

A framework for a multi-agent planning support system : principles and illustrations

Citation for published version (APA):

Saarloos, D. J. M. (2006). *A framework for a multi-agent planning support system : principles and illustrations*. [Phd Thesis 1 (Research TU/e / Graduation TU/e), Built Environment]. Technische Universiteit Eindhoven. <https://doi.org/10.6100/IR607386>

DOI:

[10.6100/IR607386](https://doi.org/10.6100/IR607386)

Document status and date:

Published: 01/01/2006

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

A Framework for a Multi-Agent Planning Support System

Principles and Illustrations

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
Rector Magnificus, prof.dr.ir. C.J. van Duijn, voor een
commissie aangewezen door het College voor
Promoties in het openbaar te verdedigen
op woensdag 15 maart 2006 om 16.00 uur

door

Dirk Johannes Maria Saarloos

geboren te Rotterdam

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr. H.J.P. Timmermans
en
prof.dr. M. Batty

Copromotor:
dr. T.A. Arentze

Copyright © 2006 D.J.M. Saarloos
Technische Universiteit Eindhoven
Faculteit Bouwkunde, Capaciteitsgroep Stedebouw

Printed by the Eindhoven University Press Facilities

Cover design: Ton van Gennip, Tekenstudio Faculteit Bouwkunde
Cover photograph: Historic work of art at Feyenoord Stadium, Rotterdam

CIP-DATA KONINKLIJKE BIBLIOTHEEK, DEN HAAG
ISBN 90-6814-590-8

PREFACE

With writing the preface of this book, the research that I conducted over the last several years has reached its final stage. As part of a research program aimed at developing a system named *MASQUE*, i.e., *Multi-Agent System for supporting the Quest for Urban Excellence*, my focus has been on defining a generally applicable system framework that exploits the opportunities for planning support provided by multi-agent technology. These opportunities have been further detailed with the development and demonstration of a prototype multi-agent application for generating alternative plans, a key aspect in planning for which decision support has always been lagging behind in comparison to forecasting or evaluation. Hopefully, the proposals made in this book are sufficiently cogent to persist as structuring principles and to guide the subsequent efforts needed to accomplish the intended comprehensive system.

Obviously, the work reported in this book would not have been possible without the help and support of others. First of all, I owe many thanks to my first promotor Professor Harry Timmermans for providing me the chance to start up an academic career under his first-class and pleasant guidance. It has been a true honour to work with such a great scientist and personality. I also wish to express my gratitude to my second promotor Professor Mike Batty, one of the most prominent researchers in the field of planning support, for carefully reviewing the entire manuscript. His comments and suggestions have been highly valued.

Throughout the project I have received excellent supervision from my co-promotor Theo Arentze and Aloys Borgers. Together with Harry, they have formed a team of great support and guidance. With their undisputable knowledge, expertise and patience they have been able to help me find answers and solutions through interesting discussions. My appreciation for all their efforts made and time spend goes beyond description. In addition, I would like to thank Joran Jessurun for helping me to get started with Java programming and for setting up the initial structure of the prototype application.

The pleasant working environment of the Urban Planning Group has been of great importance to me. The relaxed and cheerful atmosphere made it a privilege to be part of the group. Therefore, I would like to say thank you to everyone I have worked with, in particular to Mandy van de Sande – van Kasteren for her secretarial support and caring personality, Leo van Veghel for the enjoyable discussions about football and Peter van

der Waerden for his mental support and wonderful sense of humour. A special word of thanks goes to Qi Han, who has been my roommate for the last few years, for the interesting talks we had about research, different cultures and everyday life, her valuable suggestions and her encouragement in times of need.

Finally, I would like to thank my mother and sister for their support and endless patience with me. Although at times it did seem hard for them to understand what I was actually busy with for so long, they have always been interested in hearing about my research experiences. Now that this book is printed, I know they feel very relieved and happy. This means a lot to me...

