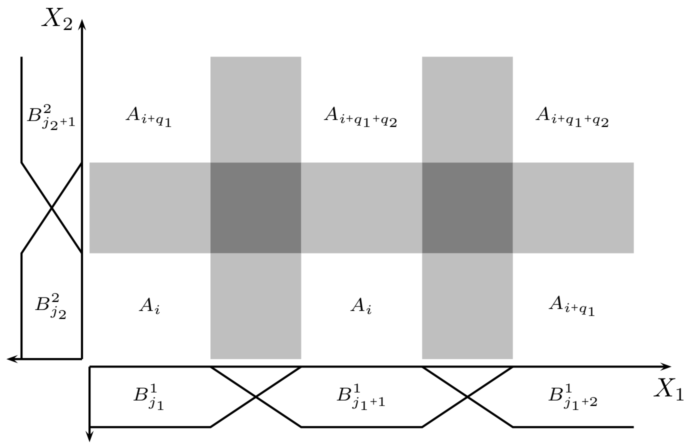


Figure 9.5: Representation of the monotone non-smooth rule base discussed in Section 9.5.2.

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+q_1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS $A_{i+q_1+q_2}$
IF	X_1 IS $B^1_{j_1+2}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_{i+q_1}
IF	X_1 IS $B^1_{j_1+2}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS $A_{i+q_1+q_2}$

in order to withhold the combination of the t-norm $T_{\mathbf{P}}$ and the MOM defuzzification method for the design of monotone models with a non-smooth rule base.

To that end, the following chain of inequalities should hold

$$y^*_{\mathrm{MOM},\mathbf{2P},1} \leq y^*_{\mathrm{MOM},\mathbf{2P},2} \leq y^*_{\mathrm{MOM},\mathbf{2P},3}\,, \tag{9.125}$$

with

$$y^*_{\mathrm{MOM}} = \begin{cases} y^*_{\mathrm{MOM},\mathbf{2P},1} & \text{, if } B^1_{j_1}(x_1) = B^1_{j_1+1}(x_1) = 0.5, \\ & B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2) = 0.5\,, \\ y^*_{\mathrm{MOM},\mathbf{2P},2} & \text{, if } B^1_{j_1}(x_1) < 0.5, B^1_{j_1+1}(x_1) > 0.5, \\ & B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2) = 0.5\,, \\ y^*_{\mathrm{MOM},\mathbf{2P},3} & \text{, if } B^1_{j_1+1}(x_1) = B^1_{j_1+2}(x_1) = 0.5, \\ & B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2) = 0.5\ . \end{cases}$$

The crisp outputs $y^*_{\mathrm{MOM},\mathbf{2P},i}$ are derived from fuzzy outputs with corresponding sets $I_{\max,i}$,

- $I_{\max,1} = \{i, i{+}q_1, i{+}q_1{+}q_2\}$,
- $I_{\max,2} = \{i, i{+}q_1{+}q_2\}$, and

- $I_{\max,3} = \{i, i{+}q_1, i{+}q_1{+}q_2\}$.

In models applying $T_{\mathbf{P}}$ the crisp model output y^*_{MOM} depends on which linguistic output values are fired to the maximum fulfilment degree $\alpha_{\max}$, but does not depend on the value as such, of the maximum fulfilment degree as shown by Eq. (9.24). Thus for the rules represented in Fig. 9.5 it holds that

$$y^*_{\mathrm{MOM},\mathbf{2P},1} = y^*_{\mathrm{MOM},\mathbf{2P},3} \,, \tag{9.126}$$

and Eq. (9.125)can only hold if

$$y^*_{\mathrm{MOM},\mathbf{2P},1} = y^*_{\mathrm{MOM},\mathbf{2P},2} \,. \tag{9.127}$$

Case a If k_i, k_{i+q_1} and $k_{i+q_1+q_2}$ are all strictly positive, Eq. (9.127) is equivalent with

$$\begin{aligned}
& \frac{k_i c_i + k_{i+q_1} c_{i+q_1} + k_{i+q_1+q_2} c_{i+q_1+q_2}}{k_i + k_{i+q_1} + k_{i+q_1+q_2}} = \frac{k_i c_i + k_{i+q_1+q_2} c_{i+q_1+q_2}}{k_i + k_{i+q_1+q_2}} \\
\Leftrightarrow \quad & k_i k_{i+q_1} c_{i+q_1} + k_{i+q_1} k_{i+q_1+q_2} c_{i+q_1} = k_i k_{i+q_1} c_i + k_{i+q_1} k_{i+q_1+q_2} c_{i+q_1+q_2} \\
\Leftrightarrow \quad & k_i (c_{i+q_1} - c_i) = k_{i+q_1+q_2} (c_{i+q_1+q_2} - c_{i+q_1}) \,.
\end{aligned} \tag{9.128}$$

Case b If k_i and k_{i+q_1} are strictly positive and $k_{i+q_1+q_2}$ is equal to zero, Eq. (9.127) does **not** hold as $c_i < c_{i+q_1}$

$$\begin{aligned}
y^*_{\mathrm{MOM},\mathbf{2P},1} &= \frac{k_i c_i + k_{i+q_1} c_{i+q_1}}{k_i + k_{i+q_1}} \,, \\
y^*_{\mathrm{MOM},\mathbf{2P},2} &= c_i \,.
\end{aligned} \tag{9.129}$$

Case c If k_i and $k_{i+q_1+q_2}$ are strictly positive and k_{i+q_1} is equal to zero, Eq. (9.127) holds as

$$y^*_{\mathrm{MOM},\mathbf{2P},1} = \frac{k_i c_i + k_{i+q_1+q_2} c_{i+q_1+q_2}}{k_i + k_{i+q_1+q_2}} = y^*_{\mathrm{MOM},\mathbf{2P},2} \,. \tag{9.130}$$

Case d If k_i is strictly positive and k_{i+q_1} and $k_{i+q_1+q_2}$ are equal to zero, Eq. (9.127) holds as

$$y^*_{\mathrm{MOM},\mathbf{2P},1} = c_i = y^*_{\mathrm{MOM},\mathbf{2P},2} \,. \tag{9.131}$$

Case e If k_{i+q_1} and $k_{i+q_1+q_2}$ are strictly positive and k_i is equal to zero, Eq. (9.127) does **not** hold as $c_{i+q_1} < c_{i+q_1+q_2}$

$$\begin{aligned}
y^*_{\mathrm{MOM},\mathbf{2P},1} &= \frac{k_{i+q_1} c_{i+q_1} + k_{i+q_1+q_2} c_{i+q_1+q_2}}{k_{i+q_1} + k_{i+q_1+q_2}} \,, \\
y^*_{\mathrm{MOM},\mathbf{2P},2} &= c_{i+q_1+q_2} \,.
\end{aligned} \tag{9.132}$$

Case f If k_{i+q_1} is strictly positive and k_i and $k_{i+q_1+q_2}$ are equal to zero, Eq. (9.127) is equivalent with

$$\begin{aligned} & c_{i+q_1} = \frac{1}{2}(c_i + c_{i+q_1+q_2}) \\ \Leftrightarrow \quad & c_{i+q_1} - c_i = c_{i+q_1+q_2} - c_{i+q_1} \, . \end{aligned} \tag{9.133}$$

Case g If $k_{i+q_1+q_2}$ is strictly positive and k_i and k_{i+q_1} are equal to zero, Eq. (9.127) holds as

$$y^*_{\mathrm{MOM},\mathbf{2P},1} = c_{i+q_1+q_2} = y^*_{\mathrm{MOM},\mathbf{2P},2} \, . \tag{9.134}$$

Case h If k_i, k_{i+q_1} and $k_{i+q_1+q_2}$ are all equal to zero, Eq. (9.127) is equivalent with

$$\begin{aligned} & \frac{1}{3}(c_i + c_{i+q_1} + c_{i+q_1+q_2}) = \frac{1}{2}(c_i + c_{i+q_1+q_2}) \\ \Leftrightarrow \quad & c_{i+q_1} - c_i = c_{i+q_1+q_2} - c_{i+q_1} \, . \end{aligned} \tag{9.135}$$

As the model designer should have the freedom to apply the above set of six rules to any combination of three different linguistic output values A_i, A_{i+q_1} and $A_{i+q_1+q_2}$, fuzzy output partitions that contain a triangular membership function preceded by two or more trapezial membership functions as well as fuzzy output partitions that contain a triangular membership function followed by two or more trapezial membership functions should be discarded when monotonicity is required, as a non-monotone input-output behaviour is obtained for Cases b and e. These findings restrict the output membership functions to fuzzy partitions that satisfy one of the following conditions:

1. n trapezial membership functions,
2. n triangular membership functions,
3. trapezial membership functions A_1 and A_n and n-2 triangular membership functions A_i for $i \in \{2, \ldots, n\text{-}1\}$, or
4. one trapezial membership function and n-1 triangular membership functions.

In order to withhold the combination of the t-norm $T_{\mathbf{P}}$ and the MOM defuzzification method for the design of monotone models with a monotone non-smooth rule base, monotonicity should also be obtained for the set of rules used in Section 9.5.1 to support the recommendation to use a monotone smooth rule base when $T = T_{\mathbf{M}}$ and presented in a schematic way in Fig. 9.3. In the following paragraphs the chain of inequalities in Eq. (9.125) is investigated for the seven cases listed in Table 9.1.

Cases a and g If k_i, k_{i+q_1}, k_{i+q_2} and $k_{i+q_1+q_2}$ are all strictly positive or all equal to zero, all weights w_i, w_{i+q_1}, w_{i+q_2} and $w_{i+q_1+q_2}$ in Eq. (9.24) are strictly positive. In this case, the differences between the crisp outputs $y^*_{\mathrm{MOM},\mathbf{2P},i}$ are identical to those in Eqs. (9.122–9.123) and Eq. (9.124) proves that non-monotonicity is obtained in this situation.

Table 9.1: Cases to be considered when investigating the chain of inequalities in Eq. (9.125) for the rules represented in Fig. 9.3 and models applying $T_{\mathbf{P}}$.

	k_i	k_{i+q_1}	k_{i+q_2}	$k_{i+q_1+q_2}$		k_i	k_{i+q_1}	k_{i+q_2}	$k_{i+q_1+q_2}$
a	>0	>0	>0	>0	e	$=0$	$=0$	>0	$=0$
b	>0	$=0$	$=0$	>0	f	$=0$	$=0$	$=0$	>0
c	>0	$=0$	$=0$	$=0$	g	$=0$	$=0$	$=0$	$=0$
d	$=0$	>0	$=0$	$=0$					

Case b If k_i and $k_{i+q_1+q_2}$ are strictly positive and k_{i+q_1} and k_{i+q_2} are equal to zero, Eq. (9.125) holds since

$$y^*_{\mathrm{MOM},\mathbf{2P},1} = y^*_{\mathrm{MOM},\mathbf{2P},2} = y^*_{\mathrm{MOM},\mathbf{2P},3} = \frac{k_i c_i + k_{i+q_1+q_2} c_{i+q_1+q_2}}{k_i + k_{i+q_1+q_2}} \, . \tag{9.136}$$

Case c If k_i is strictly positive and k_{i+q_1}, k_{i+q_2} and $k_{i+q_1+q_2}$ are equal to zero, Eq. (9.125) holds since

$$y^*_{\mathrm{MOM},\mathbf{2P},1} = y^*_{\mathrm{MOM},\mathbf{2P},2} = y^*_{\mathrm{MOM},\mathbf{2P},3} = c_i \, . \tag{9.137}$$

Case d If k_{i+q_1} is strictly positive and k_i, k_{i+q_2} and $k_{i+q_1+q_2}$ are equal to zero, Eq. (9.125) does **not** hold. Indeed,

$$\begin{aligned} & y^*_{\mathrm{MOM},\mathbf{2P},1} \leq y^*_{\mathrm{MOM},\mathbf{2P},2} \\ \Leftrightarrow \quad & \frac{1}{3}(c_i + c_{i+q_2} + c_{i+q_1+q_2}) \leq \frac{1}{2}(c_i + c_{i+q_1+q_2}) \\ \Leftrightarrow \quad & 2c_{i+q_2} \leq c_i + c_{i+q_1+q_2} \, , \end{aligned} \tag{9.138}$$

and

$$\begin{aligned} & y^*_{\mathrm{MOM},\mathbf{2P},2} \leq y^*_{\mathrm{MOM},\mathbf{2P},3} \\ \Leftrightarrow \quad & \frac{1}{2}(c_i + c_{i+q_1+q_2}) \leq c_{i+q_1} \\ \Leftrightarrow \quad & c_i + c_{i+q_1+q_2} \leq 2c_{i+q_1} \, , \end{aligned} \tag{9.139}$$

together with $c_{i+q_1} < c_{i+q_2}$, imply that Eq. (9.125) does not hold.

Case e If k_{i+q_2} is strictly positive and k_i, k_{i+q_1} and $k_{i+q_1+q_2}$ are equal to zero, Eq. (9.125) does **not** hold. Indeed,

$$\begin{aligned} & y^*_{\mathrm{MOM},\mathbf{2P},1} \leq y^*_{\mathrm{MOM},\mathbf{2P},2} \\ \Leftrightarrow \quad & c_{i+q_2} \leq \frac{1}{2}(c_i + c_{i+q_1+q_2}) \\ \Leftrightarrow \quad & 2c_{i+q_2} \leq c_i + c_{i+q_1+q_2} \, , \end{aligned} \tag{9.140}$$

and

$$
\begin{aligned}
& y^*_{\mathrm{MOM},\mathbf{2P},2} \leq y^*_{\mathrm{MOM},\mathbf{2P},3} \\
\Leftrightarrow \quad & \frac{1}{2}(c_i + c_{i+q_1+q_2}) \leq \frac{1}{3}(c_i + c_{i+q_1} + c_{i+q_1+q_2}) \\
\Leftrightarrow \quad & c_i + c_{i+q_1+q_2} \leq 2c_{i+q_1} \,,
\end{aligned}
\tag{9.141}
$$

together with $c_{i+q_1} < c_{i+q_2}$, imply that Eq. (9.125) does not hold.

Case f If $k_{i+q_1+q_2}$ is strictly positive and k_i, k_{i+q_1} and k_{i+q_2} are equal to zero, Eq. (9.125) holds since

$$
y^*_{\mathrm{MOM},\mathbf{2P},1} = y^*_{\mathrm{MOM},\mathbf{2P},2} = y^*_{\mathrm{MOM},\mathbf{2P},3} = c_{i+q_1+q_2} \,. \tag{9.142}
$$

If all output membership functions are trapezial, the number of linguistic output values should be smaller than or equal to three, since for $q_1 = 1$ and $q_2 = 2$ non-monotonicity is obtained if the rule base contains six rules as in Fig. 9.3 and there are four consecutive trapezial membership functions (Case a). If the number of output membership functions is three, they should satisfy $k_1(c_2 - c_1) = k_3(c_3 - c_2)$ (Eq. (9.128)). If the number of output membership functions is smaller than three, no monotone, but non-smooth rule base can be constructed. Analogously, if all output membership functions are triangular, the number of linguistic output values should be smaller than or equal to three (Case g) and if the number of linguistic output values is equal to three, the membership functions should satisfy $c_2 - c_1 = c_3 - c_2$ (Eq. (9.135)). The results obtained for Cases a and g also restrict the number of linguistic output values for fuzzy output partitions with trapezial membership functions for the first and last membership function and triangular membership functions for the intermediate membership functions. In this case the number of linguistic output values should be at most five and if the number of linguistic output values is five the condition $c_3 - c_2 = c_4 - c_3$ (Eq. (9.135)) should be satisfied. Using the results obtained for Cases d and e, among the fuzzy output partitions consisting of only one trapezial membership function combined with triangular membership functions, only four types could be withheld: two types of fuzzy partitions with four membership functions, once with the first membership function and once with the fourth membership function being trapezial, and two types of fuzzy partitions with three membership functions with the trapezial membership function being either the first or the second membership functions. Summarizing, it can be concluded that monotonicity is obtained for the two sets of rules represented in Figs. 9.3 and 9.5 that might occur in a monotone, but non-smooth rule base, for fuzzy output partitions with five membership functions if the order of triangular and trapezial membership functions is

$$
\{\text{trapezial}, \text{triangular}, \text{triangular}, \text{triangular}, \text{trapezial}\} \quad \text{with} \quad c_3 - c_2 = c_4 - c_3
$$

with four membership functions if

$$\begin{array}{lll} \{\text{trapezial}, \text{triangular}, \text{triangular}, \text{trapezial}\} & & \\ \{\text{trapezial}, \text{triangular}, \text{triangular}, \text{triangular}\} & \text{with} & c_3 - c_2 = c_4 - c_3 \\ \{\text{triangular}, \text{triangular}, \text{triangular}, \text{trapezial}\} & \text{with} & c_2 - c_1 = c_3 - c_2 \end{array}$$

and three membership functions if

$$\begin{array}{lll} \{\text{trapezial}, \text{trapezial}, \text{trapezial}\} & \text{with} & k_1(c_2 - c_1) = k_3(c_3 - c_2) \\ \{\text{triangular}, \text{triangular}, \text{triangular}\} & \text{with} & c_2 - c_1 = c_3 - c_2 \\ \{\text{trapezial}, \text{triangular}, \text{trapezial}\} & & \\ \{\text{trapezial}, \text{triangular}, \text{triangular}\} & & \\ \{\text{triangular}, \text{trapezial}, \text{triangular}\} & \text{with} & c_2 - c_1 = c_3 - c_2 \end{array}$$

Moreover, it can be proved that for these types of fuzzy partitions a monotone input-output behaviour is obtained for any monotone rule base.

9.6 Conclusion

In this chapter, it was proved that a Mamdani–Assilian model applying the MOM defuzzification method is monotone if it corresponds to one of the six model types listed in Table 9.2, characterized by a number of input variables m, a t-norm T, an either monotone or monotone smooth rule base and additional properties of the membership functions appearing in the rule consequents. For the t-norm $T_{\mathbf{M}}$, models with a single input variable show a monotone input-output behaviour for any monotone rule base when the linguistic output values in the consequents of the rules are defined by trapezial or triangular membership functions with intervals of changing membership degree of equal length, whereas for the t-norms $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$, models with a single input variable show a monotone input-output behaviour for any monotone rule base and any fuzzy output partition. The monotonicity of models with two input variables applying $T_{\mathbf{P}}$ is only guaranteed for any monotone rule base when using one of the nine types of fuzzy output partitions defined in Table 9.3. Finally, it is proved that a monotone input-output behaviour is always obtained for models with a monotone smooth rule base applying $T_{\mathbf{M}}$ when the linguistic output values in the consequents of the rules are defined by trapezial or triangular membership functions with intervals of changing membership degree of equal length and for models with a monotone smooth rule base applying $T_{\mathbf{P}}$ for any fuzzy output partition.

The interpolation procedure presented in Section 8.6 between a user-defined fuzzy output partition and a second fuzzy partition satisfying the constraints required to guarantee monotonicity, can also be incorporated in models applying the MOM defuzzification method. Therefore, monotonicity can be guaranteed for all models with a monotone rule base, one input variable and applying either $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$, as well as for all models with a monotone smooth rule base, an unrestricted number of input variables and applying either $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$, regardless of the fuzzy output partition.

Table 9.2: Mamdani–Assilian models for which monotonicity is guaranteed if applying the MOM defuzzification method characterized by a number of input variables m, a t-norm T, an either monotone or monotone smooth rule base and additional properties of the membership functions appearing in the rule consequents.

	m	T	rule base	additional properties A_{i_s}
1	1	$T_{\mathbf{M}}$	monotone	$(\forall s \in \{1,\ldots,r\})(i_s \notin \{1,n\})$ $(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)$
2	1	$T_{\mathbf{P}}$	monotone	
3	1	$T_{\mathbf{L}}$	monotone	
4	2	$T_{\mathbf{P}}$	monotone	restricted class of nine types of fuzzy partitions given in Table 9.3
5		$T_{\mathbf{M}}$	monotone and smooth	$(\forall s \in \{1,\ldots,r\})(i_s \notin \{1,n\})$ $(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)$
6		$T_{\mathbf{P}}$	monotone and smooth	

Table 9.3: Characteristics of the nine partitions for which monotonicity can be guaranteed for models with two input variables and a monotone rule base, applying $T_{\mathbf{P}}$ and the MOM defuzzification method.

n	k_1	k_2	k_3	k_4	k_5	additional properties A_{i_s}
5	>0	0	0	0	>0	$l_2 = l_3$
4	>0	0	0	>0		
4	>0	0	0	0		$l_2 = l_3$
4	0	0	0	>0		$l_1 = l_2$
3	>0	>0	>0			$k_1 l_1 = k_3 l_2$
3	0	0	0			$l_1 = l_2$
3	>0	0	>0			
3	>0	0	0			
3	0	>0	0			$l_1 = l_2$

However, for models with two input variables and a monotone non-smooth rule base applying $T_{\mathbf{P}}$, the interpolation procedure does not allow the user to use any fuzzy output partition. The nine types of fuzzy partitions in Table 9.3 for which monotonicity is guaranteed cannot be extended to all fuzzy partitions as in the nine types of fuzzy partitions

1. the maximum number of output membership functions is five and,
2. the second to second last membership functions in all partitions with four or five membership functions are triangular.

The relationship between the crisp output y'^* returned by the inference procedure using the second fuzzy partition and the crisp output y^* in the output domain defined by the model designer is functional, *i.e.* it maps each crisp output y'^* to one crisp output y^*. If a trapezial membership function in the user-defined fuzzy partition is represented by a triangular membership function in the second partition, only one value within its kernel can be returned as model output y^* rendering the other elements of the kernel of this trapezial membership function redundant and leading to an (additional) discontinuity in the model output.

Thus, if the monotonicity of a model with two input variables, a monotone rule base and applying $T_{\mathbf{P}}$ should be guaranteed and one wants to use four or five linguistic output values, one is only free to choose the shape, either triangular or trapezial, of the extreme membership functions, and should define the intermediate linguistic values by triangular membership functions. If one wants to use only three linguistic output values, the shape of all output membership functions can be chosen freely as a fuzzy partition of three trapezial membership functions guaranteeing monotonicity can be used to determine y'^*. One easily verifies that when the second fuzzy partition is chosen such that all trapezial membership functions have kernels of equal length, and that all membership functions have intervals of changing membership degree of equal length as well, then the class of user-defined fuzzy output partitions for which monotonicity is guaranteed, for models with a monotone rule base, two input variables and applying the t-norm $T_{\mathbf{P}}$ can be summarized as { *, triangular, triangular, triangular, * }, { *, triangular, triangular, * } or { *, *, * } with * a membership function that might be either triangular or trapezial.

CHAPTER 10

ATL–ATM models

Le seul véritable voyage n'est pas d'aller vers d'autres paysages, mais d'avoir d'autres yeux.
(Marcel Proust)

10.1 Introduction

In this chapter the monotonicity is investigated of linguistic fuzzy models applying plain implicator-based inference or ATL–ATM inference. It is verified for the three t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ and the three R-implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$ whether a monotone input-output behaviour is obtained for any monotone rule base, or at least for any monotone smooth rule base. The models are assumed to hold additional properties apart from the properties described in Section 7.2: the linguistic output values in the rule consequents are assumed to be defined by trapezial or triangular membership functions of identical shape, *i.e.*

$$(\forall s \in \{1, \ldots, r\})(i_s \notin \{1, n\})\,, \tag{10.1}$$

$$(\exists l > 0)(\forall i \in I \setminus \{n\})(l_i = l)\,, \tag{10.2}$$

$$(\exists k \geq 0)(\forall i \in I \setminus \{1, n\})(k_i = k)\,. \tag{10.3}$$

However, the auxiliary interpolation procedure described in Section 8.6 allows to extend the results obtained in this chapter to any fuzzy output partition as defined in Section 7.2.1.

In Section 10.2 the monotonicity of linguistic fuzzy models with a monotone rule base applying implicator-based inference without using the ATL and ATM modifiers is described. As for this *plain* implicator-based inference procedure monotonicity cannot be guaranteed for models with two or more input variables, but only for models with a single input variable and a smooth monotone rule base, this section justifies the introduction of the new implicator-based inference procedure.

Since the ATL–ATM inference procedure has not been described in literature, Section 10.3 is dedicated to some general remarks on the new inference procedure.

Sections 10.4–10.6 discuss ATL–ATM models with up to two input variables, applying the t-norm $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$, the R-implicator $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ and the Mean of Maxima (MOM) defuzzification method. In Section 10.4, the monotonicity of models with a single input variable is studied for the R-implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$. In Section 10.5, the monotonicity of models with two input variables and a monotone smooth rule base is discussed for the nine combinations of the t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ and the three implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$. As the research pointed out that when applying the implicator $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ monotonicity cannot be guaranteed for models with a single input variable and any monotone rule base, Section 10.6 deals with the monotonicity of models with two input variables and a monotone rule base for the implicator $I_{\mathbf{L}}$ only. The chapter concludes with a summary of the obtained results in Section 10.7.

10.2 Motivation for the use of ATL and ATM modifiers

As discussed in Section 7.4 a prerequisite for a monotone model is to return a non-empty fuzzy output different from the universal set for any input vector $\mathbf{x}$. In the counterexample below it is shown that obtaining a meaningful fuzzy output for any input vector $\mathbf{x}$ cannot be guaranteed for models with two input variables and any monotone (smooth) rule base when applying plain implicator-based inference.

The set of four rules

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_{i+1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+2}

can occur in a monotone smooth rule base as well as in a monotone non-smooth rule base. For an input vector $\mathbf{x} = (x_1, x_2)$ not firing any other rule than the four rules above

$$\gamma_1 = 1 - B^1_{j_1}(x_1) = B^1_{j_1+1}(x_1)\,, \tag{10.4}$$
$$\gamma_2 = 1 - B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2)\,, \tag{10.5}$$

the fulfilment degrees of the linguistic output values A_i, A_{i+1} and A_{i+2} are obtained by

$$\alpha_i = T(1-\gamma_1, 1-\gamma_2)\,, \tag{10.6}$$
$$\alpha_{i+1} = \max(T(1-\gamma_1, \gamma_2), T(\gamma_1, 1-\gamma_2))\,, \tag{10.7}$$
$$\alpha_{i+2} = T(\gamma_1, \gamma_2)\,, \tag{10.8}$$

with the t-norm T either $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$. The t-norm $T_{\mathbf{L}}$ is not taken into consideration as for $\gamma_1 = \gamma_2 = 0.5$, the fulfilment degrees of all linguistic output values are equal to zero and the universal set is obtained as fuzzy model output (see Section 7.4.2).

In the following paragraphs the model behaviour for input vectors $\mathbf{x}$ satisfying Eqs. (10.4)–(10.5) is first discussed for the implicators $I_{\mathbf{M}}$ and $I_{\mathbf{P}}$ and then for the implicator $I_{\mathbf{L}}$. Before starting the discussion, the reader is reminded that in this study, a linguistic output value A_i originates from a fuzzy partition of trapezial membership functions as shown in Fig. 7.1 and its support and kernel are given by (note that $i \in I \setminus \{1, n\}$)

$$\mathrm{supp}(A_i) =]a_{2i\text{-}2}, a_{2i+1}[\qquad \mathrm{kern}(A_i) = [a_{2i\text{-}1}, a_{2i}] \,. \tag{10.9}$$

Models applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ When a linguistic output value A_i is fired, *i.e.* if its fulfilment degree α_i is strictly positive, only output values y belonging to the support of A_i have a non-zero membership degree to the adapted membership function A'_i when using the implicator $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(y \leq a_{2i\text{-}2} \Rightarrow I_T(\alpha_i, A_i(y)) = 0) \,, \tag{10.10}$$
$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(a_{2i\text{-}2} < y < a_{2i+1} \Rightarrow I_T(\alpha_i, A_i(y)) > 0) \,, \tag{10.11}$$
$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(y \geq a_{2i+1} \Rightarrow I_T(\alpha_i, A_i(y)) = 0) \,. \tag{10.12}$$

Thus, when the adapted output membership functions are obtained with $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$, the minimum of the membership degrees to the adapted membership functions of two fired, non-consecutive linguistic output values A_i and A_{i+p} ($p > 1$) is equal to zero for all output values y

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(p > 1 \Rightarrow \min(I_T(\alpha_i, A_i(y)), I_T(\alpha_{i+p}, A_{i+p}(y))) = 0) \,, \tag{10.13}$$

since

$$a_{2i+1} \leq a_{2i+2p\text{-}2} \,, \tag{10.14}$$

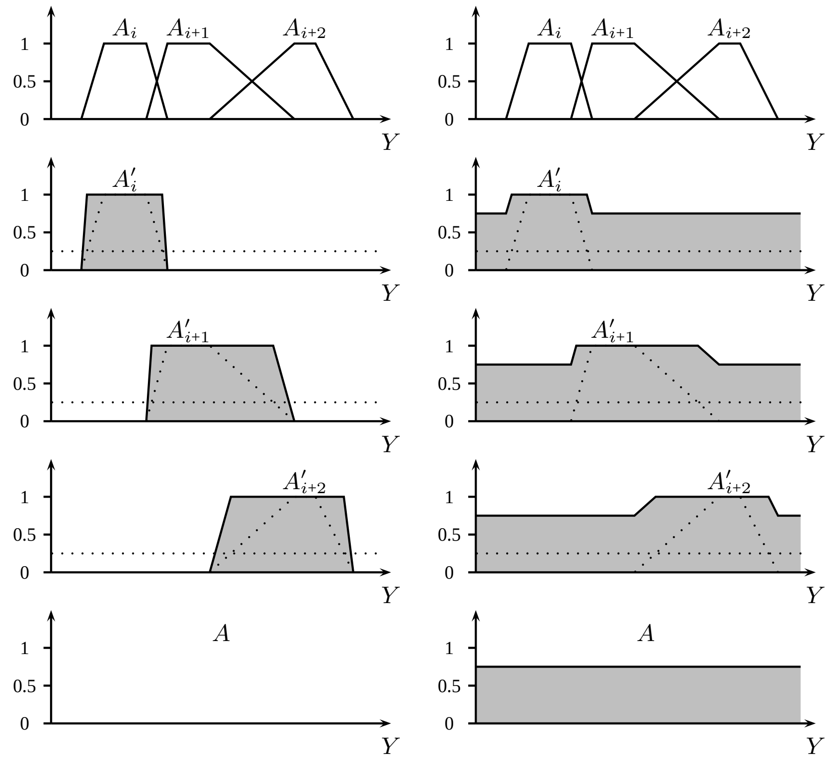
and

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(y \leq a_{2i+2p\text{-}2} \Rightarrow I_T(\alpha_{i+p}, A_{i+p}(y)) = 0) \,, \tag{10.15}$$
$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(y \geq a_{2i+1} \Rightarrow I_T(\alpha_i, A_i(y)) = 0) \,. \tag{10.16}$$

When applying plain implicator-based inference, adapted membership functions of linguistic output values that are not fired are identical to the universal set and the membership degree to the global fuzzy output A is the minimum of the membership degrees to the n adapted membership functions A'_i. Therefore, for input vectors $\mathbf{x}$ firing two non-consecutive linguistic output values, as for example the input vectors $\mathbf{x}$ firing the four rules above, the empty set is obtained as fuzzy output. In Fig. 10.1(a) the adapted membership functions A'_i, A'_{i+1} and A'_{i+2} as well as the fuzzy output A are shown for a model applying the t-norm $T_{\mathbf{P}}$ and the implicator $I_{\mathbf{P}}$, and $\gamma_1 = \gamma_2 = 0.5$.

Models applying $I_{\mathbf{L}}$ When a linguistic output value A_i is fired, *i.e.* if its fulfilment degree α_i is strictly positive, the membership degree of an output value y not belonging to the support of A_i is equal to $1 - \alpha_i$, while the membership degree of an output value y belonging to the support of A_i is greater than $1 - \alpha_i$ when using the implicator $I_{\mathbf{L}}$

$$(\forall y \in \mathbf{Y})(y \leq a_{2i\text{-}2} \Rightarrow I_{\mathbf{L}}(\alpha_i, A_i(y)) = 1 - \alpha_i) \,, \tag{10.17}$$
$$(\forall y \in \mathbf{Y})(a_{2i\text{-}2} < y < a_{2i+1} \Rightarrow I_{\mathbf{L}}(\alpha_i, A_i(y)) > 1 - \alpha_i) \,, \tag{10.18}$$
$$(\forall y \in \mathbf{Y})(y \geq a_{2i+1} \Rightarrow I_{\mathbf{L}}(\alpha_i, A_i(y)) = 1 - \alpha_i) \,. \tag{10.19}$$



(a) $I_{\mathbf{P}}$ (b) $I_{\mathbf{L}}$

Figure 10.1: Fuzzy outputs obtained for input vectors $\mathbf{x}$ ($\gamma_1 = \gamma_2 = 0.5$) firing the four rules considered in the discussion about models applying plain implicator-based inference. The t-norm $T_{\mathbf{P}}$ and implicators (a) $I_{\mathbf{P}}$ and (b) $I_{\mathbf{L}}$ were applied.

Thus, when the adapted output membership functions are obtained with $I_{\mathbf{L}}$, the minimum of the membership degrees to the adapted membership functions of two non-consecutive linguistic output values A_i and A_{i+p} ($p > 1$) which are fired to a same non-zero membership degree α_i is equal to $1 - \alpha_i$ for all output values y

$$(\forall y \in \mathbf{Y})(p > 1 \Rightarrow \min(I_{\mathbf{L}}(\alpha_i, A_i(y)), I_{\mathbf{L}}(\alpha_i, A_{i+p}(y))) = 1 - \alpha_i)\,, \tag{10.20}$$

since

$$a_{2i+1} \leq a_{2i+2p\text{-}2}\,, \tag{10.21}$$

and

$$(\forall y \in \mathbf{Y})(y \leq a_{2i+2p\text{-}2} \Rightarrow (I_{\mathbf{L}}(\alpha_i, A_i(y)) \geq 1 - \alpha_i \wedge I_{\mathbf{L}}(\alpha_i, A_{i+p}(y)) = 1 - \alpha_i))\,, \tag{10.22}$$

$$(\forall y \in \mathbf{Y})(y \geq a_{2i+1} \Rightarrow (I_{\mathbf{L}}(\alpha_i, A_{i+p}(y)) \geq 1 - \alpha_i \wedge I_{\mathbf{L}}(\alpha_i, A_i(y)) = 1 - \alpha_i))\,. \tag{10.23}$$

As for input vectors $\mathbf{x}$ firing the four rules above, the same fulfilment degrees are obtained for the linguistic output values A_i, A_{i+1} and A_{i+2} when $\gamma_1 = \gamma_2 = 0.5$

$$\alpha_i = \alpha_{i+1} = \alpha_{i+2} = T(0.5, 0.5)\,, \tag{10.24}$$

the fuzzy output obtained in this case is as meaningless as the empty set or the universal set since all linguistic output values y have a same membership degree $1 - T(0.5, 0.5)$ (0.5 for $T_{\mathbf{M}}$ and 0.75 for $T_{\mathbf{P}}$) to the fuzzy output A. In Fig. 10.1(b) the adapted membership functions A'_i, A'_{i+1} and A'_{i+2} as well as the fuzzy output A are shown for a model applying the t-norm $T_{\mathbf{P}}$ and the implicator $I_{\mathbf{L}}$, and $\gamma_1 = \gamma_2 = 0.5$.

Conclusion From the discussion above, summarized in Eqs. (10.13) and (10.20), it follows that for models with a single input variable and a monotone non-smooth rule base a constant fuzzy set is obtained for some input vectors $\mathbf{x}$, either the empty set for models applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ or a fuzzy set to which all linguistic output values have a same membership degree for models applying $I_{\mathbf{L}}$, since these models contain a set of rules corresponding to

IF	X_1 IS B^1_j	THEN	Y IS A_i
IF	X_1 IS B^1_{j+1}	THEN	Y IS A_{i+p}

with $p > 1$. One can easily verify that models with a single input variable and a monotone smooth rule base always return a non-empty fuzzy output for the three considered implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$. As in practice, models with a single input variable are of minor importance, monotonicity aspects of models applying *plain* implicator-based inference were not investigated in more detail in this study.

10.3 Some general remarks on ATL–ATM models

10.3.1 Adapted output membership functions

From the definition of the modifiers ATL and ATM

$$\mathrm{ATL}(A)(x) = \sup\{A(t) \mid t \leq x\}\,, \tag{10.25}$$

$$\mathrm{ATM}(A)(x) = \sup\{A(t) \mid t \geq x\}\,, \tag{10.26}$$

it follows that the original membership functions $\mathrm{ATL}(A_i)$ and $\mathrm{ATM}(A_i)$ in the consequents of the rules of ATL and ATM models are defined by respectively increasing and decreasing membership functions. In ATL and ATM models implicator-based inference is applied: the adapted output membership functions $(\mathrm{ATL}(A_i))'$ and $(\mathrm{ATM}(A_i))'$ are given by

$$(\mathrm{ATL}(A_i))'(y) = I_T(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y))\,, \tag{10.27}$$

$$(\mathrm{ATM}(A_i))'(y) = I_T(\alpha_{\mathrm{ATM},i}, \mathrm{ATM}(A_i)(y))\,. \tag{10.28}$$

As the three implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$ considered in this dissertation satisfy

$$(\forall x, y, z \in [0,1])(y \leq z \Rightarrow I_T(x,y) \leq I_T(x,z))\,, \tag{10.29}$$

the adapted output membership functions $(\mathrm{ATL}(A_i))'$ and $(\mathrm{ATM}(A_i))'$ are respectively increasing and decreasing functions.

When a linguistic output value $\mathrm{ATL}(A_i)$ (resp. $\mathrm{ATM}(A_i)$) is not fired, *i.e.* if its fulfilment degree $\alpha_{\mathrm{ATL},i}$ (resp. $\alpha_{\mathrm{ATM},i}$) is equal to zero, the corresponding adapted membership function $(\mathrm{ATL}(A_i))'$ (resp. $(\mathrm{ATM}(A_i))'$) obtained with $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ is the universal set

$$I_T(0, \mathrm{ATL}(A_i)(y)) = 1\,, \tag{10.30}$$

$$I_T(0, \mathrm{ATM}(A_i)(y)) = 1\,, \tag{10.31}$$

whereas, when a linguistic output value $\mathrm{ATL}(A_i)$ (resp. $\mathrm{ATM}(A_i)$) has a fulfilment degree $\alpha_{\mathrm{ATL},i}$ (resp. $\alpha_{\mathrm{ATM},i}$) equal to one, the corresponding adapted membership function $(\mathrm{ATL}(A_i))'$ (resp. $(\mathrm{ATM}(A_i))'$) obtained with $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ is identical to the original membership function

$$I_T(1, \mathrm{ATL}(A_i)(y)) = \mathrm{ATL}(A_i)(y)\,, \tag{10.32}$$

$$I_T(1, \mathrm{ATM}(A_i)(y)) = \mathrm{ATM}(A_i)(y)\,. \tag{10.33}$$

In this study, a linguistic output value A_i originates from a fuzzy partition of trapezial membership functions as shown in Fig. 7.1 and the supports and kernels of the corresponding linguistic values $\mathrm{ATL}(A_i)$ and $\mathrm{ATM}(A_i)$ are given by

$$\mathrm{supp}(\mathrm{ATL}(A_i)) = \,]a_{2i\text{-}2}, +\infty[\qquad \mathrm{kern}(\mathrm{ATL}(A_i)) = [a_{2i\text{-}1}, +\infty[\;, \tag{10.34}$$

$$\mathrm{supp}(\mathrm{ATM}(A_i)) = \,]-\infty, a_{2i+1}[\qquad \mathrm{kern}(\mathrm{ATM}(A_i)) = \,]-\infty, a_{2i}]\;. \tag{10.35}$$

When a linguistic output value $\mathrm{ATL}(A_i)$ (resp. $\mathrm{ATM}(A_i)$) is fired, *i.e.* if its fulfilment degree $\alpha_{\mathrm{ATL},i}$ (resp. $\alpha_{\mathrm{ATM},i}$) is strictly positive, only output values y belonging to the support of $\mathrm{ATL}(A_i)$ (resp. $\mathrm{ATM}(A_i)$) have a non-zero membership degree to the adapted membership function $(\mathrm{ATL}(A_i))'$ (resp. $(\mathrm{ATM}(A_i))'$) when using $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ as implicator

$$\begin{aligned}&(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(\forall \alpha_{\mathrm{ATL},i} > 0)\\ &\quad (y \leq a_{2i\text{-}2} \Rightarrow I_T(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) = 0)\,,\end{aligned} \tag{10.36}$$

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(y > a_{2i\text{-}2} \Rightarrow I_T(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) > 0)\,, \tag{10.37}$$

respectively,

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_\mathbf{M}, I_\mathbf{P}\})(\forall \alpha_{\mathrm{ATM},i} > 0)$$
$$(y \geq a_{2i+1} \Rightarrow I_T(\alpha_{\mathrm{ATM},i}, \mathrm{ATM}(A_i)(y)) = 0)\,, \tag{10.38}$$
$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_\mathbf{M}, I_\mathbf{P}\})(y < a_{2i+1} \Rightarrow I_T(\alpha_{\mathrm{ATM},i}, \mathrm{ATM}(A_i)(y)) > 0)\,. \tag{10.39}$$

When applying the implicator $I_\mathbf{L}$, the membership degree to the adapted membership function $(\mathrm{ATL}(A_i))'$ (resp. $(\mathrm{ATM}(A_i))'$) with $\alpha_{\mathrm{ATL},i}$ (resp. $\alpha_{\mathrm{ATM},i}$) being strictly positive, of an output value y not belonging to the support of $\mathrm{ATL}(A_i)$ (resp. $\mathrm{ATM}(A_i)$) is given by $1 - \alpha_{\mathrm{ATL},i}$ (resp. $1 - \alpha_{\mathrm{ATM},i}$), while the membership degree of an output value y belonging to the support of $\mathrm{ATL}(A_i)$ (resp. $\mathrm{ATM}(A_i)$) is greater than $1 - \alpha_{\mathrm{ATL},i}$ (resp. $1 - \alpha_{\mathrm{ATM},i}$)

$$(\forall y \in \mathbf{Y})(y \leq a_{2i\text{-}2} \Rightarrow I_\mathbf{L}(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) = 1 - \alpha_{\mathrm{ATL},i})\,, \tag{10.40}$$
$$(\forall y \in \mathbf{Y})(\forall \alpha_{\mathrm{ATL},i} > 0)(y > a_{2i\text{-}2} \Rightarrow I_\mathbf{L}(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) > 1 - \alpha_{\mathrm{ATL},i})\,, \tag{10.41}$$

respectively,

$$(\forall y \in \mathbf{Y})(y \geq a_{2i+1} \Rightarrow I_\mathbf{L}(\alpha_{\mathrm{ATM},i}, \mathrm{ATM}(A_i)(y)) = 1 - \alpha_{\mathrm{ATM},i})\,, \tag{10.42}$$
$$(\forall y \in \mathbf{Y})(\forall \alpha_{\mathrm{ATM},i} > 0)(y < a_{2i+1} \Rightarrow I_\mathbf{L}(\alpha_{\mathrm{ATM},i}, \mathrm{ATM}(A_i)(y)) > 1 - \alpha_{\mathrm{ATM},i})\,. \tag{10.43}$$

Finally, since it holds for the three considered implicators I_T that

$$(\forall\, x \in [0,1])(I_T(x,1) = 1)\,, \tag{10.44}$$

output values belonging to the kernel of $\mathrm{ATL}(A_i)$ (resp. $\mathrm{ATM}(A_i)$) also belong to the kernel of the adapted membership function $(\mathrm{ATL}(A_i))'$ (resp. $(\mathrm{ATM}(A_i))'$)

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_\mathbf{M}, I_\mathbf{P}, I_\mathbf{L}\})(y \geq a_{2i\text{-}1} \Rightarrow I_T(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) = 1)\,, \tag{10.45}$$
$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_\mathbf{M}, I_\mathbf{P}, I_\mathbf{L}\})(y \leq a_{2i} \Rightarrow I_T(\alpha_{\mathrm{ATM},i}, \mathrm{ATM}(A_i)(y)) = 1)\,. \tag{10.46}$$

10.3.2 Fuzzy output of the ATL and ATM model

The fuzzy output of ATL and ATM models is the intersection of the individual adapted membership functions $(\mathrm{ATL}(A_i))'$ and $(\mathrm{ATM}(A_i))'$, *i.e.*

$$A_{\mathrm{ATL}}(y) = \min_{i=1}^{n} (\mathrm{ATL}(A_i))'(y)\,, \tag{10.47}$$
$$A_{\mathrm{ATM}}(y) = \min_{i=1}^{n} (\mathrm{ATM}(A_i))'(y)\,. \tag{10.48}$$

As $(\mathrm{ATL}(A_i))'$ and $(\mathrm{ATM}(A_i))'$ are respectively increasing and decreasing functions, it follows from their definition that A_{ATL} and A_{ATM} are respectively increasing and decreasing functions.

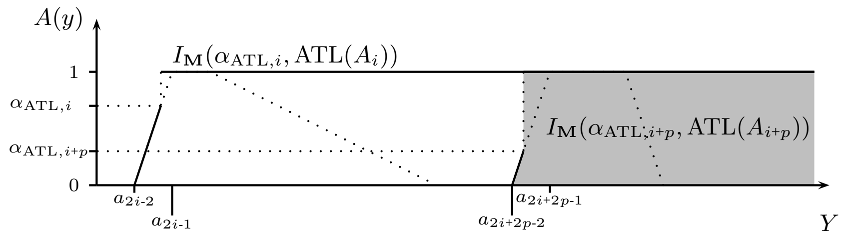


Figure 10.2: Illustration of the property defined in Eq. (10.49) of adapted membership functions in an ATL model obtained with $I_{\mathbf{M}}$ (or $I_{\mathbf{P}}$).

10.3.2.1 Models applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$

In the following paragraphs it is shown that for a strictly positive fulfilment degree $\alpha_{\mathrm{ATL},i+p}$ and the implicators $I_{\mathbf{M}}$ and $I_{\mathbf{P}}$, the membership degree of any output value y to an adapted linguistic value $(\mathrm{ATL}(A_i))'$ is greater than or equal to its membership degree to an adapted linguistic value $(\mathrm{ATL}(A_{i+p}))'$, *i.e.*

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(\forall \alpha_{\mathrm{ATL},i+p} > 0)$$
$$(p \geq 1 \Rightarrow I_T(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) \geq I_T(\alpha_{\mathrm{ATL},i+p}, \mathrm{ATL}(A_{i+p})(y)))\,. \quad (10.49)$$

This property is illustrated in Fig. 10.2.

As A_i and A_{i+p} originate from a same fuzzy partition as shown in Fig. 7.1 the parameters defining the corresponding membership functions satisfy

$$a_{2i\text{-}1} \leq a_{2i+2p\text{-}2}\,. \quad (10.50)$$

From Eq. (10.36) it follows that for all values smaller than or equal to the lower bound of the support of $\mathrm{ATL}(A_{i+p})$, Eq. (10.49) holds. Since

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(\forall \alpha_{\mathrm{ATL},i+p} > 0)$$
$$(y \leq a_{2i+2p\text{-}2} \Rightarrow I_T(\alpha_{\mathrm{ATL},i+p}, \mathrm{ATL}(A_{i+p})(y)) = 0)\,, \quad (10.51)$$

it also holds that

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(\forall \alpha_{\mathrm{ATL},i+p} > 0)$$
$$(y \leq a_{2i+2p\text{-}2} \Rightarrow I_T(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) \geq I_T(\alpha_{\mathrm{ATL},i+p}, \mathrm{ATL}(A_{i+p})(y)))\,. \quad (10.52)$$

From Eq. (10.45) it follows that for all values larger than or equal to the lower bound of the kernel of $\mathrm{ATL}(A_i)$, Eq. (10.49) holds. Since

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\})(y \geq a_{2i\text{-}1} \Rightarrow I_T(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) = 1)\,, \quad (10.53)$$

it also holds that

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_\mathbf{M}, I_\mathbf{P}\})$$
$$(y \geq a_{2i\text{-}1} \Rightarrow I_T(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) \geq I_T(\alpha_{\mathrm{ATL},i+p}, \mathrm{ATL}(A_{i+p})(y)))\,. \tag{10.54}$$

Since the lower bound $a_{2i\text{-}1}$ of the kernel of $\mathrm{ATL}(A_i)$ is smaller than or equal to the lower bound $a_{2i+2p\text{-}2}$ of the support of $\mathrm{ATL}(A_{i+p})$ it follows from Eqs. (10.52) and (10.54) that Eq. (10.49) holds.

Analogously, for a strictly positive fulfilment degree $\alpha_{\mathrm{ATM},i}$, $p \geq 1$ and the implicators $I_\mathbf{M}$ and $I_\mathbf{P}$, the membership degree of any output value y to an adapted linguistic value $(\mathrm{ATM}(A_i))'$ is smaller than or equal to its membership degree to an adapted linguistic value $(\mathrm{ATM}(A_{i+p}))'$, *i.e.*

$$(\forall y \in \mathbf{Y})(\forall I_T \in \{I_\mathbf{M}, I_\mathbf{P}\})(\forall \alpha_{\mathrm{ATM},i} > 0)$$
$$(p \geq 1 \Rightarrow I_T(\alpha_{\mathrm{ATM},i}, \mathrm{ATM}(A_i)(y)) \leq I_T(\alpha_{\mathrm{ATM},i+p}, \mathrm{ATM}(A_{i+p})(y)))\,. \tag{10.55}$$

Since the ATL–ATM inference procedure is an implicator-based inference procedure, the adapted membership functions of linguistic output values that are not fired are identical to the universal set and do not contribute to the global fuzzy output if there exists an adapted membership function which is different from the universal set. In Section 7.3.2 it is shown that for any input vector $\mathbf{x}$ at least one rule of an ATL (resp. ATM) model is fired to a fulfilment degree equal to one and from Eqs. (10.32–10.33) it follows that the adapted linguistic output value corresponding to this rule is not defined by the universal set. From Eq. (10.49) it then follows that, when applying $I_\mathbf{M}$ or $I_\mathbf{P}$, the fuzzy output A_{ATL} of the ATL model is given by the adapted membership function $(\mathrm{ATL}(A_{i_{\mathrm{max,zero}}}))'$ of the linguistic output value with the largest index $i_{\mathrm{max,zero}}$ among all fired linguistic output values, *i.e.*

$$(\forall I_T \in \{I_\mathbf{M}, I_\mathbf{P}\})(A_{\mathrm{ATL}}(y) = I_T(\alpha_{\mathrm{ATL},i_{\mathrm{max,zero}}}, \mathrm{ATL}(A_{\mathrm{max,zero}})(y)))\,, \tag{10.56}$$

with

$$i_{\mathrm{max,zero}} = \max\{i \in I \mid \alpha_{\mathrm{ATL},i} > 0\}\,. \tag{10.57}$$

Analogously, it follows from Eq. (10.55) that the fuzzy output A_{ATM} of the ATM model is given by the adapted membership function $(\mathrm{ATM}(A_{i_{\mathrm{min,zero}}}))'$ of the linguistic output value with the smallest index $i_{\mathrm{min,zero}}$ among all fired linguistic output values, *i.e.*

$$(\forall I_T \in \{I_\mathbf{M}, I_\mathbf{P}\})(A_{\mathrm{ATM}}(y) = I_T(\alpha_{\mathrm{ATM},i_{\mathrm{min,zero}}}, \mathrm{ATM}(A_{\mathrm{min,zero}})(y)))\,, \tag{10.58}$$

with

$$i_{\mathrm{min,zero}} = \min\{i \in I \mid \alpha_{\mathrm{ATM},i} > 0\}\,. \tag{10.59}$$

Thus, given Eqs. (10.36–10.39), only output values smaller than or equal to the lower bound of the support of $\mathrm{ATL}(A_{i_{\mathrm{max,zero}}})$ do not belong to the fuzzy output A_{ATL}

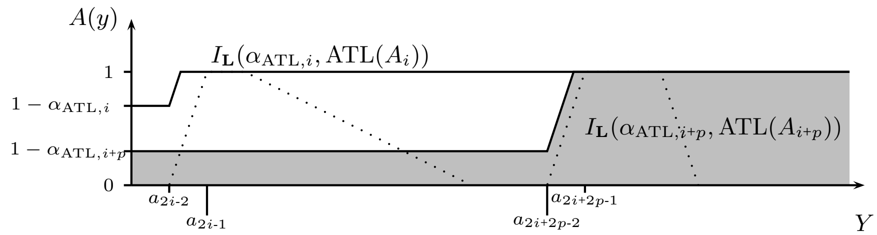


Figure 10.3: Illustration of the property defined in Eq. (10.64) of adapted membership functions in an ATL model obtained with $I_{\mathbf{L}}$.

of the ATL model, *i.e.*

$$(\forall y\in\mathbf{Y})(\forall I_T\in\{I_{\mathbf{M}},I_{\mathbf{P}}\})(y\le a_{2i_{\mathrm{max,zero}}\text{-}2}\Rightarrow A_{\mathrm{ATL}}(y)=0)\,, \tag{10.60}$$
$$(\forall y\in\mathbf{Y})(\forall I_T\in\{I_{\mathbf{M}},I_{\mathbf{P}}\})(y> a_{2i_{\mathrm{max,zero}}\text{-}2}\Rightarrow A_{\mathrm{ATL}}(y)>0)\,, \tag{10.61}$$

and only output values greater than or equal to the upper bound of the support of $\mathrm{ATM}(A_{i_{\mathrm{min,zero}}})$ do not belong to the fuzzy output A_{ATM} of the ATM model, *i.e.*

$$(\forall y\in\mathbf{Y})(\forall I_T\in\{I_{\mathbf{M}},I_{\mathbf{P}}\})(y\ge a_{2i_{\mathrm{min,zero}}+1}\Rightarrow A_{\mathrm{ATM}}(y)=0)\,, \tag{10.62}$$
$$(\forall y\in\mathbf{Y})(\forall I_T\in\{I_{\mathbf{M}},I_{\mathbf{P}}\})(y> a_{2i_{\mathrm{min,zero}}+1}\Rightarrow A_{\mathrm{ATM}}(y)>0)\,. \tag{10.63}$$

10.3.2.2 Models applying $I_{\mathbf{L}}$

In the following paragraphs it is shown that when applying $I_{\mathbf{L}}$ the membership degree of any output value y to an adapted linguistic value $(\mathrm{ATL}(A_i))'$ is greater than or equal to its membership degree to an adapted linguistic value $(\mathrm{ATL}(A_{i+p}))'$ if the linguistic value $(\mathrm{ATL}(A_i))$ is fired to a smaller or the same fulfilment degree as the linguistic value $(\mathrm{ATL}(A_{i+p}))$, *i.e.*

$$\begin{aligned}&(\forall y\in\mathbf{Y})((\alpha_{\mathrm{ATL},i}\le\alpha_{\mathrm{ATL},i+p}\wedge p\ge 1)\\&\quad\Rightarrow I_{\mathbf{L}}(\alpha_{\mathrm{ATL},i},\mathrm{ATL}(A_i)(y))\ge I_{\mathbf{L}}(\alpha_{\mathrm{ATL},i+p},\mathrm{ATL}(A_{i+p})(y)))\,.\end{aligned} \tag{10.64}$$

This property is illustrated in Fig. 10.3.

As A_i and A_{i+p} originate from a same fuzzy partition as shown in Fig. 7.1 the parameters defining the corresponding membership functions satisfy the inequality in Eq. (10.50). From Eqs. (10.40–10.41) it follows that for all values smaller than or equal to the lower bound of the support of $\mathrm{ATL}(A_{i+p})$, Eq. (10.64) holds. Since

$$(\forall y\in\mathbf{Y})(y\le a_{2i+2p\text{-}2}\Rightarrow I_{\mathbf{L}}(\alpha_{\mathrm{ATL},i+p},\mathrm{ATL}(A_{i+p})(y))=1-\alpha_{\mathrm{ATL},i+p})\,, \tag{10.65}$$
$$(\forall y\in\mathbf{Y})(I_{\mathbf{L}}(\alpha_{\mathrm{ATL},i},\mathrm{ATL}(A_i)(y))\ge 1-\alpha_{\mathrm{ATL},i})\,, \tag{10.66}$$

with $\alpha_{\mathrm{ATL},i}\le\alpha_{\mathrm{ATL},i+p}$, it also holds that

$$\begin{aligned}&(\forall y\in\mathbf{Y})(y\le a_{2i+2p\text{-}2}\\&\quad\Rightarrow I_{\mathbf{L}}(\alpha_{\mathrm{ATL},i},\mathrm{ATL}(A_i)(y))\ge I_{\mathbf{L}}(\alpha_{\mathrm{ATL},i+p},\mathrm{ATL}(A_{i+p})(y)))\,.\end{aligned} \tag{10.67}$$

From Eq. (10.45) it follows that for all values larger than or equal to the lower bound of the kernel of $\mathrm{ATL}(A_i)$, Eq. (10.64) holds. Since

$$(\forall y \in \mathbf{Y})(y \geq a_{2i\text{-}1} \Rightarrow I_{\mathbf{L}}(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) = 1)\,, \tag{10.68}$$

it also holds that

$$\begin{aligned}&(\forall y \in \mathbf{Y})(y \geq a_{2i\text{-}1}\\ &\quad \Rightarrow I_{\mathbf{L}}(\alpha_{\mathrm{ATL},i}, \mathrm{ATL}(A_i)(y)) \geq I_{\mathbf{L}}(\alpha_{\mathrm{ATL},i+p}, \mathrm{ATL}(A_{i+p})(y)))\,.\end{aligned} \tag{10.69}$$

Since the lower bound $a_{2i\text{-}1}$ of the kernel of $\mathrm{ATL}(A_i)$ is smaller than or equal to the lower bound $a_{2i+2p\text{-}2}$ of the support of $\mathrm{ATL}(A_{i+p})$ it follows from Eqs. (10.67) and (10.69) that Eq. (10.64) holds.

Analogously, the membership degree of any output value y to an adapted linguistic value $(\mathrm{ATM}(A_i))'$ is smaller than or equal to its membership degree to an adapted linguistic value $(\mathrm{ATM}(A_{i+p}))'$ if the linguistic value $(\mathrm{ATM}(A_i))$ is fired to a greater or the same fulfilment degree as the linguistic value $(\mathrm{ATM}(A_{i+p}))$, *i.e.*

$$\begin{aligned}&(\forall y \in \mathbf{Y})((\alpha_{\mathrm{ATM},i} \geq \alpha_{\mathrm{ATM},i+p} \wedge p \geq 1)\\ &\quad \Rightarrow I_{\mathbf{L}}(\alpha_{\mathrm{ATM},i}, \mathrm{ATM}(A_i)(y)) \leq I_{\mathbf{L}}(\alpha_{\mathrm{ATM},i+p}, \mathrm{ATM}(A_{i+p})(y)))\,.\end{aligned} \tag{10.70}$$

In Section 7.3.2 it is shown that for any input vector $\mathbf{x}$ at least one rule of the ATL (resp. ATM) model is fired to a fulfilment degree equal to one. As the membership function defining an adapted linguistic value with a corresponding fulfilment degree equal to one is identical to the membership function defining the original linguistic value and given the property described in Eq. (10.64), only output values smaller than or equal to the lower bound of the support of $\mathrm{ATL}(A_{i_{\mathrm{max,one}}})$ do not belong to the fuzzy output A_{ATL} of the ATL model with $i_{\mathrm{max,one}}$ the greatest index for which the corresponding fulfilment degree is equal to one, *i.e.*

$$(\forall y \in \mathbf{Y})((I_T = I_{\mathbf{L}} \wedge y \leq a_{2i_{\mathrm{max,one}}\text{-}2}) \Rightarrow A_{\mathrm{ATL}}(y) = 0)\,, \tag{10.71}$$

$$(\forall y \in \mathbf{Y})((I_T = I_{\mathbf{L}} \wedge y > a_{2i_{\mathrm{max,one}}\text{-}2}) \Rightarrow A_{\mathrm{ATL}}(y) > 0)\,, \tag{10.72}$$

with

$$i_{\mathrm{max,one}} = \max\{i \in I \mid \alpha_{\mathrm{ATL},i} = 1\}\,. \tag{10.73}$$

and only output values greater than or equal to the upper bound of the support of $\mathrm{ATM}(A_{i_{\mathrm{min,one}}})$ do not belong to the fuzzy output A_{ATM} of the ATM model with $i_{\mathrm{min,one}}$ the smallest index for which the corresponding fulfilment degree is equal to one, *i.e.*

$$(\forall y \in \mathbf{Y})((I_T = I_{\mathbf{L}} \wedge y \geq a_{2i_{\mathrm{min,one}}+1}) \Rightarrow A_{\mathrm{ATM}}(y) = 0)\,, \tag{10.74}$$

$$(\forall y \in \mathbf{Y})((I_T = I_{\mathbf{L}} \wedge y > a_{2i_{\mathrm{min,one}}+1}) \Rightarrow A_{\mathrm{ATM}}(y) > 0)\,. \tag{10.75}$$

with

$$i_{\mathrm{min,one}} = \min\{i \in I \mid \alpha_{\mathrm{ATM},i} = 1\}\,. \tag{10.76}$$

10.3.3 Fuzzy output of the ATL–ATM model

Thus, both for models applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ and for models applying $I_{\mathbf{L}}$ there exist indices $i_{\max}$ and $i_{\min}$, *i.e.*

$$i_{\max} = \begin{cases} i_{\max,\text{zero}} & \text{, if } I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\}\,, \\ i_{\max,\text{one}} & \text{, if } I_T = I_{\mathbf{L}}\,, \end{cases} \tag{10.77}$$

$$i_{\min} = \begin{cases} i_{\min,\text{zero}} & \text{, if } I_T \in \{I_{\mathbf{M}}, I_{\mathbf{P}}\}\,, \\ i_{\min,\text{one}} & \text{, if } I_T = I_{\mathbf{L}}\,. \end{cases} \tag{10.78}$$

with $i_{\max,\text{zero}}$, $i_{\min,\text{zero}}$, $i_{\max,\text{one}}$ and $i_{\min,\text{one}}$ respectively defined in Eqs. (10.57), (10.59), (10.73) and (10.76). The supports of $\mathrm{ATL}(A_{i_{\max}})$ and $\mathrm{ATM}(A_{i_{\min}})$ coincide with the support of the fuzzy output of the ATL and ATM model respectively, given by

$$\text{supp}(A_{\text{ATL}}) = \,]a_{2i_{\max}\text{-}2}, +\infty[\;, \tag{10.79}$$

$$\text{supp}(A_{\text{ATM}}) = \,]-\infty, a_{2i_{\min}+1}[\;. \tag{10.80}$$

The fuzzy output A of an ATL–ATM model is given by the intersection of the fuzzy output of the ATL and ATM model, *i.e.*

$$A(y) = \min(A_{\text{ATL}}(y), A_{\text{ATM}}(y))\,. \tag{10.81}$$

When $A_{i_{\min}}$ is smaller than $A_{i_{\max}}$, and $A_{i_{\min}}$ and $A_{i_{\max}}$ are non-consecutive linguistic output values, *i.e.*

$$i_{\min} < i_{\max} - 1\,, \tag{10.82}$$

the lower bound of the support of A_{ATL} is greater than or equal to the upper bound of the support of A_{ATM}, *i.e.*

$$a_{2i_{\min}+1} \leq a_{2i_{\max}\text{-}2}\,, \tag{10.83}$$

and the fuzzy output A is the empty set, *i.e.*

$$(\forall y \in \mathbf{Y})(A(y) = 0)\,, \tag{10.84}$$

as illustrated in Fig. 10.4(a).

When $A_{i_{\min}}$ is smaller than $A_{i_{\max}}$ and $A_{i_{\min}}$ and $A_{i_{\max}}$ are consecutive linguistic output values, or when $A_{i_{\min}}$ is larger than $A_{i_{\max}}$ *i.e.*

$$i_{\min} \geq i_{\max} - 1\,, \tag{10.85}$$

the lower bound of the support of A_{ATL} is smaller than the upper bound of the support of A_{ATM}, *i.e.*

$$a_{2i_{\min}+1} > a_{2i_{\max}\text{-}2}\,, \tag{10.86}$$

and the support of the fuzzy output A is given by,

$$\text{supp}(A) = \,]a_{2i_{\max}\text{-}2}, a_{2i_{\min}+1}[\;, \tag{10.87}$$

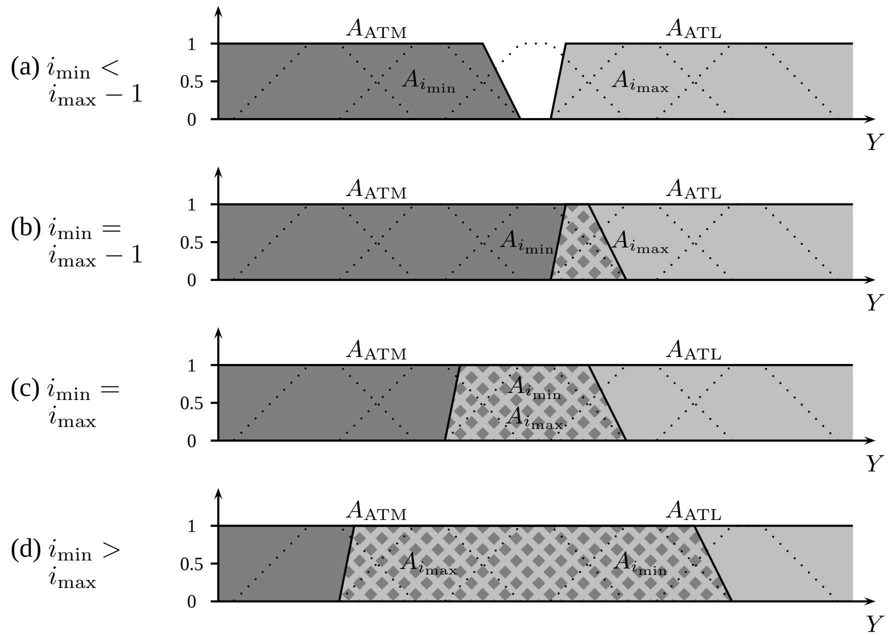


Figure 10.4: Fuzzy output A (crosshatched) of an ATL–ATM model.

as illustrated in Fig. 10.4(b–d). Since the endpoints of the support of A are finite, the defuzzification method introduced by Dvořák and Jedelský (1999) coincides with the COG defuzzification method defined in Eq. (2.44). As furthermore, A_{ATL} and A_{ATM} are respectively increasing and decreasing functions in y, the core of A is a single interval which is a very attractive property when applying the MOM defuzzification method.

10.4 Models with a single input variable

In Section 7.3.2 it is shown that the fuzzy output of an ATL–ATM model is determined by exactly those rules that are fired when applying Mamdani–Assilian or plain implicator-based inference. Thus, two rules should be considered when determining the fuzzy output of an ATL–ATM model with a single input variable: the rule corresponding to some linguistic value B^1_j to which the input x_1 has a membership degree $1 - \gamma_1$ and the rule corresponding to the linguistic input value B^1_{j+1} to which the input x_1 has a membership degree γ_1. In case of a monotone rule base, B^1_j and B^1_{j+1} can either be mapped to

1. a same linguistic output value A_i: the *constant* case,

2. two consecutive output values A_i and A_{i+1}: the *smooth* case, or

3. two non-consecutive output values A_i and A_{i+p} ($p \in \mathbb{N}$, $p > 1$, $i + p \leq n$): the *non-smooth* case.

In the ATL model, the fulfilment degrees of the two rules considered

$$\begin{array}{llll} R_1: & \text{IF} \quad X_1 \text{ IS } \mathrm{ATL}(B_j^1) & \text{THEN} & Y \text{ IS } \mathrm{ATL}(A_i) \\ R_2: & \text{IF} \quad X_1 \text{ IS } \mathrm{ATL}(B_{j+1}^1) & \text{THEN} & Y \text{ IS } \mathrm{ATL}(A_{i+p}) \end{array}$$

are given by

$$\beta_{\mathrm{ATL},1} = 1\,, \tag{10.88}$$

$$\beta_{\mathrm{ATL},2} = \gamma_1\,. \tag{10.89}$$

In the ATM model, the fulfilment degrees of the two rules considered

$$\begin{array}{llll} R_1: & \text{IF} \quad X_1 \text{ IS } \mathrm{ATM}(B_j^1) & \text{THEN} & Y \text{ IS } \mathrm{ATM}(A_i) \\ R_2: & \text{IF} \quad X_1 \text{ IS } \mathrm{ATM}(B_{j+1}^1) & \text{THEN} & Y \text{ IS } \mathrm{ATM}(A_{i+p}) \end{array}$$

are given by

$$\beta_{\mathrm{ATM},1} = 1 - \gamma_1\,, \tag{10.90}$$

$$\beta_{\mathrm{ATM},2} = 1\,. \tag{10.91}$$

10.4.1 The *constant* case

As discussed in Section 8.3, considering the *constant* case for a model with a single input variable might seem in disaccord with the aim to safeguard the model interpretability, but is nevertheless meaningful as interpretable models with more than one input variable might behave as a model with a single input variable in the *constant* case in some parts of the input space. When B_j^1 and B_{j+1}^1 are mapped to a same linguistic output value A_i, the fulfilment degree $\alpha_{\mathrm{ATL},i}$ of the linguistic output value $\mathrm{ATL}(A_i)$ is the maximum of the fulfilment degrees $\beta_{\mathrm{ATL},1}$ and $\beta_{\mathrm{ATL},2}$

$$\alpha_{\mathrm{ATL},i} = \max(\beta_{\mathrm{ATL},1}, \beta_{\mathrm{ATL},2}) = \max(1, \gamma_1) = 1\,, \tag{10.92}$$

and the fulfilment degree $\alpha_{\mathrm{ATM},i}$ of the linguistic output value $\mathrm{ATM}(A_i)$ is the maximum of the fulfilment degrees $\beta_{\mathrm{ATM},1}$ and $\beta_{\mathrm{ATM},2}$

$$\alpha_{\mathrm{ATM},i} = \max(\beta_{\mathrm{ATM},1}, \beta_{\mathrm{ATM},2}) = \max(1 - \gamma_1, 1) = 1\,. \tag{10.93}$$

The fuzzy outputs of the ATL and ATM model are thus respectively given by $\mathrm{ATL}(A_i)$ and $\mathrm{ATM}(A_i)$ and the fuzzy output of the ATL–ATM model, obtained as the intersection of the fuzzy outputs of the ATL and ATM model, is the linguistic output value A_i

$$A(y) = \min(I_T(1, \mathrm{ATL}(A_i)(y)), I_T(1, \mathrm{ATM}(A_i)(y))) \tag{10.94}$$

$$= \min(\mathrm{ATL}(A_i)(y), \mathrm{ATM}(A_i)(y)) \tag{10.95}$$

$$= A_i(y)\,. \tag{10.96}$$

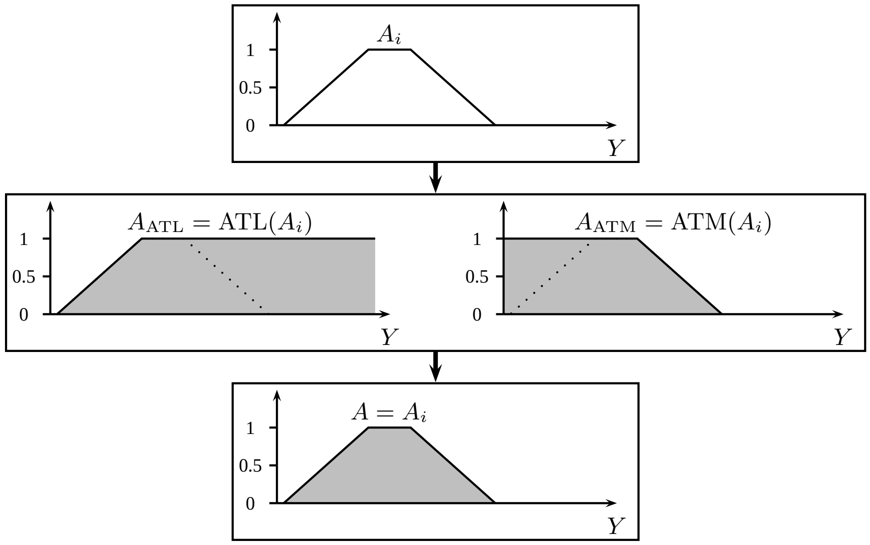


Figure 10.5: Inference procedure applied in an ATL–ATM model with a single input variable when two adjacent linguistic input values are mapped to a same linguistic output value A_i.

The inference procedure is illustrated in Fig. 10.5. Thus, for all three implicators considered, the linguistic output value A_i is obtained as fuzzy output of the ATL–ATM model in case of a constant model output in a model with a single input variable. As the obtained fuzzy output is independent of γ_1, monotonicity is guaranteed for any defuzzification method. When applying the MOM defuzzification method, the crisp output y^*_{MOM} is given by

$$y^*_{\text{MOM}} = c_i \,. \tag{10.97}$$

10.4.2 The *smooth* case

When B^1_j and B^1_{j+1} are mapped to the linguistic output values A_i and A_{i+1} respectively, the fulfilment degrees $\alpha_{\text{ATL},i}$ and $\alpha_{\text{ATL},i+1}$ of the linguistic output values $\text{ATL}(A_i)$ and $\text{ATL}(A_{i+1})$ are given by

$$\alpha_{\text{ATL},i} = \beta_{\text{ATL},1} = 1 \,, \tag{10.98}$$

$$\alpha_{\text{ATL},i+1} = \beta_{\text{ATL},2} = \gamma_1 \,, \tag{10.99}$$

and the fulfilment degrees $\alpha_{\text{ATM},i}$ and $\alpha_{\text{ATM},i+1}$ of the linguistic output values $\text{ATM}(A_i)$ and $\text{ATM}(A_{i+1})$ by

$$\alpha_{\text{ATM},i} = \beta_{\text{ATM},1} = 1 - \gamma_1 \,, \tag{10.100}$$

$$\alpha_{\text{ATM},i+1} = \beta_{\text{ATM},2} = 1 \,. \tag{10.101}$$

The fuzzy output of the ATL–ATM model (Eqs. (7.31–7.32)) is given by

$$A(y) = \min(\mathrm{ATL}(A_i)(y), I_T(\gamma_1, \mathrm{ATL}(A_{i+1}))(y), I_T(1-\gamma_1, \mathrm{ATM}(A_i)(y)), \mathrm{ATM}(A_{i+1})(y)) \,. \tag{10.102}$$

For the boundary values of γ_1, the fuzzy output A is equal to A_i or A_{i+1}

$$A = \begin{cases} A_i & \text{, if } \gamma_1 = 0 \,, \\ A_{i+1} & \text{, if } \gamma_1 = 1 \,, \end{cases} \tag{10.103}$$

with corresponding crisp outputs y^*_{MOM} given by

$$y^*_{\mathrm{MOM}} = \begin{cases} c_i & \text{, if } \gamma_1 = 0 \,, \\ c_{i+1} = c_i + k + l & \text{, if } \gamma_1 = 1 \,. \end{cases} \tag{10.104}$$

In Figs. 10.6–10.7 the inference procedure is illustrated for the implicators $I_{\mathbf{M}}$ and $I_{\mathbf{L}}$ respectively. Fig. 10.6 clearly illustrates that when applying $I_{\mathbf{M}}$ the fuzzy output A_{ATL} (resp. A_{ATM}) of the ATL (resp. ATM) model is given by the adapted membership function $(\mathrm{ATL}(A_{i_{\mathrm{max,zero}}}))'$ (resp. $(\mathrm{ATM}(A_{i_{\mathrm{min,zero}}}))'$) of the linguistic output value with the largest (resp. smallest) index among all fired linguistic output values, as was expressed earlier in Eqs. (10.56) and (10.58). The same observation was made for the implicator $I_{\mathbf{P}}$. Thus, for models applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ the expression for the fuzzy output A in Eq. (10.102) can be simplified to

$$A(y) = \min(I_T(\gamma_1, \mathrm{ATL}(A_{i+1})(y)), I_T(1-\gamma_1, \mathrm{ATM}(A_i)(y))) \,. \tag{10.105}$$

In Fig. 10.7 one can see that $\mathrm{ATL}(A_i)$, $I_T(\gamma_1, \mathrm{ATL}(A_{i+1}))$, $I_T(1-\gamma_1, \mathrm{ATM}(A_i))$ and $\mathrm{ATM}(A_{i+1})$ all contribute to the shape of the fuzzy output A. Moreover, as $\mathrm{ATL}(A_i)$ and $\mathrm{ATM}(A_{i+1})$ are fired to a fulfilment degree equal to one, the support of the fuzzy output A is bounded even if the adaptation of membership functions using $I_{\mathbf{L}}$ results in membership functions with an unbounded support. This property of the fuzzy output A of an ATL–ATM model applying $I_{\mathbf{L}}$ was discussed earlier in Sections 10.3.2–10.3.3.

In Fig. 10.8 a schematic representation is given of the fuzzy output A obtained for the three considered implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$. The crisp output y^*_{MOM} is given by the same expression for the three implicators, *i.e.*

$$y^*_{\mathrm{MOM}} = c_i + \frac{1}{2}k + \gamma_1 l \,. \tag{10.106}$$

As the derivative of y^*_{MOM} to γ_1 is positive, *i.e.* $l > 0$, and the crisp output obtained for $\gamma_1 \in \,]0,1[$ is greater than the crisp output obtained for $\gamma_1 = 0$ and smaller than the crisp output obtained for $\gamma_1 = 1$, monotonicity is guaranteed.

Approximating the function $Y = X$ is a quite frequently addressed issue in fuzzy modelling articles, for instance in the works by Park et al. (1992) and Cordón et al. (1997). From the expressions for y^*_{MOM} in Eqs. (10.104) and (10.106), it follows that the function $Y = X$ can be easily obtained by applying ATL–ATM inference to a model with in its rule base r rules

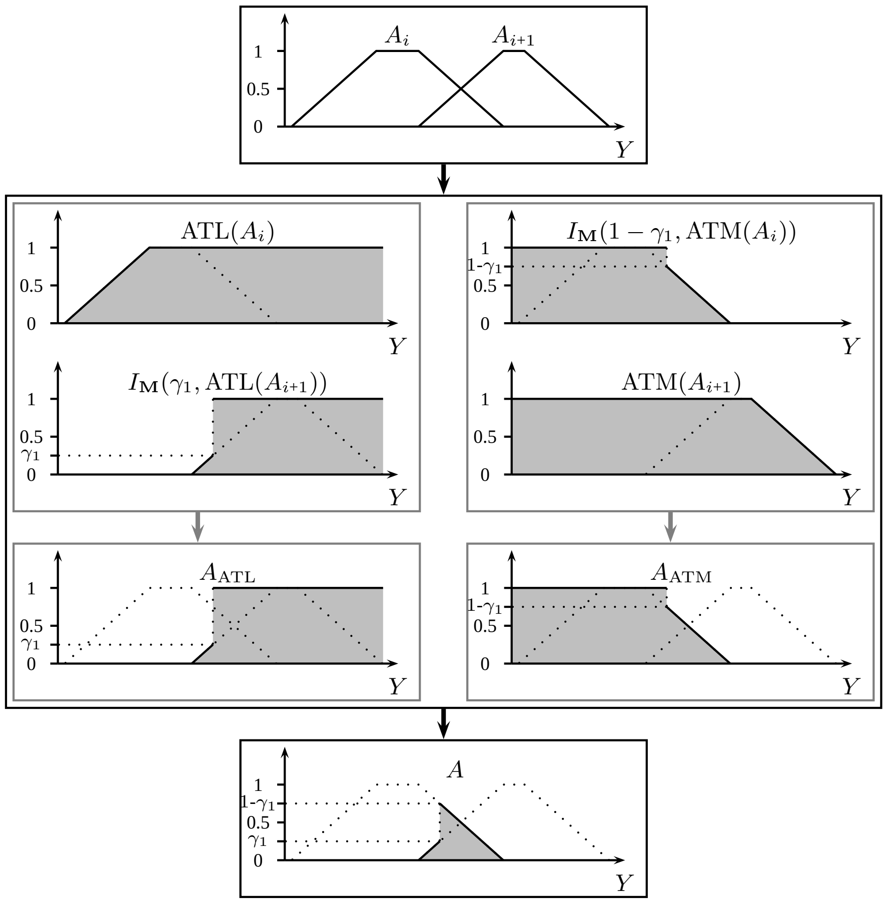


Figure 10.6: Inference procedure applied in an ATL–ATM model with a single input variable when two adjacent linguistic input values are mapped to two consecutive linguistic output values A_i and A_{i+1} with $I_T = I_{\mathbf{M}}$.

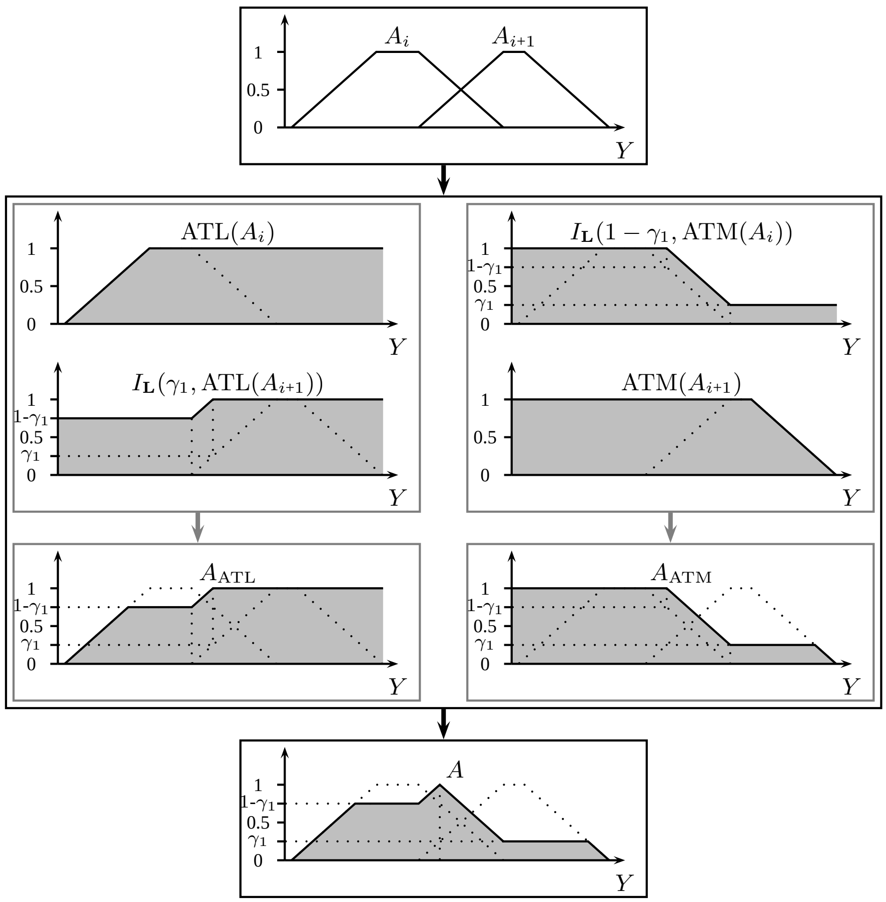


Figure 10.7: Inference procedure applied in an ATL–ATM model with a single input variable when two adjacent linguistic input values are mapped to two consecutive linguistic output values A_i and A_{i+1} with $I_T = I_{\mathbf{L}}$.

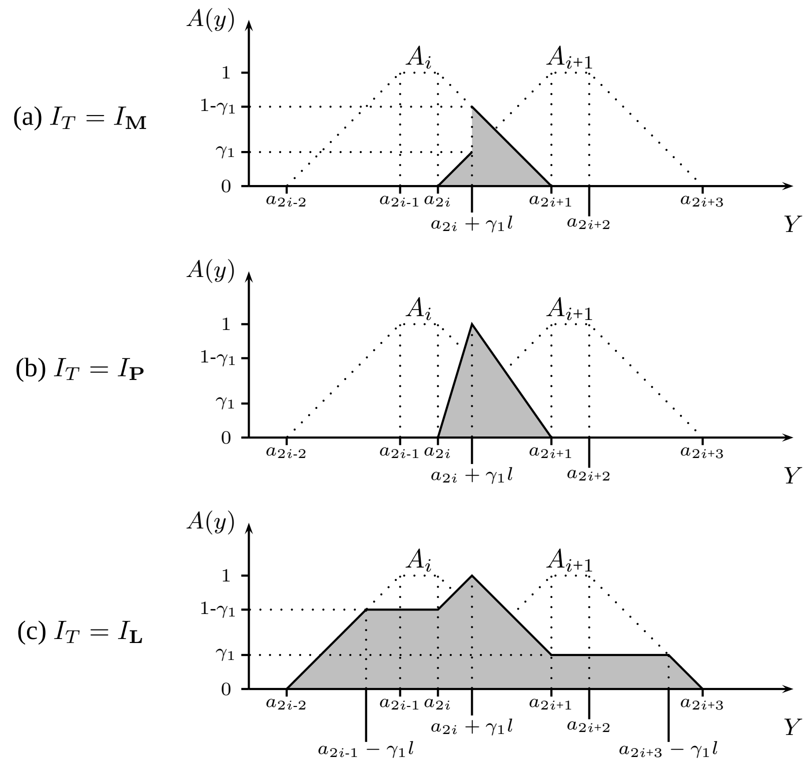


Figure 10.8: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to two consecutive linguistic output values A_i and A_{i+1} for (a) $I_T = I_{\mathbf{M}}$, (b) $I_T = I_{\mathbf{P}}$ and (c) $I_T = I_{\mathbf{L}}$.

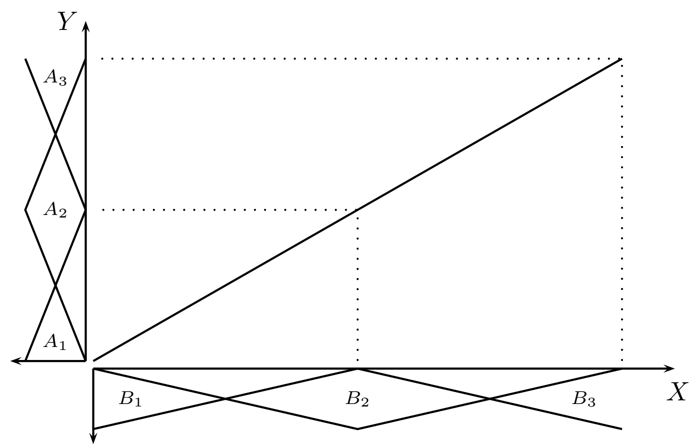


Figure 10.9: The function $Y = X$ can be obtained with ATL–ATM inference.

$$R_s: \quad \text{IF} \quad X \text{ IS } B_s \quad \text{THEN} \quad Y \text{ IS } A_s$$

and using the same fuzzy partition in the input domain $\mathbf{X}$ and the output domain $\mathbf{Y}$

$$(\forall x \in \mathbf{X})(\forall y \in \mathbf{Y})(\forall s \in \{1, \ldots, r\})(x = y \Rightarrow B_s(x) = A_s(y)) \tag{10.107}$$

This is illustrated in Fig. 10.9.