Dick Saarloos

Eindhoven, January 2006

TABLE OF CONTENTS

PREFACE

LIST OF FIGURES

LIST OF TABLES

1	INTRODUCTION.....	1
1.1	Urban planning and the need for decision support	1
1.1.1	Instruments.....	1
1.1.2	Process	2
1.1.3	Complications	4
1.2	The supply of decision support tools	6
1.3	Aim and objectives	7
1.4	Outline of the thesis	8

PART I: INVENTORY

2	PLANNING SUPPORT SYSTEMS: TOWARDS AN IDEAL	13
2.1	Introduction.....	13
2.2	Requirements	16
2.2.1	System usage.....	16
2.2.2	System capabilities.....	18
2.3	Developments	20
2.3.1	Information	21
2.3.2	Models.....	24
2.3.3	Visualization	27
2.4	Intelligent planning support	29
2.5	Conclusions and discussion	33
3	MULTI-AGENT TECHNOLOGY AND PLANNING SUPPORT.....	37
3.1	Introduction.....	37
3.1.1	Agents	38
3.1.2	Multi-agent systems	40

3.2	Multi-agent applications.....	41
3.2.1	Agents for user assistance	42
3.2.2	Agents for micro-simulation	43
3.2.3	Agents for software design.....	44
3.3	Multi-agents and planning support.....	45
3.3.1	Opportunities	45
3.3.2	Multi-agents and the urban system.....	48
3.4	Conclusions and discussion.....	50

PART II: PRINCIPLES

4 FRAMEWORK FOR A MULTI-AGENT PLANNING SUPPORT SYSTEM.55

4.1	Introduction	55
4.2	<i>MASQUE</i> system architecture	56
4.3	Agent typology	59
4.3.1	Agents for human-computer interaction	60
4.3.2	Agents for model use and management	61
4.3.3	Agents for use of specialized knowledge.....	63
4.4	<i>MASQUE</i> knowledge component	66
4.5	Conclusions	68

5 A MULTI-AGENT MODEL FOR ALTERNATIVE PLAN GENERATION...69

5.1	Introduction	69
5.2	Context	69
5.2.1	Organization	70
5.2.2	Assumptions	70
5.3	Agent specification.....	72
5.3.1	Beliefs.....	77
5.3.1.1	Goal state belief.....	77
5.3.1.2	Future state belief	81
5.3.2	Strategic actions	84
5.3.2.1	Claims.....	85
5.3.2.2	Assignments	86
5.3.2.3	Assessment of assignments	88
5.3.3	Selection of alternative plans	89
5.4	Interaction protocols.....	90

5.4.1	Simultaneous contributions.....	91
5.4.2	Sequential contributions.....	94
5.5	Conclusions.....	96

PART III: ILLUSTRATIONS

6 DESIGN AND IMPLEMENTATION OF *MASQUE* ALTERNATIVE PLAN GENERATOR..... 99

6.1	Introduction.....	99
6.2	Design	100
6.2.1	Environment.....	101
6.2.2	Agent organization.....	102
6.2.3	Mental constructs	103
6.3	Implementation	105
6.3.1	File input	106
6.3.2	Behaviour of agents	107
6.3.3	Decision-making of agents	110
6.4	Running the application	111
6.5	Conclusions.....	115

7 TESTING THE MODEL: SENSITIVITY ANALYSIS..... 117

7.1	Introduction.....	117
7.2	Benchmark test case.....	118
7.2.1	Environment and team	118
7.2.2	Agent behaviour.....	119
7.2.3	Agent decision-making	121
7.2.4	Benchmark output.....	122
7.3	Analysis of case characteristics	124
7.3.1	Surroundings	124
7.3.2	Plan program.....	126
7.3.3	Resolution	128
7.4	Analysis of decision models	133
7.4.1	Accessibility-based utility.....	133
7.4.2	Adjacency-based utility	135
7.4.3	Population-based utility	136
7.5	Analysis of claiming behaviour	138