10.4.3 The *non-smooth* case

When B_j^1 and B_{j+1}^1 are mapped to two non-consecutive output values A_i and A_{i+p} ($p \in \mathbb{N}$, $p > 1$, $i + p \leq n$) respectively, the fulfilment degrees $\alpha_{\mathrm{ATL},i}$ and $\alpha_{\mathrm{ATL},i+p}$ of the linguistic output values $\mathrm{ATL}(A_i)$ and $\mathrm{ATL}(A_{i+p})$ are given by

$$\alpha_{\mathrm{ATL},i} = \beta_{\mathrm{ATL},1} = 1\,, \tag{10.108}$$
$$\alpha_{\mathrm{ATL},i+p} = \beta_{\mathrm{ATL},2} = \gamma_1\,, \tag{10.109}$$

and the fulfilment degrees $\alpha_{\mathrm{ATM},i}$ and $\alpha_{\mathrm{ATM},i+p}$ of the linguistic output values $\mathrm{ATM}(A_i)$ and $\mathrm{ATM}(A_{i+p})$ by

$$\alpha_{\mathrm{ATM},i} = \beta_{\mathrm{ATM},1} = 1 - \gamma_1\,, \tag{10.110}$$
$$\alpha_{\mathrm{ATM},i+p} = \beta_{\mathrm{ATM},2} = 1\,. \tag{10.111}$$

The fuzzy output of the ATL–ATM model (Eqs. (7.31–7.32)) is given by

$$A(y) = \min(\mathrm{ATL}(A_i)(y), I_T(\gamma_1, \mathrm{ATL}(A_{i+p})(y)), I_T(1 - \gamma_1, \mathrm{ATM}(A_i)(y)), \mathrm{ATM}(A_{i+p})(y))\,. \tag{10.112}$$

For the boundary values of γ_1, the fuzzy output A is equal to A_i or A_{i+p}

$$A = \begin{cases} A_i & \text{, if } \gamma_1 = 0\,, \\ A_{i+p} & \text{, if } \gamma_1 = 1\,, \end{cases} \tag{10.113}$$

with corresponding crisp outputs y^*_{MOM} given by

$$y^*_{\mathrm{MOM}} = \begin{cases} c_i & \text{, if } \gamma_1 = 0\,, \\ c_{i+p} = c_i + p\,k + p\,l & \text{, if } \gamma_1 = 1\,. \end{cases} \tag{10.114}$$

Given Eqs. (10.57), (10.59) and (10.77–10.78), for models applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$, the index $i_{\max}$ of the linguistic value $\mathrm{ATL}(A_{i_{\max}}))$ of which the support coincides with the support of A_{ATL} is given by

$$i_{\max} = \max(i, i+p) = i + p\,, \tag{10.115}$$

and the index $i_{\min}$ of the linguistic value $\mathrm{ATM}(A_{i_{\min}}))$ of which the support coincides with the support of A_{ATM} is given by

$$i_{\min} = \min(i, i+p) = i\,. \tag{10.116}$$

Since Eqs. (10.82–10.83) are satisfied, *i.e.* the lower bound of the support of A_{ATL} is greater than or equal to the upper bound of the support of A_{ATM}, the fuzzy output A is the empty set. Thus, monotonicity cannot be guaranteed for models with a single input variable and any monotone rule base when applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$.

For models applying $I_{\mathbf{L}}$ the indices $i_{\max}$ and $i_{\min}$ are given by (with Eqs. (10.73) and (10.76–10.78))

$$i_{\max} = i\,, \tag{10.117}$$

$$i_{\min} = i + p\,. \tag{10.118}$$

Since Eqs. (10.85–10.86) are satisfied, *i.e.* the lower bound of the support of A_{ATL} is smaller than the upper bound of the support of A_{ATM}, the fuzzy output A is a non-empty set. In Fig. 10.10 the fuzzy output A is represented for $\gamma_1 \in \,]0, 0.5[$, $\gamma_1 = 0.5$ and $\gamma_1 \in \,]0.5, 1[$, respectively. The corresponding crisp outputs are given by

$$y^*_{\mathrm{MOM}} = \begin{cases} c_i + \frac{1}{2}\gamma_1 l & \text{, if } \gamma_1 \in \,]0, 0.5[\,, \\ \frac{1}{2}(c_i + c_{i+p}) = c_i + \frac{1}{2}p\,k + \frac{1}{2}p\,l & \text{, if } \gamma_1 = 0.5\,, \\ c_{i+p} - \frac{1}{2}(1-\gamma_1)l = c_i + p\,k + \frac{1}{2}(\gamma_1 + 2p - 1)l & \text{, if } \gamma_1 \in \,]0.5, 1[\,. \end{cases} \tag{10.119}$$

Since the expressions for y^*_{MOM} in Eqs. (10.114) and (10.119) satisfy the following chain of inequalities

$$\begin{aligned} &y^*_{\mathrm{MOM}}(\gamma_1 = 0) \leq y^*_{\mathrm{MOM}}(\gamma_1 \in \,]0, 0.5[) \leq y^*_{\mathrm{MOM}}(\gamma_1 = 0.5) \\ &\quad \leq y^*_{\mathrm{MOM}}(\gamma_1 \in \,]0.5, 1[) \leq y^*_{\mathrm{MOM}}(\gamma_1 = 1)\,, \end{aligned} \tag{10.120}$$

and as the derivatives to γ_1 of the expressions in Eq. (10.119) are all positive, monotonicity is guaranteed for models with a single input variable and any monotone rule base when applying $I_{\mathbf{L}}$.

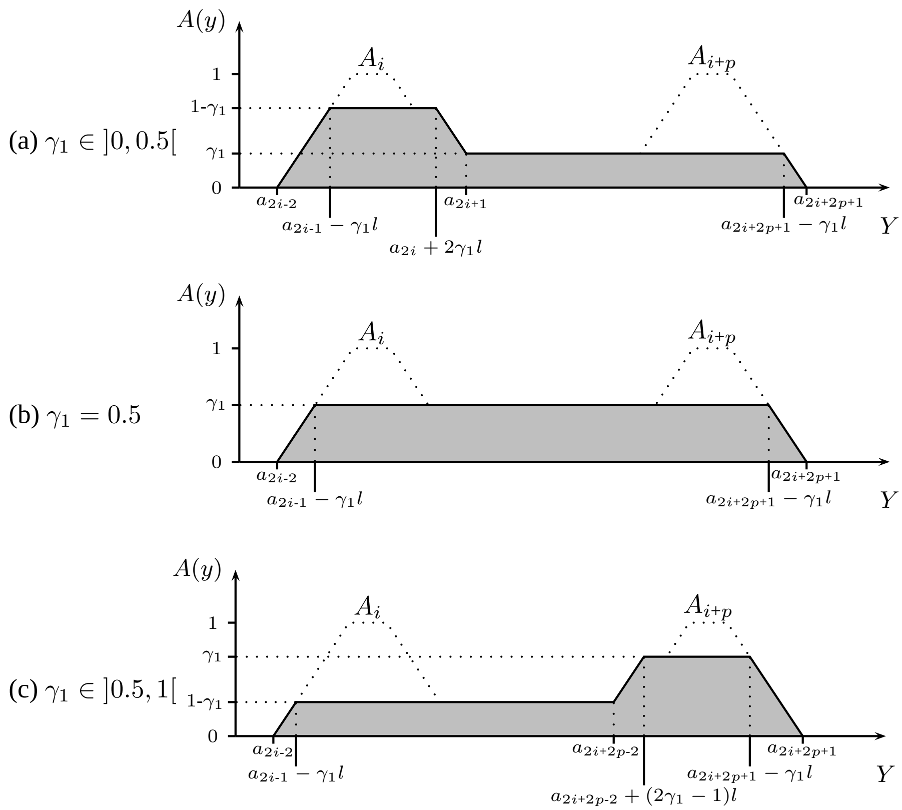


Figure 10.10: Schematic representation of the output of a model with a single input variable when two adjacent linguistic input values are mapped to two non-consecutive linguistic output values A_i and A_{i+p} for $I_T = I_{\mathbf{L}}$.

10.5 Models with two input variables and a monotone smooth rule base

For an input vector $\mathbf{x} = (x_1, x_2)$ satisfying

$$\gamma_1 = 1 - B^1_{j_1}(x_1) = B^1_{j_1+1}(x_1)\,, \tag{10.121}$$
$$\gamma_2 = 1 - B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2)\,, \tag{10.122}$$

the four rules that need to be considered when determining the model output of an ATL–ATM model with two input variables can be represented as

$$\begin{array}{llllllll}
R_1: & \text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{THEN} & Y \text{ IS } A_i \\
R_2: & \text{IF} & X_1 \text{ IS } B^1_{j_1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2} \\
R_3: & \text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2} & \text{THEN} & Y \text{ IS } A_{i+p_1} \\
R_4: & \text{IF} & X_1 \text{ IS } B^1_{j_1+1} & \text{AND} & X_2 \text{ IS } B^2_{j_2+1} & \text{THEN} & Y \text{ IS } A_{i+p_1+p_2+p_3}
\end{array}$$

When the rule base is smooth, the values of p_1, p_2 and p_3 in the rules above are restricted to

$$(p_1, p_2, p_3) \in \{(0,0,0), (0,0,1), (0,1,0), (1,0,0), (1,0,1)\}\,. \tag{10.123}$$

In the following these triplets will respectively be referred to as Case I, II, III, IV and V.

The fulfilment degrees of the four corresponding rules in the ATL model

$$\begin{array}{lllllll}
R_1: & \text{IF } X_1 \text{ IS ATL}(B^1_{j_1}) & \text{AND } X_2 \text{ IS ATL}(B^2_{j_2}) & \text{THEN } Y \text{ IS ATL}(A_i) \\
R_2: & \text{IF } X_1 \text{ IS ATL}(B^1_{j_1}) & \text{AND } X_2 \text{ IS ATL}(B^2_{j_2+1}) & \text{THEN } Y \text{ IS ATL}(A_{i+p_1+p_2}) \\
R_3: & \text{IF } X_1 \text{ IS ATL}(B^1_{j_1+1}) & \text{AND } X_2 \text{ IS ATL}(B^2_{j_2}) & \text{THEN } Y \text{ IS ATL}(A_{i+p_1}) \\
R_4: & \text{IF } X_1 \text{ IS ATL}(B^1_{j_1+1}) & \text{AND } X_2 \text{ IS ATL}(B^2_{j_2+1}) & \text{THEN } Y \text{ IS ATL}(A_{i+p_1+p_2+p_3})
\end{array}$$

are given by

$$\beta_{\text{ATL},1} = 1\,, \tag{10.124}$$
$$\beta_{\text{ATL},2} = \gamma_2\,, \tag{10.125}$$
$$\beta_{\text{ATL},3} = \gamma_1\,, \tag{10.126}$$
$$\beta_{\text{ATL},4} = T(\gamma_1, \gamma_2)\,. \tag{10.127}$$

The fulfilment degrees of the four corresponding rules in the ATM model

$$\begin{array}{lllllll}
R_1: & \text{IF } X_1 \text{ IS ATM}(B^1_{j_1}) & \text{AND } X_2 \text{ IS ATM}(B^2_{j_2}) & \text{THEN } Y \text{ IS ATM}(A_i) \\
R_2: & \text{IF } X_1 \text{ IS ATM}(B^1_{j_1}) & \text{AND } X_2 \text{ IS ATM}(B^2_{j_2+1}) & \text{THEN } Y \text{ IS ATM}(A_{i+p_1+p_2}) \\
R_3: & \text{IF } X_1 \text{ IS ATM}(B^1_{j_1+1}) & \text{AND } X_2 \text{ IS ATM}(B^2_{j_2}) & \text{THEN } Y \text{ IS ATM}(A_{i+p_1}) \\
R_4: & \text{IF } X_1 \text{ IS ATM}(B^1_{j_1+1}) & \text{AND } X_2 \text{ IS ATM}(B^2_{j_2+1}) & \text{THEN } Y \text{ IS ATM}(A_{i+p_1+p_2+p_3})
\end{array}$$

are given by

$$\beta_{\text{ATM},1} = T(1-\gamma_1, 1-\gamma_2)\,, \tag{10.128}$$
$$\beta_{\text{ATM},2} = 1-\gamma_1\,, \tag{10.129}$$
$$\beta_{\text{ATM},3} = 1-\gamma_2\,, \tag{10.130}$$
$$\beta_{\text{ATM},4} = 1\,. \tag{10.131}$$

10.5.1 General discussion of Case I

For Case I, with $(p_1, p_2, p_3) = (0, 0, 0)$, all four considered rules contain a same linguistic output A_i value in their consequent. For input vectors corresponding to any $(\gamma_1, \gamma_2) \in \,]0, 1[^2$, the fulfilment degree $\alpha_{\mathrm{ATL},i}$ is given by

$$\begin{aligned}\alpha_{\mathrm{ATL},i} &= \max(\beta_{\mathrm{ATL},1}, \beta_{\mathrm{ATL},2}, \beta_{\mathrm{ATL},3}, \beta_{\mathrm{ATL},4}) \\ &= \max(1, \gamma_2, \gamma_1, T(\gamma_1, \gamma_2)) = 1\,, \end{aligned} \tag{10.132}$$

and the fulfilment degree $\alpha_{\mathrm{ATM},i}$ is given by

$$\begin{aligned}\alpha_{\mathrm{ATM},i} &= \max(\beta_{\mathrm{ATM},1}, \beta_{\mathrm{ATM},2}, \beta_{\mathrm{ATM},3}, \beta_{\mathrm{ATM},4}) \\ &= \max(T(1-\gamma_1, 1-\gamma_2), 1-\gamma_1, 1-\gamma_2, 1) = 1\,. \end{aligned} \tag{10.133}$$

The same fulfilment degrees are obtained for the linguistic output values if γ_1 or γ_2 are equal to zero or one. For Case I the same fulfilment degrees are obtained for the linguistic output values of the ATL and ATM model as for a model with a single input variable in the constant case (Section 10.4.1). Therefore the crisp output y^*_{MOM} is given by the expression obtained in the latter case, *i.e.*

$$y^*_{\mathrm{MOM}} = c_i\,, \tag{10.134}$$

and thus, monotonicity is guaranteed.

10.5.2 General discussion of Case III

For Case III, with $(p_1, p_2, p_3) = (0, 1, 0)$, the four considered rules contain linguistic output values derived from A_i and A_{i+1} in their consequent. The fulfilment degrees $\alpha_{\mathrm{ATL},i}$ and $\alpha_{\mathrm{ATL},i+1}$ are given by

$$\alpha_{\mathrm{ATL},i} = \max(\beta_{\mathrm{ATL},1}, \beta_{\mathrm{ATL},3}) = \max(1, \gamma_1) = 1\,, \tag{10.135}$$

$$\alpha_{\mathrm{ATL},i+1} = \max(\beta_{\mathrm{ATL},2}, \beta_{\mathrm{ATL},4}) = \max(\gamma_2, T(\gamma_1, \gamma_2)) = \gamma_2\,, \tag{10.136}$$

and the fulfilment degrees $\alpha_{\mathrm{ATM},i}$ and $\alpha_{\mathrm{ATM},i+1}$ are given by

$$\alpha_{\mathrm{ATM},i} = \max(\beta_{\mathrm{ATM},1}, \beta_{\mathrm{ATM},3}) = \max(T(1-\gamma_1, 1-\gamma_2), 1-\gamma_2) = 1-\gamma_2\,, \tag{10.137}$$

$$\alpha_{\mathrm{ATM},i+1} = \max(\beta_{\mathrm{ATM},2}, \beta_{\mathrm{ATM},4}) = \max(1-\gamma_1, 1) = 1\,. \tag{10.138}$$

The fulfilment degrees obtained for the boundary conditions are shown in Table 10.1.

The fulfilment degrees obtained for $(\gamma_1, \gamma_2) \in [0, 1] \times \,]0, 1[$ correspond to those obtained for a model with a single input variable in the smooth case (Section 10.4.2). Thus, for these input vectors the crisp output y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = c_i + \frac{1}{2}k + \gamma_2 l \qquad \text{, if } (\gamma_1, \gamma_2) \in [0, 1] \times \,]0, 1[\,. \tag{10.139}$$

Table 10.1: Fulfilment degrees $\alpha_{\mathrm{ATL},i}$, $\alpha_{\mathrm{ATL},i+1}$, $\alpha_{\mathrm{ATM},i}$ and $\alpha_{\mathrm{ATM},i+1}$ for the boundary conditions of Case III.

γ_1	γ_2	$\alpha_{\mathrm{ATL},i}$	$\alpha_{\mathrm{ATL},i+1}$	$\alpha_{\mathrm{ATM},i}$	$\alpha_{\mathrm{ATM},i+1}$
0	$]0,1[$	1	γ_2	$1-\gamma_2$	1
1	$]0,1[$	1	γ_2	$1-\gamma_2$	1
$]0,1[$	0	1	(0)	1	(1)
$]0,1[$	1	(1)	1	(0)	1

It follows from the discussion in Section 10.3 that for $\gamma_1 \in]0,1[$ and $\gamma_2 = 0$ (resp. $\gamma_2 = 1$) the fuzzy output of the ATM model (resp. ATL model) is identical to $\mathrm{ATM}(A_i)$ (resp. $\mathrm{ATL}(A_{i+1})$) and the fuzzy output A of the ATL–ATM model is given by A_i (resp. A_{i+1}). In Table 10.1 fulfilment degrees corresponding to an adapted membership function that do not influence the fuzzy output A_{ATL} or A_{ATM} are put in round brackets. Thus, for these input vectors the crisp output y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = \begin{cases} c_i & \text{, if } \gamma_1 \in]0,1[\text{ and } \gamma_2 = 0\,, \\ c_i + k + l & \text{, if } \gamma_1 \in]0,1[\text{ and } \gamma_2 = 1\,. \end{cases} \tag{10.140}$$

In Case III, monotonicity is guaranteed, since the derivatives to γ_1 or γ_2 of all expressions for y^*_{MOM} are positive and since the following chains of inequalities holds for any $\gamma_2^* \in]0,1[$ and $\gamma_1^* \in]0,1[$

$$y^*_{\mathrm{MOM}}(\gamma_1 = 0, \gamma_2 = \gamma_2^*) \le y^*_{\mathrm{MOM}}(\gamma_1 = \gamma_1^*, \gamma_2 = \gamma_2^*) \le y^*_{\mathrm{MOM}}(\gamma_1 = 1, \gamma_2 = \gamma_2^*)\,, \tag{10.141}$$

$$y^*_{\mathrm{MOM}}(\gamma_1 = \gamma_1^*, \gamma_2 = 0) \le y^*_{\mathrm{MOM}}(\gamma_1 = \gamma_1^*, \gamma_2 = \gamma_2^*) \le y^*_{\mathrm{MOM}}(\gamma_1 = \gamma_1^*, \gamma_2 = 1)\,, \tag{10.142}$$

as

$$c_i + \frac{1}{2}k + \gamma_2^* l = c_i + \frac{1}{2}k + \gamma_2^* l = c_i + \frac{1}{2}k + \gamma_2^* l\,, \tag{10.143}$$

$$c_i < c_i + \frac{1}{2}k + \gamma_2^* l < c_i + k + l\,. \tag{10.144}$$

10.5.3 Models applying $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$ combined with $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$

For Case V, with $(p_1, p_2, p_3) = (1, 0, 1)$, the four considered rules contain linguistic output values derived from A_i, A_{i+1} and A_{i+2} in their consequent. The fulfilment degrees $\alpha_{\mathrm{ATL},i}$, $\alpha_{\mathrm{ATL},i+1}$ and $\alpha_{\mathrm{ATL},i+2}$ are given by

$$\alpha_{\mathrm{ATL},i} = \beta_{\mathrm{ATL},1} = 1\,, \tag{10.145}$$

$$\alpha_{\mathrm{ATL},i+1} = \max(\beta_{\mathrm{ATL},2}, \beta_{\mathrm{ATL},3}) = \max(\gamma_2, \gamma_1)\,, \tag{10.146}$$

$$\alpha_{\mathrm{ATL},i+2} = \beta_{\mathrm{ATL},4} = T(\gamma_1, \gamma_2)\,, \tag{10.147}$$

and the fulfilment degrees $\alpha_{\mathrm{ATM},i}$, $\alpha_{\mathrm{ATM},i+1}$ and $\alpha_{\mathrm{ATM},i+2}$ are given by

$$\alpha_{\mathrm{ATM},i} = \beta_{\mathrm{ATM},1} = T(1-\gamma_1, 1-\gamma_2)\,, \tag{10.148}$$
$$\alpha_{\mathrm{ATM},i+1} = \max(\beta_{\mathrm{ATM},2}, \beta_{\mathrm{ATM},3}) = \max(1-\gamma_1, 1-\gamma_2)\,, \tag{10.149}$$
$$\alpha_{\mathrm{ATM},i+2} = \beta_{\mathrm{ATM},4} = 1\,. \tag{10.150}$$

When applying the t-norm $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$, the fulfilment degrees $\alpha_{\mathrm{ATL},i+2}$ and $\alpha_{\mathrm{ATM},i}$ are strictly positive for any $(\gamma_1,\gamma_2) \in]0,1[^2$, *i.e.*

$$(\forall(\gamma_1,\gamma_2) \in]0,1[^2)(\forall T \in \{T_{\mathbf{M}}, T_{\mathbf{P}}\})(T(\gamma_1,\gamma_2) > 0) \tag{10.151}$$
$$(\forall(\gamma_1,\gamma_2) \in]0,1[^2)(\forall T \in \{T_{\mathbf{M}}, T_{\mathbf{P}}\})(T(1-\gamma_1,1-\gamma_2) > 0) \tag{10.152}$$

Thus, for models applying $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$ and input vectors characterized by $(\gamma_1,\gamma_2) \in]0,1[^2$, the index $i_{\max}$ (defined in Eqs. (10.57) and (10.77)) of the linguistic value $\mathrm{ATL}(A_{i_{\max}})$ of which the support coincides with the support of A_{ATL} is given by

$$i_{\max} = \max(i, i+1, i+2) = i+2\,, \tag{10.153}$$

and the index $i_{\min}$ (defined in Eqs. (10.59) and (10.78)) of the linguistic value $\mathrm{ATM}(A_{i_{\min}})$ of which the support coincides with the support of A_{ATM} is given by

$$i_{\min} = \min(i, i+1, i+2) = i\,. \tag{10.154}$$

Since Eqs. (10.82–10.83) are satisfied, *i.e.* the lower bound of the support of A_{ATL} is greater than or equal to the upper bound of the support of A_{ATM}, the fuzzy output A is the empty set. Thus, monotonicity cannot be guaranteed for models with two input variables and any monotone rule base when applying $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$ combined with $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$.

10.5.4 Models applying $T_{\mathbf{M}}$ combined with $I_{\mathbf{L}}$

For Case V, with $(p_1,p_2,p_3) = (1,0,1)$, the four considered rules contain linguistic output values derived from A_i, A_{i+1} and A_{i+2} in their consequent. The corresponding fulfilment degrees are given in Eqs. (10.145–10.150). For the t-norm $T_{\mathbf{M}}$ they are obtained by

$$\begin{aligned}
\alpha_{\mathrm{ATL},i} &= 1 & \alpha_{\mathrm{ATM},i} &= \min(1-\gamma_1, 1-\gamma_2)\,,\\
\alpha_{\mathrm{ATL},i+1} &= \max(\gamma_1,\gamma_2) & \alpha_{\mathrm{ATM},i+1} &= \max(1-\gamma_1, 1-\gamma_2)\,,\\
\alpha_{\mathrm{ATL},i+2} &= \min(\gamma_1,\gamma_2) & \alpha_{\mathrm{ATM},i+2} &= 1\,.
\end{aligned} \tag{10.155}$$

For two input vectors $\mathbf{x_1}$ and $\mathbf{x_2}$ characterized by $(\gamma_1,\gamma_2) = (\eta_1,\eta_2)$ and $(\gamma_1,\gamma_2) = (\eta_2,\eta_2)$ respectively with

$$0 < \eta_1 < \eta_2 < 0.5\,, \tag{10.156}$$

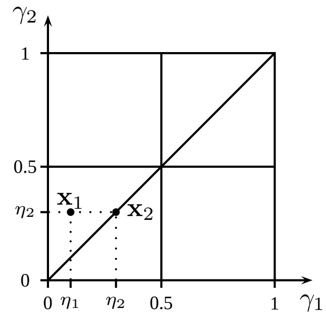


Figure 10.11: Indication of the two input vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ considered in the discussion about models with two input variables applying $T_{\mathbf{M}}$ combined with $I_{\mathbf{L}}$.

as indicated in the (γ_1, γ_2)-plane in Fig. 10.11, the following inequality should hold in order to obtain a monotone input-output behaviour

$$y^*_{\text{MOM}}(\mathbf{x}_1) \leq y^*_{\text{MOM}}(\mathbf{x}_2)\,. \tag{10.157}$$

For $\mathbf{x}_1$, the fulfilment degrees are given by

$$\begin{aligned} \alpha_{\text{ATL},i} &= 1 & \alpha_{\text{ATM},i} &= 1 - \eta_2\,, \\ \alpha_{\text{ATL},i+1} &= \eta_2 & \alpha_{\text{ATM},i+1} &= 1 - \eta_1\,, \\ \alpha_{\text{ATL},i+2} &= \eta_1 & \alpha_{\text{ATM},i+2} &= 1\,, \end{aligned} \tag{10.158}$$

and a fuzzy output A as illustrated in Fig. 10.12(a) is obtained. The crisp output $y^*_{\text{MOM}}(\mathbf{x}_1)$ is given by

$$y^*_{\text{MOM}}(\mathbf{x}_1) = c_i + \frac{1}{2}k + \eta_2 l\,. \tag{10.159}$$

For $\mathbf{x}_2$, the fulfilment degrees are given by

$$\begin{aligned} \alpha_{\text{ATL},i} &= 1 & \alpha_{\text{ATM},i} &= 1 - \eta_2\,, \\ (\alpha_{\text{ATL},i+1} &= \eta_2) & (\alpha_{\text{ATM},i+1} &= 1 - \eta_2)\,, \\ \alpha_{\text{ATL},i+2} &= \eta_2 & \alpha_{\text{ATM},i+2} &= 1\,. \end{aligned} \tag{10.160}$$

The fulfilment degrees $\alpha_{\text{ATL},i+1}$ and $\alpha_{\text{ATM},i+1}$ are put in round brackets as it follows from Eqs. (10.64) and (10.70) that the corresponding adapted membership functions do not determine the fuzzy outputs A_{ATL} and A_{ATM}. Thus, the fuzzy output obtained for $\mathbf{x}_2$ corresponds to the fuzzy output obtained for a model with a single input variable in the non-smooth case (Section 10.4.3) and, as illustrated in Fig. 10.12(b) the crisp output $y^*_{\text{MOM}}(\mathbf{x}_2)$ is given by

$$y^*_{\text{MOM}}(\mathbf{x}_2) = c_i + \frac{1}{2}\eta_2 l\,. \tag{10.161}$$

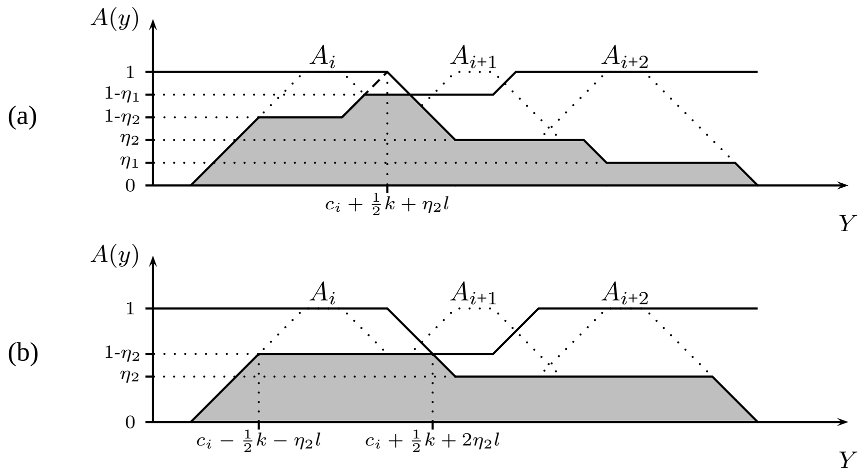


Figure 10.12: Schematic representation of the output obtained for the input vectors (a) $\mathbf{x}_1$ and (b) $\mathbf{x}_2$ considered in the discussion about models with two input variables applying $T_{\mathbf{M}}$ combined with $I_{\mathbf{L}}$.

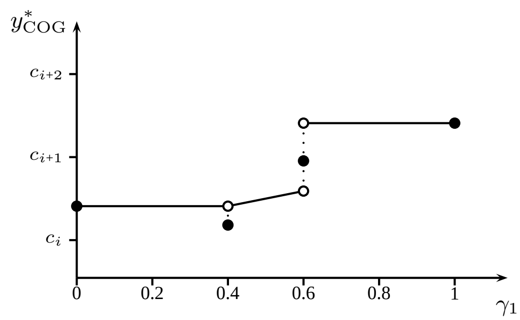


Figure 10.13: Crisp output y^*_{MOM} as a function of γ_1 for input vectors firing the four rules 'Case V' with $\gamma_2 = 0.4$, $k = 0.1$, $l = 1$, $T = T_{\mathbf{M}}$ and $I_T = I_{\mathbf{L}}$.

Since l and η_2 are strictly positive and k is positive, it holds that

$$c_i + \frac{1}{2}k + \eta_2 l > c_i + \frac{1}{2}\eta_2 l\,. \tag{10.162}$$

Thus, Eq. (10.157) does not hold and a non-monotone input-output behaviour is obtained for Case V when applying $T_{\mathbf{M}}$ combined with $I_{\mathbf{L}}$. In Fig. 10.13 the crisp output y^*_{MOM} is shown as a function of γ_1 for input vectors firing the four rules 'Case V'. Thus, monotonicity cannot be guaranteed for models with two input variables and any monotone rule base when applying $T_{\mathbf{M}}$ combined with $I_{\mathbf{L}}$.

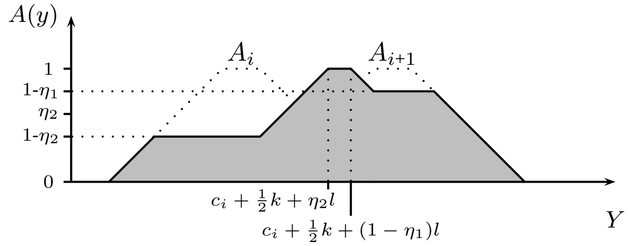


Figure 10.14: Schematic representation of the model output type '2input-$I_{\mathbf{L}}$-1' for models applying $I_{\mathbf{L}}$.

10.5.5 Models applying $T_{\mathbf{P}}$ combined with $I_{\mathbf{L}}$

In order to prove that monotonicity is guaranteed for models with any monotone smooth rule base applying $T_{\mathbf{P}}$ combined with $I_{\mathbf{L}}$ it still needs to be proved that monotonicity is guaranteed in Cases II, IV and V. Before starting the discussion about the model behaviour is these cases, first the crisp model output y^*_{MOM} is determined of a model applying $I_{\mathbf{L}}$ in case the fuzzy output is the intersection of adapted membership functions corresponding to the following fulfilment degrees

$$\begin{aligned} \alpha_{\mathrm{ATL},i} &= 1 \qquad & \alpha_{\mathrm{ATM},i} &= \eta_1\,, \\ \alpha_{\mathrm{ATL},i+1} &= \eta_2 \qquad & \alpha_{\mathrm{ATM},i+1} &= 1\,. \end{aligned} \tag{10.163}$$

with $0 < \eta_2 < 1 - \eta_1 < 1$. In the following, this type of model output will be referred to as '2input-$I_{\mathbf{L}}$-1'. The corresponding fuzzy output A is shown in Fig. 10.14. The core of the fuzzy output A is given by

$$\mathrm{core}(A) = \left[c_i + \frac{1}{2}k + \eta_2 l, c_i + \frac{1}{2}k + (1-\eta_1)l\right]\,, \tag{10.164}$$

and the crisp output y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = c_i + \frac{1}{2}k + \frac{1}{2}(1-\eta_1+\eta_2)l\,. \tag{10.165}$$

For Cases II and IV, with $(p_1, p_2, p_3) = (0, 0, 1)$ and $(p_1, p_2, p_3) = (1, 0, 0)$ respectively, the four considered rules contain linguistic output values derived from A_i and A_{i+1} in their consequent. For Case II, the fulfilment degrees $\alpha_{\mathrm{ATL},i}$ and $\alpha_{\mathrm{ATL},i+1}$ are given by

$$\alpha_{\mathrm{ATL},i} = \max(\beta_{\mathrm{ATL},1}, \beta_{\mathrm{ATL},2}, \beta_{\mathrm{ATL},3}) = \max(1, \gamma_2, \gamma_1) = 1\,, \tag{10.166}$$

$$\alpha_{\mathrm{ATL},i+1} = \beta_{\mathrm{ATL},4} = T(\gamma_1, \gamma_2)\,, \tag{10.167}$$

and the fulfilment degrees $\alpha_{\text{ATM},i}$ and $\alpha_{\text{ATM},i+1}$ are given by

$$\begin{aligned} \alpha_{\text{ATM},i} &= \max(\beta_{\text{ATM},1}, \beta_{\text{ATM},2}, \beta_{\text{ATM},3}) \\ &= \max(T(1-\gamma_1, 1-\gamma_2), 1-\gamma_1, 1-\gamma_2) \\ &= \max(1-\gamma_1, 1-\gamma_2)\,, \qquad (10.168) \\ \alpha_{\text{ATM},i+1} &= \beta_{\text{ATM},4} = 1\,. \qquad (10.169) \end{aligned}$$

For Case IV, the fulfilment degrees $\alpha_{\text{ATL},i}$ and $\alpha_{\text{ATL},i+1}$ are given by

$$\alpha_{\text{ATL},i} = \beta_{\text{ATL},1} = 1\,, \qquad (10.170)$$

$$\alpha_{\text{ATL},i+1} = \max(\beta_{\text{ATL},2}, \beta_{\text{ATL},3}, \beta_{\text{ATL},4}) = \max(\gamma_2, \gamma_1, T(\gamma_1, \gamma_2)) = \max(\gamma_2, \gamma_1)\,, \qquad (10.171)$$

and the fulfilment degrees $\alpha_{\text{ATM},i}$ and $\alpha_{\text{ATM},i+1}$ are given by

$$\alpha_{\text{ATM},i} = \beta_{\text{ATM},1} = T(1-\gamma_1, 1-\gamma_2)\,, \qquad (10.172)$$

$$\alpha_{\text{ATM},i+1} = \max(\beta_{\text{ATM},2}, \beta_{\text{ATM},3}, \beta_{\text{ATM},4}) = \max(1-\gamma_1, 1-\gamma_2, 1) = 1\,. \qquad (10.173)$$

The fulfilment degrees $\alpha_{\text{ATL},i}$, $\alpha_{\text{ATL},i+1}$, $\alpha_{\text{ATM},i}$ and $\alpha_{\text{ATM},i+1}$ obtained in the different parts of the input space for Cases II and IV are listed in Table 10.2. The derivatives to γ_1 or γ_2 of all expressions for y^*_{MOM} in Table 10.2 are positive. Furthermore, y^*_{MOM} increases for any $\gamma_2 \in \,]0,1[$ (resp. $\gamma_1 \in \,]0,1[$) when γ_1 (resp. γ_2) is increased from 0 to 1, as illustrated in Fig. 10.15. Thus, monotonicity is guaranteed for Cases II and IV.

For Case V, with $(p_1, p_2, p_3) = (1, 0, 1)$, the four considered rules contain linguistic output values derived from A_i, A_{i+1} and A_{i+2} in their consequent. The corresponding fulfilment degrees are given in Eqs. (10.145–10.150). For the t-norm $T_{\mathbf{P}}$ they are given by

$$\begin{aligned} \alpha_{\text{ATL},i} &= 1 & \alpha_{\text{ATM},i} &= (1-\gamma_1)(1-\gamma_2)\,, \\ \alpha_{\text{ATL},i+1} &= \max(\gamma_1, \gamma_2) & \alpha_{\text{ATM},i+1} &= \max(1-\gamma_1, 1-\gamma_2)\,, \\ \alpha_{\text{ATL},i+2} &= \gamma_1\gamma_2 & \alpha_{\text{ATM},i+2} &= 1\,. \qquad (10.174) \end{aligned}$$

In Fig. 10.16 the fuzzy output A is shown for input vectors $\mathbf{x}$ characterized by $\gamma_1 < \gamma_2$ and

(a) $\gamma_1 + \gamma_2 < 1$,

(b) $\gamma_1 + \gamma_2 = 1$, or,

(c) $\gamma_1 + \gamma_2 > 1$.

Similar fuzzy outputs are obtained for input vectors $\mathbf{x}$ for which $\gamma_1 = \gamma_2$ or $\gamma_1 > \gamma_2$. An overview of the obtained expressions for y^*_{MOM} in the different parts of the input space is given in Fig. 10.17. The derivatives to γ_1 or γ_2 of all expressions for y^*_{MOM}

Table 10.2: Fulfilment degrees $\alpha_{\mathrm{ATL},i}$, $\alpha_{\mathrm{ATL},i+1}$, $\alpha_{\mathrm{ATM},i}$ and $\alpha_{\mathrm{ATM},i+1}$ and crisp output y^*_{MOM} obtained in the different parts of the input space for Cases II and IV with a model applying $T_{\mathbf{P}}$ combined with $I_{\mathbf{L}}$.

Case	conditions on (γ_1, γ_2)	$\alpha_{\mathrm{ATL},i}$	$\alpha_{\mathrm{ATL},i+1}$	$\alpha_{\mathrm{ATM},i}$	$\alpha_{\mathrm{ATM},i+1}$	corresponds to	y^*_{MOM}
II a	$\gamma_1 = 0, \gamma_2 \in]0,1[$	1	(0)	1	(1)	1input-constant	c_i
II b	$\gamma_1 \in]0,1[, \gamma_2 = 1$	1	γ_1	$1-\gamma_1$	1	1input-smooth	$c_i + \frac{1}{2}k + \gamma_1 l$
II c	$\gamma_1 = 1, \gamma_2 \in]0,1[$	1	γ_2	$1-\gamma_2$	1	1input-smooth	$c_i + \frac{1}{2}k + \gamma_2 l$
II d	$\gamma_1 \in]0,1[, \gamma_2 = 0$	1	(0)	1	(1)	1input-constant	c_i
II e	$\gamma_1, \gamma_2 \in]0,1[, \gamma_1 \leq \gamma_2$	1	$\gamma_1\gamma_2$	$1-\gamma_1$	1	2input-$I_{\mathbf{L}}$-1	$c_i + \frac{1}{2}k + \frac{1}{2}\gamma_1(1+\gamma_2)l$
II f	$\gamma_1, \gamma_2 \in]0,1[, \gamma_1 \geq \gamma_2$	1	$\gamma_1\gamma_2$	$1-\gamma_2$	1	2input-$I_{\mathbf{L}}$-1	$c_i + \frac{1}{2}k + \frac{1}{2}(1+\gamma_1)\gamma_2 l$
IV a	$\gamma_1 = 0, \gamma_2 \in]0,1[$	1	γ_2	$1-\gamma_2$	1	1input-smooth	$c_i + \frac{1}{2}k + \gamma_2 l$
IV b	$\gamma_1 \in]0,1[, \gamma_2 = 1$	(1)	1	(0)	1	1input-constant	$c_i + k + l$
IV c	$\gamma_1 = 1, \gamma_2 \in]0,1[$	(1)	1	(0)	1	1input-constant	$c_i + k + l$
IV d	$\gamma_1 \in]0,1[, \gamma_2 = 0$	1	γ_1	$1-\gamma_1$	1	1input-smooth	$c_i + \frac{1}{2}k + \gamma_1 l$
IV e	$\gamma_1, \gamma_2 \in]0,1[, \gamma_1 \leq \gamma_2$	1	γ_2	$(1-\gamma_1)(1-\gamma_2)$	1	2input-$I_{\mathbf{L}}$-1	$c_i + \frac{1}{2}k + \frac{1}{2}(\gamma_1 + 2\gamma_2 - \gamma_1\gamma_2)l$
IV f	$\gamma_1, \gamma_2 \in]0,1[, \gamma_1 \geq \gamma_2$	1	γ_1	$(1-\gamma_1)(1-\gamma_2)$	1	2input-$I_{\mathbf{L}}$-1	$c_i + \frac{1}{2}k + \frac{1}{2}(2\gamma_1 + \gamma_2 - \gamma_1\gamma_2)l$

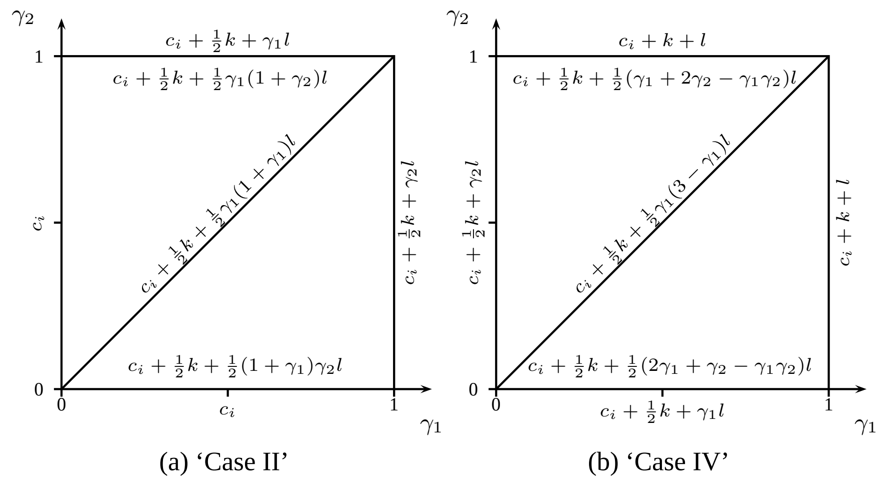


Figure 10.15: Overview of the expressions obtained for y^*_{MOM} with a model applying $T_{\mathbf{P}}$ combined with $I_{\mathbf{L}}$ and input vectors firing the four rules (a) 'Case II' and (b) 'Case IV'.

are positive. Furthermore, y^*_{MOM} increases for any $\gamma_2 \in \,]0,1[$ (resp. $\gamma_1 \in \,]0,1[$) when γ_1 (resp. γ_2) is increased from 0 to 1, as illustrated in Fig. 10.17. Thus, monotonicity is guaranteed for Case V.

Summarizing, the results in Sections 10.5.1–10.5.2 and in this section show that monotonicity is guaranteed for models with any monotone smooth rule base when applying $T_{\mathbf{P}}$ combined with $I_{\mathbf{L}}$.

10.5.6 Models applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$

In order to prove that monotonicity is guaranteed for models with any monotone smooth rule base applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ it still needs to be proved that monotonicity is guaranteed in Cases II, IV and V. Before starting the discussion about the model behaviour is these cases, first the crisp model output y^*_{MOM} is determined of a model applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ for four model output types. For all four model output types the same expression is obtained for y^*_{MOM} regardless of the implicator applied, *i.e.* either $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$.

The first type are fuzzy outputs that are the intersection of adapted membership functions corresponding to the following fulfilment degrees

$$\alpha_{\mathrm{ATL},i} = 1 \qquad \alpha_{\mathrm{ATM},i} = \eta_1 \,. \tag{10.175}$$

with $0 < \eta_1 < 1$. In the following, this type of model output will be referred to as '2input-$I_{\mathbf{M}}$/$I_{\mathbf{P}}$-1'. The corresponding fuzzy output A is shown in Fig. 10.18(a). The

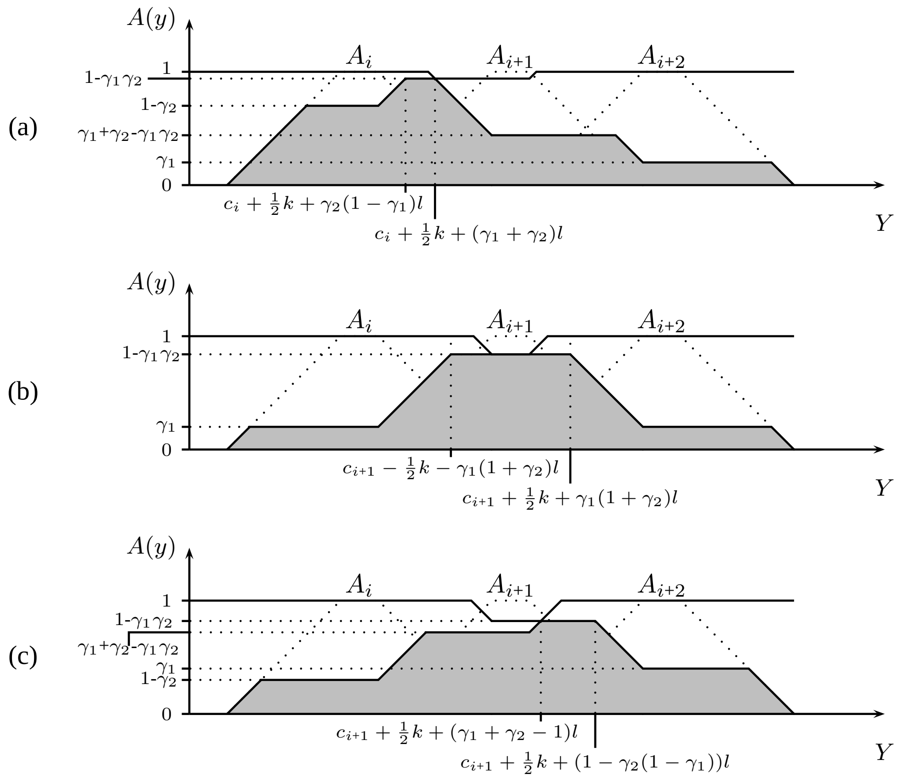


Figure 10.16: Schematic representation of the output obtained with a model applying $T_{\mathbf{P}}$ combined with $I_{\mathbf{L}}$ for input vectors firing the four rules 'Case V' and characterized by $\gamma_1 < \gamma_2$ and (a) $\gamma_1 + \gamma_2 < 1$, (b) $\gamma_1 + \gamma_2 = 1$ or (c) $\gamma_1 + \gamma_2 > 1$.

crisp output y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = c_i + \frac{1}{2}(1 - \eta_1)l\,. \tag{10.176}$$

The second type are fuzzy outputs that are the intersection of adapted membership functions corresponding to the following fulfilment degrees

$$\alpha_{\mathrm{ATL},i} = \eta_1 \qquad \alpha_{\mathrm{ATM},i} = 1\,. \tag{10.177}$$

with $0 < \eta_1 < 1$. In the following, this type of model output will be referred to as '2input-$I_{\mathbf{M}}$/$I_{\mathbf{P}}$-2'. The corresponding fuzzy output A is shown in Fig. 10.18(b). The crisp output y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = c_i - \frac{1}{2}(1 - \eta_1)l\,. \tag{10.178}$$

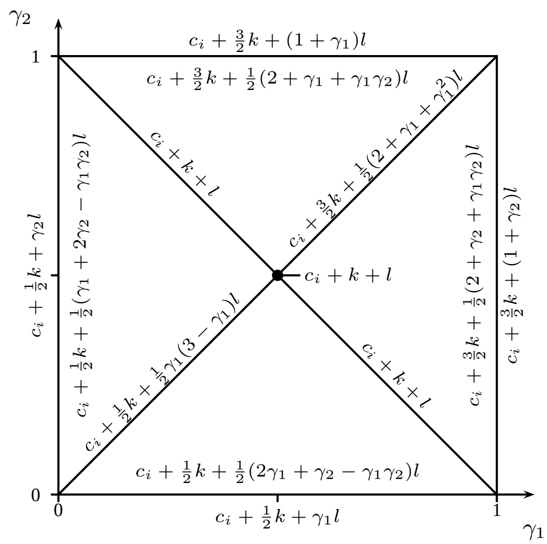


Figure 10.17: Overview of the expressions obtained for y^*_{MOM} with a model applying $T_{\mathbf{P}}$ combined with $I_{\mathbf{L}}$ and input vectors firing the four rules 'Case V'.

The third type are fuzzy outputs that are the intersection of adapted membership functions corresponding to the following fulfilment degrees

$$\alpha_{\mathrm{ATL},i} = \eta_1 \qquad \alpha_{\mathrm{ATM},i} = \eta_1 \,. \tag{10.179}$$

with $0 < \eta_1 < 1$. In the following, this type of model output will be referred to as '2input-$I_{\mathbf{M}}$/$I_{\mathbf{P}}$-3'. The corresponding fuzzy output A is shown in Fig. 10.18(c). The crisp output y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = c_i \,. \tag{10.180}$$

The fourth type are fuzzy outputs that are the intersection of adapted membership functions corresponding to the following fulfilment degrees

$$\alpha_{\mathrm{ATL},i+1} = \eta_2 \qquad \alpha_{\mathrm{ATM},i} = \eta_1 \,. \tag{10.181}$$

with $0 < \eta_2 < 1 - \eta_1 < 1$. In the following, this type of model output will be referred to as '2input-$I_{\mathbf{M}}$/$I_{\mathbf{P}}$-4'. The corresponding fuzzy output A is shown in Fig. 10.18(d). The crisp output y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = c_i + \frac{1}{2}k + \frac{1}{2}(1 - \eta_1 + \eta_2)l \,. \tag{10.182}$$

For Cases II and IV, with $(p_1, p_2, p_3) = (0, 0, 1)$ and $(p_1, p_2, p_3) = (1, 0, 0)$ respectively, the four considered rules contain linguistic output values derived from A_i and A_{i+1} in their consequent. The corresponding fulfilment degrees obtained for

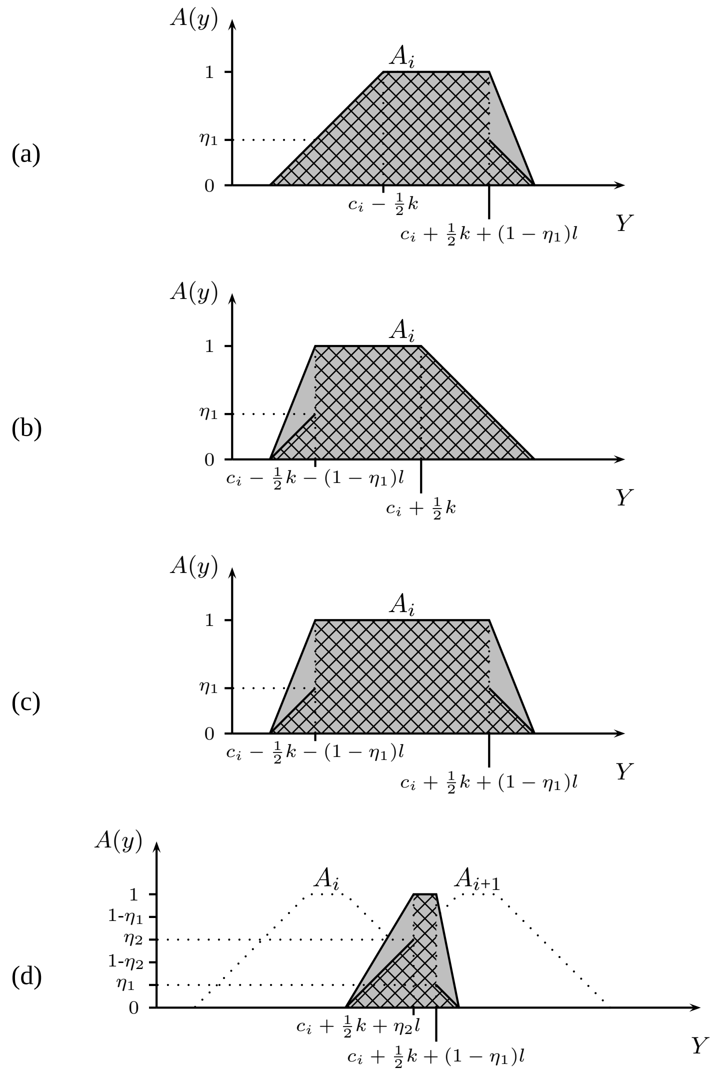


Figure 10.18: Schematic representation of the three types of model outputs (a) '2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-1', (b) '2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-2', (c) '2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-3' and (d) '2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-4' for models applying $I_{\mathbf{M}}$ (crosshatched) or $I_{\mathbf{P}}$ (in gray).

Table 10.3: Definitions of the regions of the input space where fulfilment degrees are described by different functions of γ_1 and γ_2.

Case	conditions on (γ_1, γ_2)
a	$\gamma_1 = 0$, $\gamma_2 \in]0, 1[$
b	$\gamma_1 \in]0, 1[$, $\gamma_2 = 1$
c	$\gamma_1 = 1$, $\gamma_2 \in]0, 1[$
d	$\gamma_1 \in]0, 1[$, $\gamma_2 = 0$
e	$\gamma_1 \in]0, 0.5[$, $\gamma_2 \in]0, 1[$, $\gamma_1 \le \gamma_2$, $\gamma_1 + \gamma_2 < 1$
f	$\gamma_1 \in]0, 0.5]$, $\gamma_2 \in [0.5, 1[$, $\gamma_1 \le \gamma_2$, $\gamma_1 + \gamma_2 = 1$
g	$\gamma_1 \in]0, 1[$, $\gamma_2 \in]0.5, 1[$, $\gamma_1 \le \gamma_2$, $\gamma_1 + \gamma_2 > 1$
h	$\gamma_1 \in]0.5, 1[$, $\gamma_2 \in]0, 1[$, $\gamma_1 \ge \gamma_2$, $\gamma_1 + \gamma_2 > 1$
i	$\gamma_1 \in [0.5, 1[$, $\gamma_2 \in]0, 0.5]$, $\gamma_1 \ge \gamma_2$, $\gamma_1 + \gamma_2 = 1$
j	$\gamma_1 \in]0, 1[$, $\gamma_2 \in]0, 0.5[$, $\gamma_1 \ge \gamma_2$, $\gamma_1 + \gamma_2 < 1$

Case II are given in Eqs. (10.166–10.169). For the t-norm $T_{\mathbf{L}}$ the fulfilment degrees are given by

$$\begin{aligned} \alpha_{\mathrm{ATL},i} &= 1 & \alpha_{\mathrm{ATM},i} &= \max(1-\gamma_1, 1-\gamma_2)\,, \\ \alpha_{\mathrm{ATL},i+1} &= \max(\gamma_1+\gamma_2-1, 0) & \alpha_{\mathrm{ATM},i+1} &= 1\,. \end{aligned} \tag{10.183}$$

The corresponding fulfilment degrees obtained for Case IV are given in Eqs. (10.170–10.173). For the t-norm $T_{\mathbf{L}}$ the fulfilment degrees are given by

$$\begin{aligned} \alpha_{\mathrm{ATL},i} &= 1 & \alpha_{\mathrm{ATM},i} &= \max(1-\gamma_1-\gamma_2, 0)\,, \\ \alpha_{\mathrm{ATL},i+1} &= \max(\gamma_1, \gamma_2) & \alpha_{\mathrm{ATM},i+1} &= 1\,. \end{aligned} \tag{10.184}$$

The fulfilment degrees $\alpha_{\mathrm{ATL},i}$, $\alpha_{\mathrm{ATL},i+1}$, $\alpha_{\mathrm{ATM},i}$ and $\alpha_{\mathrm{ATM},i+1}$ obtained for Cases II and IV in the different parts of the input space defined in Table 10.3 are listed in Table 10.4. The fulfilment degrees corresponding to an adapted membership function that does not determine the fuzzy output A_{ATL} or A_{ATM} of models applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ are put in *round or square* brackets. In all subcases the same expressions are obtained for y^*_{MOM} regardless if $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ is applied as implicator. Models applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{L}}$ are discussed in Section 10.5.7. The derivatives to γ_1 or γ_2 of all expressions for y^*_{MOM} in Table 10.4 are positive. Furthermore, y^*_{MOM} increases for any $\gamma_2 \in]0, 1[$ (resp. $\gamma_1 \in]0, 1[$) when γ_1 (resp. γ_2) is increased from 0 to 1, as illustrated in Fig. 10.19. Thus, monotonicity is guaranteed for Cases II and IV.

For Case V, with $(p_1, p_2, p_3) = (1, 0, 1)$, the four considered rules contain linguistic output values derived from A_i, A_{i+1} and A_{i+2} in their consequent. The corresponding fulfilment degrees are given in Eqs. (10.145–10.150). For the t-norm $T_{\mathbf{L}}$ they are given by

$$\begin{aligned} \alpha_{\mathrm{ATL},i} &= 1 & \alpha_{\mathrm{ATM},i} &= \max(1-\gamma_1-\gamma_2, 0)\,, \\ \alpha_{\mathrm{ATL},i+1} &= \max(\gamma_1, \gamma_2) & \alpha_{\mathrm{ATM},i+1} &= \max(1-\gamma_1, 1-\gamma_2)\,, \\ \alpha_{\mathrm{ATL},i+2} &= \max(\gamma_1+\gamma_2-1, 0) & \alpha_{\mathrm{ATM},i+2} &= 1\,. \end{aligned} \tag{10.185}$$

Table 10.4: Fulfilment degrees $\alpha_{\mathrm{ATL},i}$, $\alpha_{\mathrm{ATL},i+1}$, $\alpha_{\mathrm{ATM},i}$ and $\alpha_{\mathrm{ATM},i+1}$ and crisp output y^*_{MOM} obtained in the different parts of the input space defined in Table 10.3 for Cases II and IV with a model applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$.

Case	$\alpha_{\mathrm{ATL},i}$	$\alpha_{\mathrm{ATL},i+1}$	$\alpha_{\mathrm{ATM},i}$	$\alpha_{\mathrm{ATM},i+1}$	$I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ corresponds to	$I_{\mathbf{L}}$ corresponds to	y^*_{MOM}
II a	1	[0]	1	[1]	1input-constant	1input-constant	c_i
II b	(1)	γ_1	$1-\gamma_1$	(1)	1input-smooth	1input-smooth	$c_i+\frac{1}{2}k+\gamma_1 l$
II c	(1)	γ_2	$1-\gamma_2$	(1)	1input-smooth	1input-smooth	$c_i+\frac{1}{2}k+\gamma_2 l$
II d	1	[0]	1	[1]	1input-constant	1input-constant	c_i
II e–f	1	[0]	$1-\gamma_1$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-1	2input-$I_{\mathbf{L}}$-2	$c_i+\frac{1}{2}\gamma_1 l$
II g	(1)	$\gamma_1+\gamma_2-1$	$1-\gamma_1$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-4	2input-$I_{\mathbf{L}}$-1	$c_i+\frac{1}{2}k+\frac{1}{2}(2\gamma_1+\gamma_2-1)l$
II h	(1)	$\gamma_1+\gamma_2-1$	$1-\gamma_2$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-4	2input-$I_{\mathbf{L}}$-1	$c_i+\frac{1}{2}k+\frac{1}{2}(\gamma_1+2\gamma_2-1)l$
II i–j	1	[0]	$1-\gamma_2$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-1	2input-$I_{\mathbf{L}}$-2	$c_i+\frac{1}{2}\gamma_2 l$
IV a	(1)	γ_2	$1-\gamma_2$	(1)	1input-smooth	1input-smooth	$c_i+\frac{1}{2}k+\gamma_2 l$
IV b	[1]	1	[0]	1	1input-constant	1input-constant	c_i+k+l
IV c	[1]	1	[0]	1	1input-constant	1input-constant	c_i+k+l
IV d	(1)	γ_1	$1-\gamma_1$	(1)	1input-smooth	1input-smooth	$c_i+\frac{1}{2}k+\gamma_1 l$
IV e	(1)	γ_2	$1-\gamma_1-\gamma_2$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-4	2input-$I_{\mathbf{L}}$-1	$c_i+\frac{1}{2}k+\frac{1}{2}(\gamma_1+2\gamma_2)l$
IV f–g	(1)	γ_2	[0]	1	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-2	2input-$I_{\mathbf{L}}$-3	$c_i+k+\frac{1}{2}(1+\gamma_2)l$
IV h–i	(1)	γ_1	[0]	1	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-2	2input-$I_{\mathbf{L}}$-3	$c_i+k+\frac{1}{2}(1+\gamma_1)l$
IV j	(1)	γ_1	$1-\gamma_1-\gamma_2$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-4	2input-$I_{\mathbf{L}}$-1	$c_i+\frac{1}{2}k+\frac{1}{2}(2\gamma_1+\gamma_2)l$

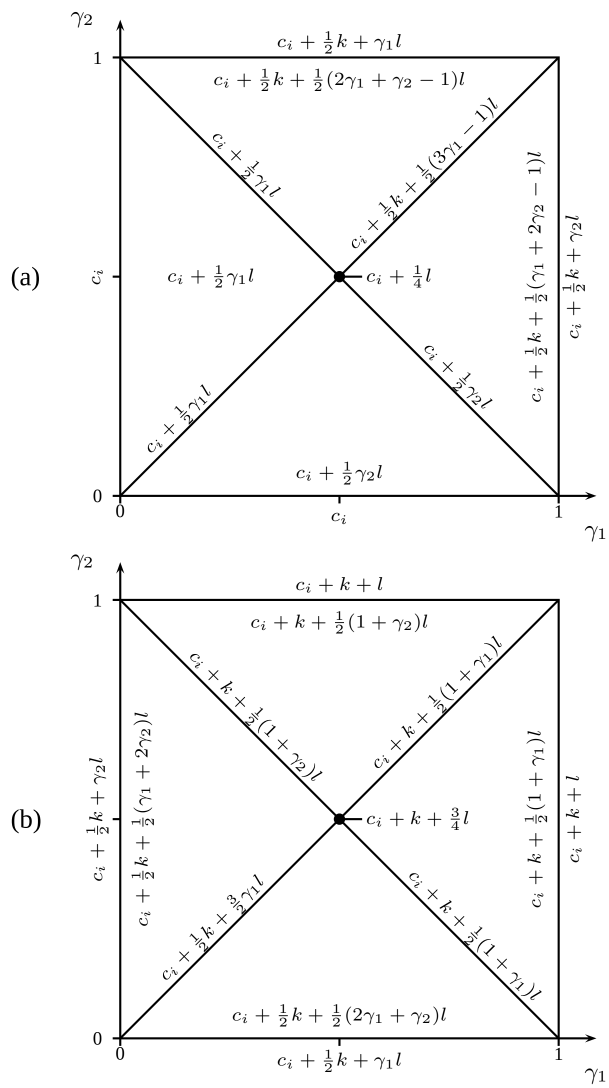


Figure 10.19: Overview of the expressions obtained for y^*_{MOM} with a model applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ and input vectors firing the four rules (a) ‘Case II’ and (b) ‘Case IV’.

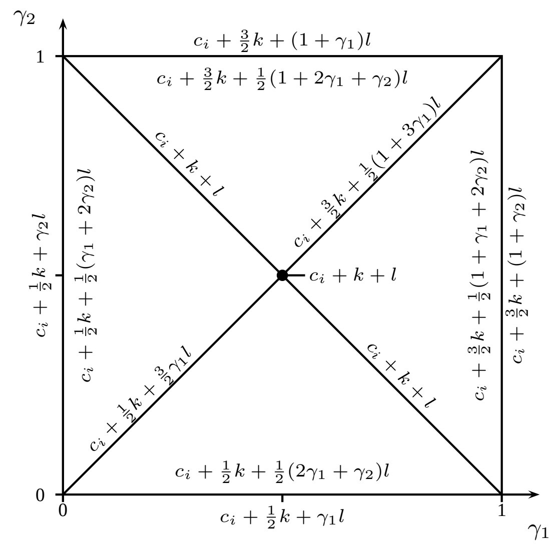


Figure 10.20: Overview of the expressions obtained for y^*_{MOM} with a model applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ and input vectors firing the four rules 'Case V'.

The fulfilment degrees $\alpha_{\mathrm{ATL},i}$, $\alpha_{\mathrm{ATL},i+1}$, $\alpha_{\mathrm{ATL},i+2}$, $\alpha_{\mathrm{ATM},i}$, $\alpha_{\mathrm{ATM},i+1}$ and $\alpha_{\mathrm{ATM},i+2}$ obtained for Case V in the different parts of the input space defined in Table 10.3 are listed in Table 10.5. The fulfilment degrees corresponding to an adapted membership function that does not determine the fuzzy output A_{ATL} or A_{ATM} of models applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$ are put in round brackets. The derivatives to γ_1 or γ_2 of all expressions for y^*_{MOM} are positive. Furthermore, y^*_{MOM} increases for any $\gamma_2 \in]0,1[$ (resp. $\gamma_1 \in]0,1[$) when γ_1 (resp. γ_2) is increased from 0 to 1, as illustrated in Fig. 10.20. Thus, monotonicity is guaranteed for Case V.

Summarizing, the results in Sections 10.5.1–10.5.2 and in this section show that monotonicity is guaranteed for models with any monotone smooth rule base when applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$.

10.5.7 Models applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{L}}$

In order to prove that monotonicity is guaranteed for models with any monotone smooth rule base applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{L}}$ it still needs to be proved that monotonicity is guaranteed in Cases II, IV and V. Before starting the discussion about the model behaviour is these cases, first the crisp model output y^*_{MOM} is determined of a model applying $I_{\mathbf{L}}$ for two model output types.

The first type are fuzzy outputs that are the intersection of adapted membership

Table 10.5: Fulfilment degrees $\alpha_{\mathrm{ATL},i}$, $\alpha_{\mathrm{ATL},i+1}$, $\alpha_{\mathrm{ATL},i+2}$, $\alpha_{\mathrm{ATM},i}$, $\alpha_{\mathrm{ATM},i+1}$ and $\alpha_{\mathrm{ATM},i+2}$ and crisp output y^*_{MOM} in the different parts of the input space defined in Table 10.3 for Case V with a model applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$.

Case	$\alpha_{\mathrm{ATL},i}$	$\alpha_{\mathrm{ATL},i+1}$	$\alpha_{\mathrm{ATL},i+2}$	$\alpha_{\mathrm{ATM},i}$	$\alpha_{\mathrm{ATM},i+1}$	$\alpha_{\mathrm{ATM},i+2}$	corresponds to	y^*_{MOM}
V a	(1)	γ_2	(0)	$1-\gamma_2$	(1)	(1)	1input-smooth	$c_i + \frac{1}{2}k + \gamma_2 l$
V b	(1)	(1)	γ_1	(0)	$1-\gamma_1$	(1)	1input-smooth	$c_i + \frac{3}{2}k + (1+\gamma_1)l$
V c	(1)	(1)	γ_2	(0)	$1-\gamma_2$	(1)	1input-smooth	$c_i + \frac{3}{2}k + (1+\gamma_2)l$
V d	(1)	γ_1	(0)	$1-\gamma_1$	(1)	(1)	1input-smooth	$c_i + \frac{1}{2}k + \gamma_1 l$
V e	(1)	γ_2	(0)	$1-\gamma_1-\gamma_2$	$(1-\gamma_1)$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-4	$c_i + \frac{1}{2}k + \frac{1}{2}(\gamma_1+2\gamma_2)l$
V f	(1)	γ_2	(0)	(0)	γ_2	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-3	$c_i + k + l$
V g	(1)	(γ_2)	$\gamma_1+\gamma_2-1$	(0)	$1-\gamma_1$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-4	$c_i + \frac{3}{2}k + \frac{1}{2}(1+2\gamma_1+\gamma_2)l$
V h	(1)	(γ_1)	$\gamma_1+\gamma_2-1$	(0)	$1-\gamma_2$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-4	$c_i + \frac{3}{2}k + \frac{1}{2}(1+\gamma_1+2\gamma_2)l$
V i	(1)	γ_1	(0)	(0)	γ_1	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-3	$c_i + k + l$
V j	(1)	γ_1	(0)	$1-\gamma_1-\gamma_2$	$(1-\gamma_2)$	(1)	2input-$I_{\mathbf{M}}/I_{\mathbf{P}}$-4	$c_i + \frac{1}{2}k + \frac{1}{2}(2\gamma_1+\gamma_2)l$

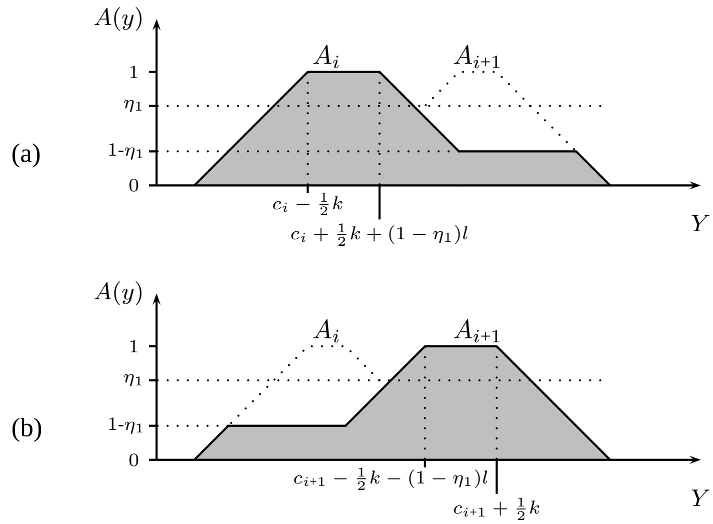


Figure 10.21: Schematic representation of the two types of model outputs (a) '2input-$I_{\mathbf{L}}$-2' and (b) '2input-$I_{\mathbf{L}}$-3' for models applying $I_{\mathbf{L}}$.

functions corresponding to the following fulfilment degrees

$$\begin{aligned} \alpha_{\mathrm{ATL},i} &= 1 \qquad & \alpha_{\mathrm{ATM},i} &= \eta_1\,, \\ \alpha_{\mathrm{ATL},i+1} &= 0 \qquad & \alpha_{\mathrm{ATM},i+1} &= 1\,. \end{aligned} \tag{10.186}$$

In the following, this type of model output will be referred to as '2input-$I_{\mathbf{L}}$-2'. The corresponding fuzzy output A is shown in Fig. 10.21(a). The crisp output y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = c_i + \frac{1}{2}(1-\eta_1)l\,. \tag{10.187}$$

The second type are fuzzy outputs that are the intersection of adapted membership functions corresponding to the following fulfilment degrees

$$\begin{aligned} \alpha_{\mathrm{ATL},i} &= 1 \qquad & \alpha_{\mathrm{ATM},i} &= 0\,, \\ \alpha_{\mathrm{ATL},i+1} &= \eta_1 \qquad & \alpha_{\mathrm{ATM},i+1} &= 1\,. \end{aligned} \tag{10.188}$$

In the following, this type of model output will be referred to as '2input-$I_{\mathbf{L}}$-3'. The corresponding fuzzy output A is shown in Fig. 10.21(b). The crisp output y^*_{MOM} is given by

$$y^*_{\mathrm{MOM}} = c_i + k + \frac{1}{2}(1+\eta_1)l\,. \tag{10.189}$$

For Cases II and IV, with $(p_1, p_2, p_3) = (0, 0, 1)$ and $(p_1, p_2, p_3) = (1, 0, 0)$ respectively, the four considered rules contain linguistic output values derived from

A_i and A_{i+1} in their consequent. The corresponding fulfilment degrees obtained in the different parts of the input space defined in Table 10.3 are listed in Table 10.4. The fulfilment degrees corresponding to an adapted membership function that does not determine the fuzzy output A_{ATL} or A_{ATM} of models applying $I_{\mathbf{L}}$ are put in *round* brackets. The derivatives to γ_1 or γ_2 of all expressions for y^*_{MOM} in Table 10.4 are positive. Furthermore, y^*_{MOM} increases for any $\gamma_2 \in\,]0,1[$ (resp. $\gamma_1 \in\,]0,1[$) when γ_1 (resp. γ_2) is increased from 0 to 1, as illustrated in Fig. 10.19. Thus, monotonicity is guaranteed for Cases II and IV.

For Case V, with $(p_1, p_2, p_3) = (1, 0, 1)$, the four considered rules contain linguistic output values derived from A_i, A_{i+1} and A_{i+2} in their consequent. The corresponding fulfilment degrees are given in Eqs. (10.145–10.150). For the t-norm $T_{\mathbf{L}}$ they are given by

$$
\begin{aligned}
\alpha_{\mathrm{ATL},i} &= 1 & \alpha_{\mathrm{ATM},i} &= \max(1-\gamma_1-\gamma_2, 0)\,,\\
\alpha_{\mathrm{ATL},i+1} &= \max(\gamma_1,\gamma_2) & \alpha_{\mathrm{ATM},i+1} &= \max(1-\gamma_1, 1-\gamma_2)\,,\\
\alpha_{\mathrm{ATL},i+2} &= \max(\gamma_1+\gamma_2-1, 0) & \alpha_{\mathrm{ATM},i+2} &= 1\,. \qquad (10.190)
\end{aligned}
$$

In Fig. 10.22 the fuzzy output A is shown for input vectors $\mathbf{x}$ characterized by $\gamma_1 < \gamma_2$ and

(a) $\gamma_1 + \gamma_2 < 1$,

(b) $\gamma_1 + \gamma_2 = 1$, or,

(c) $\gamma_1 + \gamma_2 > 1$.

Similar fuzzy outputs are obtained for input vectors $\mathbf{x}$ for which $\gamma_1 = \gamma_2$ or $\gamma_1 > \gamma_2$. An overview of the obtained expressions for y^*_{MOM} in the different parts of the input space is given in Fig. 10.20. The derivatives to γ_1 or γ_2 of all expressions for y^*_{MOM} are positive. Furthermore, y^*_{MOM} increases for any $\gamma_2 \in\,]0,1[$ (resp. $\gamma_1 \in\,]0,1[$) when γ_1 (resp. γ_2) is increased from 0 to 1, as illustrated in Fig. 10.20. Thus, monotonicity is guaranteed for Case V.

Summarizing, the results in Sections 10.5.1–10.5.2 and in this section show that monotonicity is guaranteed for models with any monotone smooth rule base when applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{L}}$.

10.5.8 Overview

Below an overview is given of the results obtained in Section 10.5. Combinations of t-norm and implicator for which monotonicity can be guaranteed for models with two input variables and any monotone smooth rule base are indicated with a ‘yes’. Combinations for which monotonicity cannot be guaranteed for any monotone smooth rule base are indicated with a ‘no’.

	$T_{\mathbf{M}}$	$T_{\mathbf{P}}$	$T_{\mathbf{L}}$
$I_{\mathbf{M}}$	no	no	yes
$I_{\mathbf{P}}$	no	no	yes
$I_{\mathbf{L}}$	no	yes	yes

(a)

(b)

(c)

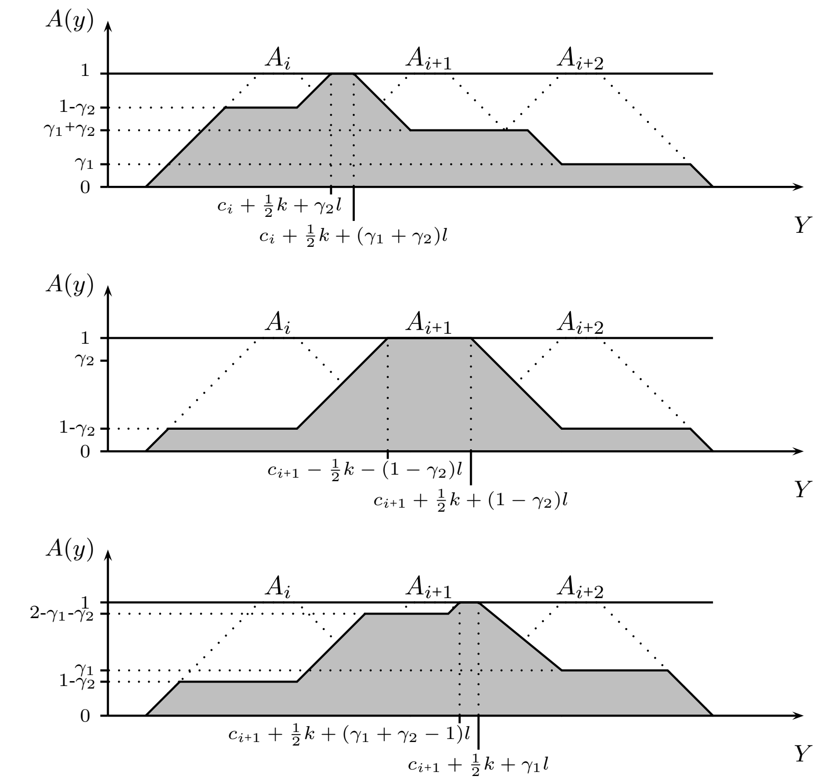


Figure 10.22: Schematic representation of the output obtained with a model applying $T_{\mathbf{L}}$ combined with $I_{\mathbf{L}}$ for input vectors firing the four rules 'Case V' and characterized by $\gamma_1 < \gamma_2$ and (a) $\gamma_1 + \gamma_2 < 1$, (b) $\gamma_1 + \gamma_2 = 1$ or (c) $\gamma_1 + \gamma_2 > 1$.

10.6 Models with two input variables and a monotone smooth rule base

As shown by the counterexample below, monotonicity cannot be guaranteed for any monotone rule base when applying ATL–ATM inference to models with two input variables using one of the nine considered combinations of t-norm and implicator. For the nine combinations of the t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ and implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$ only two combinations should be considered when studying the monotonicity of models with two input variables and a monotone (non-smooth) rule base, *i.e.* $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$ combined with $I_{\mathbf{L}}$. Since, firstly, in Section 10.4.3 models with a single input variable are shown to return the empty set as fuzzy output in the non-smooth case when applying $I_{\mathbf{M}}$ or $I_{\mathbf{P}}$, and, secondly, in Section 10.5.4 it is shown that monotonicity cannot be guaranteed for models with two input variables and any monotone smooth rule base when applying $T_{\mathbf{M}}$ combined with $I_{\mathbf{L}}$.

The set of four rules

IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+1}
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2}$	THEN	Y IS A_i
IF	X_1 IS $B^1_{j_1+1}$	AND	X_2 IS $B^2_{j_2+1}$	THEN	Y IS A_{i+2}

can occur in a monotone non-smooth rule base. For all inputs $\mathbf{x} = (x_1, x_2)$ satisfying

$$\gamma_1 = 1 - B^1_{j_1}(x_1) = B^1_{j_1+1}(x_1)\,, \tag{10.191}$$

$$\gamma_2 = 1 - B^2_{j_2}(x_2) = B^2_{j_2+1}(x_2)\,, \tag{10.192}$$

with $(\gamma_1, \gamma_2) \in [0,1]^2$, the fulfilment degrees $\alpha_{\text{ATL},i}$, $\alpha_{\text{ATL},i+1}$ and $\alpha_{\text{ATL},i+2}$ are given by

$$\alpha_{\text{ATL},i} = \max(\beta_{\text{ATL},1}, \beta_{\text{ATL},3}) = \max(1, \gamma_1) = 1\,, \tag{10.193}$$

$$\alpha_{\text{ATL},i+1} = \beta_{\text{ATL},2} = \gamma_2\,, \tag{10.194}$$

$$\alpha_{\text{ATL},i+2} = \beta_{\text{ATL},4} = T(\gamma_1, \gamma_2)\,, \tag{10.195}$$

and the fulfilment degrees $\alpha_{\text{ATM},i}$, $\alpha_{\text{ATM},i+1}$ and $\alpha_{\text{ATM},i+2}$ are given by

$$\alpha_{\text{ATM},i} = \max(\beta_{\text{ATM},1}, \beta_{\text{ATM},3}) = \max(T(1-\gamma_1, 1-\gamma_2), 1-\gamma_2) = 1-\gamma_2\,, \tag{10.196}$$

$$\alpha_{\text{ATM},i+1} = \beta_{\text{ATM},2} = 1-\gamma_1\,, \tag{10.197}$$

$$\alpha_{\text{ATM},i+2} = \beta_{\text{ATM},4} = 1\,, \tag{10.198}$$

For two input vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ defined by $(\gamma_1, \gamma_2) = (\eta_1, \eta_2)$ and $(\gamma_1, \gamma_2) = (1, \eta_2)$ respectively with

$$\eta_2 < 0.5 < \eta_1\,, \tag{10.199}$$

$$2\eta_1 + \eta_2 > 2\,, \tag{10.200}$$

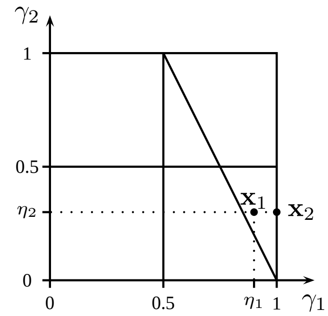


Figure 10.23: Indication of the two input vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ considered in the discussion about models with two input variables and a monotone non-smooth rule base.

as indicated in the (γ_1, γ_2)-plane in Fig. 10.23, the following inequality should hold in order to obtain a monotone input-output behaviour

$$y^*_{\mathrm{MOM}}(\mathbf{x}_1) \leq y^*_{\mathrm{MOM}}(\mathbf{x}_2)\,. \tag{10.201}$$

For $\mathbf{x}_1$, the fulfilment degrees are given by

$$\alpha_{\mathrm{ATL},i} = 1 \qquad \alpha_{\mathrm{ATM},i} = 1 - \eta_2\,, \tag{10.202}$$
$$\alpha_{\mathrm{ATL},i+1} = \eta_2 \qquad (\alpha_{\mathrm{ATM},i+1} = 1 - \eta_1)\,, \tag{10.203}$$
$$\alpha_{\mathrm{ATL},i+2} = T(\eta_1, \eta_2) \qquad \alpha_{\mathrm{ATM},i+2} = 1\,. \tag{10.204}$$

with $T_{\mathbf{P}}(\eta_1, \eta_2) = \eta_1\eta_2$ and $T_{\mathbf{L}}(\eta_1, \eta_2) = \eta_1 + \eta_2 - 1$. Both for $T = T_{\mathbf{P}}$ and $T = T_{\mathbf{L}}$, the following chain of inequalities holds for $\mathbf{x}_1$

$$\eta_2 < 0.5 < 1 - \eta_2 < 1 - T(\eta_1, \eta_2)\,, \tag{10.205}$$

and a fuzzy output A as illustrated in Fig. 10.24 is obtained, with corresponding crisp output $y^*_{\mathrm{MOM}}(\mathbf{x}_1)$ given by

$$y^*_{\mathrm{MOM}}(\mathbf{x}_1) = c_i + \frac{1}{2}k + \eta_2 l\,. \tag{10.206}$$

For $\mathbf{x}_2$ the fulfilment degrees are given by

$$\alpha_{\mathrm{ATL},i} = 1 \qquad \alpha_{\mathrm{ATM},i} = 1 - \eta_2\,, \tag{10.207}$$
$$(\alpha_{\mathrm{ATL},i+1} = \eta_2) \qquad (\alpha_{\mathrm{ATM},i+1} = 0)\,, \tag{10.208}$$
$$\alpha_{\mathrm{ATL},i+2} = \eta_2 \qquad \alpha_{\mathrm{ATM},i+2} = 1\,. \tag{10.209}$$

From Eq. (10.64) it follows that the adapted membership function $(\mathrm{ATL}(A_{i+1}))'$ will not determine the fuzzy output A_{ATL}. Furthermore, from Eq. (10.31) it follows that the adapted membership function $(\mathrm{ATM}(A_{i+1}))'$ is the universal set and will therefore

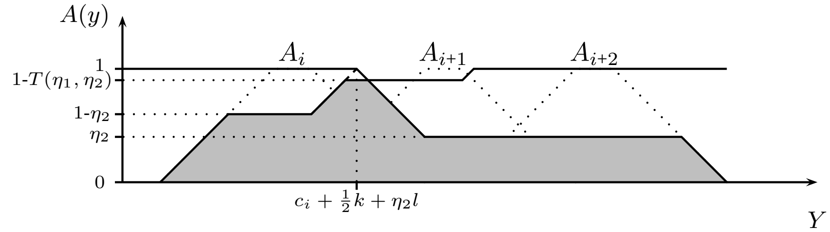


Figure 10.24: Schematic representation of the output obtained for the input vector $\mathbf{x}_1$ considered in the discussion about models with two input variables and a monotone non-smooth rule base.

not determine the fuzzy output A_{ATM}. Thus, the fuzzy output A obtained for $\mathbf{x}_2$ corresponds to the fuzzy output obtained for models with a single input variable in the non-smooth case (Section 10.4.3). The crisp output $y^*_{\mathrm{MOM}}(\mathbf{x}_2)$ is therefore given by

$$y^*_{\mathrm{MOM}}(\mathbf{x}_2) = c_i + \frac{1}{2}\eta_2 l\,. \tag{10.210}$$

Since l and η_2 are strictly positive and k is positive, it holds that

$$c_i + \frac{1}{2}k + \eta_2 l > c_i + \frac{1}{2}\eta_2 l\,. \tag{10.211}$$

Thus, Eq. (10.201) does not hold and a non-monotone input-output behaviour is obtained for inputs firing the four rules mentioned at the beginning of this section. Thus, monotonicity cannot be guaranteed for models with two input variables and any monotone rule base when applying one of the three t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ combined with one of the three implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$.

10.7 Conclusion

In this chapter, it was proved that an ATL–ATM model applying the MOM defuzzification method is monotone when the linguistic output values in the consequents of the rules are defined by trapezial or triangular membership functions of identical shape if it corresponds to one of the seven model types listed in Table 10.6, characterized by a number of input variables m, a t-norm T, an implicator I_T and an either monotone or monotone smooth rule base. For the implicators $I_{\mathbf{M}}$ and $I_{\mathbf{P}}$, models with a single input variable show a monotone input-output behaviour for any monotone *smooth* rule base, whereas for the implicator $I_{\mathbf{L}}$, models with a single input variable show a monotone input-output behaviour for any monotone rule base. When designing a monotone model with two input variables, one should opt for a monotone smooth rule base and apply the t-norm $T_{\mathbf{P}}$ combined with the implicator $I_{\mathbf{L}}$ or the t-norm $T_{\mathbf{L}}$ combined with one of the three considered implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$.

Table 10.6: ATL–ATM models for which monotonicity is guaranteed when applying the MOM defuzzification method characterized by a number of input variables m, a t-norm T, an implicator I_T and an either monotone or monotone smooth rule base.

	m	T	I_T	rule base
1	1		$I_{\mathbf{M}}$	monotone and smooth
2	1		$I_{\mathbf{P}}$	monotone and smooth
3	1		$I_{\mathbf{L}}$	monotone
4	2	$T_{\mathbf{P}}$	$I_{\mathbf{L}}$	monotone and smooth
5	2	$T_{\mathbf{L}}$	$I_{\mathbf{M}}$	monotone and smooth
6	2	$T_{\mathbf{L}}$	$I_{\mathbf{P}}$	monotone and smooth
7	2	$T_{\mathbf{L}}$	$I_{\mathbf{L}}$	monotone and smooth

The fact that the model behaviour of ATL–ATM models was studied for models with linguistic output values in the rule consequents defined by trapezial or triangular membership functions of identical shape, does not restrict the practical implementation of the results obtained in this chapter to models satisfying Eqs. (10.1–10.3). With the auxiliary interpolation procedure described in Section 8.6 a model designer can apply any fuzzy output partition in a monotone model when the model corresponds to one of the seven model types defined in Table 10.6.