7.5.1	Clustering	139
7.5.2	Claiming strategy	141
7.5.2.1	Opportunism.....	142
7.5.2.2	Accentuation.....	143
7.5.3	Scaling policy	146
7.6	Analysis of assigning behaviour	148
7.6.1	Initial weight sets	148
7.6.2	Weight adjustment.....	150
7.6.3	Priority policy.....	151
7.7	Conclusions	152
8	CONCLUSIONS AND DISCUSSION	159
8.1	Introduction	159
8.2	Conclusions from part I.....	159
8.3	Conclusions from part II	161
8.4	Conclusions from part III	163
8.5	Discussion	164
8.6	Possible directions of future research.....	165
	BIBLIOGRAPHY	167
	AUTHOR INDEX.....	177
	SUBJECT INDEX.....	181
	SAMENVATTING (Dutch summary).....	185
	CURRICULUM VITAE	193

LIST OF FIGURES

Figure 3-1.	The mindset of a multi-agent system	41
Figure 4-1.	System architecture: (a) PSS in general and (b) <i>MASQUE</i>	57
Figure 4-2.	<i>MASQUE</i> : plan development process	58
Figure 4-3.	<i>MASQUE</i> : knowledge component	67
Figure 5-1.	Cooperation of <i>domain managers</i> in the plan development process	71
Figure 5-2.	Utility function to determine the expected utility of a cell based on accessibility	74
Figure 5-3.	Utility function to determine the expected utility of a cell based on adjacency	75
Figure 5-4.	Utility function to determine the expected utility of a cell based on population	76
Figure 5-5.	Model building blocks: components for reasoning and interaction	76
Figure 5-6.	Sensitivity of parameters α and β that define the S-shaped curve for converting expected utility of cells into preliminary probability	79
Figure 5-7.	Interaction protocol: simultaneous contributions	92
Figure 5-8.	Interaction protocol: sequential contributions	95
Figure 6-1.	Static structure of the environment	101
Figure 6-2.	Static structure of the agent organization	103
Figure 6-3.	Static structure of the mental constructs	104
Figure 6-4.	<i>MASQUE</i> Alternative Plan Generator: GUI components	105
Figure 6-5.	Program control unit: opening a project input file (example)	106
Figure 6-6.	Format of a project input file (example)	107
Figure 6-7.	Setting the behaviour of agents (incl. default values)	108
Figure 6-8.	Setting the decision model of the services agent (incl. default values)	111
Figure 6-9.	Program control unit: generating plans	112
Figure 6-10.	Generated set of alternative plans (example)	113
Figure 6-11.	Output of assignments data (example: plan by services agent)	114
Figure 6-12.	Single plan animation (example: plan by services agent)	114
Figure 6-13.	All plan animation (example)	115
Figure 7-1.	Benchmark test case: study area and its surroundings	119
Figure 7-2.	Benchmark test case: initial best sets of agents	123
Figure 7-3.	Benchmark test case: plan outcomes and evaluations	123

Figure 7-4.	Results of testing model sensitivity to changes in surroundings	124
Figure 7-5.	Settings for testing model sensitivity to changes in plan program	126
Figure 7-6.	Results of testing model sensitivity to changes in plan program	127
Figure 7-7.	Results of testing model sensitivity to changes in resolution: plansets and Kappa statistic	130
Figure 7-8.	Shift in location of first finalization due to a change in resolution	132
Figure 7-9.	Results of testing model sensitivity to changes in resolution: plan evaluations	132
Figure 7-10.	Results of testing the green space agent's rule for accessibility of other green space (example)	134
Figure 7-11.	Results of testing the business agent's rule for accessibility of other business (example)	134
Figure 7-12.	Results of testing the housing agent's rule for adjacency of business (example)	137
Figure 7-13.	Results of testing the business agent's rule for adjacency of green space (example)	137
Figure 7-14.	Results of testing the services agent's rule for population (example)	137
Figure 7-15.	Results of testing the clustering intensity for large-task agents	140
Figure 7-16.	Results of testing the clustering intensity for small-task agents	141
Figure 7-17.	Results of testing the opportunism for large-task agents	143
Figure 7-18.	Results of testing the opportunism for small-task agents	144
Figure 7-19.	Results of testing the claim accentuation for large-task agents	145
Figure 7-20.	Results of testing the claim accentuation for small-task agents	146
Figure 7-21.	Results of testing the team-based claim scaling policy	147
Figure 7-22.	Results of testing the initial weight sets of initiators: plansets	149
Figure 7-23.	Results of testing the weight adjustment used by initiators	151
Figure 7-24.	Results of testing the team-based priority policy for initiators	152