Monotonicity of models with more than two input variables was not investigated in this study, but the obtained results show that for models with more than two input variables and one of the nine considered combinations of t-norm and implicator, only models should be considered with a monotone smooth rule base applying $T_{\mathbf{P}}$ combined with the implicator $I_{\mathbf{L}}$ or the t-norm $T_{\mathbf{L}}$ combined with either $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$. Furthermore, monotonicity of ATL–ATM models applying COG defuzzification and ATL–ATM models applying other implicators such as the S-implicators defined in Eq. (2.37), could be the subject of further investigation.

Part IV

Conclusion

CHAPTER 11

Conclusions and future research

Alles Wissen und alle Vermehrung unseres Wissens endet nicht mit einem Schlusspunkt, sondern mit Fragezeichen.
(Herman Hesse)

11.1 General Conclusions

This section gives an overview of the main conclusions of the research concerning the computational aspects of the Center of Gravity defuzzification method in Mamdani–Assilian models, the ecological case study and the research on monotone linguistic fuzzy models.

11.1.1 Computational aspects of Center of Gravity defuzzification

The Center of Gravity defuzzification method results in a crisp model output that changes continuously when the input values change continuously, a desirable property in modelling and control applications. However, the Center of Gravity defuzzification method has a high computational burden. In this dissertation two computational methods, the slope-based method and the modified transformation function method, were introduced to determine the crisp output of Mamdani–Assilian models using a fuzzy output partition of trapezial membership functions and applying the Center of Gravity defuzzification method. The accuracy, computational cost and implementational complexity of these two methods and the commonly applied discretization method were discussed for the basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$. Its easy implementation appears to be the only advantage of the discretization method. The two other methods to compute the Center of Gravity defuzzification method are not as straightforward to implement but allow both a quicker and more accurate computation. Of the three methods presented, the modified transformation function method has the smallest computational cost while being as accurate as the slope-based method.

11.1.2 Ecological case study

Fuzzy classifiers were applied to a modelling problem concerning the habitat suitability of river sites along springs to small rivers in the Central and Western Plains of Europe for 86 macroinvertebrate species. For each species, four models were developed, an A-, N-, P-, and C-model. The fuzzy classifiers take a certain width, velocity and either ammonium (A), nitrate (N) or phosphate (P) concentration or electrical conductivity (C) as input and return four values between 0 and 1 as output, indicating the degree to which the river site is considered 'not suitable' respectively 'lowly', 'moderately' and 'highly suitable' for the species to establish a population. With the developed models the influence on the habitat suitability can be assessed for the stream width and stream velocity, two variables determining the river type and reflecting the water quantity conditions at a river site, as well as for one aspect of the impact of human activities, *i.e.* the nutrient and organic load.

Field data collected at 445 sites in the Province of Overijssel (the Netherlands), referred to as the EKOO data set, allowed for an objective evaluation of the four developed models for 12 selected species. The fact that among them only one is an indicator for reference conditions, indicates that given the present environmental conditions of rivers in EU Member States, shifts in abundance levels of more common species are more suitable to detect gradual changes in water quality. With an improving water quality, the follow-up of indicator species with more narrow niches will gain importance. Of these 48 models, 16 models turned out to have a good model performance expressed by the performance measure % CFCI. These 16 good performing and objectively evaluated models are all, except one model, N- or P-models.

For the four models of the 12 selected species an optimization of the membership function parameters of the input variables was carried out. One type of interpretability-preserving data-driven optimization, as well as an accuracy-oriented optimization, were applied using both a binary-coded and a real-coded genetic algorithm. As fitness function the average deviation (AD), a new performance measure for fuzzy ordered classification, was used. For four models the binary-coded genetic algorithms returned less accurate solutions for the accuracy-oriented optimization than for the constrained optimization, due to the fact that the optimized membership function parameters only take values from a limited set of values. A shortcoming which, as shown by the experiments, can be remedied by applying a real-valued representation instead of a binary representation. The real-coded genetic algorithms applied in this study, however, showed maladjusted to eight of the 96 addressed membership function optimization problems, as an exhaustive investigation of the control structures of the genetic algorithms was outside the scope of this study. A purely accuracy-oriented optimization is no option when one wants to preserve the interpretability of the habitat suitability models under study with the EKOO data set. In this case, expert knowledge is a prerequisite to build interpretable models in order to define the rule bases and determine the optimization intervals of the membership function parameters. The accuracy-oriented optimization, however, gives a better insight in the driving force during the bounded optimization, *i.e.* the tendency to classify as much data points as possible in the abundance class $absent$ by increasing the regions where the input is mapped to $absent$, and stresses the importance of uniformly distributed and unambiguous training data for model optimization.

Fuzzy rule-based modelling showed to be of great value as a knowledge-based habitat suitability modelling technique in river management. The fuzzy sets allow working with vague information which makes them very suitable for the variables and criteria used in this application field. Moreover, the labels attached to the fuzzy sets are relevant for river management as they were inspired by the existing classifications used nowadays in bio-assessment and river typologies required by the Water Framework Directive. The structure of a fuzzy rule base allows for the incorporation of the information summarized in the knowledge base into an inference system for habitat suitability modelling, by expressing non-linear relations in terms of if-then rules. The degrees of membership to the different output classes provide the end-user with a quantification of the uncertainty associated with the model output. This information has an added value in decision support.

11.1.3 Monotone linguistic fuzzy models

A fuzzy model can be identified by combining quantitative with qualitative knowledge. First of all, qualitative knowledge allows us to obtain meaningful, interpretable models. Moreover, it permits a reduction of the search space of the data-driven model identification which renders the model identification process less vulnerable to noise and inconsistencies in the data and suppresses overfitting. This dissertation focussed on a common property of evaluation and selection procedures, namely on the monotonicity of the model output with respect to an input variable, *i.e.* the fact that the model output is either increasing or decreasing in the variable for all combinations of values of other input variables. More specifically, monotone models were studied in this work, *i.e.* models that are monotone in all input variables.

Models were assumed to apply a fuzzy partition of trapezial membership functions in all input domains as well as in the output domain, which imposes a natural order on the linguistic values of all variables, and to have a monotone rule base, *i.e.* to use a set of if-then rules describing a monotone relation between the input variables and the output variable. The monotonicity of linguistic fuzzy models under different inference procedures was discussed: Mamdani–Assilian inference, *plain* implicator-based inference and ATL–ATM inference. Mamdani–Assilian models applying one of the three basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$ combined with either the Center of Gravity or the Mean of Maxima defuzzification method were considered. Furthermore, models applying plain implicator-based inference or ATL–ATM inference, one of the three basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$, one of the three R-implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ and the Mean of Maxima defuzzification method, were studied. The objective of this study was to select, for each inference procedure, combinations of t-norm, implicator or defuzzification method resulting in a monotone input-output behaviour for any monotone rule base, or at least for any monotone smooth rule base.

For the assumed model properties, the input-output behaviour of models with m input variables reduces to the input-output behaviour of models with m^* ($m^* < m$) input variables in those regions of the input space where the inputs belong to the kernel of the same linguistic value in all but m^* input domains. Thus, if certain model properties are necessary to guarantee monotonicity for models with m^* input variables, these model properties are also required to guarantee a monotone input-output behaviour for models with more than m^* input variables. Furthermore, an auxiliary interpolation procedure was presented which allows for the extension of results obtained for models for which all linguistic output values in the rule consequents are defined by trapezial membership functions of identical shape to models with any fuzzy output partition of trapezial or triangular membership functions.

For a model with two input variables and a monotone rule base monotonicity cannot be guaranteed for the considered combinations of inference procedures, t-norms, implicators and defuzzification methods, except for Mamdani–Assilian inference combined with the t-norm $T_{\mathbf{P}}$ and the Mean of Maxima defuzzification method if, at least, the model satisfies additional constraints. For Mamdani–Assilian models with two input variables and any monotone rule base applying the Mean of Maxima defuzzification method, a monotone input-output behaviour can be guaranteed when using a fuzzy output partition corresponding to one of the following schemata { *, triangular, triangular, triangular, * }, { *, triangular, triangular, * } or { *, *, * } with * a membership function that might be either triangular or trapezial. When a system with two input variables is described by a monotone smooth rule base a wider range of inference procedures can be applied: Mamdani–Assilian inference with the t-norm $T_{\mathbf{P}}$ and the Center of Gravity or Mean of Maxima defuzzification method, Mamdani–Assilian inference with the t-norm $T_{\mathbf{M}}$ and the Mean of Maxima defuzzification method, ATL–ATM inference with the t-norm $T_{\mathbf{P}}$, the implicator $I_{\mathbf{L}}$ and the Mean of Maxima defuzzification method or ATL–ATM inference with the t-norm $T_{\mathbf{L}}$, the implicator $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ and the Mean of Maxima defuzzification method.

The monotonicity of ATL–ATM models with three or more input variables was not studied in this dissertation. For Mamdani–Assilian models applying the Center of Gravity defuzzification method, models with up to three input variables were investigated. It was proved that a monotone input-output behaviour is always obtained for Mamdani–Assilian models with three input variables and a monotone smooth rule base applying the t-norm $T_{\mathbf{P}}$ and the Center of Gravity defuzzification method when the linguistic output values in the consequents of the rules are defined by trapezial or triangular membership functions of identical shape. Furthermore, for Mamdani–Assilian models applying the Mean of Maxima defuzzification method, it was shown that monotonicity is guaranteed for models with a monotone smooth rule base applying $T_{\mathbf{M}}$ when the linguistic output values in the consequents of the rules are defined by trapezial or triangular membership functions with intervals of changing membership degree of equal length and for models with a monotone smooth rule base applying $T_{\mathbf{P}}$ for any fuzzy output partition.

11.2 Indications for future research

Monotone models in habitat suitability modelling Univariate preferences functions are commonly applied in habitat suitability modelling (Schneider, 2001). They describe the preference of a species for the values taken by a physical or chemical variable by mapping them to values between 0 (for completely unsuitable conditions) and 1 (for perfectly suitable conditions). The suitability of a site as a habitat for a certain species is then obtained by the product, arithmetic or geometric mean or minimum of the univariate preferences of the considered variables characterizing the site. In the fuzzy models developed in the ecological case study, if-then rules describe the relationship between a site's suitability and the variables characterizing the site. The habitat suitability of a site for a species can however also be expressed as a function of the univariate preferences of the species for the different variables characterizing the site, which results in a model with a monotone rule base.

Performance measures for fuzzy ordered classifiers The newly introduced performance measure for fuzzy ordered classifiers, average deviation, takes the order of the output classes into account by returning the average deviation between the position of the class obtained with the model and the position of the class stored in the data set. However, average deviation does not differentiate between deviations resulting from an over-classification, *i.e.* a classification in a too high class by the model compared to the data, and those resulting from an under-classification, *i.e.* a classification in a too low class by the model compared to the data. In the ecological case study over- and under-classification are however, in fact, not considered to be of equal importance when determining the global model performance since over-classifications could be the result of lower abundances for individual species due to competition between several species having similar habitat requirements and do therefore not necessarily indicate that the model badly describes the habitat suitability of a certain species.

ATL–ATM models with other implicators Apart from Mamdani–Assilian models applying the t-norm $T_\mathbf{P}$ and the Mean of Maxima defuzzification method and using a fuzzy output partition belonging to a restricted class of nine types of fuzzy partitions, monotonicity cannot be guaranteed for models with two or more input variables and any monotone non-smooth rule base. Most systems, however, are described by a set of if-then rules forming a monotone non-smooth rule base as was illustrated by the cited applications from the bioscience engineering field. It would therefore be interesting to investigate if monotonicity is guaranteed for ATL–ATM models with two or more input variables and any monotone rule base when applying other implicators, such as the S-implicators defined in Eq. (2.37), or the Center of Gravity defuzzification method.

Computational aspects of defuzzification in ATL–ATM models The modified transformation function method, allowing for an accurate computation of the crisp output of Mamdani–Assilian models applying the Center of Gravity defuzzification method, showed to be an essential tool when studying the monotonicity of these models. An accurate determination of the model output makes useful numerical experiments guiding the analytical analysis possible and hereby facilitates the analytical analysis of the model behaviour. Therefore, I recommend the development of a procedure to accurately determine the model output of ATL–ATM models before continuing (resp. starting) the study on the monotonicity of ATL–ATM models applying the Mean of Maxima defuzzification method (resp. Center of Gravity defuzzification method).

Identification of monotone linguistic fuzzy models The selection made in this dissertation of combinations of Mamdani–Assilian and ATL–ATM inference, t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$, implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ and $I_{\mathbf{L}}$ and the Mean of Maxima and Center of Gravity defuzzification method which guarantee monotonicity of models with a monotone rule base, could be used when developing a data-driven identification method for monotone linguistic fuzzy models.

11.3 Main contributions of this dissertation

This dissertation has tried to make contributions to both the ecological modelling as well as to the fuzzy modelling domain. These contributions are listed below.

To the ecological modelling community

- a fuzzy ordered classifier was applied for habitat suitability modelling
- the need of a data set including a similar number of examples for the different phenomena described by the model was illustrated

To the fuzzy modelling community

- an accurate and fast computational method was introduced for determining the crisp output of Mamdani–Assilian models applying the Center of Gravity defuzzification method and using fuzzy output partitions of trapezial membership functions
- a new performance measure for fuzzy ordered classifiers was presented taking the ordering of the output classes into account
- guidelines were formulated for designers of monotone linguistic fuzzy models
- a new inference procedure, called ATL–ATM inference, was introduced for linguistic fuzzy models with a monotone rule base

Part V

Appendices

APPENDIX A
List of macroinvertebrate taxa

In Table A.1 all 86 macroinvertebrate taxa considered in this study are listed. In the first column the index is given as used in this manuscript, followed by the full taxon name and the abbreviation used in this study in the second and third column. The twelve taxa selected for optimization of the membership functions are indicated in bold.

Table A.1: Macroinvertebrate taxa

	Taxon name	Taxon code		Taxon name	Taxon code
		Indicator species			
1	*Agabus didymus*	agabdidy	2	*Agabus guttatus*	agabgutt
3	*Agabus paludosus*	agabpalu	4	*Amphinemura sulcicolis*	amphsulc
5	*Anacaena globulus*	anacglob	6	*Ancyclus fluviatilus*	ancyfluv
7	*Baetis rhodani*	baetrhod	8	*Brillia longifurca*	brillong
9	*Crunoecia irrorata*	crunirro	10	*Dugesia gonocephala*	dugegono
11	*Elmis aenea*	elmiaena	12	*Elodes minuta*	elodminu
13	*Ephemera vulgata*	epravulg	14	*Gammarus roesellii*	gammroes
15	*Halesus radiatus*	haledira	16	*Hydroporus nigrita*	hyponigr
17	*Hydropsyche pellucidula*	hypspell	18	*Ironoquia dubia*	irondubi
19	*Limnephilus extricates*	liluextr	20	*Limnephilus fuscifornis*	lilufusc
21	*Limnephilus lunatus*	lilulluna	22	*Notidobia ciliaris*	nodocili
23	*Odontomesa fulva*	odmefulv	24	*Orectochillus villosus*	orecvill
25	***Physa fontinalis***	physfont	26	*Platambus maculatus*	pltamacu
27	*Plectrocnemia conspersa*	pltrcons	28	*Nebrioporus depressus*	ponedepr
29	*Rheocricotopus group fuscipes*	rhcrgfus	30	*Sericostoma personatum*	setopers
		Non-indicator species			
31	*Acroloxus lacustris*	aclolacu	32	*Agabus affinis*	agabaffi
33	*Agabus bipustulatus*	agabbipu	34	*Anabolia nervosa*	anabnerv

continued on next page

continued from previous page

	Taxon name	Taxon code		Taxon name	Taxon code
35	*Anacaena bipustulatus*	anacbipu	36	***Anisus vortex***	ansuvote
37	***Asellus aquaticus***	aselaqua	38	*Corixa punctata*	coripunc
39	*Dugesia lugubris/polychroa*	dugelupo	40	***Erpobdella octoculata***	erpoocto
41	*Galba trunculata*	galbtrun	42	***Gammarus pulex***	gammpule
43	*Gerris lacustris*	gerrlacu	44	*Glossiphonia complanata*	glsicomp
45	***Glossiphonia heteroclita***	glsihete	46	*Glyphotaelius pellucidus*	glphpell
47	*Haliplus flavicollis*	haliflav	48	*Haliplus fluviatilis*	halifluv
49	*Haliplus lineatocollis*	halilito	50	*Haementaria costata*	hamecost
51	***Helobdella stagnalis***	hebdstag	52	*Hemiclepsis marginata*	heclmarg
53	*Helophorus aquaticus/grandis*	heruaqgr	54	*Helophorus brevipalpis*	herubrev
55	*Hydroporus palustris*	hypopalu	56	*Hydropsyche angustipennis*	hypsangu
57	*Hygrotus inaequalis*	hytuinae	58	*Ilybius fenestratus*	ilybfene
59	*Ilybius fuliginosus*	ilybfuli	60	*Limnephilus rhombicus*	lilurhom
61	*Lype reducta*	lyperedu	62	*Notonecta glauca*	notoglau
63	*Physa acuta*	physacut	64	*Piscicola geometra*	piscgeom
65	*Planorbis carinatus*	plbicari	66	***Planorbis planorbis***	plbiplan
67	*Plectrocnemia geniculata*	pltrgeni	68	***Proasellus meridianus***	proameri
69	***Radix peregra***	radipere	70	*Sialis fuliginosa*	sialfuli
71	*Sialis lutaria*	sialluta	72	*Sigara falleni*	sigafall
73	*Sigara lateralis*	sigalate	74	*Sigara semistriata*	sigasemi
75	***Sigara striata***	sigastri	76	*Stagnicola palustris*	stagpalu
77	***Valvata piscinalis***	valvpisc	78	*Velia caprai*	velicapr
79	*Brillia modesta*	brilmode	80	*Aspectrotanypus trifascipennis*	apsetrif
81	*Dicrotendipes group notatus*	ditegnot	82	*Polypedilum laetum agg.*	popelaea
83	*Parametriocnemus stylatus*	paocstyl	84	*Aplexa hypnorum*	aplehypn
85	*Prodiamesa olivacea*	prodoliv	86	*Rhantus suturalis*	rhansura

Acroloxus lacustris (aclolacu,31)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
		oligosaprobic / oligotrophic / oligoionic			
stream velocity	low	Absent	Absent	Low	Low
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Absent	Low	Moderate	Moderate
	moderate	Absent	Absent	Low	Low
	high	Absent	Absent	Absent	Absent
		mesosaprobic / α-mesotrophic / mesoionic			
	low	Low	Moderate	High	High
	moderate	Absent	Low	Moderate	Moderate
	high	Absent	Absent	Low	Low
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Absent	Absent	Low	Low
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		polysaprobic / hypertrophic / polyionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Agabus affinis (agabaffi,32)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
		oligosaprobic / oligotrophic / oligoionic			
stream velocity	low	Absent	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Absent	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		mesosaprobic / α-mesotrophic / mesoionic			
	low	Low	Moderate	Low	Absent
	moderate	Absent	Low	Absent	Absent
	high	Absent	Absent	Absent	Absent
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Moderate	High	Moderate	Low
	moderate	Low	Moderate	Low	Absent
	high	Absent	Low	Absent	Absent
		polysaprobic / hypertrophic / polyionic			
	low	Absent	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Agabus guttatus (agabgutt, 2) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Low	Low	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Absent
high	Moderate	Moderate	Low	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	High	Moderate	Low	Absent
moderate	High	High	Moderate	Low
high	High	High	Moderate	Low
polysaprobic / hypertrophic / polyionic				
low	Low	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Moderate	Moderate	Low	Absent

Agabus bipustulatus (agabbipu,33) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Low	Low
moderate	Moderate	Moderate	Moderate	Moderate
high	Low	Low	Low	Low
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	Moderate	Moderate	Moderate	Moderate
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Low	Low	Low	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Agabus paludosus (agabpalu, 3) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Absent	Low	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Absent
high	Low	Moderate	Moderate	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	High	Moderate	Low	Absent
moderate	High	High	Moderate	Low
high	Moderate	High	High	Moderate
polysaprobic / hypertrophic / polyionic				
low	Low	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Low	Moderate	Moderate	Low

Agabus didymus (agabdidy, 1) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Low	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Low
moderate	Low	Low	Low	Low
high	Moderate	Moderate	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	Moderate	Moderate	Moderate	Moderate
high	High	High	High	High
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Low	Low

Anacaena bipustulatus (anacbipu,35)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Low	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Low	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Moderate	High	Moderate
moderate	Low	Low	Moderate	Low
high	Absent	Absent	Low	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Low	Low	Moderate	Low
moderate	Absent	Absent	Low	Absent
high	Absent	Absent	Low	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Low	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Amphinemura sulcicolis (amphsulc, 4)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Moderate	Moderate	Moderate
moderate	Moderate	Moderate	Moderate	Moderate
high	Moderate	High	High	High
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Moderate	Moderate	Moderate
moderate	Moderate	Moderate	Moderate	Moderate
high	Moderate	High	High	High
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Moderate	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Low	Low	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Anacaena globulus (anacglob, 5)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Absent
high	Absent	Low	Low	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Low	Moderate	Low
moderate	Low	Moderate	High	Moderate
high	Absent	Low	Moderate	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Low	Absent
moderate	Absent	Low	Moderate	Low
high	Absent	Absent	Low	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Anabolia nervosa (anabnerv,34)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Low	Low	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Low	Low	Absent
moderate	Low	Moderate	Moderate	Low
high	Absent	Low	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Moderate	Moderate	Low
moderate	Moderate	High	High	Moderate
high	Low	Moderate	Moderate	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Moderate	Moderate	Low
high	Low	Moderate	Moderate	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Low	Low	Absent
high	Absent	Low	Low	Absent

Aplexa hypnorum (aplehypn,84) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	High	High	Moderate	Low
moderate	Moderate	Moderate	Low	Absent
high	Low	Low	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	High	High	Moderate	Low
moderate	Moderate	Moderate	Low	Absent
high	Low	Low	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Moderate	Low	Absent
moderate	Low	Low	Absent	Absent
high	Low	Low	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Low	Low	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Ancyclus fluviatilus (ancyfluv, 6) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Low	Low
moderate	Absent	Absent	Low	Low
high	Absent	Low	Moderate	Moderate
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Moderate	Moderate
moderate	Low	Low	Moderate	Moderate
high	Low	Moderate	High	High
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Low	Low
moderate	Absent	Absent	Low	Low
high	Low	Moderate	High	High
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Low	Moderate	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Low	Low

Asellus aquaticus (aselaqua,37) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Low
moderate	Moderate	Moderate	Moderate	Moderate
high	Low	Low	Low	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	Moderate	Moderate	Moderate	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Low	Low	Low	Low

Anisus vortex (ansuvote,36) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Low	Low	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Moderate	Moderate	Low
moderate	Absent	Low	Low	Absent
high	Absent	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	High	High	Moderate
moderate	Low	Moderate	Moderate	Low
high	Absent	Low	Low	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Low	Low	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Brillia longifurca (brillong, 8) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Moderate	Moderate	Moderate
moderate	Low	Moderate	Moderate	Moderate
high	Moderate	High	High	High
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Low	Low	Low
moderate	Absent	Low	Low	Low
high	Moderate	High	High	High
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Low	Low	Low
moderate	Absent	Low	Low	Low
high	Low	High	High	High
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Moderate	Moderate	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Low	Low	Low

Aspectrotanypus trifascipennis (apsetrif,80) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	High	Moderate	Low	Absent
moderate	High	Moderate	Low	Absent
high	High	Moderate	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	High	Moderate	Low	Absent
moderate	High	Moderate	Low	Absent
high	High	Moderate	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Low	Absent	Absent
moderate	Moderate	Low	Absent	Absent
high	High	Moderate	Low	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Low	Absent	Absent	Absent
moderate	Low	Absent	Absent	Absent
high	Moderate	Low	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Absent	Absent	Absent

Brillia modesta (brilmode,79) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Moderate	Low	Absent	Absent
moderate	Moderate	Low	Absent	Absent
high	High	Moderate	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Moderate	Low	Absent	Absent
moderate	Moderate	Low	Absent	Absent
high	High	Moderate	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Low	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Baetis rhodani (baetrhod, 7) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Low	Low
moderate	Low	Low	Low	Low
high	Low	Moderate	Moderate	Moderate
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	Moderate	Moderate	Moderate	Moderate
high	Moderate	High	High	High
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Low
moderate	Low	Low	Low	Low
high	Moderate	High	High	High
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Moderate	Moderate	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Low	Low	Low

Dicrotendipes gr. notatus (`ditegnot,81`)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity		oligosaprobic / oligotrophic / oligoionic			
	low	Absent	Low	Low	Low
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Low	Moderate	Moderate	Moderate
	moderate	Absent	Low	Low	Low
	high	Absent	Absent	Absent	Absent
		mesosaprobic / α-mesotrophic / mesoionic			
	low	Moderate	High	High	High
	moderate	Low	Moderate	Moderate	Moderate
	high	Absent	Low	Low	Low
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Low	Moderate	Moderate	Moderate
	moderate	Absent	Low	Low	Low
	high	Absent	Absent	Absent	Absent
		polysaprobic / hypertrophic / polyionic			
	low	Absent	Low	Low	Low
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Corixa punctata (`coripunc,38`)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity		oligosaprobic / oligotrophic / oligoionic			
	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		mesosaprobic / α-mesotrophic / mesoionic			
	low	High	High	Moderate	Low
	moderate	Moderate	Moderate	Low	Absent
	high	Low	Low	Absent	Absent
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		polysaprobic / hypertrophic / polyionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Dugesia gonocephala (`dugegono,10`)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity		oligosaprobic / oligotrophic / oligoionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Low	Absent	Absent
	high	Low	Moderate	Low	Low
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Absent	Low	Absent	Absent
	moderate	Low	Moderate	Low	Absent
	high	Moderate	High	Moderate	Low
		mesosaprobic / α-mesotrophic / mesoionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Low	Moderate	Low	Absent
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Low	Absent	Absent
		polysaprobic / hypertrophic / polyionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Crunoecia irrorata (`crunirro, 9`)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity		oligosaprobic / oligotrophic / oligoionic			
	low	Absent	Absent	Absent	Absent
	moderate	Low	Absent	Absent	Absent
	high	Moderate	Low	Absent	Absent
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Low	Absent	Absent	Absent
	moderate	Moderate	Low	Absent	Absent
	high	High	Moderate	Low	Absent
		mesosaprobic / α-mesotrophic / mesoionic			
	low	Absent	Absent	Absent	Absent
	moderate	Low	Absent	Absent	Absent
	high	High	Moderate	Low	Absent
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Moderate	Low	Absent	Absent
		polysaprobic / hypertrophic / polyionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Low	Absent	Absent	Absent

Elodes minuta (elodminu,12)	stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Low	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic	low	Absent	Absent	Absent	Absent
	moderate	Low	Absent	Absent	Absent
	high	Moderate	Low	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic	low	Low	Absent	Absent	Absent
	moderate	Moderate	Low	Absent	Absent
	high	High	Moderate	Low	Absent
polysaprobic / hypertrophic / polyionic	low	Absent	Absent	Absent	Absent
	moderate	Low	Absent	Absent	Absent
	high	Moderate	Low	Absent	Absent

Dugesia lugubris/polychroa (dugelupo,39)	stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic	low	Absent	Absent	Low	Absent
	moderate	Absent	Low	Moderate	Low
	high	Absent	Absent	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic	low	Absent	Low	Moderate	Low
	moderate	Low	Moderate	High	Moderate
	high	Absent	Low	Moderate	Low
mesosaprobic / α-mesotrophic / mesoionic	low	Absent	Low	Moderate	Low
	moderate	Low	Moderate	High	Moderate
	high	Absent	Low	Moderate	Low
α-mesosaprobic / eutrophic / α-mesoionic	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Low	Absent
	high	Absent	Absent	Low	Absent
polysaprobic / hypertrophic / polyionic	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Ephemera vulgata (epravulg,13)	stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic	low	Absent	Absent	Low	Low
	moderate	Absent	Low	Low	Low
	high	Low	Moderate	High	High
β,α-oligosaprobic / β-mesotrophic / β-mesoionic	low	Low	Low	Moderate	Moderate
	moderate	Low	Moderate	Moderate	Moderate
	high	Low	Moderate	High	High
mesosaprobic / α-mesotrophic / mesoionic	low	Low	Low	Moderate	Moderate
	moderate	Low	Moderate	Moderate	Moderate
	high	Low	Moderate	High	High
α-mesosaprobic / eutrophic / α-mesoionic	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Low	Moderate	Moderate
polysaprobic / hypertrophic / polyionic	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Low	Low

Elmis aenea (elmiaena,11)	stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic	low	Moderate	Moderate	Moderate	Moderate
	moderate	Moderate	Moderate	Moderate	Moderate
	high	High	High	High	High
β,α-oligosaprobic / β-mesotrophic / β-mesoionic	low	Low	Low	Low	Low
	moderate	Low	Low	Low	Low
	high	High	High	High	High
mesosaprobic / α-mesotrophic / mesoionic	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Moderate	Moderate	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Low	Low	Low	Low
polysaprobic / hypertrophic / polyionic	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Gammarus pulex (`gammpule,42`) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Moderate	Moderate	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Absent
high	High	High	Moderate	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Absent
high	High	High	Moderate	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Moderate	Low	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Absent	Absent

Erpobdella octoculata (`erpoocto,40`) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Low
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Low
high	Absent	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Low
moderate	Absent	Low	Low	Moderate
high	Absent	Absent	Absent	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Low	Low	Moderate
moderate	Low	Moderate	Moderate	High
high	Absent	Low	Low	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Low
high	Absent	Absent	Absent	Low

Gammarus roesellii (`gammroes,14`) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Low
moderate	Absent	Absent	Absent	Low
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Low
moderate	Absent	Absent	Absent	Low
high	Low	Low	Low	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Moderate	High
moderate	Moderate	Moderate	Moderate	High
high	High	High	High	High
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Low	Moderate
moderate	Low	Low	Low	Moderate
high	Moderate	Moderate	Moderate	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Low
moderate	Absent	Absent	Absent	Low
high	Low	Low	Low	Low

Galba trunculata (`galbtrun,41`) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Absent
high	Low	Low	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Absent
high	Low	Low	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Moderate	Low	Absent
moderate	High	High	Moderate	Low
high	Moderate	Moderate	Low	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Low	Low	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Glossiphonia heteroclita (glsihete,45)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Low
high	Low	Low	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Low
high	Low	Low	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Moderate	Low	Low
moderate	High	High	Moderate	Moderate
high	Moderate	Moderate	Low	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Low	Low	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Gerris lacustris (gerrlacu,43)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Low	Low
moderate	Absent	Low	Moderate	Moderate
high	Absent	Absent	Low	Low
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Low	Moderate	Moderate
moderate	Low	Moderate	High	High
high	Absent	Low	Moderate	Moderate
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Low	Low
moderate	Absent	Low	Moderate	Moderate
high	Absent	Low	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Low
high	Absent	Absent	Low	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Glyphotaelius pellucidus (glphpell,46)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Moderate	Low	Low	Absent
moderate	High	Moderate	Moderate	Low
high	Moderate	Low	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Moderate	Low	Low	Absent
moderate	High	Moderate	Moderate	Low
high	Moderate	Low	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Low	Low	Absent
moderate	High	Moderate	Moderate	Low
high	Moderate	Low	Low	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Absent	Absent	Absent
high	Low	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Glossiphonia complanata (glsicomp,44)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Low
high	Low	Low	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Moderate	Low	Low
moderate	High	High	Moderate	Moderate
high	Moderate	Moderate	Low	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Low	Low	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Haliplus flavicollis (haliflav,47)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	High	High	Moderate	Low
moderate	Moderate	Moderate	Low	Absent
high	Low	Low	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Low	Low	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Haementaria costata (hamecost,50)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Absent	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Absent	Absent	Absent
moderate	Moderate	Low	Absent	Absent
high	Low	Absent	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Moderate	Low	Absent	Absent
moderate	High	Moderate	Low	Absent
high	Moderate	Low	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Absent	Absent	Absent
high	Low	Absent	Absent	Absent

Haliplus fluviatilis (halifluv,48)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Low	Low
moderate	Low	Low	Moderate	Moderate
high	Absent	Absent	Low	Low
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Low	Low
moderate	Low	Low	Moderate	Moderate
high	Absent	Absent	Low	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Moderate	Moderate
moderate	Moderate	Moderate	High	High
high	Low	Low	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Low
high	Absent	Absent	Low	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Halesus radiatus (haledira,15)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Low	Absent
moderate	Moderate	Moderate	Moderate	Low
high	High	High	High	Moderate
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Absent
high	High	High	High	Moderate
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Moderate	Moderate	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Low	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Helophorus aquaticus/grandis (heruaqgr,53)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
		oligosaprobic / oligotrophic / oligoionic			
stream velocity	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Moderate	Moderate	Low	Absent
	moderate	Low	Low	Absent	Absent
	high	Absent	Absent	Absent	Absent
		mesosaprobic / α-mesotrophic / mesoionic			
	low	High	High	Moderate	Low
	moderate	Moderate	Moderate	Low	Absent
	high	Low	Low	Absent	Absent
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Moderate	Moderate	Low	Absent
	moderate	Low	Low	Absent	Absent
	high	Low	Low	Absent	Absent
		polysaprobic / hypertrophic / polyionic			
	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Haliplus lineatocollis (halilito,49)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
		oligosaprobic / oligotrophic / oligoionic			
stream velocity	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		mesosaprobic / α-mesotrophic / mesoionic			
	low	High	High	Moderate	Moderate
	moderate	Moderate	Moderate	Low	Low
	high	Low	Low	Absent	Absent
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
		polysaprobic / hypertrophic / polyionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Helophorus brevipalpis (herubrev,54)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
		oligosaprobic / oligotrophic / oligoionic			
stream velocity	low	Absent	Absent	Absent	Absent
	moderate	Low	Low	Absent	Absent
	high	Absent	Absent	Absent	Absent
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Low	Low	Absent	Absent
	moderate	Moderate	Moderate	Low	Absent
	high	Low	Low	Absent	Absent
		mesosaprobic / α-mesotrophic / mesoionic			
	low	Moderate	Moderate	Low	Absent
	moderate	High	High	Moderate	Low
	high	Moderate	Moderate	Low	Absent
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Low	Low	Absent	Absent
	moderate	Moderate	Moderate	Low	Absent
	high	Moderate	Moderate	Low	Absent
		polysaprobic / hypertrophic / polyionic			
	low	Absent	Absent	Absent	Absent
	moderate	Low	Low	Absent	Absent
	high	Low	Low	Absent	Absent

Helobdella stagnalis (hebdstag,51)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
		oligosaprobic / oligotrophic / oligoionic			
stream velocity	low	Absent	Absent	Absent	Absent
	moderate	Absent	Low	Low	Low
	high	Absent	Absent	Absent	Absent
		β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
	low	Absent	Low	Low	Low
	moderate	Low	Moderate	Moderate	Moderate
	high	Absent	Low	Low	Low
		mesosaprobic / α-mesotrophic / mesoionic			
	low	Low	Moderate	Moderate	Moderate
	moderate	Moderate	High	High	High
	high	Low	Moderate	Moderate	Moderate
		α-mesosaprobic / eutrophic / α-mesoionic			
	low	Absent	Low	Low	Low
	moderate	Low	Moderate	Moderate	Moderate
	high	Low	Moderate	Moderate	Moderate
		polysaprobic / hypertrophic / polyionic			
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Low	Low	Low
	high	Absent	Low	Low	Low

Hydroporus palustris (hypopalu,55)	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic			
low	High	High	Moderate	Low
moderate	Moderate	Moderate	Low	Absent
high	Low	Low	Absent	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Moderate	Moderate	Low	Absent
moderate	Low	Low	Absent	Absent
high	Low	Low	Absent	Absent
	mesosaprobic / α-mesotrophic / mesoionic			
low	Low	Low	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Hemiclepsis marginata (heclmarg,52)	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic			
low	Low	Low	Low	Low
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Moderate	Moderate	Moderate	Moderate
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
	mesosaprobic / α-mesotrophic / mesoionic			
low	High	High	High	High
moderate	Moderate	Moderate	Moderate	Moderate
high	Low	Low	Low	Low
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Low	Low	Low	Low
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Hydropsyche angustipennis (hypsangu,56)	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Low
high	Low	Low	Moderate	Moderate
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Low
high	Low	Low	Moderate	Moderate
	mesosaprobic / α-mesotrophic / mesoionic			
low	Absent	Absent	Low	Low
moderate	Low	Low	Moderate	Moderate
high	Moderate	Moderate	High	High
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Low
high	Moderate	Moderate	High	High
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Moderate	Moderate

Hydroporus nigrita (hyponigr,16)	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic			
low	High	Moderate	Low	Absent
moderate	Moderate	Low	Absent	Absent
high	Low	Absent	Absent	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Moderate	Low	Absent	Absent
moderate	Low	Absent	Absent	Absent
high	Low	Absent	Absent	Absent
	mesosaprobic / α-mesotrophic / mesoionic			
low	Low	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Ilybius fenestratus (ilybfene,58)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Low	Low
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	High	High	High	High
moderate	Moderate	Moderate	Moderate	Moderate
high	Low	Low	Low	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	Low	Low	Low	Low
high	Low	Low	Low	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Low	Low	Low	Low
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Hydropsyche pellucidula (hypspell,17)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Low	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Low	Absent
moderate	Absent	Low	Low	Absent
high	Low	Moderate	Moderate	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Low	Moderate	Low
moderate	Low	Moderate	Moderate	Low
high	Moderate	High	High	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Moderate	Moderate	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Low	Low	Absent

Ilybius fuliginosus (ilybfuli,59)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	Moderate	Moderate	Moderate	Moderate
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Low
moderate	Moderate	Moderate	Moderate	Moderate
high	Moderate	Moderate	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Low	Low	Low	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Hygrotus inaequalis (hytuinae,57)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	High	High	Moderate	Moderate
moderate	Moderate	Moderate	Low	Low
high	Low	Low	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Limnephilus fusciformis (lilufusc,20)	stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic					
	low	Absent	Low	Low	Low
	moderate	Absent	Low	Low	Low
	high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic					
	low	Low	Moderate	Moderate	Moderate
	moderate	Low	Moderate	Moderate	Moderate
	high	Absent	Low	Low	Low
mesosaprobic / α-mesotrophic / mesoionic					
	low	Moderate	High	High	High
	moderate	Moderate	High	High	High
	high	Low	Moderate	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic					
	low	Absent	Low	Low	Low
	moderate	Absent	Low	Low	Low
	high	Absent	Low	Low	Low
polysaprobic / hypertrophic / polyionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Ironoquia dubia (irondubi,18)	stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic					
	low	Low	Low	Absent	Absent
	moderate	Moderate	Moderate	Low	Absent
	high	Low	Low	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic					
	low	Moderate	Moderate	Low	Absent
	moderate	High	High	Moderate	Low
	high	Moderate	Moderate	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic					
	low	Absent	Absent	Absent	Absent
	moderate	Low	Low	Absent	Absent
	high	Low	Low	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Limnephilus lunatus (lilulun a,21)	stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Low	Absent
	high	Absent	Absent	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic					
	low	Absent	Absent	Low	Absent
	moderate	Absent	Low	Moderate	Low
	high	Absent	Absent	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic					
	low	Absent	Low	Moderate	Low
	moderate	Low	Moderate	High	Moderate
	high	Absent	Low	Moderate	Low
α-mesosaprobic / eutrophic / α-mesoionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Low	Absent
	high	Absent	Absent	Low	Absent
polysaprobic / hypertrophic / polyionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Limnephilus extricates (liluextr,19)	stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Low	Low	Low
	high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic					
	low	Absent	Low	Low	Low
	moderate	Low	Moderate	Moderate	Moderate
	high	Absent	Low	Low	Low
mesosaprobic / α-mesotrophic / mesoionic					
	low	Low	Moderate	Moderate	Moderate
	moderate	Moderate	High	High	High
	high	Low	Moderate	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Low	Low	Low
	high	Absent	Low	Low	Low
polysaprobic / hypertrophic / polyionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Nebrioporus depressus (ponedepr,28)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
	oligosaprobic / oligotrophic / oligoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Low	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Absent	Absent	Low	Absent
moderate	Absent	Absent	Low	Absent
high	Absent	Low	Moderate	Low
	mesosaprobic / α-mesotrophic / mesoionic			
low	Absent	Low	Moderate	Low
moderate	Absent	Low	Moderate	Low
high	Low	Moderate	High	Moderate
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Low	Moderate	Low
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Low	Absent

Limnephilus rhombicus (lilurhom,60)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
	oligosaprobic / oligotrophic / oligoionic			
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Absent	Absent	Absent	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Moderate	Moderate	Low	Low
moderate	High	High	Moderate	Moderate
high	Moderate	Moderate	Low	Low
	mesosaprobic / α-mesotrophic / mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Low	Low	Absent	Absent
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Notidobia ciliaris (nodocili,22)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
	oligosaprobic / oligotrophic / oligoionic			
low	Moderate	Moderate	Low	Absent
moderate	Moderate	Moderate	Low	Absent
high	High	High	Moderate	Low
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Moderate	Low	Absent
	mesosaprobic / α-mesotrophic / mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Absent	Absent
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Lype reducta (lyperedu,61)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
	oligosaprobic / oligotrophic / oligoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Absent	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Absent
high	High	High	Moderate	Low
	mesosaprobic / α-mesotrophic / mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Moderate	Moderate	Absent	Absent
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Absent	Absent
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Orectochillus villosus (`orecvill,24`) stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	High	High	High	High
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	High	High	High	High
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Low
moderate	Moderate	Moderate	Moderate	Moderate
high	High	High	High	High
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Moderate	Moderate	Moderate	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Low	Low

Notonecta glauca (`notoglau,62`) stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Low	Low
moderate	Low	Low	Moderate	Moderate
high	Absent	Absent	Low	Low
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Moderate	Moderate
moderate	Moderate	Moderate	High	High
high	Low	Low	Moderate	Moderate
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Moderate	Moderate
moderate	Moderate	Moderate	High	High
high	Low	Low	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Low
high	Absent	Absent	Low	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Parametriocnemus stylatus (`paocstyl,83`) stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Moderate	Low	Absent	Absent
moderate	Moderate	Low	Absent	Absent
high	High	Moderate	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Moderate	Low	Absent	Absent
moderate	Moderate	Low	Absent	Absent
high	High	Moderate	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Low	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Odontomesa fulva (`odmefulv,23`) stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Absent
moderate	Moderate	Moderate	Moderate	Low
high	High	High	High	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Moderate	Moderate	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Low	Absent

Piscicola geometra (piscgeom,64)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic				
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Low
	high	Absent	Absent	Absent	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
	low	Absent	Absent	Absent	Low
	moderate	Absent	Low	Low	Moderate
	high	Absent	Absent	Absent	Low
	mesosaprobic / α-mesotrophic / mesoionic				
	low	Absent	Low	Low	Moderate
	moderate	Low	Moderate	Moderate	High
	high	Absent	Low	Low	Moderate
	α-mesosaprobic / eutrophic / α-mesoionic				
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Low
	high	Absent	Absent	Absent	Low
	polysaprobic / hypertrophic / polyionic				
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Physa acuta (physacut,63)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic				
	low	Moderate	Moderate	High	High
	moderate	Low	Low	Moderate	Moderate
	high	Absent	Absent	Low	Low
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
	low	Moderate	Moderate	High	High
	moderate	Low	Low	Moderate	Moderate
	high	Absent	Absent	Low	Low
	mesosaprobic / α-mesotrophic / mesoionic				
	low	Moderate	Moderate	High	High
	moderate	Low	Low	Moderate	Moderate
	high	Absent	Absent	Low	Low
	α-mesosaprobic / eutrophic / α-mesoionic				
	low	Low	Low	Moderate	Moderate
	moderate	Absent	Absent	Low	Low
	high	Absent	Absent	Low	Low
	polysaprobic / hypertrophic / polyionic				
	low	Absent	Absent	Low	Low
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Planorbis carinatus (plbicari,65)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic				
	low	Moderate	Moderate	High	High
	moderate	Low	Low	Moderate	Moderate
	high	Absent	Absent	Low	Low
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
	low	Absent	Absent	Low	Low
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
	mesosaprobic / α-mesotrophic / mesoionic				
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
	α-mesosaprobic / eutrophic / α-mesoionic				
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic				
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Physa fontinalis (physfont,25)		stream width			
		spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic				
	low	Low	Moderate	Low	Absent
	moderate	Absent	Low	Absent	Absent
	high	Absent	Low	Absent	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
	low	Low	Moderate	Low	Absent
	moderate	Absent	Low	Absent	Absent
	high	Absent	Low	Absent	Absent
	mesosaprobic / α-mesotrophic / mesoionic				
	low	Moderate	High	Moderate	Low
	moderate	Low	Moderate	Low	Absent
	high	Absent	Low	Absent	Absent
	α-mesosaprobic / eutrophic / α-mesoionic				
	low	Absent	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic				
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Plectrocnemia conspersa (pltrcons,27)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Low	Moderate	Low
moderate	Absent	Low	Moderate	Low
high	Low	Moderate	High	Moderate
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Low	Moderate	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Low	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Planorbis planorbis (plbiplan,66)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	High	High	High	High
moderate	Moderate	Moderate	Moderate	Moderate
high	Low	Low	Low	Low
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	High	High	High	High
moderate	Moderate	Moderate	Moderate	Moderate
high	Low	Low	Low	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	Low	Low	Low	Low
high	Low	Low	Low	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Low	Low	Low	Low
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Plectrocnemia geniculata (pltrgeni,67)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Absent	Absent	Absent
moderate	Moderate	Low	Absent	Absent
high	High	Moderate	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Moderate	Low	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Platambus maculatus (pltamacu,26)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	Moderate	Moderate	Moderate	Moderate
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	Moderate	Moderate	Moderate	Moderate
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Low
moderate	Moderate	Moderate	Moderate	Moderate
high	Moderate	Moderate	Moderate	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Low	Low	Low	Low
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Prodiamesa olivacea (prodoliv,85)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Absent
high	Moderate	Moderate	Moderate	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Absent
moderate	Moderate	Moderate	Moderate	Low
high	High	High	High	Moderate
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Absent
high	High	High	High	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Moderate	Moderate	Low

Polypedilum laetum agg. (popelaea,82)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Moderate	Low	Absent	Absent
moderate	Moderate	Low	Absent	Absent
high	High	Moderate	Low	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Moderate	Low	Absent	Absent
moderate	Moderate	Low	Absent	Absent
high	High	Moderate	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Low	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Radix peregra (radipere,69)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Low	Low	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Low	Absent
moderate	Low	Moderate	Moderate	Low
high	Low	Low	Low	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Moderate	Moderate	Low
moderate	Moderate	High	High	Moderate
high	Moderate	Moderate	Moderate	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Low	Low	Absent
high	Low	Low	Low	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Proasellus meridianus (proameri,68)	stream width			
stream velocity	spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Low
moderate	Moderate	Moderate	Moderate	Moderate
high	Low	Low	Low	Low
α-mesosaprobic / eutrophic / α-mesoionic				
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	Moderate	Moderate	Moderate	Moderate
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Low	Low	Low	Low

Sericostoma personatum (setopers,30) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Moderate	Moderate	Moderate	Low
moderate	Moderate	Moderate	Moderate	Low
high	High	High	Moderate	Low
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Low	Absent
moderate	Low	Low	Low	Absent
high	High	High	Moderate	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Moderate	Low	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Rhantus suturalis (rhansura,86) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic				
low	Moderate	Moderate	Low	Absent
moderate	Low	Low	Absent	Absent
high	Absent	Absent	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	High	High	Moderate	Low
moderate	Moderate	Moderate	Low	Absent
high	Low	Low	Absent	Absent
polysaprobic / hypertrophic / polyionic				
low	Moderate	Moderate	Low	Absent
moderate	Low	Low	Absent	Absent
high	Low	Low	Absent	Absent

Sialis fuliginosa (sialfuli,70) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Absent
high	High	High	Moderate	Low
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Low	Low	Absent	Absent
moderate	Moderate	Moderate	Low	Absent
high	High	High	Moderate	Low
mesosaprobic / α-mesotrophic / mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	High	High	Moderate	Absent
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Moderate	Low	Absent
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Absent	Absent

Rheocricotopus (gr) fuscipes (rhcrgfus,29) — stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Low	Low	Low	Low
β,α-oligosaprobic / β-mesotrophic / β-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Moderate	Moderate	Moderate	Moderate
mesosaprobic / α-mesotrophic / mesoionic				
low	Low	Low	Low	Low
moderate	Moderate	Moderate	Moderate	Moderate
high	High	High	High	High
α-mesosaprobic / eutrophic / α-mesoionic				
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	High	High	High	High
polysaprobic / hypertrophic / polyionic				
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Moderate	Moderate	Moderate	Moderate

Sigara lateralis (`sigalate,73`)	stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic					
	low	High	High	Moderate	Low
	moderate	Moderate	Moderate	Low	Absent
	high	Low	Low	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic					
	low	High	High	Moderate	Low
	moderate	Moderate	Moderate	Low	Absent
	high	Low	Low	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic					
	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Sialis lutaria (`sialluta,71`)	stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic					
	low	Moderate	Moderate	Moderate	Moderate
	moderate	Low	Low	Low	Low
	high	Absent	Absent	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic					
	low	High	High	High	High
	moderate	Moderate	Moderate	Moderate	Moderate
	high	Low	Low	Low	Low
mesosaprobic / α-mesotrophic / mesoionic					
	low	High	High	High	High
	moderate	Moderate	Moderate	Moderate	Moderate
	high	Low	Low	Low	Low
α-mesosaprobic / eutrophic / α-mesoionic					
	low	Moderate	Moderate	Moderate	Moderate
	moderate	Low	Low	Low	Low
	high	Low	Low	Low	Low
polysaprobic / hypertrophic / polyionic					
	low	Low	Low	Low	Low
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Sigara semistriata (`sigasemi,74`)	stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic					
	low	High	High	Moderate	Low
	moderate	Moderate	Moderate	Low	Absent
	high	Low	Low	Absent	Absent
β,α-oligosaprobic / β-mesotrophic / β-mesoionic					
	low	High	High	Moderate	Low
	moderate	Moderate	Moderate	Low	Absent
	high	Low	Low	Absent	Absent
mesosaprobic / α-mesotrophic / mesoionic					
	low	Low	Low	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Sigara falleni (`sigafall,72`)	stream velocity	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
oligosaprobic / oligotrophic / oligoionic					
	low	Moderate	Moderate	High	High
	moderate	Low	Low	Moderate	Moderate
	high	Absent	Absent	Low	Low
β,α-oligosaprobic / β-mesotrophic / β-mesoionic					
	low	Moderate	Moderate	High	High
	moderate	Low	Low	Moderate	Moderate
	high	Absent	Absent	Low	Low
mesosaprobic / α-mesotrophic / mesoionic					
	low	Absent	Absent	Low	Low
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
α-mesosaprobic / eutrophic / α-mesoionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent
polysaprobic / hypertrophic / polyionic					
	low	Absent	Absent	Absent	Absent
	moderate	Absent	Absent	Absent	Absent
	high	Absent	Absent	Absent	Absent

Valvata piscinalis (valvpisc,77)	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic			
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
	mesosaprobic / α-mesotrophic / mesoionic			
low	High	High	High	High
moderate	Moderate	Moderate	Moderate	Moderate
high	Low	Low	Low	Low
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Sigara striata (sigastri,75)	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic			
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	Moderate	Moderate	Moderate	Moderate
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Moderate	Moderate	Moderate	Moderate
moderate	High	High	High	High
high	Moderate	Moderate	Moderate	Moderate
	mesosaprobic / α-mesotrophic / mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Low	Low
high	Low	Low	Low	Low
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Velia caprai (velicapr,78)	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic			
low	Moderate	Moderate	Low	Absent
moderate	High	High	Moderate	Low
high	Moderate	Moderate	Low	Absent
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Moderate	Moderate	Low	Absent
moderate	High	High	Moderate	Low
high	Moderate	Moderate	Low	Absent
	mesosaprobic / α-mesotrophic / mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Low	Low	Absent	Absent
high	Low	Low	Absent	Absent
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

Stagnicola palustris (stagpalu,76)	stream width: spring / small stream	upper course stream	middle course stream	lower course stream / small river
stream velocity	oligosaprobic / oligotrophic / oligoionic			
low	Absent	Low	Moderate	Moderate
moderate	Low	Moderate	High	High
high	Absent	Low	Moderate	Moderate
	β,α-oligosaprobic / β-mesotrophic / β-mesoionic			
low	Absent	Low	Moderate	Moderate
moderate	Low	Moderate	High	High
high	Absent	Low	Moderate	Moderate
	mesosaprobic / α-mesotrophic / mesoionic			
low	Absent	Absent	Low	Low
moderate	Absent	Low	Moderate	Moderate
high	Absent	Low	Moderate	Moderate
	α-mesosaprobic / eutrophic / α-mesoionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Low	Low
high	Absent	Absent	Low	Low
	polysaprobic / hypertrophic / polyionic			
low	Absent	Absent	Absent	Absent
moderate	Absent	Absent	Absent	Absent
high	Absent	Absent	Absent	Absent

APPENDIX C

Appendix to Section 8.4.3

The terms $C_1, C_2, C_3, C_4, C_5, C_6, C_7$ in Eq. (8.97) are functions of $\gamma_1 \in [0.5, 1]$, $l_{i\text{-}1}, l_i, l_{i+1}, l_{i+2} \in \mathbb{R}_0^+$ and $k_i, k_{i+1}, k_{i+2} \in \mathbb{R}^+$.

$$
\begin{aligned}
C_1 = {} & -(3\gamma_1-1)(1-\gamma_1)(2\gamma_1-1)^2 l_i(l_{i\text{-}1}^2 + 3l_{i\text{-}1}k_i + l_{i+2}^2 + 3l_{i+2}k_{i+1} + 3l_{i+2}k_{i+2} \\
& + 3k_i^2 + 6k_{i+1}k_{i+2} + 3k_{i+2}^2) + (1-\gamma_1)(1+\gamma_1)(2\gamma_1-1)^4 l_{i+1}(l_{i\text{-}1}^2 + 3l_{i\text{-}1}l_i \\
& + 3l_{i\text{-}1}k_i + 3l_{i\text{-}1}k_{i+1} + 6l_ik_i + l_{i+2}^2 + 3l_{i+2}k_{i+2} + 3k_i^2 + 6k_ik_{i+1} + 3k_{i+2}^2) \\
& + (2\gamma_1-1)^4 k_{i+1}(2l_{i\text{-}1}^2 + 6l_{i\text{-}1}l_i + 6l_{i\text{-}1}k_i + 3l_{i\text{-}1}k_{i+1} + 12l_ik_i + 6l_{i+1}l_{i+2} \\
& + 12l_{i+1}k_{i+2} + 2l_{i+2}^2 + 3l_{i+2}k_{i+1} + 6l_{i+2}k_{i+2} + 6k_i^2 + 6k_ik_{i+1} + 6k_{i+1}k_{i+2} \\
& + 6k_{i+2}^2) - 2(1-\gamma_1)(2\gamma_1-1)(7\gamma_1^2 - 5\gamma_1 + 1)l_{i\text{-}1}l_i^2 + (2\gamma_1+1)(\gamma_1-1)^2 \\
& (2\gamma_1-1)^4(l_{i\text{-}1} + 2k_i)l_{i+1}^2 - (1-\gamma_1)\gamma_1(15\gamma_1^2 - 15\gamma_1 + 4)l_i^3 + \gamma_1(1-\gamma_1) \\
& (2\gamma_1-1)(18\gamma_1^3 - 17\gamma_1^2 - \gamma_1 + 4)l_i^2 l_{i+1} + (4\gamma_1-1)(2\gamma_1-1)(\gamma_1-1)^2 l_i^2 \\
& (l_{i+2} + 2k_{i+2}) - 4(1-\gamma_1)(2\gamma_1-1)(7\gamma_1^2 - 5\gamma_1 + 1)l_i^2 k_i + 4\gamma_1(2\gamma_1-1) \\
& (7\gamma_1^2 - 9\gamma_1 + 3)l_i^2 k_{i+1} - \gamma_1(1-\gamma_1)(12\gamma_1^3 - 27\gamma_1^2 + 23\gamma_1 - 4)(2\gamma_1-1)^2 l_i l_{i+1}^2 \\
& - 3(3\gamma_1-1)(1-\gamma_1)(2\gamma_1-1)^2 l_i l_{i+1}(l_{i+2} + 2k_{i+2}) - 18\gamma_1^2(2\gamma_1-1)^2 \\
& (\gamma_1-1)^2 l_i l_{i+1} k_{i+1} + 3\gamma_1(3\gamma_1-2)(2\gamma_1-1)^2 l_i k_{i+1}^2 + \gamma_1(1-\gamma_1) \\
& (\gamma_1^2 - \gamma_1 + 4)(2\gamma_1-1)^4 l_{i+1}^3 + 2(1-\gamma_1)(\gamma_1^2 + \gamma_1 + 1)(2\gamma_1-1)^4 l_{i+1}^2 \\
& (l_{i+2} + 2k_{i+2}) + 4\gamma_1(\gamma_1^2 - 3\gamma_1 + 3)(2\gamma_1-1)^4 l_{i+1}^2 k_{i+1} - 3\gamma_1(\gamma_1-2) \\
& (2\gamma_1-1)^4 l_{i+1} k_{i+1}^2 \, ,
\end{aligned}
\tag{C.1}
$$

$$
\begin{aligned}
C_2 = {}& (1-\gamma_1)(2\gamma_1-1)(16\gamma_1^2-9\gamma_1+1)(l_{i\text{-}1}^2+3l_{i\text{-}1}k_i+3k_i^2)l_i-3(6\gamma_1-1) \\
& (2\gamma_1-1)^3(2l_{i\text{-}1}l_ik_{i+1}+2l_{i\text{-}1}k_ik_{i+1}+l_{i\text{-}1}k_{i+1}^2+4l_ik_ik_{i+1}+2k_i^2k_{i+1}+2k_ik_{i+1}^2) \\
& -(1-\gamma_1)(6\gamma_1-1)(\gamma_1+1)(2\gamma_1-1)^3l_{i+1}(l_{i\text{-}1}^2+3l_{i\text{-}1}l_i+3l_{i\text{-}1}k_i+3l_{i\text{-}1}k_{i+1} \\
& +6l_ik_i+3k_i^2+6k_ik_{i+1})-2(6\gamma_1-1)(2\gamma_1-1)^3l_{i\text{-}1}^2k_{i+1}-2(\gamma_1-1) \\
& (3\gamma_1-1)(13\gamma_1^2-8\gamma_1+1)(l_{i\text{-}1}+2k_i)l_i^2-(2\gamma_1+1)(6\gamma_1-1)(\gamma_1-1)^2 \\
& (2\gamma_1-1)^3(l_{i\text{-}1}+2k_i)l_{i+1}^2+(1-\gamma_1)(45\gamma_1^2-37\gamma_1+8)\gamma_1l_i^3-4\gamma_1(1-\gamma_1) \\
& (25\gamma_1^4-21\gamma_1^3-9\gamma_1^2+13\gamma_1-3)l_i^2l_{i+1}-2\gamma_1(5\gamma_1-2)(\gamma_1-1)^2(l_{i+2}+2k_{i+2}) \\
& l_i^2-4(45\gamma_1^3-70\gamma_1^2+36\gamma_1-6)l_i^2k_{i+1}\gamma_1+\gamma_1(1-\gamma_1)(2\gamma_1-1)(60\gamma_1^4 \\
& -126\gamma_1^3+91\gamma_1^2-9\gamma_1-4)l_il_{i+1}^2+\gamma_1(1-\gamma_1)(2\gamma_1-1)(5\gamma_1-2)2l_i \\
& (3l_{i+1}l_{i+2}+6l_{i+1}k_{i+2}+l_{i+2}^2+3l_{i+2}k_{i+1}+3l_{i+2}k_{i+2}+6k_{i+1}k_{i+2}+3k_{i+2}^2) \\
& -3\gamma_1(1-\gamma_1)(2\gamma_1-1)(30\gamma_1^3-23\gamma_1^2-7\gamma_1+4)l_il_{i+1}k_{i+1}-3\gamma_1(2\gamma_1-1) \\
& (20\gamma_1^2-19\gamma_1+4)l_ik_{i+1}^2-4(1-\gamma_1)(\gamma_1^2-\gamma_1+4)(2\gamma_1-1)^3\gamma_1^2l_{i+1}^3-8\gamma_1 \\
& (1-\gamma_1)(\gamma_1^2+\gamma_1+1)(2\gamma_1-1)^3l_{i+1}^2(l_{i+2}+2k_{i+2})-16(\gamma_1^2-3\gamma_1+3) \\
& (2\gamma_1-1)^3\gamma_1^2l_{i+1}^2k_{i+1}-4\gamma_1(1-\gamma_1)(1+\gamma_1)(2\gamma_1-1)^3l_{i+1}(l_{i+2}^2+3l_{i+2}k_{i+2} \\
& +3k_{i+2}^2)-(2\gamma_1-1)^3\gamma_1(24l_{i+1}l_{i+2}k_{i+1}+48l_{i+1}k_{i+1}k_{i+2}+8l_{i+2}^2k_{i+1} \\
& +12l_{i+2}k_{i+1}^2+24l_{i+2}k_{i+1}k_{i+2}+24k_{i+1}^2k_{i+2}+24k_{i+1}k_{i+2}^2)-12\gamma_1^2(2-\gamma_1) \\
& (2\gamma_1-1)^3l_{i+1}k_{i+1}^2\,,
\end{aligned}
\tag{C.2}
$$

$$
\begin{aligned}
C_3 = {}& -\gamma_1(1-\gamma_1)(31\gamma_1^2-23\gamma_1+4)(l_{i\text{-}1}^2+3l_{i\text{-}1}k_i+3k_i^2)l_i+\gamma_1(1-\gamma_1) \\
& (7\gamma_1-2)(1+\gamma_1)(2\gamma_1-1)^2(2l_{i\text{-}1}^2l_{i+1}+6l_{i\text{-}1}l_il_{i+1}+6l_{i\text{-}1}l_{i+1}k_i+6l_{i\text{-}1}l_{i+1}k_{i+1} \\
& +12l_il_{i+1}k_i+6l_{i+1}k_i^2+12l_{i+1}k_ik_{i+1})+\gamma_1(7\gamma_1-2)(2\gamma_1-1)^2(4l_{i\text{-}1}^2k_{i+1} \\
& +12l_{i\text{-}1}l_ik_{i+1}+12l_{i\text{-}1}k_ik_{i+1}+6l_{i\text{-}1}k_{i+1}^2+24l_ik_ik_{i+1}+12k_i^2k_{i+1}+12k_ik_{i+1}^2) \\
& -4\gamma_1(1-\gamma_1)(4\gamma_1-1)(5\gamma_1-2)(l_{i\text{-}1}+2k_i)l_i^2+2\gamma_1(2\gamma_1+1)(7\gamma_1-2) \\
& (2\gamma_1-1)^2(\gamma_1-1)^2(l_{i\text{-}1}+2k_i)l_{i+1}^2-(1-\gamma_1)\gamma_1(49\gamma_1^2-29\gamma_1+4)l_i^3+\gamma_1 \\
& (1-\gamma_1)(113\gamma_1^4-37\gamma_1^3-78\gamma_1^2+48\gamma_1-6)l_i^2l_{i+1}+\gamma_1^2(\gamma_1-1)^23l_i^2 \\
& (l_{i+2}+2k_{i+2})+4\gamma_1(55\gamma_1^3-69\gamma_1^2+27\gamma_1-3)l_i^2k_{i+1}-\gamma_1(1-\gamma_1)(124\gamma_1^5 \\
& -247\gamma_1^4+147\gamma_1^3+5\gamma_1^2-19\gamma_1+2)l_il_{i+1}^2-\gamma_1^2(11\gamma_1-5)(1-\gamma_1)l_i \\
& (3l_{i+1}l_{i+2}+6l_{i+1}k_{i+2}+l_{i+2}^2+3l_{i+2}k_{i+1}+3l_{i+2}k_{i+2}+6k_{i+1}k_{i+2}+3k_{i+2}^2) \\
& +3\gamma_1(1-\gamma_1)(62\gamma_1^4-36\gamma_1^3-29\gamma_1^2+19\gamma_1-2)l_il_{i+1}k_{i+1}+3\gamma_1(51\gamma_1^3 \\
& -60\gamma_1^2+21\gamma_1-2)l_ik_{i+1}^2+6(1-\gamma_1)(\gamma_1^2-\gamma_1+4)(2\gamma_1-1)^2l_{i+1}^3\gamma_1^3+12\gamma_1^2 \\
& (1-\gamma_1)(\gamma_1^2+\gamma_1+1)(2\gamma_1-1)^2l_{i+1}^2(l_{i+2}+2k_{i+2})+24\gamma_1^3(\gamma_1^2-3\gamma_1+3)
\end{aligned}
$$

$$(2\gamma_1-1)^2 l_{i+1}^2 k_{i+1} + 6\gamma_1^2(1-\gamma_1)(\gamma_1+1)(2\gamma_1-1)^2 l_{i+1}(l_{i+2}^2 + 3l_{i+2}k_{i+2} + 3k_{i+2}^2) + 6\gamma_1^2(2\gamma_1-1)^2 k_{i+1}(6l_{i+1}l_{i+2} + 12l_{i+1}k_{i+2} + 2l_{i+2}^2 + 3l_{i+2}k_{i+1} + 6l_{i+2}k_{i+2} + 6k_{i+1}k_{i+2} + 6k_{i+2}^2) + 18(2-\gamma_1)(2\gamma_1-1)^2 l_{i+1}k_{i+1}^2\gamma_1^3\,, \tag{C.3}$$

$$C_4 = \gamma_1^2(13\gamma_1-5)(1-\gamma_1)(l_{i\text{-}1}^2 + 3l_{i\text{-}1}k_i + 3k_i^2)l_i - \gamma_1^2(1-\gamma_1)(2\gamma_1-1)(8\gamma_1-3)(1+\gamma_1)2l_{i+1}(l_{i\text{-}1}^2 + 6l_{i\text{-}1}k_{i+1} + 3l_{i\text{-}1}k_i + 2l_i^2 + 6l_ik_i + 3k_i^2 + 6k_ik_{i+1}) - 2\gamma_1^2(8\gamma_1-3)(2\gamma_1-1)k_{i+1}(2l_{i\text{-}1}^2 + 6l_{i\text{-}1}l_i + 6l_{i\text{-}1}k_i + 3l_{i\text{-}1}k_{i+1} + 4l_i^2 + 12l_ik_i + 6k_i^2 + 6k_ik_{i+1}) - 12\gamma_1^2(3\gamma_1-1)(\gamma_1-1)(l_{i\text{-}1} + 2k_i)l_i^2 - 6\gamma_1^2(1-\gamma_1)(2\gamma_1-1)(8\gamma_1-3)(1+\gamma_1)l_{i\text{-}1}l_{i+1}(l_i - k_{i+1}) - 2\gamma_1^2(8\gamma_1-3)(2\gamma_1+1)(2\gamma_1-1)(\gamma_1-1)^2 l_{i\text{-}1}l_{i+1}^2 + (23\gamma_1-7)(1-\gamma_1)\gamma_1^2 l_i^3 + \gamma_1^2(1-\gamma_1)(66\gamma_1^4 - 95\gamma_1^3 + 17\gamma_1^2 + 21\gamma_1 - 5)l_il_{i+1}^2 + 2\gamma_1^3(1-\gamma_1)l_i(3l_{i+1}l_{i+2} + 6l_{i+1}k_{i+2} + l_{i+2}^2 + 3l_{i+2}k_{i+1} + 3l_{i+2}k_{i+2} + 6k_{i+1}k_{i+2} + 3k_{i+2}^2) - 3\gamma_1^2(1-\gamma_1)(33\gamma_1^3 + \gamma_1^2 - 21\gamma_1 + 5)l_il_{i+1}k_{i+1} - 3\gamma_1^2(31\gamma_1^2 - 26\gamma_1 + 5)l_ik_{i+1}^2 - 4\gamma_1^4(1-\gamma_1)(2\gamma_1-1)(\gamma_1^2 - \gamma_1 + 4)l_{i+1}^3 - 8\gamma_1^3(1-\gamma_1)(2\gamma_1-1)(\gamma_1^2 + \gamma_1 + 1)l_{i+1}^2(l_{i+2} + 2k_{i+2}) - 4\gamma_1^2(8\gamma_1-3)(2\gamma_1+1)(2\gamma_1-1)(\gamma_1-1)^2 l_{i+1}^2k_i - 16\gamma_1^4(2\gamma_1-1)(\gamma_1^2 - 3\gamma_1 + 3)l_{i+1}^2k_{i+1} - 4\gamma_1^3(1-\gamma_1)(2\gamma_1-1)(\gamma_1+1)l_{i+1}(l_{i+2}^2 + 3l_{i+2}k_{i+2} + 3k_{i+2}^2) - 4\gamma_1^3(2\gamma_1-1)k_{i+1}(6l_{i+1}l_{i+2} + 12l_{i+1}k_{i+2} + 2l_{i+2}^2 + 3l_{i+2}k_{i+1} + 6l_{i+2}k_{i+2} + 6k_{i+1}k_{i+2} + 6k_{i+2}^2) + 12\gamma_1^4(2\gamma_1-1)(\gamma_1-2)l_{i+1}k_{i+1}^2\,, \tag{C.4}$$

$$C_5 = -2\gamma_1^3(1-\gamma_1)l_i(l_{i\text{-}1}^2 + 3l_{i\text{-}1}l_i + 3l_{i\text{-}1}k_i + 2l_i^2 + 6l_ik_i + 3k_i^2) + \gamma_1^3(1-\gamma_1)(9\gamma_1-4)(\gamma_1+1)l_{i+1}(l_{i\text{-}1}^2 + 3l_{i\text{-}1}l_i + 3l_{i\text{-}1}k_i + 3l_{i\text{-}1}k_{i+1} + 2l_i^2 + 6l_ik_i + 3k_{i+1}l_i + 3k_i^2 + 6k_ik_{i+1}) + \gamma_1^3(9\gamma_1-4)k_{i+1}(3l_{i\text{-}1}k_{i+1} + 3k_{i+1}l_i + 2l_{i\text{-}1}^2 + 6l_{i\text{-}1}l_i + 6l_{i\text{-}1}k_i + 4l_i^2 + 12l_ik_i) + \gamma_1^3(2\gamma_1+1)(9\gamma_1-4)(\gamma_1-1)^2 l_{i+1}^2(2k_i + l_{i\text{-}1} + l_i) + \gamma_1^5(1-\gamma_1)(\gamma_1^2 - \gamma_1 + 4)l_{i+1}^3 + 2\gamma_1^4(1-\gamma_1)(\gamma_1^2 + \gamma_1 + 1)l_{i+1}^2(l_{i+2} + 2k_{i+2}) + 4\gamma_1^5(\gamma_1^2 - 3\gamma_1 + 3)l_{i+1}^2k_{i+1} + \gamma_1^4(1-\gamma_1)(1+\gamma_1)l_{i+1}(l_{i+2}^2 + 3l_{i+2}k_{i+2} + 3k_{i+2}^2) + \gamma_1^4k_{i+1}(6l_{i+1}l_{i+2} + 12l_{i+1}k_{i+2} + 2l_{i+2}^2 + 3l_{i+2}k_{i+1} + 6l_{i+2}k_{i+2} + 6k_{i+1}k_{i+2} + 6k_{i+2}^2) + 3\gamma_1^5(2-\gamma_1)l_{i+1}k_{i+1}^2 + 6\gamma_1^3(9\gamma_1-4)(k_i + k_{i+1})k_{i+1}k_i\,, \tag{C.5}$$

$$\begin{aligned} C_6 = {} & \gamma_1^4(1-\gamma_1)(1+\gamma_1)l_{i+1}(l_{i\text{-}1}^2 + 3l_{i\text{-}1}l_i + 3l_{i\text{-}1}k_i + 3l_{i\text{-}1}k_{i+1} + 2l_i^2 + 6l_ik_i \\ & + 3k_{i+1}l_i + 3k_i^2 + 6k_ik_{i+1}) - \gamma_1^4 k_{i+1}(2l_{i\text{-}1}^2 + 6l_{i\text{-}1}l_i + 6l_{i\text{-}1}k_i + 3l_{i\text{-}1}k_{i+1} + 4l_i^2 \\ & + 12l_ik_i + 3k_{i+1}l_i + 6k_i^2 + 6k_ik_{i+1}) - \gamma_1^4(2\gamma_1+1)(\gamma_1-1)^2(l_{i\text{-}1} + l_i + 2k_i) \\ & l_{i+1}^2\,, \end{aligned} \tag{C.6}$$

$$\begin{aligned} C_7 = {} & 6(2\gamma_1\gamma_2 - \gamma_1 - \gamma_2)(-(1-\gamma_2)(2\gamma_1\gamma_2 - \gamma_1 - \gamma_2)\gamma_1(l_{i\text{-}1} + 2k_i) + (3\gamma_1^2\gamma_2^2 \\ & - 3\gamma_1^2\gamma_2 + \gamma_1^2 - 3\gamma_1\gamma_2^2 + \gamma_1\gamma_2 + \gamma_2^2)l_i - (2\gamma_1\gamma_2 - \gamma_1 - \gamma_2)\gamma_2((\gamma_1^2 - \gamma_1 + 1) \\ & l_{i+1} + 2(1-\gamma_1)k_{i+1}) - (2\gamma_1\gamma_2 - \gamma_1 - \gamma_2)\gamma_1\gamma_2(l_{i+2} + 2k_{i+2}))\,. \end{aligned} \tag{C.7}$$

APPENDIX D

Appendix to Section 8.5.2

The terms C_1 and C_2, functions of α_i, α_{i+1} and α_{i+2}, in Eqs. (8.138)-(8.140) are given by

$$\begin{aligned}
C_1(\alpha_i,\alpha_{i+1},\alpha_{i+2}) = \Big[&(8\alpha_{i+1}^7 + 44\alpha_{i+1}^6\alpha_i + 24\alpha_{i+1}^6\alpha_{i+2} + 71\alpha_{i+1}^5\alpha_i^2 + 134\alpha_{i+1}^5\alpha_i \\
&\alpha_{i+2} + 31\alpha_{i+1}^5\alpha_{i+2}^2 + 44\alpha_{i+1}^4\alpha_i^3 + 222\alpha_{i+1}^4\alpha_i^2\alpha_{i+2} + 176\alpha_{i+1}^4 \\
&\alpha_i\alpha_{i+2}^2 + 14\alpha_{i+1}^4\alpha_{i+2}^3 + 8\alpha_{i+1}^3\alpha_i^4 + 144\alpha_{i+1}^3\alpha_i^3\alpha_{i+2} + 300\alpha_{i+1}^3 \\
&\alpha_i^2\alpha_{i+2}^2 + 80\alpha_{i+1}^3\alpha_i\alpha_{i+2}^3 + 30\alpha_{i+1}^2\alpha_i^4\alpha_{i+2} + 204\alpha_{i+1}^2\alpha_i^3\alpha_{i+2}^2 \\
&+ 138\alpha_{i+1}^2\alpha_i^2\alpha_{i+2}^3 + 48\alpha_{i+1}\alpha_i^4\alpha_{i+2}^2 + 96\alpha_{i+1}\alpha_i^3\alpha_{i+2}^3 + 24\alpha_i^4 \\
&\alpha_{i+2}^3)l^3 + (\alpha_{i+1} + \alpha_i)(29\alpha_{i+1}^6 + 105\alpha_{i+1}^5\alpha_i + 95\alpha_{i+1}^5\alpha_{i+2} \\
&+ 105\alpha_{i+1}^4\alpha_i^2 + 357\alpha_{i+1}^4\alpha_i\alpha_{i+2} + 121\alpha_{i+1}^4\alpha_{i+2}^2 + 29\alpha_{i+1}^3\alpha_i^3 \\
&+ 369\alpha_{i+1}^3\alpha_i^2\alpha_{i+2} + 459\alpha_{i+1}^3\alpha_i\alpha_{i+2}^2 + 53\alpha_{i+1}^3\alpha_{i+2}^3 + 111\alpha_{i+1}^2 \\
&\alpha_i^3\alpha_{i+2} + 486\alpha_{i+1}^2\alpha_i^2\alpha_{i+2}^2 + 201\alpha_{i+1}^2\alpha_i\alpha_{i+2}^3 + 156\alpha_{i+1}\alpha_i^3\alpha_{i+2}^2 \\
&+ 216\alpha_{i+1}\alpha_i^2\alpha_{i+2}^3 + 72\alpha_i^3\alpha_{i+2}^3)l^2k + 3(\alpha_{i+1} + \alpha_{i+2})(\alpha_{i+1} \\
&+ \alpha_i)^2(11\alpha_{i+1}^4 + 24\alpha_{i+1}^3\alpha_i + 29\alpha_{i+1}^3\alpha_{i+2} + 11\alpha_{i+1}^2\alpha_i^2 \\
&+ 66\alpha_{i+1}^2\alpha_i\alpha_{i+2} + 21\alpha_{i+1}^2\alpha_{i+2}^2 + 32\alpha_{i+1}\alpha_i^2\alpha_{i+2} + 48\alpha_{i+1}\alpha_i \\
&\alpha_{i+2}^2 + 24\alpha_i^2\alpha_{i+2}^2)lk^2 + 12(\alpha_{i+1} + 2\alpha_{i+2})(\alpha_{i+1} + \alpha_{i+2})^2(\alpha_{i+1} \\
&+ \alpha_i)^4k^3\Big] \times \Big[3(\alpha_i + \alpha_{i+1})^2((2\alpha_{i+1}^3 + 3\alpha_{i+1}^2(\alpha_i + \alpha_{i+2}) \\
&+ 2\alpha_{i+1}(\alpha_i + \alpha_{i+2})^2 + 2\alpha_i\alpha_{i+2}(\alpha_i + \alpha_{i+2}))l + 2(\alpha_{i+1} \\
&+ \alpha_{i+2})(\alpha_{i+1} + \alpha_i)(\alpha_{i+1} + \alpha_i + \alpha_{i+2})k)^2\Big]^{-1},
\end{aligned} \tag{D.1}$$

$$
\begin{aligned}
C_2(\alpha_i,\alpha_{i+1},\alpha_{i+2}) = \Big[&(8\alpha_{i+1}^7 + 44\alpha_{i+1}^6(\alpha_i+\alpha_{i+2}) + 71\alpha_{i+1}^5(\alpha_i^2+\alpha_{i+2}^2) + 206\alpha_{i+1}^5 \\
&\alpha_i\alpha_{i+2} + 44\alpha_{i+1}^4\alpha_i^3 + 308\alpha_{i+1}^4\alpha_i\alpha_{i+2}(\alpha_i+\alpha_{i+2}) + 44\alpha_{i+1}^4 \\
&\alpha_{i+2}^3 + 8\alpha_{i+1}^3\alpha_i^4 + 181\alpha_{i+1}^3\alpha_i\alpha_{i+2}(\alpha_i^2+\alpha_{i+2}^2) + 430\alpha_{i+1}^3\alpha_i^2 \\
&\alpha_{i+2}^2 + 8\alpha_{i+1}^3\alpha_{i+2}^4 + 32\alpha_{i+1}^2\alpha_i^4\alpha_{i+2} + 232\alpha_{i+1}^2\alpha_i^2\alpha_{i+2}^2(\alpha_i \\
&+\alpha_{i+2}) + 32\alpha_{i+1}^2\alpha_i\alpha_{i+2}^4 + 36\alpha_{i+1}\alpha_i^2\alpha_{i+2}^2(\alpha_i^2+\alpha_{i+2}^2) + 108 \\
&\alpha_{i+1}\alpha_i^3\alpha_{i+2}^3 + 12\alpha_i^3\alpha_{i+2}^3(\alpha_i+\alpha_{i+2}))\alpha_{i+1}l^3 + (\alpha_{i+1}+\alpha_i) \\
&(\alpha_{i+1}+\alpha_{i+2})(29\alpha_{i+1}^6 + 105\alpha_{i+1}^5(\alpha_i+\alpha_{i+2}) + 105\alpha_{i+1}^4(\alpha_i^2 \\
&+\alpha_{i+2}^2) + 351\alpha_{i+1}^4\alpha_i\alpha_{i+2} + 29\alpha_{i+1}^3(\alpha_i^3+\alpha_{i+2}^3) + 329\alpha_{i+1}^3\alpha_i \\
&\alpha_{i+2}(\alpha_i+\alpha_{i+2}) + 87\alpha_{i+1}^2\alpha_i\alpha_{i+2}(\alpha_i^2+\alpha_{i+2}^2) + 282\alpha_{i+1}^2\alpha_i^2 \\
&\alpha_{i+2}^2 + 66\alpha_{i+1}\alpha_i^2\alpha_{i+2}^2(\alpha_i+\alpha_{i+2}) + 12\alpha_i^3\alpha_{i+2}^3)l^2k + 3(\alpha_{i+1} \\
&+\alpha_i)^2(\alpha_{i+1}+\alpha_{i+2})^2(11\alpha_{i+1}^4 + 24\alpha_{i+1}^3(\alpha_i+\alpha_{i+2}) + 11\alpha_{i+1}^2 \\
&(\alpha_i^2+\alpha_{i+2}^2) + 51\alpha_{i+1}^2\alpha_i\alpha_{i+2} + 22\alpha_{i+1}\alpha_i\alpha_{i+2}(\alpha_i+\alpha_{i+2}) + 8 \\
&\alpha_i^2\alpha_{i+2}^2)lk^2 + 12(\alpha_{i+1}+\alpha_i)^4(\alpha_{i+1}+\alpha_{i+2})^4k^3\Big] \times \Big[3(\alpha_i \\
&+\alpha_{i+1})^2(\alpha_{i+1}+\alpha_{i+2})^2((2\alpha_{i+1}^3 + 3\alpha_{i+1}^2(\alpha_i+\alpha_{i+2}) + 2\alpha_{i+1} \\
&(\alpha_i+\alpha_{i+2})^2 + 2\alpha_i\alpha_{i+2}(\alpha_i+\alpha_{i+2}))l + 2(\alpha_{i+1}+\alpha_{i+2})(\alpha_{i+1} \\
&+\alpha_i)(\alpha_{i+1}+\alpha_i+\alpha_{i+2})k)^2\Big]^{-1}. \qquad \text{(D.2)}
\end{aligned}
$$

For the first $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet in Table 8.9 the derivative of y^*_{COG} to γ_1 is given by

$$
\begin{aligned}
\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} = \Big[&(l^3(\gamma_1^3(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2))(\gamma_2^2\gamma_3^2(2-\gamma_1) + (1-\gamma_3)(1-\gamma_2)(3\gamma_3 \\
&\gamma_2 + 4(1-\gamma_3)(1-\gamma_2) + \gamma_3\gamma_2(1-\gamma_1))) + 2\gamma_1(\gamma_3\gamma_2 + (1-\gamma_3)(1 \\
&-\gamma_2))(1-\gamma_3)(1-\gamma_2)(9(1-\gamma_2)(1-\gamma_3)(1-\gamma_1) + 3(1-\gamma_3)(1 \\
&-\gamma_2) + 12\gamma_3\gamma_2(1-\gamma_1) + 2\gamma_3\gamma_2) + \gamma_1\gamma_2^2\gamma_3^2(12(1-\gamma_3)(1-\gamma_2)(1 \\
&-\gamma_1) + 4(1-\gamma_3)(1-\gamma_2) + 13\gamma_3\gamma_2(1-\gamma_1) + 5\gamma_3\gamma_2) + 14(1-\gamma_2) \\
&(1-\gamma_3)(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2))^2 + 3\gamma_2\gamma_3(1-\gamma_2)(1-\gamma_3)(\gamma_3\gamma_2 \\
&+(1-\gamma_3)(1-\gamma_2)) + \gamma_2^2\gamma_3^2(6(1-\gamma_3)(1-\gamma_2)(1-\gamma_1) + (1-\gamma_3) \\
&(1-\gamma_2) + 6\gamma_3\gamma_2(1-\gamma_1) + 2\gamma_3\gamma_2)) + l^2k(4\gamma_1^3(1-\gamma_2)(1-\gamma_3)(\gamma_3\gamma_2 \\
&+(1-\gamma_3)(1-\gamma_2))^2 + 27(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2))^2(1-\gamma_3)(1 \\
&-\gamma_2)\gamma_1(1-\gamma_1) + 6(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2))(3\gamma_3\gamma_2 + 2(1-\gamma_3) \\
&(1-\gamma_2))\gamma_3\gamma_2\gamma_1(1-\gamma_1) + 15(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2))(1-\gamma_3)^2(1-
\end{aligned}
$$

$$\gamma_2)^2\gamma_1 + 3(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2))\gamma_3(1-\gamma_3)\gamma_2(1-\gamma_2)\gamma_1(5-\gamma_1)$$
$$+ 53(1-\gamma_2)(1-\gamma_3)(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2))^2 + 15\gamma_2\gamma_3(1-\gamma_2)$$
$$(1-\gamma_3)(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2)) + \gamma_2^2\gamma_3^2(29\gamma_3\gamma_2 + 27(1-\gamma_3)(1$$
$$-\gamma_2))) + 3lk^2(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2))(((2\gamma_3\gamma_2 + 3(1-\gamma_3)(1$$
$$-\gamma_2))\gamma_1(1-\gamma_1) + 3(1-\gamma_3)(1-\gamma_2)(\gamma_1+7))(\gamma_3\gamma_2 + (1-\gamma_3)(1$$
$$-\gamma_2)) + \gamma_2\gamma_3(11\gamma_3\gamma_2 + 8(1-\gamma_3)(1-\gamma_2))) + k^3 12(\gamma_3\gamma_2 + 2(1$$
$$-\gamma_3)(1-\gamma_2))(\gamma_3\gamma_2 + (1-\gamma_3)(1-\gamma_2))^2)\gamma_2\gamma_3\Big] \times \Big[3(l(2(1-\gamma_2)(1$$
$$-\gamma_3)(1-\gamma_1) - 2(1-\gamma_1)\gamma_3(1-\gamma_3) - 2(1-\gamma_1)\gamma_2(1-\gamma_2) - (6(1$$
$$-\gamma_1) + 1 + \gamma_1^2)\gamma_2\gamma_3(\gamma_2+\gamma_3) + (5(1-\gamma_1) + 2(1+\gamma_1^2))\gamma_2^2\gamma_3^2$$
$$+ (8(1-\gamma_1) + 1 + \gamma_1^2)\gamma_2\gamma_3) + 2k(\gamma_2\gamma_3 + (1-\gamma_2)(1-\gamma_3))(\gamma_2\gamma_3$$
$$+ (1-\gamma_1)(1-\gamma_2)(1-\gamma_3)))^2\Big]^{-1}, \tag{D.3}$$

while the derivative of y^*_{COG} to γ_2 is given by

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_2} = \Big[(1-\gamma_1)(1-\gamma_3)\gamma_3(l^3(((1-\gamma_2)(1-\gamma_3) + \gamma_2\gamma_3)^4(-1\gamma_1^3 + 7\gamma_1^2 + 8)$$
$$+ (1-\gamma_2)^4(1-\gamma_3)^4(-1\gamma_1^2 + 5\gamma_1 + 6)\gamma_1 + 4(1-\gamma_2)^3(1-\gamma_3)^3\gamma_2\gamma_3$$
$$(-1\gamma_1^3 + 5\gamma_1^2 + 3\gamma_1 + 3) + (1-\gamma_2)^2(1-\gamma_3)^2\gamma_2^2\gamma_3^2(-5\gamma_1^3 + 27\gamma_1^2$$
$$+ 9\gamma_1 + 23) + 2(1-\gamma_2)(1-\gamma_3)\gamma_2^3\gamma_3^3(-1\gamma_1^3 + 6\gamma_1^2 + 1\gamma_1 + 6))$$
$$+ l^2k(\gamma_2\gamma_3 + (1-\gamma_2)(1-\gamma_3))(((1-\gamma_2)(1-\gamma_3) + \gamma_2\gamma_3)^3(-2\gamma_1^3$$
$$+ 18\gamma_1^2 + 8\gamma_1 + 29) + (1-\gamma_2)^3(1-\gamma_3)^3(3\gamma_1^2 + 16\gamma_1) + (1-\gamma_2)^2$$
$$(1-\gamma_3)^2\gamma_2\gamma_3(9\gamma_1^2 + 30\gamma_1 + 18) + 6(1-\gamma_2)(1-\gamma_3)\gamma_2^2\gamma_3^2(\gamma_1^2 + 3\gamma_1$$
$$+ 3)) + lk^2 3((1-\gamma_2)(1-\gamma_3) + \gamma_2\gamma_3)^2((1-\gamma_2-\gamma_3)^2(3\gamma_1^2 + 7\gamma_1$$
$$+ 11) + 3\gamma_1(1-\gamma_3)^2(1-\gamma_2)^2 + 2\gamma_2\gamma_3(1-\gamma_2)(1-\gamma_3)(6\gamma_1^2 + 16\gamma_1$$
$$+ 23)) + k^3 12(\gamma_2\gamma_3 + (1-\gamma_3)(1-\gamma_2))^4(1+\gamma_1))\Big] \times \Big[3(\gamma_2\gamma_3 + (1$$
$$-\gamma_2)(1-\gamma_3))^2(l(2(1-\gamma_2)(1-\gamma_3)(1-\gamma_1) - 2(1-\gamma_1)\gamma_3(1-\gamma_3)$$
$$- 2(1-\gamma_1)\gamma_2(1-\gamma_2) - (6(1-\gamma_1) + 1 + \gamma_1^2)\gamma_2\gamma_3(\gamma_2+\gamma_3)$$
$$+ (5(1-\gamma_1) + 2(1+\gamma_1^2))\gamma_2^2\gamma_3^2 + (8(1-\gamma_1) + 1 + \gamma_1^2)\gamma_2\gamma_3) + 2k$$
$$(\gamma_2\gamma_3 + (1-\gamma_2)(1-\gamma_3))(\gamma_2\gamma_3 + (1-\gamma_1)(1-\gamma_2)(1-\gamma_3)))^2\Big]^{-1}. \tag{D.4}$$