LIST OF TABLES

Table 2-1.	Synergetic contributions of DSS and ES to integrated systems	31
Table 4-1.	<i>MASQUE</i> Interface Agents	61
Table 4-2.	<i>MASQUE</i> Tool Agents	62
Table 4-3.	IMRO land use classification	64
Table 4-4.	<i>MASQUE</i> Domain Agents	65
Table 5-1.	CPT for including clustering for cell (i, j) at time t (example)	80
Table 5-2.	CPT for a logit-based update of $Bel\{SAF\}$ for cell (i, j) at time t (example)	83
Table 7-1.	Benchmark test case: land use weight sets of initiators (default settings)	119
Table 7-2.	Benchmark test case: agents' behaviour (default settings)	120
Table 7-3.	Benchmark test case: agents' decision models (default settings)	121
Table 7-4.	Stages at which all land uses have experienced one or more finalizations	127

1 INTRODUCTION

This first chapter explains the motivation and organization of the undertaken study. First, the current practice of urban plan development is briefly described to reveal the considerable degree of complexity involved. Subsequently, the apparent inadequacy of the currently available multitude of decision support tools is identified. Along this line, the aims and objectives of the research project are explicated and the outline of this thesis is detailed.

1.1 URBAN PLANNING AND THE NEED FOR DECISION SUPPORT

Urban planning is concerned with assembling and shaping the urban – i.e., local or municipal – environment by deciding about the composition and configuration of geographical objects in the space-time continuum (e.g., Lynch and Hack, 1984). As such, it sets the opportunities and constraints for individuals, households, companies and institutions with regard to conducting activities in the urban environment. However, problems – such as traffic congestions, sub-optimal performance of services, and general discontent among the population – can occur if the shaped environment does not sufficiently meet the needs and desires of its users regarding these activities. Thus, the task of urban planners is to provide an urban environment that, on a long-term basis, is expected to sufficiently meet the needs and desires of users residing in and interacting with that environment.

1.1.1 Instruments

In order to perform their task urban planners need a clear insight into users' demands and well-equipped instruments for optimizing the supplied environment given these demands. As urban planning embraces many different interests and deals with issues at different levels of scale – even when merely focussing on the local level – various 'policy instruments' or 'urban planning instruments' are used, each one serving a particular purpose. In the Netherlands, the traditional instruments used at the local level have been:

- *Structure plan*. An urban plan that indicates the outline of the overall urban development policy, covering the whole municipal territory or a substantial part of it.
- *Thematic plan*. An urban plan pointing out the development policy with

regard to one specific land use, e.g. green space or traffic, and covering the whole municipal territory or an even larger area.¹

- *Land use plan.* An urban plan that lays down legally-binding regulations for permissible land use in designated zones, either generally or more detailed, and covers specific parts of the municipal territory that can range in size from a city district to a building block.

The general purpose of these instruments is to document the policy in maps and text documents and to make it operational. However, in order to implement actual changes in the urban environment based on those plans further elaboration is required by means of urban designs. There are some ‘design instruments’ available like the *(re)development plan* that, based on given land use plan regulations, graphically shows the intended (re)development of an area by indicating the arrangement of buildings and public spaces.

Having reached their operational status, urban plans are not only guiding the current and future activities in the environment, they also define guidelines for further elaborations and future plans at lower scale levels. Due to this function, urban plans are frequently consulted by and exchanged between the municipal urban planning department that applies the plans and those that have to conform to the plans, i.e., the community and other governmental departments or (lower-scale) layers.