For the fourth $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet in Table 8.9 the derivative of y^*_{COG} to γ_1 is equal to zero

$$\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} = 0\,, \tag{D.5}$$

while the derivative of y^*_{COG} to γ_2 is given by

$$\begin{aligned}\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_2} = \Big[&\gamma_3(l^3(\gamma_2^2\gamma_3(1-\gamma_2)(1+\gamma_2) + 4\gamma_2^3(1-\gamma_3) + \gamma_2\gamma_3^3(2-\gamma_2) + 6\gamma_2\gamma_3^2\\ &(1-\gamma_2) + 11(1-\gamma_3) + 3(1-\gamma_2^2) + 9\gamma_2(1-\gamma_3) + 11\gamma_2(1-\gamma_2)(1\\ &-\gamma_3) + 4\gamma_2(1-\gamma_2) + 2\gamma_2^3\gamma_3^2 + \gamma_3^3 + 4\gamma_3^2) + l^2k(2\gamma_3^3 + 3\gamma_3^2\gamma_2(2\\ &-\gamma_2) + 12\gamma_3^2 + 4\gamma_3\gamma_2^2(3-\gamma_2) + 30\gamma_2(1-\gamma_3) + 38(1-\gamma_3) + 4\gamma_2^3\\ &+12\gamma_2(1-\gamma_2) + 15(1-\gamma_2^2)) + lk^2(9\gamma_3^2 + 3\gamma_3\gamma_2^2 + 12\gamma_2(1-\gamma_3)\\ &+39(1-\gamma_3) + 9(1-\gamma_2^2) + 6\gamma_2 + 15) + 12k^3(2-\gamma_3))\Big] \times \Big[3(l(\gamma_2^2\gamma_3\\ &+\gamma_2\gamma_3 + (2-\gamma_3+\gamma_3^2)(1-\gamma_2)) + 2k(\gamma_2\gamma_3 + (1-\gamma_2)))^2\Big]^{-1}\,,\end{aligned} \tag{D.6}$$

and the derivative of y^*_{COG} to γ_3 is given by

$$\begin{aligned}\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_3} = \Big[&(1-\gamma_2)(l^3((1-\gamma_3)^2(1-\gamma_2)\gamma_3(2-\gamma_3) + 4(1-\gamma_3)^3\gamma_2 + (1-\gamma_3)\\ &(1-\gamma_2)^3(1+\gamma_3) + 6(1-\gamma_3)(1-\gamma_2)^2\gamma_3 + 11\gamma_2 + 3(2-\gamma_3)\gamma_3\\ &+9(1-\gamma_3)\gamma_2 + 11(1-\gamma_3)\gamma_3\gamma_2 + 4(1-\gamma_3)\gamma_3 + 2(1-\gamma_3)^3(1\\ &-\gamma_2)^2 + (1-\gamma_2)^3 + 4(1-\gamma_2)^2) + l^2k(2(1-\gamma_2)^3 + 3(1-\gamma_2)^2\\ &(1-\gamma_3^2) + 12(1-\gamma_2)^2 + 4(1-\gamma_2)(1-\gamma_3)^2(2+\gamma_3) + 30(1-\gamma_3)\gamma_2\\ &+38\gamma_2 + 4(1-\gamma_3)^3 + 12(1-\gamma_3)\gamma_3 + 15(2-\gamma_3)\gamma_3) + lk^2(9(1\\ &-\gamma_2)^2 + 3(1-\gamma_2)(1-\gamma_3)^2 + 12(1-\gamma_3)\gamma_2 + 39\gamma_2 + 9(2-\gamma_3)\gamma_3\\ &+6(2-\gamma_3)\gamma_3 + 6(1-\gamma_3) + 15) + 12k^3(1+\gamma_2))\Big] \times \Big[3(l((2-\gamma_3)\\ &(1-\gamma_3)(1-\gamma_2) + (2-\gamma_2+\gamma_2^2)\gamma_3) + 2k((1-\gamma_2)+\gamma_2\gamma_3))^2\Big]^{-1}\,.\end{aligned} \tag{D.7}$$

For the twelfth $(\alpha_i, \alpha_{i+1}, \alpha_{i+2})$-triplet in Table 8.9 the derivative of y^*_{COG} to γ_1 is given by

$$
\begin{aligned}
\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} = \Big[&\gamma_2\gamma_3(\gamma_2+\gamma_3-1)(l^3(1-\gamma_1)(12\gamma_1(\gamma_2\gamma_3-\gamma_1(\gamma_2+\gamma_3-1))^4+(\gamma_1^2\\
&-2\gamma_1+6)(2-\gamma_1)\gamma_1(1-\gamma_2)(1-\gamma_3)(\gamma_2\gamma_3-\gamma_1(\gamma_2+\gamma_3-1))^3\\
&+(-3\gamma_1^3+9\gamma_1^2-16\gamma_1+17)\gamma_1(1-\gamma_2)^2(1-\gamma_3)^2(\gamma_2\gamma_3-\gamma_1(\gamma_2\\
&+\gamma_3-1))^2+(-\gamma_1^4+2\gamma_1^3-\gamma_1^2+8)(1-\gamma_1)^3(\gamma_2+\gamma_3-1)^4+32(1\\
&-\gamma_1)^2(1-\gamma_2)(1-\gamma_3)(\gamma_2+\gamma_3-1)^3+(10\gamma_1^4-31\gamma_1^3+26\gamma_1^2-17\\
&\gamma_1+48)(1-\gamma_1)(1-\gamma_2)^2(1-\gamma_3)^2(\gamma_2+\gamma_3-1)^2+(3\gamma_1^5+15\gamma_1^4\\
&-52\gamma_1^3+48\gamma_1^2-34\gamma_1+32)(1-\gamma_2)^3(1-\gamma_3)^3(\gamma_2+\gamma_3-1)+(4\gamma_1^3\\
&+13\gamma_1^2-\gamma_1+8)(1-\gamma_1)(1-\gamma_2)^4(1-\gamma_3)^4)+l^2k(\gamma_2\gamma_3-\gamma_1(\gamma_2\\
&+\gamma_3-1))(12(\gamma_2\gamma_3-\gamma_1(\gamma_2+\gamma_3-1))^3+18(1-\gamma_1)\gamma_1(\gamma_2\gamma_3-\gamma_1\\
&(\gamma_2+\gamma_3-1))^3+2(-2\gamma_1^3+7\gamma_1+2)(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)(\gamma_2\gamma_3\\
&-\gamma_1(\gamma_2+\gamma_3-1))^2+17(1-\gamma_1)(\gamma_2+\gamma_3-1)(\gamma_2\gamma_3-\gamma_1(\gamma_2+\gamma_3\\
&-1))^2+(-4\gamma_1^2+8\gamma_1+13)(1-\gamma_1)^2(1-\gamma_2)(1-\gamma_3)(\gamma_2+\gamma_3-1)^2\\
&+(\gamma_1^4-4\gamma_1^3-26\gamma_1^2+16\gamma_1+26)(1-\gamma_1)(1-\gamma_2)^2(1-\gamma_3)^2(\gamma_2\\
&+\gamma_3-1)+(-5\gamma_1^3-\gamma_1^2+21\gamma_1+13)(1-\gamma_1)(1-\gamma_2)^3(1-\gamma_3)^3)\\
&+3lk^2(\gamma_2\gamma_3-\gamma_1(\gamma_2+\gamma_3-1))^2(2(-\gamma_1^2+\gamma_1+4)(\gamma_2\gamma_3-\gamma_1(\gamma_2\\
&+\gamma_3-1))^2+3((1-\gamma_2)(1-\gamma_3)+\gamma_2\gamma_3-\gamma_1(\gamma_2+\gamma_3-1))(1-\gamma_1)\\
&(\gamma_2+\gamma_3-1)+(2+\gamma_1)\gamma_1(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)((1-\gamma_2)(1-\gamma_3)\\
&+\gamma_2\gamma_3-\gamma_1(\gamma_2+\gamma_3-1))+(1-\gamma_2)^2(1-\gamma_3)^2(3+\gamma_1)(1-\gamma_1))\\
&+12k^3(\gamma_2\gamma_3-\gamma_1(\gamma_2+\gamma_3-1))^4)\Big]\times\Big[3(\gamma_2\gamma_3-\gamma_1(\gamma_2+\gamma_3-1))^2\\
&(l((\gamma_1^2-\gamma_1+2)(1-\gamma_1)(\gamma_2+\gamma_3-1)^2+(-\gamma_1^3+2\gamma_1^2-3\gamma_1+4)(1\\
&-\gamma_2)(1-\gamma_3)(\gamma_2+\gamma_3-1)+2(\gamma_1^2+1)(1-\gamma_2)^2(1-\gamma_3)^2)+k(2(1\\
&-\gamma_1)(\gamma_2+\gamma_3-1)^2+2(-\gamma_1^2+2)(1-\gamma_2)(1-\gamma_3)(\gamma_2+\gamma_3-1)\\
&+2(1+\gamma_1)(1-\gamma_2)^2(1-\gamma_3)^2))^2\Big]^{-1},
\end{aligned}
\tag{D.8}
$$

while the derivative of y^*_{COG} to γ_2 is given by

$$
\begin{aligned}
\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_2} = \Big[&\gamma_1\gamma_3(1-\gamma_3)(l^3((-\gamma_1^3+7\gamma_1^2+8)(1-\gamma_1)^4(\gamma_2+\gamma_3-1)^4+2(-\gamma_1^4\\
&+4\gamma_1^3+15\gamma_1^2+6\gamma_1+16)(1-\gamma_1)^3(1-\gamma_2)(1-\gamma_3)(\gamma_2+\gamma_3-1)^3\\
&+(\gamma_1^5-15\gamma_1^4+33\gamma_1^3+35\gamma_1^2+36\gamma_1+48)(1-\gamma_1)^2(1-\gamma_2)^2(1\\
&-\gamma_3)^2(\gamma_2+\gamma_3-1)^2+4(\gamma_1^5-6\gamma_1^4+10\gamma_1^3+2\gamma_1^2+9\gamma_1+8)(1-\gamma_1)
\end{aligned}
$$

$$
\begin{aligned}
&(1-\gamma_2)^3(1-\gamma_3)^3(\gamma_2+\gamma_3-1)+4(\gamma_1^2+1)(\gamma_1^3-3\gamma_1^2+3\gamma_1+2)(1\\
&-\gamma_2)^4(1-\gamma_3)^4)+l^2k(\gamma_3\gamma_2-\gamma_1(\gamma_2+\gamma_3-1))((-2\gamma_1^3+18\gamma_1^2\\
&+8\gamma_1+29)(1-\gamma_1)^3(\gamma_2+\gamma_3-1)^3+3(1-\gamma_2)(1-\gamma_3)(24\gamma_1^2+14\\
&\gamma_1+29)(1-\gamma_1)^2(\gamma_2+\gamma_3-1)^2+3(1-\gamma_2)^2(1-\gamma_3)^2(-\gamma_1^4+24\gamma_1^2\\
&+20\gamma_1+29)(1-\gamma_1)(\gamma_2+\gamma_3-1)+(1-\gamma_2)^3(1-\gamma_3)^3(\gamma_1^4-2\gamma_1^3\\
&+18\gamma_1^2+26\gamma_1+29))+3lk^2(\gamma_3\gamma_2-\gamma_1(\gamma_2+\gamma_3-1))^2((3\gamma_1^2+7\gamma_1\\
&+11)(1-\gamma_1)^2(\gamma_2+\gamma_3-1)^2+2(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)(5\gamma_1^2+8\\
&\gamma_1+11)(\gamma_2+\gamma_3-1)+(1-\gamma_2)^2(1-\gamma_3)^2(-\gamma_1^3+5\gamma_1^2+9\gamma_1+11))\\
&+12k^3(1+\gamma_1)(\gamma_3\gamma_2-\gamma_1(\gamma_2+\gamma_3-1))^4)\Bigg]\times\Bigg[3(\gamma_3\gamma_2-\gamma_1(\gamma_2+\gamma_3\\
&-1))^2(l((\gamma_1^2-\gamma_1+2)(1-\gamma_1)(\gamma_2+\gamma_3-1)^2+(-\gamma_1^3+2\gamma_1^2-3\gamma_1\\
&+4)(1-\gamma_2)(1-\gamma_3)(\gamma_2+\gamma_3-1)+2(\gamma_1^2+1)(1-\gamma_2)^2(1-\gamma_3)^2)\\
&+k(2(1-\gamma_1)(\gamma_2+\gamma_3-1)^2+2(-\gamma_1^2+2)(1-\gamma_2)(1-\gamma_3)(\gamma_2+\gamma_3\\
&-1)+2(1+\gamma_1)(1-\gamma_2)^2(1-\gamma_3)^2))^2\Bigg]^{-1}. \qquad \text{(D.9)}
\end{aligned}
$$

For the first $(\alpha_{i+1}, \alpha_{i+2})$-pair in Table 8.10 the derivative of y^*_{COG} to γ_1 is given by

$$
\begin{aligned}
\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_1} = &\Bigg[(1-\gamma_2)(1-\gamma_3)(l^3(((-\gamma_1^3-5\gamma_1+12)\gamma_1+2\gamma_1^3(1+\gamma_2)+3\gamma_1^2\gamma_2\\
&+2\gamma_2^2)(1-\gamma_2)(1-\gamma_3)+(2\gamma_1^3\gamma_2\gamma_3+8(1-\gamma_1^2)+1\gamma_1^2\gamma_2^3+6\gamma_1^2\gamma_2\gamma_3\\
&+10\gamma_2^2+2\gamma_2\gamma_3^2)(1-\gamma_3)+(-4\gamma_1^3+\gamma_1^2+6)\gamma_2+14(1-\gamma_1^2)\gamma_2\gamma_3\\
&+\gamma_1^2\gamma_2\gamma_3^3+16\gamma_2\gamma_3^2)+l^2k(\gamma_2(4(6-\gamma_1^3)+\gamma_3(-27\gamma_1^2+31\gamma_3+53))\\
&+(1-\gamma_3)(3\gamma_1^2\gamma_2(2-\gamma_2+2\gamma_3)+18(1-\gamma_1^2)+2(9\gamma_1+\gamma_2^2)(1-\gamma_2)\\
&+19\gamma_2^2+2\gamma_2\gamma_3^2+11))+3lk^2((2(1-\gamma_1)(1+\gamma_1)+2\gamma_1(1-\gamma_2)\\
&+3\gamma_2^2+9)(1-\gamma_3)+\gamma_2(1-\gamma_1)(1+\gamma_1)(1+4\gamma_3)+\gamma_2(5\gamma_3^2+17\gamma_3\\
&+9))+12k^3(\gamma_2(1+2\gamma_3)+1(1-\gamma_3)))\Bigg]\times\Bigg[3(l((\gamma_1^2-\gamma_1+2)\\
&(1-\gamma_2)(1-\gamma_3)+1\gamma_1\gamma_2(\gamma_3^2+\gamma_2(1-\gamma_3)+1))+2k(\gamma_1\gamma_2\\
&+(1-\gamma_2)(1-\gamma_3)))^2\Bigg]^{-1}, \qquad \text{(D.10)}
\end{aligned}
$$

while the derivative of y^*_{COG} to γ_2 is given by

$$\begin{aligned}
\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_2} = \Bigg[&\gamma_1(1-\gamma_3)(l^3((11\gamma_2^2\gamma_3 + 3\gamma_3 + 13\gamma_3^2 + 3\gamma_2^2\gamma_3)(1-\gamma_1) + 11(\gamma_2+1)\\
&(1-\gamma_1)(1-\gamma_2) + (2\gamma_1\gamma_2^3\gamma_3 + 4\gamma_2^3 + 20\gamma_2)(1-\gamma_1)(1-\gamma_3) + 3(\gamma_2\\
&+1)(1-\gamma_2) + (\gamma_1^3\gamma_2 + 6\gamma_1^2\gamma_2 + \gamma_1\gamma_2^3 + \gamma_1\gamma_2^2 + 4\gamma_2)(1-\gamma_2)(1-\gamma_3)\\
&+ (\gamma_1^3\gamma_2 + 2\gamma_1^2\gamma_2^3\gamma_3 + 2\gamma_1^2\gamma_2^3 + \gamma_1^2\gamma_3^2 + 2\gamma_3^2)(1-\gamma_3) + (3\gamma_3+1)\gamma_3\\
&(1-\gamma_1\gamma_2^2) + 4\gamma_1(\gamma_1 - \gamma_2^3\gamma_3) + \gamma_1^3\gamma_3^2 + \gamma_1^3 + 5\gamma_1^2\gamma_3^2 + 3\gamma_1^2\gamma_3 + \gamma_1\gamma_2^2\\
&\gamma_3^3 + \gamma_1\gamma_3^3 + 2\gamma_3) + l^2k((-12\gamma_1 + 4\gamma_2^3 + 18\gamma_3^2 + 9\gamma_3 + 53)(1-\gamma_1)\\
&+ 4(6\gamma_1 + 6\gamma_1\gamma_2\gamma_3 + \gamma_2^2\gamma_3 + \gamma_2\gamma_3 + \gamma_3)(1-\gamma_2) + 3\gamma_1(1-\gamma_2)^2\\
&+ (3\gamma_1^2\gamma_2 + 15\gamma_2 + 3\gamma_1\gamma_2^2 + 3\gamma_1\gamma_2^2\gamma_3 + 2\gamma_3^2)(1-\gamma_3) + 3(\gamma_1^2+9)\gamma_2\\
&(1-\gamma_2)(1-\gamma_3) + 2\gamma_1^3 + 6\gamma_1^2\gamma_3^2 + 9\gamma_1^2\gamma_3 + 6\gamma_1\gamma_2^2 + 6\gamma_1\gamma_2\gamma_3 + 13\gamma_3^2\\
&+ 11\gamma_3) + 3lk^2((3\gamma_2 + 2\gamma_3^2 + 2\gamma_3 + 13)(1-\gamma_1) + (4\gamma_1\gamma_2\gamma_3 + 2\gamma_2\\
&+ 6\gamma_3)(1-\gamma_2) + \gamma_2(1-\gamma_1)(1-\gamma_2) + 2\gamma_1^2\gamma_3 + 3\gamma_1^2 + 3\gamma_2^2\gamma_3 + 3\gamma_3^2\\
&+ 2\gamma_3 + 8) + 12k^3(2-\gamma_1+\gamma_3))\Bigg] \times \Bigg[3(l((\gamma_1^2 - \gamma_1 + 2)(1-\gamma_2)\\
&(1-\gamma_3) + \gamma_1\gamma_2^2(1-\gamma_3) + \gamma_1\gamma_2(1+\gamma_3^2)) + 2k(\gamma_1\gamma_2 + (1-\gamma_2)\\
&(1-\gamma_3)))^2\Bigg]^{-1},
\end{aligned} \tag{D.11}$$

and the derivative of y^*_{COG} to γ_3 is given by

$$\begin{aligned}
\frac{\partial y^*_{\mathrm{COG}}}{\partial \gamma_3} = \Bigg[&\gamma_1\gamma_2(l^3((9\gamma_3+14)(1-\gamma_1)(1-\gamma_2) + 14\gamma_3(1-\gamma_1)(1-\gamma_2)(1-\gamma_3)\\
&+ (\gamma_1^3\gamma_3 + \gamma_1^3 + 2\gamma_1^2\gamma_3^3 + 3\gamma_1^2\gamma_3 + 7\gamma_1^2 + 4\gamma_3^3 + 1\gamma_3 + 6)(1-\gamma_2)\\
&+ (1-\gamma_1\gamma_2)(1-\gamma_2)\gamma_3 + (1-\gamma_1\gamma_2^2)(1-\gamma_2)\gamma_3 + (\gamma_1^3\gamma_3 + 9\gamma_1^2\gamma_3\\
&+ 10\gamma_3)(1-\gamma_2)(1-\gamma_3) + (\gamma_1\gamma_2^3\gamma_3 + 6\gamma_1\gamma_2^2\gamma_3 + \gamma_1\gamma_2\gamma_3^3 + 8\gamma_1\gamma_2\gamma_3\\
&+ 2\gamma_1\gamma_3^2)(1-\gamma_3) + 3\gamma_1(\gamma_2-\gamma_3) + 2\gamma_3^3\gamma_1\gamma_2^2 + \gamma_3^3\gamma_1\gamma_2 + \gamma_1\gamma_2^3\\
&+ 4\gamma_1\gamma_2^2) + l^2k((47+15\gamma_3)(1-\gamma_1)(1-\gamma_2) + (2\gamma_1^3 + 6\gamma_1^2\gamma_3 + 21\gamma_1^2\\
&+ 4\gamma_3^3 + 12\gamma_3 + 24)(1-\gamma_2) + 6(1-\gamma_1\gamma_3)(1-\gamma_2) + (6\gamma_1^2\gamma_3 + 39\gamma_3)\\
&(1-\gamma_2)(1-\gamma_3) + 3\gamma_1\gamma_3\gamma_2(11+\gamma_2)(1-\gamma_3) + 3(2\gamma_2+5)\gamma_1\\
&(\gamma_2-\gamma_3) + 18\gamma_1\gamma_3^2 + 3\gamma_1\gamma_3\gamma_2^2 + 2\gamma_1\gamma_2^3 + 6\gamma_1\gamma_2^2) + 3lk^2(15(1-\gamma_1)\\
&(1-\gamma_2) + (5\gamma_1^2 + 5\gamma_3 + 16)(1-\gamma_2) + 5\gamma_3(1-\gamma_2)(1-\gamma_3) + 4\gamma_1\gamma_2
\end{aligned}$$

$$\gamma_3(1-\gamma_3)+4\gamma_1(2\gamma_2-\gamma_3)+3\gamma_1\gamma_2^2+2\gamma_1\gamma_3\gamma_2+2\gamma_1\gamma_3^2)+12k^3(\gamma_1$$
$$\gamma_2+(3-\gamma_1)(1-\gamma_2)))\Big]\times\Big[3(l((\gamma_1^2-\gamma_1+2)(1-\gamma_2)(1-\gamma_3)+\gamma_1$$
$$\gamma_2^2(1-\gamma_3)+\gamma_1\gamma_2(1+\gamma_3^2))+2k(\gamma_1\gamma_2+(1-\gamma_2)(1-\gamma_3)))^2\Big]^{-1}. \tag{D.12}$$

BIBLIOGRAPHY

Adriaenssens, V., 2004. Knowledge-based macroinvertebrate habitat suitability models for use in ecological river management. Ph.D. thesis, Ghent University, Gent, Belgium.

Adriaenssens, V., De Baets, B., Goethals, P., De Pauw, N., 2004. Fuzzy rule-based models for decision support in ecosystem management. Sci. Total Environ. 319, 1–12.

Antonisse, J., 1989. A new interpretation of schema notation that overturns the binary encoding constraints. In: Schaffer, J. (Ed.), Proceedings of the Third International Conference on Genetic Algorithms. Morgan Kaufmann, Los Altos, CA, USA, pp. 86–91.

Arslan, A., Kaya, M., 2001. Determination of fuzzy logic membership functions using genetic algorithms. Fuzzy Sets Syst. 118, 297–306.

Ascia, G., Catania, V., Panno, D., 2006. An integrated fuzzy-GA approach for buffer management. IEEE Trans. Fuzzy Syst. 14, 528–531.

Assilian, S., 1974. Artificial intelligence in the control of real dynamical systems. Ph.D. thesis, London University, Great Britain.

Babuška, R., 1998. Fuzzy Modeling for Control. International Series in Intelligent Technologies. Kluwer Academic Publishers, Boston, USA.

Babuška, R., Verbruggen, H., 1997. Constructing fuzzy models by product space clustering. In: Hellendoorn, H., Driankov, D. (Eds.), Fuzzy Model Identification: Selected Approaches. Springer, Berlin, Germany, pp. 53–90.

Bäck, T., Fogel, D., Michalewicz, Z. (Eds.), 1997. Handbook of Evolutionary Computation. Oxford University Press, New York and Institute of Physics, London, UK.

Baker, J., 1985. Adaptive selection methods for genetic algorithms. In: Grefenstette, J. (Ed.), Proceedings of the First International Conference on Genetic Algorithms. Lawrence Erlbaum Associates, Hillsdale, MA, USA, pp. 101–111.

Baker, J., 1987. Reducing bias and inefficiency in the selection algorithm. In: Grefenstette, J. (Ed.), Proceedings of the Second International Conference on Genetic Algorithms. Lawrence Erlbaum Associates, Cambridge, MA, USA, pp. 14–21.

Baptist, M., van der Lee, G., Kerle, F., Mosselman, E., 2002. Modelling of morphodynamics, vegetation development and fish habitat in man-made secondary channels in the River Rhine, the Netherlands. In: Proceedings of Environmental Flows for River Systems Working Conference and Fourth International Ecohydraulics Symposium. Cape Town, South Africa, pp. 1–8.

Bastian, A., 2000. Identifying fuzzy models utilizing genetic programming. Fuzzy Sets Syst. 113, 333–350.

Bağiş, A., 2003. Determining fuzzy membership functions with tabu search — an application to control. Fuzzy Sets Syst. 139, 209–225.

Bayerisches Landesamt für Wasserwirtschaft, 1996. Ökologische Typisierung der aquatischen Makrofauna. Informationsberichte des Bayerischen Landesamtes für Wasserwirtschaft. Heft 4/96.

Ben-David, A., 1995. Monotonicity maintenance in information-theoretic machine learning algorithms. Mach. Learn. 19, 29–43.

Blickle, T., Thiele, L., 1995. A comparison of selection schemes used in genetic algorithms. TIK Report 11, Computer Engineering and Communication Networks Lab, Swiss Federal Institute of Technology, Zurich, Switzerland.

Bock, W., Salski, A., 1998. A fuzzy knowledge-based model of population dynamics of the yellow-necked mouse (*Apodemus flavicollis*) in a beech forest. Ecol. Model. 108, 155–161.

Bodenhofer, U., 1999. A Similarity-based Generalization of Fuzzy Orderings. Vol. 26 of Schrift der Johannes-Kepler-Universität Linz, Reihe C - Technik und Naturwissenschaften. Universitätsverlag Rudolf Trauner, Linz, Austria.

Bodenhofer, U., Bauer, P., 2005. Interpretability of linguistic variables: a formal account. Kybernetika 41 (2), 227–248.

Bodenhofer, U., Klement, E., 2001. Genetic optimization of fuzzy classification systems — A case study. In: Reusch, B., Temme, K.-H. (Eds.), Computational Intelligence in Theory and Practice. Advances in Soft Computing. Physica Verlag, Heidelberg, pp. 183–200.

Boone, R., Krohn, W., 1999. Modelling the occurrence of bird species. Are the errors predictable ? Ecol. Appl. 9, 835–848.

Borsuk, M., 2003. A Bayesian network of eutrophication models for synthesis, prediction and uncertainty analysis. Ecol. Model. 173, 219–239.

Botzheim, J., Cabrita, C., Kóczy, L., Ruano, A., 2004. Estimating fuzzy membership functions parameters by the Levenberg-Marquardt algorithm. In: Proceedings of the IEEE International Conference of Fuzzy Systems (FUZZ-IEEE). Budapest, Hungary, pp. 1667–1672.

Buckley, J., 1993. Sugeno type controllers are universal controllers. Fuzzy Sets Syst. 53, 299–303.

Calvo, T., Kolesárová, A., Komorníková, M., Mesiar, R., 2002. Aggregation operators: properties, classes and construction methods. In: Calvo, T., Mayor, G., Mesiar, R. (Eds.), Aggregation Operators. Vol. 97 of Studies in Fuzziness and Soft Computing. Springer Verlag, Heidelberg, Germany, pp. 3–104.

Campello, R., do Amaral, W., 2006. Hierarchical fuzzy relational models: linguistic interpretation and universal approximation. IEEE Trans. Fuzzy Syst. 14, 446–453.

Cao-Van, K., De Baets, B., 2003. Growing decision trees in an ordinal setting. Int. J. Intell. Syst. 18, 733–750.

Carmona, P., Castro, J., Zurita, J., 2005. Commutativity as prior knowledge in fuzzy modeling. Fuzzy Sets Syst. 152, 565–585.

Casillas, J., Cordón, O., del Jesus, M., Herrera, F., 2005. Genetic tuning of fuzzy rule deep structures preserving interpretability and its interaction with fuzzy rule set reduction. IEEE Trans. Fuzzy Syst. 13 (1), 13–29.

Casillas, J., Cordón, O., Herrera, F., Magdalena, L. (Eds.), 2003a. Accuracy Improvements in Fuzzy Modeling. Vol. 129 of Studies in Fuzziness and Soft Computing. Springer Verlag, Heidelberg.

Casillas, J., Cordón, O., Herrera, F., Magdalena, L. (Eds.), 2003b. Interpretability Issues in Fuzzy Modeling. Vol. 128 of Studies in Fuzziness and Soft Computing. Springer Verlag, Heidelberg.

Chang, T., Hasegawa, K., Ibbs, C., 1991. The effects of membership function on fuzzy reasoning. Fuzzy Sets Syst. 44, 169–186.

Chave, P., 2001. The EU Water Framework Directive, an introduction. IWA Publishing. 208 p.

Chiou, Y.-C., Lan, L., 2005. Genetic fuzzy logic controller: an iterative evolution algorithm with new encoding method. Fuzzy Sets Syst. 152, 617–635.

Cordón, O., Gomide, F., Herrera, F., Hoffmann, F., Magdalena, L., 2004. Ten years of genetic fuzzy systems: current framework and new trends. Fuzzy Sets Syst. 141 (1), 5–31.

Cordón, O., Herrera, F., Hoffmann, F., Magdalena, L., 2001. Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases. Vol. 19 of Advances in Fuzzy Systems — Applications and Theory. World Scientific, Singapore.

Cordón, O., Herrera, F., Peregrín, A., 1997. Applicability of the fuzzy operators in the design of fuzzy logic controllers. Fuzzy Sets Syst. 86, 15–41.

Cowley, M., Wilson, R., Léon-Cortès, J., Gutierrez, D., Bulman, C., Thomas, C., 2000. Habitat-based statistical models for predicting the spatial distribution of butterflies and day-flying moths in a fragmented landscape. J. Appl. Ecol. 37, 60–72.

Damousis, I., Satsios, K., Labridis, D., Dokopoulos, P., 2002. Combined fuzzy logic and genetic algorithm techniques — application to an electromagnetic field problem. Fuzzy Sets Syst. 129, 371–386.

Daniels, H., Velikova, M., 2006. Derivation of monotone decision models from noisy data. IEEE Trans. Syst. Man Cybern. Part C-Appl. Rev. 36, 705–710.

Davidson, V., Ryks, J., Chu, T., 2001. Fuzzy models to predict consumer ratings for biscuits based on digital image features. IEEE Trans. Fuzzy Syst. 9, 62–67.

De Baets, B., 1996. A note on Mamdani controllers. In: Ruan, D., D'hondt, P., Govaerts, P., Kerre, E. (Eds.), Proceedings of the 2nd International FLINS Workshop. Intelligent Systems and Soft Computing for Nuclear Science and Industry. Mol, Belgium, pp. 22–28.

De Baets, B., Kerre, E., 1993. The generalized modus ponens and the triangular fuzzy data model. Fuzzy Sets Syst. 59, 305–317.

De Cooman, W., Detemmerman, L., 2004. Waterbodemkwaliteit 2003. Tech. rep., Vlaamse Milieumaatschappij, Aalst, Belgium, in Dutch.

De Jong, K., 1975. An analysis of the behavior of a class of genetic adaptive systems. Ph.D. thesis, University of Michigan, MI, USA.

De Loose, L., Verheyen, R., Bervoets, L., Weiss, L., Schneiders, A., Wils, C., Van Maele, T., De Pauw, N., Baillieul, M., Blust, R., Vanderborght, O., 1995. Ecological targets for surface waters. Antwerp, Belgium, group Applied Ecology, 185 p. + appendices (in Dutch).

de Mol, R., Ouweltjes, W., 2000. Detection model for estrus and mastitis in cows milked in an automatic milking system. In: Proceedings of the International Symposium on Robotic Milking. Lelystad, the Netherlands, pp. 97–107.

de Mol, R., Woldt, W., 2001. Application of fuzzy logic in automated cow status monitoring. J. Dairy Sci 84, 400–410.

De Pauw, N., Vanhooren, G., 1983. Method for biological quality assessment of water courses in Belgium. Hydrobiologia 100, 153–168.

Dedecker, A., Goethals, P., De Pauw, N., 2002. Comparison of artificial neural network (ANN) model development methods for prediction of macroinvertebrate communities in the Zwalm river basin in Flanders, Belgium. TheScientificWorldJOURNAL 2, 96–104.

Dedecker, A., Goethals, P., Gabriels, W., De Pauw, N., 2004. Optimization of Artificial Neural Network (ANN) design for prediction of macroinvertebrates in the Zwalm river basin. Ecol. Model. 174, 161–173.

Delgado, M., Vila, M., Voxman, W., 1998. A fuzziness measure for fuzzy numbers: Applications. Fuzzy Sets Syst. 94, 205–216.

Driankov, D., Hellendoorn, H., Reinfrank, M., 1993. An Introduction to Fuzzy Control. Springer, Berlin, Germany.

Dubois, D., Prade, H., 1996. What are fuzzy rules and how to use them. Fuzzy Sets Syst. 84, 169–185.

Ducheyne, E., 2003. Multi-objective forest management using GIS and genetic optimisation techniques. Ph.D. thesis, Ghent University, Gent, Belgium.

Dvořák, A., Jedelský, D., 1999. Defuzzification and chaining of rules in hierarchical fuzzy systems. In: Proceedings of the EUSFLAT-ESTYLF Joint Conference. Palma de Mallorca, Spain, pp. 199–202.

Ehrlich, P., Daily, G., 1993. Science and the management of natural resources. Ecol. Appl. 3, 558–560.

Eiben, A., Smith, J., 2003. Introduction to Evolutionary computing. Natural Computing Series. Springer Verlag, Berlin.

Elith, J., Burgman, M., Regan, H., 2002. Mapping epistemic uncertainties and vague concepts in predictions of species distribution. Ecol. Model. 157, 313–329.

Eshelman, L., Schaffer, J., 1993. Real-coded genetic algorithms and interval-schemata. In: Whitley, L. (Ed.), Proceedings of the Second Workshop on Foundations of Genetic Algorithms. Morgan Kaufmann, Vail, CO, USA, pp. 187–202.

Espinosa, J., Vandewalle, J., 2000. Constructing fuzzy models with linguistic integrity from numerical data-AFRELI algorithm. IEEE Trans. Fuzzy Syst. 8 (5), 591–600.

EU, 2000. Directive of the European Parliament and of the Council 2000/60/EC establishing a framework for community action in the field of water policy. European Union, The European Parliament, The Council, PE-CONS 3639/1/00 REV 1 EN, 62 p. + annexes.

Fielding, A., Bell, J., 1997. A review of methods for the assessment of prediction errors in conservation presence/absence models. Environ. Conserv. 24, 38–49.

Filev, D., Yager, R., 1991. A generalized defuzzification method via BAD distributions. Int. J. Intell. Syst. 6, 689–697.

Fogel, L., Owens, A., Walsh, M., 1966. Artificial Intelligence throught Simulated Evolution. John Wiley, New York, USA.

Freitas, A., 2002. Data Mining and Knowledge Discovery with Evolutionary Algorithms. Natural Computing Series. Springer, Berlin, Germany.

Goldberg, D., 1991. Real-coded genetic algorithms, virtual alphabets, and blocking. Complex Syst. 5, 139–167.

Goldberg, D. E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Boston, USA.

Grefenstette, J., 1986. Optimization of control parameters for genetic algorithms. IEEE Trans. Syst. Man Cybern. 16 (1), 122–128.

Guély, F., La, R., Siarry, P., 1999. Fuzzy rule base learning through simulated annealing. Fuzzy Sets Syst. 105, 353–363.

Guillaume, S., 2001. Designing fuzzy inference systems from data: an interpretability-oriented review. IEEE Trans. Fuzzy Syst. 9 (3), 426–443.

Guisan, A., Zimmerman, N., 2000. Predictive habitat distribution models in ecology. Ecol. Model. 135, 147–168.

Hering, D., Moog, O., Sandin, L., Verdonschot, P., 2004. Overview and application of the AQEM assessment system. Hydrobiologia 516, 1–20.

Herrera, F., Lozano, M., 2003. Fuzzy adaptive genetic algorithms: design, taxonomy, and future directions. Soft Comput. 7, 545–562.

Herrera, F., Lozano, M., Verdegay, J., 1997. Fuzzy connectives based crossover operators to model genetic algorithms population diversity. Fuzzy Sets Syst. 92, 21–30.

Herrera, F., Lozano, M., Verdegay, J., 1998. Tackling real-coded genetic algorithms: operators and tools for behavioural analysis. Artif. Intell. Rev. 12, 265–319.

Holland, J., 1975. Adaptation in Natural and Artificial Systems. MIT Press, Ann Arbor, USA.

Holling, C., Allen, C., 2002. Adaptive inference for distinguishing credible from incredible patterns in nature. Ecosystems 5, 319–328.

Illies, J., 1978. Limnofauna Europaea, 2nd Edition. Fischer, Stuttgart.

Ishigami, H., Fukuda, T., Shibata, T., Arai, F., 1995. Structure optimization of fuzzy neural network by genetic algorithm. Fuzzy Sets Syst. 71, 257–264.

Jang, J.-S., 1993. ANFIS: Adaptive-network-based fuzzy inference system. IEEE Trans. Syst. Man Cybern. 23 (3), 665–685.

Jiang, T., Li, Y., 1996. Generalized defuzzification strategies and their parameter learning procedures. IEEE Trans. Fuzzy Syst. 4 (1), 64–71.

Jin, Y., 2003. Advanced Fuzzy Systems Design and Applications. Vol. 112 of Studies in Fuzziness and Soft Computing. Springer Verlag, Heidelberg, Germany.

Jin, Y., von Seelen, W., Sendhoff, B., 1999. On generating FC^3 fuzzy rule systems from data using evolution strategies. IEEE Trans. Syst. Man Cybern. Part B-Cybern. 29 (6), 829–845.

Kampichler, C., Barthel, J., Wieland, R., 2000. Species density of foliage-dwelling spiders in field margins: a simple, fuzzy rule-based model. Ecol. Model. 129, 87–99.

Karr, C., Gentry, E., 1993. Fuzzy control of pH using genetic algorithms. IEEE Trans. Fuzzy Syst. 1, 46–53.

Karr, J., Fausch, K., Angermeier, P., Yant, P., Schlosser, L., 1986. Assessing biological integrity in running waters: a method and its rationale. Special publication 5, Illinois Natural History Survey, Champaign, Illinois, USA.

Kerle, F., Zöllner, F., Kappus, B., Marx, W., Giesecke, J., 2001. Fish habitat and vegetation modelling in floodplains with Casimir. CFR project report 13, IWS, University of Stuttgart.

Kim, H.-S., Roschke, P., 2006. Fuzzy control of base-isolation system using multi-objective genetic algorithm. Comput.-Aided Civil Infrastruct. Eng. 21, 436–449.

Kim, M.-S., Kim, C.-H., Lee, J.-J., 2005. Evolving structure and parameters of fuzzy models with interpretable membership functions. J. Intell. Fuzzy Syst. 16, 95–105.

Kinzel, J., Klawonn, F., Kruse, R., 1994. Modifications of genetic algorithms for designing and optimizing fuzzy controllers. In: Proceedings of the First IEEE Conference on Evolutionary Computation. Orlando, FL, USA, pp. 28–33.

Kompare, B., Bratko, I., Steinman, F., Dzeroski, S., 1994. Using machine learning techniques in the construction of models. I. Introduction. Ecol. Model. 75-76, 617–628.

Koo, K., Won, J., Lee, J., 2004. Least squares identification of monotonic fuzzy systems. In: Proceedings of the Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS). Vol. 2. Banff, Canada, pp. 745–749.

Koza, J., 1993. Genetic Programming. MIT Press, Cambridge, MA, USA.

Kruse, R., Gebhardt, J., Klawonn, F., 1994. Foundations of Fuzzy Systems. John Wiley, Chichester, UK.

Lau, H., Lee, C., Ip, W., Chan, F., Leung, R., 2005. Design and implementation of a process optimizer: a case study on monitoring molding operations. Expert Syst. 22, 12–21.

Lievens, S., De Baets, B., Cao-Van, K., in press. A probabilistic framework for the design of instance-based supervised ranking algorithms. Ann. Oper. Res.

Ludwig, D., Hilborn, R., Walters, C., 1993. Uncertainty, resource exploitation, and conservation: lessons from history. Science 260 (5104), 17–36.

Mackinson, S., 2000. An adaptive fuzzy expert system for predicting structure, dynamics and distribution of herring shoals. Ecol. Model. 126, 155–178.

Mamdani, E., 1974. Application of fuzzy algorithms for control of simple dynamic plant. Proc. IEE 121 (12), 1585–1588.