In the Netherlands, urban plan application has always been based on hard copy formats of plans, until in recent years the Dutch government started the so-called DURP² initiative. This initiative aims at converting all urban plans into digitized format in order to improve the ease and efficiency of consultation and exchange of (data of) those plans. As such, it serves mostly operational and administrative needs. This is, traditionally and usually, the primary purpose of local government information systems being developed (e.g., Klosterman, 2001).

1.1.2 Process

Urban planning practice gives evidence of an *urban plan development process* – i.e., the decision-making process that leads towards an operational plan – that is semi-structured.

¹ In current planning practice, thematic plans have become rare as they are replaced by ‘comprehensive plans’ that focus on the integration of land uses.

² DURP = “Digitaal Uitwisselbare Ruimtelijke Plannen”; Dutch for “digitally exchangeable spatial plans”.

Still, it is possible to distinguish several phases constituting this process. In terms of the general description of a decision-making process given by Simon (1977), three principal phases can be distinguished:

- Intelligence
- Design
- Choice

The *Intelligence* phase is committed to the detection of problems and the collection of information that might be required in the search for solutions. In the light of urban plan development, this phase can also be referred to as the phase of survey (Rutledge, 1971), i.e. an investigative preparation to the plan development in which activities take place such as problem identification and definition, information gathering, and formulation of goals and objectives. This commonly materializes in two documents: (1) a *Plan Program* that lists the early requirements of the project ('design brief'), and (2) an *Inventory* that contains all available and possibly relevant facts and data about both the study area ('site') and its surroundings ('area of influence').

The *Design* phase covers basically two stages. First, it includes a stage of analysis, which is dedicated to making value judgments about the effects of one fact upon another, usually in two respects. The information from the *Plan Program* – indicating which items (i.e., land uses) ought to be allocated in the study area – is digested through relational analysis in order to discover the mutual, functional interdependencies between the items. Additionally, the *Inventory* data is used for site analysis in order to visualize the opportunities and constraints offered by the study area and its surroundings. Second, this phase contains a stage of developing alternatives. Every alternative is the outcome of synthesizing the results of analysis into a comprehensive form that is an organizational solution to the problem (Rutledge, 1971). The reason for developing multiple solutions is usually twofold. On the one hand, it is a necessity due to the fact that many design decisions are often hard – not to say impossible – to make, as their effects cannot be objectively quantified nor qualified. On the other hand, it is also considered good practice to have a set of alternatives facilitate the democratic involvement of other parties in the decision-making process.

The *Choice* phase involves analyzing the alternatives and selecting one – a 'best' or at least 'good' solution – for implementation. The included analysis basically concerns the process of determining the effects or impacts of the various alternatives through

detailed forecasting. The subsequent selection implies a rigorous evaluation of the relative performances of the alternatives in light of the original goals and objectives and, normally, in consultation with all parties concerned. Once consensus has been reached over this choice, phases of implementation and long-term monitoring will follow. Whenever a need for revision is detected, a new decision-making process will start with the inadequacies of the current plan as input to the *Intelligence* phase.

As an implication of the massive drift towards decentralization seen in the last twenty years in Western societies (e.g., Batty and Longley, 2003), the urban plan development process has opened up under the header of *public participation* to let basically all parties affected by planning decisions have the opportunity to become involved in the process at its various phases. This, however, has not so much affected the general structure of the process as it has intensified the communication and interaction from start to end.