Manel, S., Williams, H., Ormerod, S., 2001. Evaluating presence-absence models in ecology: the need to account for prevalence. J. Appl. Ecol. 38, 921–931.

Marín-Blázquez, J. G., Shen, Q., 2002. From approximative to descriptive fuzzy classifiers. IEEE Trans. Fuzzy Syst. 10, 484–497.

Mauch, E., 1976. Leitformen der Saprobität für die biologische Gewässeranalyse. Vol. 21 of Courier Forschungsinstitut Senckenberg. E. Schweizerbart Verlag, Stuttgart, Germany, 844 p.

Meesters, E., Bak, R., Westmacott, S., Ridgley, M., Dollar, S., 1998. A fuzzy logic model to predict coral reef development under nutrient and sediment stress. Conserv. Biol. 12, 957–965.

Mencar, C., Castellano, G., Fanelli, A., 2005. Some fundamental interpretability issues in fuzzy modeling. In: Proceedings of Joint Fourth Conference of the European Society for Fuzzy Logic and Technology and the 11th Recontres Francophones sur la Logique Flou et ses Applications (EUSFLAT-LFA 2005). Barcelona, Spain, pp. 100–105.

Metcalfe, J., 1989. Biological water quality assessment of running waters based on macroinvertebrates communities: History and present status in Europe. Environ. Pollut. 60, 101–139.

Michalewicz, Z., 1996. Genetic Algorithms + Data Structures = Evolution Programs, 3rd Edition. Springer, Berlin.

Michalewicz, Z., Fogel, D., 2000. How to Solve It: Modern Heuristics. Springer Verlag, Berlin.

Miller, G., 1956. The magical number seven, plus or minus two: some limits on our capacity for processing information. Psychol. Rev. 63, 81–97.

Ministerie van de Vlaamse Gemeenschap, 2000. Handboek voor de karakterisatie van de bodems van de Vlaamse waterlopen volgens TRIADE. Ministerie van de Vlaamse Gemeenschap, Departement leefmilieu en infrastructuur, Administratie Milieu-, Land- en Waterbeheer, Brussel, Belgium, 2nd Edition, in Dutch.

Mitra, B., Scott, H., Dixon, J., McKimmey, J., 1998. Applications of fuzzy logic to the prediction of soil erosion in a large watershed. Geoderma 86, 183–209.

Moller Pillot, H., Buskens, R., 1990. Larvae of the Dutch Chironomidae. Auto-ecology and distribution. Dutch faunal announcements. Foundation European Invertebrate Survey-the Netherlands. National Natural Historical Museum, Leiden, the Netherlands, 87p., in Dutch.

Mühlenbein, H., Schlierkamp-Voosen, D., 1993. Predictive models for the breeder genetic algorithm I: continuous parameter optimization. Evol. Comput. 1, 25–49.

Nauck, D., Kruse, R., 1997. A neuro-fuzzy method to learn fuzzy classification rules from data. Fuzzy Sets Syst. 89, 377–388.

Omlin, M., Reichert, P., 1999. A comparison of techniques for the estimation of model uncertainty. Ecol. Model. 115, 45–59.

Osyczka, A., 2002. Evolutionary Algorithms for Single and Multicriteria Design Optimization. Vol. 79 of Studies in Fuzziness and Soft Computing. Physica Verlag, Heidelberg.

Paiva, R., Dourado, A., 2004. Interpretability and learning in neuro-fuzzy systems. Fuzzy Sets Syst. 147, 17–38.

Park, D., Cao, Z., Kandel, A., 1992. Investigations on the applicability of fuzzy inference. Fuzzy Sets Syst. 49, 151–169.

Parsons, M., Norris, R., 1996. The effect of habitat-specific sampling on biological assessment of water quality using a predictive model. Freshw. Biol. 36, 419–434.

Patel, A., 2004. Transformation functions for trapezoidal membership functions. International Journal of Computational Cognition 2 (3), 115–135.

Patel, A., Mohan, B., 2002. Some numerical aspects of center of area defuzzification method. Fuzzy Sets Syst. 132, 401–409.

Pedrycz, W., 1994. Why triangular membership functions. Fuzzy Sets Syst. 64 (1), 21–30.

Pedrycz, W., Gomide, F., 1998. An Introduction to Fuzzy Sets: Analysis and Design. The MIT Press, Cambridge, USA.

Perfilieva, I., Kreinovich, V., 2002. A new universal approximation result for fuzzy systems, which reflects CNF-DNF duality. Int. J. Intell. Syst. 17, 1121–1130.

Piepers, S., 2005. Detectie van subklinische mastitis bij gebruik van een melkrobot: beschrijving van een vaag model. M.Sc. thesis, Ghent University, Gent, Belgium.

Poff, N., Allan, J., 1995. Functional organization of stream fish assemblages in relation to hydrological variability. Ecology 76, 606–627.

Radcliffe, N., 1991. Equivalence class analysis of genetic algorithms. Complex Syst. 5, 183–205.

Rademaker, M., 2004. Gebruik van genetische algoritmen voor experimenteel ontwerp. M.Sc. thesis, Ghent University, Gent, Belgium.

Rechenberg, I., 1973. Evolutionstrategie: Optimierung technisher Systeme nach Prinzipien des biologischen Evolution. Fromman-Holzboog Verlag, Stuttgart, Germany.

Regan, H., Colyvan, M., Burgman, M., 2002. A taxonomy and treatment of uncertainty for ecology and conservation biology. Ecol. Appl. 12, 618–628.

RIZA, 2000. Database disposed by the Institute for Inland Water Management and Waste Water Treatment. RIZA, the Netherlands, in Dutch.

Rojas, I., Gonzalez, J., Pomares, H., Rojas, F., Fernández, F., Prieto, A., 2001. Multidimensional and multideme genetic algorithms for the construction of fuzzy systems. Int. J. Approx. Reasoning 26, 179–210.

Rosenberg, R., Resh, V. (Eds.), 1993. Freshwater Biomonitoring and Benthic Macroinvertebrates. Chapman and Hall, New York, NY, USA.

Roychowdhury, S., Pedrycz, W., 2001. A survey of defuzzification strategies. Int. J. Intell. Syst. 16, 679–695.

Roychowdhury, S., Wang, B.-H., 1996. Cooperative neighbors in defuzzification. Fuzzy Sets Syst. 78, 37–49.

Ruspini, E., 1969. A new approach to clustering. Information and Control 15, 22–32.

Rykiel, E., 1996. Testing ecological models: the meaning of validation. Ecol. Model. 90, 239–244.

Sakly, A., Benrejeb, M., 2003. Activation of trapezoidal fuzzy subsets with different inference methods. In: Bilgiç, T., De Baets, B., Kaynak, O. (Eds.), Fuzzy Sets and Systems — Proceedings of IFSA 2003. Vol. 2715 of Lecture Notes in Artificial Intelligence. Springer, pp. 450–457.

Sánchez, L., Couso, I., Corrales, J., 2001. Combining GP operators with SA search to evolve fuzzy rule based classifiers. Inf. Sci. 136, 175–191.

Schaffer, J., Caruana, R., Eshelman, L., Das, R., 1989. A study of control parameters affecting online performance of genetic algorithms for function optimization. In: Schaffer, J. (Ed.), Proceedings of the Third International Conference on Genetic Algorithms. Morgan Kaufmann, Los Altos, CA, USA, pp. 51–60.

Schneider, M., 2001. Habitat und Abflussmodellierung fur Fließgewässer mit unscharfen Berechnungsansätzen. Dissertation am Institut für Wasserbau, Heft 108, Universität Stuttgart.

Schott, B., Whalen, T., 1996. Nonmonotonicity and discretization error in fuzzy rule-based control using COA and MOM defuzzification. In: Proceedings of the IEEE International Conference on Fuzzy Systems. New Orleans, LA, USA, pp. 450–456.

Shannon, C., Weaver, W., 1963. Mathematical Theory of Communication. University of Illinois Press, Urbana, IL, USA.

Shi, Y., Mizumoto, M., 2000. A new approach of neuro-fuzzy learning algorithm for tuning fuzzy rules. Fuzzy Sets Syst. 112, 99–116.

Shu, C., Burn, D., 2004. Homogeneous pooling group delineation for flood frequency analysis using a fuzzy expert system with genetic enhancement. J. Hydrol. 291, 132–149.

Sill, J., 1998. Monotonic networks. In: Jordan, M., Kearno, M., Solla, S. (Eds.), Proceedings of the NIPS Conference 1997. Vol. 10 of Advances in Neural Information Processing Systems. The MIT Press, Denver, CO, USA, pp. 661–667.

Simon, D., 2002. Sum normal optimization of fuzzy membership functions. Int. J. Uncertainty Fuzziness Knowl.-Based Syst. 10 (4), 363–384.

Sládecek, V., Hawkes, H., Alabaster, J., Daubner, I., Nötlich, I., Solbé, J., Uhlmann, D., 1982. Biological examination. In: Suess, M. (Ed.), Examination of Water for Pollution Control. A reference handbook. Vol. 3 of Biological, Bacteriological and Virological Examination. Pergamon Press, Oxford, UK.

Song, Q., Leland, R., 1996. Adaptive learning defuzzification techniques and applications. Fuzzy Sets Syst. 81, 321–329.

Sousa, J., Kaymak, U., 2002. Fuzzy Decision Making in Modeling and Control. Vol. 27 of World Scientific Series in Robotics and Intelligent Systems. World Scientific, New Jersey, USA.

Spears, W., De Jong, K., 1991. An analysis of multi-point crossover. In: Rawlins, G. (Ed.), Proceedings of the First Workshop on Foundations of Genetic Algorithms. Morgan Kaufmann, San Mateo, CA, USA, pp. 301–315.

Statzner, B., Gore, J., Resh, V., 1988. Hydraulic stream ecology: observed patterns and potential applications. Journal of the North American Benthological Society 7, 307–360.

Steinhardt, U., 1998. Applying the fuzzy set theory for medium and small scale landscape assessment. Landsc. Urban Plan. 41, 203–208.

Sun, C.-T., 1994. Rule-base structure identification in an adaptive-network-based fuzzy inference system. IEEE Trans. Fuzzy Syst. 2 (1), 64–73.

Surmann, H., Kanstein, A., Goser, K., 1993. Self-organizing and genetic algorithms for an automatic design of fuzzy control and decision systems. In: Proceedings of the First European Congress on Fuzzy and Intelligent Technologies (EUFIT'93). Aachen, Germany, pp. 1097–1104.

Suzuki, T., Kodama, T., Furuhashi, T., Tsutsui, H., 2001. Fuzzy modeling using genetic algorithms with fuzzy entropy as conciseness measure. Inf. Sci. 136, 53–67.

Syswerda, G., 1989. Uniform crossover in genetic algorithms. In: Schaffer, J. (Ed.), Proceedings of the Third International Conference on Genetic Algorithms. Morgan Kaufmann, Los Altos, CA, USA, pp. 2–9.

Tachet, H., Richoux, P., Bournaud, M., Usseglio-Polatera, P., 2000. Invertébrés d'eau douce, systématique, biologie, écologie. CNRS Editions, Paris, France, 587 pp.

Takagi, T., Sugeno, M., 1985. Fuzzy identification of systems and its application to modeling and control. IEEE Trans. Syst. Man Cybern. Part A-Syst. Hum. 15 (1), 116–132.

Tanaka, K., Sano, M., Watanabe, H., 1995. Modeling and control of carbon monoxide concentration using a neuro-fuzzy technique. IEEE Trans. Fuzzy Syst. 3 (3), 271–279.

Usseglio-Polatera, P., 1994. Theoretical habitat templets, species traits, and species richness: aquatic insects in the Upper Rhône River and its floodplain. Freshw. Biol. 31, 417–437.

Valente de Oliveira, J., 1999. Semantic constraints for membership functions optimization. IEEE Trans. Syst. Man Cybern. Part A-Syst. Hum. 29, 128–138.

Van Broekhoven, E., Adriaenssens, V., De Baets, B., Verdonschot, P., 2006. Fuzzy rule-based macroinvertebrate habitat suitability models for running waters. Ecol. Model. 198, 71–84.

Van Leekwijck, W., Kerre, E., 1999. Defuzzification: criteria and classification. Fuzzy Sets Syst. 108 (2), 159–178.

Vaughan, I., Ormerod, S., 2005. The continuing challenges of testing species distribution models. J. Appl. Ecol. 42, 720–730.

Verdonschot, P., 1990. Ecological characterization of surface waters in the Province of Overijssel (The Netherlands). Ph.D. thesis, Landbouwuniversiteit Wageningen, Wageningen, The Netherlands.

Verdonschot, P., 2000a. Natural communities of Dutch inland waters. Part 1, Sources. Background document at the ’Handbook Natural River Types’. Alterra, Wageningen, the Netherlands, in Dutch.

Verdonschot, P., 2000b. Natural communities of Dutch inland waters. Part 2, Brooks. Background document at the ’Handbook Natural River Types’. Alterra, Wageningen, the Netherlands, in Dutch.

Vishnupad, P., Shin, Y., 1999. Adaptive tuning of fuzzy membership functions for non-linear optimization using gradient descent method. J. Intell. Fuzzy Syst. 7, 13–25.

Walley, W., Dzeroski, S., 1995. Biological monitoring: a comparison between Bayesian, neural and machine learning methods of water quality classification. In: Denzer, R., Schimak, G., Russell, D. (Eds.), Environmental Software Systems. Chapman and Hall, London, UK, pp. 229–240.

Wang, W.-J., Luoh, L., 2000. Simple computation for the defuzzification of center of sum and center of gravity. J. Intell. Fuzzy Syst. 9, 53–59.

Wiederholm, T., 1980. Use of benthos in lake monitoring. J. Water Pollut. Control Fed. 52, 537–547.

Wolpert, D., Macready, W., 1997. No free lunch theorems for optimization. IEEE Trans. Evol. Comput. 1 (1), 67–82.

Won, J., Park, S., Lee, J., 2002. Parameter conditions for monotonic Takagi-Sugeno-Kang fuzzy system. Fuzzy Sets Syst. 132, 135–146.

Wright, A., 1991. Genetic algorithms for real parameter optimization. In: Rawlins, G. (Ed.), Proceedings of the First Workshop on Foundations of Genetic Algorithms. Morgan Kaufmann, San Mateo, CA, USA, pp. 205–218.

Yager, R., Filev, D., 1993. SLIDE: a simple adaptive defuzzification method. IEEE Trans. Fuzzy Syst. 1 (1), 69–78.

Yager, R., Filev, D., 1994. Essentials of Fuzzy Modeling and Control. John Wiley, New York, USA.

Ying, H., Ding, Y., Li, S., Shao, S., 1999. Comparison of necessary conditions for typical Takagi–Sugeno and Mamdani fuzzy systems as universal approximators. IEEE Trans. Syst. Man Cybern. Part A-Syst. Hum. 29, 508–514.

Zadeh, L., 1965. Fuzzy sets. Information and Control 8, 338–353.

Summary

In linguistic fuzzy models the knowledge about the system is expressed in words, more specifically in if-then rules such as 'IF the slope is very large AND the coverage by vegetation is low THEN the expected soil loss by erosion is high'. Hence the term *linguistic* fuzzy models. They are referred to as linguistic *fuzzy* models since fuzzy sets are used to incorporate the uncertainty in the definition of the linguistic values 'very large', 'low' and 'high' of the linguistic variables 'slope', 'coverage by vegetation' and 'expected soil loss by erosion' in the model. In contrast to classical set theory where one or zero is assigned to an object (*e.g.* a real value) depending on whether the object is in or not in a set, a fuzzy set is characterized by a membership function which assigns a grade ranging between zero and one to each object to reflect the degree to which an object is 'a member' of the fuzzy set.

The components of a linguistic fuzzy model, *i.e.* the if-then rules, membership functions and mathematical operations used to obtain a model output from an input, can all be based on knowledge from an expert familiar with the system, or can — either partially or completely — be derived from data. In general, other modelling techniques allow for a higher accuracy than linguistic fuzzy models, *i.e.* other types of models return an output that resembles the output in the data set to a higher degree. Linguistic fuzzy models, however, have an interpretable model structure: a simple reading of the if-then rules gives insight in the system's behaviour and a meaning can be assigned to the fuzzy sets. This property, setting linguistic fuzzy models apart from other modelling techniques, is considered their greatest asset. Therefore, in the identification process of a linguistic fuzzy model, the interpretability of the model should be safeguarded or at least be balanced against its accuracy. A good trade-off between accuracy and interpretability can be obtained by including as much qualitative knowledge as possible, how little this may be, in the data-driven model identification process. Monotonicity is the type of qualitative knowledge that plays a central role in this dissertation. Monotone is hereby interpreted as order-preserving.

First, however, this dissertation shortly addresses a topic from the fuzzy modelling domain which is not related with monotonicity: the computational aspects of

the Center of Gravity defuzzification method, a defuzzification method which has a high computational burden. Two computational methods, the slope-based method and the modified transformation function method, were introduced to determine the crisp output of Mamdani–Assilian models using a fuzzy output partition of trapezial membership functions. The accuracy, computational cost and implementational complexity of these two methods and the commonly applied discretization method were discussed for the basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$. Its easy implementation appears to be the only advantage of the discretization method. The two other methods to compute the Center of Gravity defuzzification method are not as straightforward to implement but allow both a quicker and more accurate computation. Of the three methods presented, the modified transformation function method has the smallest computational cost while being as accurate as the slope-based method.

In the ecological case study described in the second part of this dissertation habitat suitability models were developed. Fuzzy ordered classifiers were applied to a modelling problem concerning the habitat suitability of river sites along springs to small rivers in the Central and Western Plains of Europe for 86 macroinvertebrate species. For each species, four models were developed, an A-, N-, P-, and C-model. The fuzzy classifiers take a certain width, velocity and either ammonium (A), nitrate (N) or phosphate (P) concentration or electrical conductivity (C) as input and return four values between 0 and 1 as output, indicating the degree to which the river site is considered 'not suitable' respectively 'lowly', 'moderately' and 'highly suitable' for the species to establish a population. Ordered linguistic values were assigned to both input and output variables, but the output variable, *i.e.* the habitat suitability, was not necessarily monotone in the input variables. The models were built using expert knowledge and evaluated on the EKOO data set collected in the Province of Overijssel in the Netherlands. The data allowed for an objective evaluation of the four developed models for 12 species. The fact that among them only one is an indicator for reference conditions, indicates that given the present environmental conditions of rivers in EU Member States, shifts in abundance levels of more common species are more suitable to detect gradual changes in water quality. With an improving water quality, the follow-up of indicator species with more narrow niches will gain importance. Of these 48 objectively evaluated models, 16 models turned out to have a good model performance expressed by the performance measure % CFCI. These 16 good performing and objectively evaluated models are all, except one model, N- or P-models. For the 48 models for which the EKOO data set allowed for an objective evaluation, an interpretability-preserving as well as an accuracy-oriented genetic optimization of the membership functions in the input domains, applying once binary-coded and once real-coded genetic algorithms, was carried out. As fitness function, a new performance measure for fuzzy ordered classifiers was applied, referred to as the average deviation (AD) as it takes the order of the output classes into account by returning the average deviation between the position of the class obtained with the model and the position of the class stored in the data set. A purely accuracy-oriented optimization showed to be no option when one wants to preserve the interpretability of the habitat suitability models under study with the EKOO data set. In this case, expert knowledge is a prerequisite to build interpretable models in order to define the rule bases and determine the optimization intervals of the membership function parameters. Furthermore, the optimization results stress the importance of uniformly distributed and unambiguous training data for model optimization.

The third, more methodological issue discussed in this dissertation is the monotonicity of linguistic fuzzy models. In monotone models, ordered linguistic values are assigned to both input and output variables and the model output is monotone in all input variables. Models were assumed to apply a fuzzy partition of trapezial membership functions in all input domains as well as in the output domain, which imposes a natural order on the linguistic values of all variables, and to have a monotone rule base, *i.e.* to use a set of if-then rules describing a monotone relation between the input variables and the output variable. The monotonicity of linguistic fuzzy models under different inference procedures was discussed: two existing inference procedures, Mamdani–Assilian inference and *plain* implicator-based inference, and a new inference procedure, ATL–ATM inference. Mamdani–Assilian models applying one of the three basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ and $T_{\mathbf{L}}$ combined with either the Center of Gravity or the Mean of Maxima defuzzification method were considered. Furthermore, models applying plain implicator-based inference or ATL–ATM inference, one of the three basic t-norms $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ or $T_{\mathbf{L}}$, one of the three R-implicators $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ and the Mean of Maxima defuzzification method, were studied. The objective of this study was to select, for each inference procedure, combinations of t-norm, implicator and defuzzification method resulting in a monotone input-output behaviour for any monotone rule base, or at least for any monotone smooth rule base. A rule base is called smooth if every set of two rules differing in only one input variable in their antecedent and containing adjacent values for this variable, have equal or adjacent values in their consequent.

For the assumed model properties, the input-output behaviour of models with m input variables reduces to the input-output behaviour of models with m^* ($m^* < m$) input variables in those regions of the input space where the inputs belong to the kernel of the same linguistic value in all but m^* input domains. Thus, if certain model properties are necessary to guarantee monotonicity for models with m^* input variables, these model properties are also required to guarantee a monotone input-output behaviour for models with more than m^* input variables. Furthermore, an auxiliary interpolation procedure was presented which allows for the extension of results obtained for models for which all linguistic output values in the rule consequents are defined by trapezial membership functions of identical shape to models with any fuzzy output partition of trapezial or triangular membership functions.

For a model with two input variables and a monotone rule base monotonicity cannot be guaranteed for the considered combinations of inference procedures, t-norms, implicators and defuzzification methods, except for Mamdani–Assilian inference combined with the t-norm $T_{\mathbf{P}}$ and the Mean of Maxima defuzzification method if, at least, the model satisfies additional constraints. For Mamdani–Assilian models with two input variables and any monotone rule base applying the Mean of Maxima defuzzification method, a monotone input-output behaviour can be guaranteed when using a fuzzy output partition corresponding to one of the following schemata { *, triangular, triangular, triangular, * }, { *, triangular, triangular, * } or { *, *, * } with * a membership function that might be either triangular or trapezial. When a system with two input variables is described by a monotone smooth rule base a wider range of inference procedures can be applied: Mamdani–Assilian inference with the t-norm $T_{\mathbf{P}}$ and the Center of Gravity or Mean of Maxima defuzzification method, Mamdani–Assilian inference with the t-norm $T_{\mathbf{M}}$ and the Mean of Maxima defuzzification method, ATL–ATM inference with the t-norm $T_{\mathbf{P}}$, the implicator $I_{\mathbf{L}}$ and the Mean of Maxima defuzzification method or ATL–ATM inference with the t-norm $T_{\mathbf{L}}$, the implicator $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ or $I_{\mathbf{L}}$ and the Mean of Maxima defuzzification method. The monotonicity of ATL–ATM models with three or more input variables was not studied in this dissertation. For Mamdani–Assilian models applying the Center of Gravity defuzzification method, models with up to three input variables were investigated. It was proved that with the auxiliary interpolation procedure, a monotone input-output behaviour is always obtained for Mamdani–Assilian models with three input variables and a monotone smooth rule base applying the t-norm $T_{\mathbf{P}}$ and the Center of Gravity defuzzification method. Furthermore, for Mamdani–Assilian models applying the Mean of Maxima defuzzification method, it was shown that when applying the auxiliary interpolation procedure, monotonicity can be guaranteed for models with a monotone smooth rule base applying $T_{\mathbf{M}}$ or $T_{\mathbf{P}}$ and any fuzzy output partition.

Samenvatting

In linguïstische vage modellen wordt de kennis over het gemodelleerde systeem in woorden uitgedrukt, meer bepaald in als-dan regels zoals ‘ALS de helling heel groot is EN de bedekking door vegetatie laag is DAN is het verwachte verlies aan bodem hoog’. Vandaar de term *linguïstische* vage modellen. Ze worden linguïstische *vage* modellen genoemd daar vage verzamelingen gebruikt worden om de onzekerheid in de definitie van de linguïstische waarden ‘zeer hoog’, ‘laag’ en ‘hoog’ van de linguïstische variabelen ‘helling’, ‘bedekking door vegetatie’ en ‘verwacht verlies aan bodem’ te incorporeren in het model. In tegenstelling tot de klassieke verzamelingenleer waar een of nul wordt toegekend aan een object (bv. een reëel getal) afhankelijk of het tot de verzameling behoort of er niet toe behoort, wordt een vage verzameling gekarakteriseerd door een lidmaatschapsfunctie die een graad tussen nul en een toekent aan een object afhankelijk van de mate waarin het object ‘lid’ is van de vage verzameling.

De componenten van een linguïstisch vaag model, d.w.z. de als-dan regels, lidmaatschapsfuncties en wiskundige bewerkingen waarmee voor een modelingang een corresponderende modeluitgang bekomen wordt, kunnen allemaal gebaseerd zijn op kennis van een expert die vertrouwd is met het systeem, of kunnen — gedeeltelijk of volledig — afgeleid worden uit data. Meestal zal met een andere modelleringstechniek een nauwkeuriger model kunnen bekomen worden dan met linguïstische vage modellering, d.w.z. een ander soort model kan modeluitgangen opleveren die de uitgangen in de data set beter benaderen. Linguïstische vage modellen hebben echter een interpreteerbare modelstructuur: het eenvoudigweg lezen van de als-dan regels verschaft inzicht in het gedrag van het systeem en er kan een betekenis toegekend worden aan de vage verzamelingen. Deze eigenschap onderscheidt linguïstisch vage modellen van andere modelleringstechnieken en vormt hun grootste troef. Vandaar dat bij de identificatie van een linguïstisch vaag model, de interpreteerbaarheid van het model dient gevrijwaard te worden of op zijn minst dient afgewogen te worden tegen de nauwkeurigheid van het model. Een goed evenwicht tussen nauwkeurigheid en interpreteerbaarheid kan bereikt worden door zoveel mogelijk kwalitatieve kennis, hoe weinig dit ook mag zijn, te incorporeren in het data-gedreven optimalisatieproces. Monotoniteit is het soort kwalitatieve kennis dat een centrale rol speelt in dit proefschrift. Monotoon wordt hierbij geïnterpreteerd als rangorde-bewarend.

In het eerste deel van dit proefschrift wordt een onderwerp uit het domein van de linguïstische vage modellering behandeld dat geen verband houdt met monotoniteit: de rekenkundige aspecten van de zwaartepuntontvagingsmethode, een ontvagingsmethode met een hoge rekenkundige last. Twee berekeningsmethoden, de 'helling-gebaseerde' methode en de 'aangepaste transformatiefunctie' methode werden geïntroduceerd om de scherpe uitgang te bepalen van Mamdani–Assilian modellen die gebruik maken van een vage uitgangspartitie van trapeziumvormige lidmaatschapsfuncties. De nauwkeurigheid, rekenkundige last en complexiteit van de implementatie van deze twee methoden en de gebruikelijke discretisatiemethode werden besproken voor de drie meest toegepaste driehoeksnormen $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ en $T_{\mathbf{L}}$. Haar eenvoudige implementatie blijkt het enige voordeel te zijn van de discretisatiemethode. De twee andere methoden voor de zwaartepuntsontvaging zijn niet zo eenvoudig te implementeren maar resulteren beide in een snellere en meer nauwkeurige berekening. De 'aangepaste transformatiefunctie' methode heeft de kleinste rekenlast van de drie beschouwde methoden terwijl ze zo nauwkeurig is als de 'helling-gebaseerde' methode.

In de ecologische casestudy, beschreven in het tweede deel van dit proefschrift, werden habitatgeschiktheidsmodellen ontwikkeld. Vage geordende klassificatie werd toegepast op een modelleringsprobleem over de habitatgeschiktheid van rivierlokaties langs bronbeken tot kleine rivieren in de centrale en westelijke vlakten van Europa voor 86 macro-invertebratenspecies. Voor elk species werden vier modellen ontwikkeld, een A-, N-, P- en C-model. De vage klassificaties hebben een bepaalde breedte, snelheid en hetzij ammonium- (A), nitraat- (N) of fosfaat- (P) concentratie of elektrisch geleidingsvermogen (C) als ingang en kennen vier waarden tussen 0 en 1 toe als uitgang die aangeven in welke mate een rivierlokatie verondersteld wordt 'niet geschikt', respectievelijk 'laag', 'matig' en 'uitermate geschikt' te zijn voor het species om een populatie te ontwikkelen. Geordende linguïstische waarden werden toegekend aan zowel ingangs- als uitgangsvariabelen, maar de uitgangsvariabele, nl. de habitatgeschiktheid, is niet noodzakelijk monotoon in de ingangsvariabelen. De modellen werden ontwikkeld op basis van expertkennis en geëvalueerd op de EKOO data set verzameld in de provincie Overijssel in Nederland. De data lieten een objectieve evaluatie toe van de vier ontwikkelde modellen voor 12 species. Het feit dat onder hen slechts één indicatorspecies is voor referentieomstandigheden, geeft aan dat, gezien de huidige milieukwaliteit van rivieren in lidstaten van de Europese Unie, verschuivingen in de abundantie van meer algemene species geschikter zijn om geleidelijke veranderingen van de waterkwaliteit te detecteren. Bij een verbeterde waterkwaliteit zal de opvolging van indicatorspecies met een beperktere niche aan belang winnen. Van de 48 objectief geëvalueerde modellen, bleken 16 modellen goed te presteren op basis van de performantiemaat % CFCI. Deze 16 goed presenterende en objectief geëvalueerde modellen zijn alle, behalve een model, N- of P-modellen. Voor de 48 modellen waarvoor de EKOO data set een objectieve evaluatie toeliet, werd een interpreteerbaarheidsbewarende alsook een nauwkeurigheidsgeoriënteerde genetisch optimalisatie van de lidmaatschapsfuncties in de ingangsdomeinen uitgevoerd, waarbij eens binairgecodeerde en eens reëelgecodeerde genetische algoritmen werden toegepast. Als fitnessfunctie werd een nieuwe performatiemaat voor vage geordende klassificatie toegepast, average deviation (gemiddelde afwijking) genoemd daar het de rangorde van de uitgangsklassen in rekening brengt door de gemiddelde afwijking te geven tussen de rang van de klasse bekomen met het model en de rang van de klasse opgeslagen in de data set. De resultaten tonen aan dat een volledig nauwkeurigheidsgeoriënteerde optimalisatie niet kan toegepast worden wanneer men de interpreteerbaarheid van de bestudeerde habitatgeschiktheidsmodellen wil bewaren met de EKOO data set. Wil men in dit geval interpreteerbare modellen bekomen, dan is expertkennis vereist om enerzijds de regelbanken op te stellen en anderzijds de optimalisatieintervallen van de parameters van de lidmaatschapsfuncties te definiëren. Verder benadrukken de resultaten ook het belang van uniform verdeelde en ondubbelzinnige trainingsdata voor modeloptimalisatie.

Het derde, meer methodologisch deel van dit proefschrift handelt over de monotoniteit van linguïstische vage modellen. In monotone modellen worden geordende linguïstische waarden toegekend aan zowel ingangsvariabelen als aan de uitgangsvariabele en is de modeluitgang monotoon in alle ingangsvariabelen. Modellen werden verondersteld, ten eerste, gebruik te maken van een vage partitie van trapeziumvormige lidmaatschapsfuncties in alle ingangsdomeinen en in het uitgangsdomein, waardoor een natuurlijke rangorde wordt opgelegd aan de linguïstische waarden van alle variabelen, en ten tweede, over een monotone regelbank te beschikken, d.w.z. een set van als-dan regels te gebruiken die een monotone relatie beschrijven tussen de ingangsvariabelen en de uitgangsvariabele. De monotoniteit van linguïstische vage modellen werd onderzocht voor verschillende inferentieprocedures: twee bestaande inferentieprocedures, Mamdani–Assilian inferentie en *gewone* implicator-gebaseerde inferentie, en een nieuwe inferentie procedure, ATL–ATM inferentie. Mamdani–Assilian modellen die gebruik maken van een van de drie meest toegepaste driehoeksnormen $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ en $T_{\mathbf{L}}$ gecombineerd met ofwel de zwaartepunt- ofwel de ‘gemiddelde-van-de-maxima’-ontvagingsmethode werden beschouwd. Verder werden modellen bestudeerd die *gewone* implicator-gebaseerde inferentie of ATL–ATM inferentie, de driehoeksnorm $T_{\mathbf{M}}$, $T_{\mathbf{P}}$ of $T_{\mathbf{L}}$, de R-implicator $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ of $I_{\mathbf{L}}$ en de ‘gemiddelde-van-de-maxima’-ontvagingsmethode toepassen. Het doel van de studie was het selecteren, voor elke inferentie procedure, van combinaties van driehoeksnorm, implicator en ontvagingsmethode waarvoor voor elke monotone regelbank of op zijn minst voor elke monotone gladde regelbank, een monotoon ingangs-uitgangsgedrag bekomen wordt. Een regelbank wordt glad genoemd indien elk paar van twee regels die slechts in één ingangsvariabele verschillen en aangrenzende waarden bevatten voor deze variabele, gelijke of aangrenzende waarden bevatten in hun consequent.

Voor modellen met de vooropgestelde eigenschappen, herleidt het ingangs-uitgangsgedrag van modellen met m ingangsvariabelen zich tot het ingangs-uitgangsgedrag van modellen met m^* ($m^* < m$) ingangsvariabelen in de delen van de ingangsruimte waar $m - m^*$ reële ingangswaarden tot de kern van dezelfde linguïstische waarden behoren. Dus, als bepaalde modeleigenschappen noodzakelijk zijn om de monotoniteit van modellen met m^* ingangsvariabelen te garanderen, dan zijn deze modeleigenschappen ook vereist om een monotoon ingangs-uitgangsgedrag te waarborgen voor modellen met meer dan m^* ingangsvariabelen. Er wordt in dit proefschrift ook een interpolatieprocedure beschreven die toelaat de resultaten bekomen voor modellen waarbij alle linguïstische uitgangswaarden in de consequenten van de regels gedefinieerd zijn door gelijkvormige trapeziumvormige lidmaatschapsfuncties uit te breiden tot modellen met om het even welke vage uitgangspartitie van driehoekige of trapeziumvormige lidmaatschapsfuncties.

Voor een model met twee ingangsvariabelen en een monotone regelbank is monotoniteit niet gegarandeerd voor de beschouwde combinaties van inferentieprocedures, driehoeksnormen, implicatoren en ontvagingsmethoden, uitgezonderd voor Mamdani–Assilian inferentie gecombineerd met de driehoeksnorm $T_{\mathbf{P}}$ en de 'gemiddelde-van-de-maxima'-ontvagingsmethode als het model aan bijkomende voorwaarden voldoet. Voor Mamdani–Assilian modellen met twee ingangsvariabelen en om het even welke monotone regelbank die de 'gemiddelde-van-de-maxima'-ontvagingsmethode toepast, is een monotoon ingangs-uitgangsgedrag gegarandeerd wanneer het model gebruik maakt van een vage uitgangspartitie die overeenkomt met een van volgende schema's { *, driehoekig, driehoekig, driehoekig, * }, { *, driehoekig, driehoekig, * } or { *, *, * } met * een driehoekige of trapeziumvormige lidmaatschapsfunctie. Wanneer een systeem van twee ingangsvariabelen wordt beschreven aan de hand van een monotone gladde regelbank kan, eventueel door gebruik te maken van de interpolatieprocedure, een breder scala aan inferentieprocedures worden toegepast: Mamdani–Assilian inferentie met de driehoeksnorm $T_{\mathbf{P}}$ en de zwaartepunt- of 'gemiddelde-van-de-maxima'-ontvagingsmethode, Mamdani–Assilian inferentie met de driehoeksnorm $T_{\mathbf{M}}$ en de 'gemiddelde-van-de-maxima'-ontvagingsmethode, ATL–ATM inferentie met de driehoeksnorm $T_{\mathbf{P}}$, de implicator $I_{\mathbf{L}}$ en de 'gemiddelde-van-de-maxima'-ontvagingsmethode of ATL–ATM inferentie met de driehoeksnorm $T_{\mathbf{L}}$, de implicator $I_{\mathbf{M}}$, $I_{\mathbf{P}}$ of $I_{\mathbf{L}}$ en de 'gemiddelde-van-de-maxima'-ontvagingsmethode. De monotoniteit van ATL–ATM modellen met drie of meer ingangsvariabelen werd niet bestudeerd in dit proefschrift. Voor Mamdani–Assilian modellen die de zwaartepuntontvagingsmethode toepassen, werden modellen met tot drie ingangsvariabelen bestudeerd. Er werd aangetoond dat met de interpolatieprocedure altijd een monotoon ingangs-uitgangsgedrag bekomen wordt voor Mamdani–Assilian modellen met drie ingangsvariabelen en een monotone gladde regelbank die de driehoeksnorm $T_{\mathbf{P}}$ en de zwaartepuntontvagingsmethode toepassen. Verder werd voor Mamdani–Assilian modellen die de 'gemiddelde-van-de-maxima'-ontvagingsmethode toepassen, aangetoond dat wanneer gebruikt gemaakt wordt van de interpolatieprocedure, monotoniteit gewaarborgd is voor modellen met een monotone gladde regelbank die gebruik maken van $T_{\mathbf{M}}$ of $T_{\mathbf{P}}$ en om het even welke vage uitgangspartitie.

Curriculum vitae

Personalia

Naam	Ester Van Broekhoven
Geslacht	vrouwelijk
Nationaliteit	Belg
Geboortedatum	15/11/1979
Geboorteplaats	Gent, België
Burgelijke staat	ongehuwd
e-mail adres	ester.vanbroekhoven@ugent.be

Opleiding

okt 02 – dec 05	Doctoraatsopleiding in de Toegepaste Biologische Wetenschappen, Universiteit Gent
okt 97 – jul 02	Bio-ingenieur in de Milieutechnologie, Universiteit Gent *Scriptie*: Toepasbaarheid van vage technieken bij de procesvoering van destillatie-eenheden. Promotor: Prof. Dr. B. De Baets

Loopbaanoverzicht-werkervaring

okt 06 – dec 06	Wetenschappelijk medewerker, Universiteit Gent
okt 02 – sep 06	Doctoraatsbursaal Bijzonder Onderzoeksfonds Universiteit Gent

Buitenlandse ervaring tijdens opleiding en werk

2005	Studieverblijf 19/06-25/06, Software Competence Center Hagenberg, Hagenberg, Oostenrijk (Prof. Dr. U. Bodenhofer)
2004	Studieverblijf 2/11-31/11, Software Competence Center Hagenberg, Hagenberg, Oostenrijk (Prof. Dr. U. Bodenhofer)
2000-2001	Studieverblijf van een semester in het kader van het Erasmus programma aan de Fachhochschule für Technik und Gestaltung, Mannheim, Duitsland

Onderwijsactiviteiten

- Practica Computationele technieken (2003)
- Begeleiding van Sofie Piepers bij haar thesis 'Detectie van subklinische mastitis bij gebruik van een melkrobot: beschrijving van een vaag model' ter behaling van het diploma van dierenarts (2004-2005).

Conferenties, studiedagen en symposia

2003	• International Fuzzy Systems Association World Congress IFSA 2003, 29 juni–2 juli, Istanbul, Turkije • Studiedag 'Datamining met toepassingen in de bioinformatica', Genootschap Informatietechnologie en Toegepaste Wiskunde van het Technologisch Instituut, 16 oktober, Brussel, België • Symposium on Large Data Sets, Statistical Software Section of the Dutch Society for Statistics and Operations Research, 6 november, Amsterdam, Nederland
2004	• Annual Machine Learning Conference of Belgium and The Netherlands (BENELEARN), 8–9 januari, Brussel, België • Symposium 'Life, a Nobel Story', Sectie Biotechnologie van de Koninklijke Vlaamse Chemische Vereniging (KVCV), 28 april, Brussel, België • Symposium on Data Mining, Center for Statistics of Ghent University, 10–11 mei, Gent, België • Machine Learning for Computational Biology, Vlaams Interuniversitair Instituut voor Biotechnologie, 19 mei, Gent, België • IEEE International Joint Conference on Neural Networks and Fuzzy Systems (FUZZ-IEEE), 25–29 juli, Budapest, Hongarije
2005	• First International Workshop on Genetic Fuzzy Systems, 17–19 maart, Granada, Spanje • Joint Fourth Conference of the European Society for Fuzzy Logic and Technology and the 11th Recontres Francophones sur la Logique Flou et ses Applications (EUSFLAT-LFA 2005), 7–9 september, Barcelona, Spanje • Symposium 'Computational intelligence in water and environment', UNESCO-IHE Institute for Water Education, 15 december, Delft, Nederland
2006	• Annual Machine Learning Conference of Belgium and the Netherlands (BENELEARN), 11–12 mei, Gent, België • The 2006 Conference of the North American Fuzzy Information Processing Society (NAFIPS 2006), 3–6 juni, Montréal, Canada

Artikels in tijdschriften die opgenomen zijn in Web of Science

- E. Van Broekhoven, V. Adriaenssens, B. De Baets, and P.F.M. Verdonschot. Fuzzy rule-based macroinvertebrate habitat suitability models for running waters. *Ecological Modelling*, 198:71–84, 2006.

- E. Van Broekhoven and B. De Baets. Fast and accurate Center of Gravity defuzzification of fuzzy system outputs defined on trapezoidal fuzzy partitions. *Fuzzy Sets and Systems*, 157:904–918, 2006.

- E. Van Broekhoven, V. Adriaenssens and B. De Baets. Interpretability-preserving genetic optimization of linguistic terms in fuzzy models for fuzzy ordered classification: an ecological case study. *International Journal of Approximate Reasoning*, 44:65–90, 2007.

- V. Adriaenssens, E. Van Broekhoven, P.F.M. Verdonschot, B. De Baets, and N. De Pauw. Knowledge and rule base development for macroinvertebrate habitat suitability modelling in river management. (ingestuurd)

- E. Van Broekhoven and B. De Baets. Monotone Mamdani–Assilian models under Mean of Maxima defuzzification. (ingestuurd)

- E. Van Broekhoven and B. De Baets. Only smooth rule bases can generate monotone Mamdani–Assilian models under Center of Gravity defuzzification. (in voorbereiding)

- E. Van Broekhoven, B. De Baets and U. Bodenhofer. Monotone fuzzy models based on ATL and ATM rule bases. (in voorbereiding)

Congresproceedings

- E. Van Broekhoven, E. Volcke, O. Georgieva, and B. De Baets. A Mamdani-Assilian controller for the last phase of an operation switch on a distillation unit. In T. Bilgiç, B. De Baets, and O. Kaynak, editors, In *Proceedings of the 10th IFSA World Congress*, pp. 313-317, Istanbul, Turkije, 2003.

- E. Van Broekhoven and B. De Baets. A comparison of three methods for computing the center of gravity defuzzification. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1537-1542, Budapest, Hongarije, 2004.

- E. Van Broekhoven, V. Adriaenssens, and B. De Baets. Evaluation and interpretability-preserving optimization of a fuzzy ordered classifier. In *Proceedings of the First International Workshop on Genetic Fuzzy Systems*, pp. 130-135, Granada, Spanje, 2005.

- E. Van Broekhoven and B. De Baets. A linguistic fuzzy model with a monotone rule base is not always monotone. In *Proceedings of the Joint Fourth Conference of the European Society for Fuzzy Logic and Technology and the 11th Rencontres Francophones sur la Logique Flou et ses Applications (EUSFLAT-LFA 2005)*, pp. 530-535, Barcelona, Spanje, 2005.

- E. Van Broekhoven and B. De Baets. On the monotonicity of Mamdani-Assilian models with two input variables. In *Proceedings of the North American Fuzzy Information Processing Society Conference (NAFIPS 2006)*, Montréal, Canada, 2006, 6 pagina's, CD.