1.1.3 Complications

Urban environments are commonly considered to operate as a system (e.g., Batty and Xie, 1994c), i.e., as a unified assemblage of inter-related components (or elements). Besides dealing with the inherent complexity of ‘urban systems’, planners are faced with a high level of complexity in the process of decision-making as well, due to multi-actor, multi-goal, multi-scale, and multi-criteria facets (e.g., Ferrand, 1996). By default, they are dealing with multidisciplinary problems involving many interwoven factors and relationships (Yeh and Shi, 1999). Because one person cannot possibly oversee all of this, a planner usually works with a team of discipline-related experts and plays the role of director. “Planners rarely, if ever, solve problems or develop proposals in isolation” (Tweed, 1998, pp. 360). These multi-player situations imply a decision-making process that is complex, as the different parties will have conflicting beliefs (observations and perceptions) and desires (goals and expectations) (Ligtenberg *et al.*, 2002), which should be resolved by negotiation and cooperation.

Besides complexity, planners are also faced with high degrees of uncertainty and subjectivity. Many of the cause-effect relationships between the planner’s decisions and the quality of life are unclear in advance. The underlying mechanisms are highly profound due to the large variety of aspects involved, while the location-specific tendency of these aspects makes it fairly impossible to derive generally applicable rules. On the

other hand, ever more pragmatism has arisen in planning practice as it has attempted to respond to and embrace new fads and fashions (Batty, 1989). As a result also creativity – often implying subjectivity – has become an increasingly important prerequisite in the development of urban plans for the purpose of establishing design rationales that, on the one hand, meet the quality criteria regarding planning customs in force, and on the other hand, anticipate on the ‘genius loci’ in order to result in plan uniqueness. All together, it makes that planning decisions are founded more often on intuition rather than on actual knowledge.

When realizing that users’ demands change over time – not only in response to planning actions – we can state that the urban planning profession is one of anticipation, as any system equilibrium will be temporary at best and planners can only try to act in expectation of future changes in the demands. Fact is also that these changes take place at both aggregate and disaggregate levels, and they even tend to occur in a faster rate than most changes on the supply side – the urban environment – can possibly be implemented. This implies that the instruments planners have at their disposal show a serious degree of inertia. Consequently, urban plans are forced to incorporate a certain level of freedom to account for the uncertainty regarding future developments.

Although the plan development process outlined before presumes a straightforward structure, in practice it tends to be far more complex and irregular because the process actually is iterative, often partial and much diluted (Batty and Densham, 1996). For instance, evaluation may result in equally good alternatives or just none. The planner will then need to specify or generalize objectives, reformulate alternatives, perform additional analysis, and so on. The same holds when a plan is offered to the public for consultation and legitimate objections require adjustments to the plan. Sometimes particular phases may also be skipped. Planning practice shows that, after the need for a plan has been recognized, it seems rather tempting to minimize – or even totally omit – analysis activities and start with developing alternative plans at once. Finally, the plan development process is also nested, as all identified phases can generate new problems that will need to be solved according to similar three-phase processes. To put it briefly, the plan development process is far from a simple, structured, linear process, while planners need to deal with a matter that can be characterized by “non-autonomous, non-linear behaviour, a high sensitivity to initial conditions and hard to determine causal relations” (Ligtenberg *et al.*, 2001, pp. 22), which brings all kinds of complications to planners.

1.2 THE SUPPLY OF DECISION SUPPORT TOOLS

“Indeed, decision-makers today face an extremely complex reality where so many different factors interrelate so many facets of the problem under consideration that the support of a computer system becomes unavoidable” (Spaccapietra, 2001, p. 5).

So to say, the complex reality of urban planning practice justifies the efforts being made in the last decades to develop tools that can support urban planners in performing their professional task. Currently, there is a vast body of models and techniques to improve the quality of decision-making in shaping the urban environment. The newest generation of tools has materialized under a generic term – Planning Support Systems (PSS) – and include instruments related to geo-information technology that have been primarily developed to support different aspects of the planning process (Geertman and Stillwell, 2004). The conceptual ideal of a PSS is supposed to entail three components that can be referred to as *information*, *models*, and *visualization* (Klosterman, 1999a). Recently, an increasing use of Artificial Intelligence (AI) techniques can be observed, especially aimed at sophistication of the *models* component. Due to this trend – bringing in techniques such as genetic algorithms, neural networks, and agent-based simulation – PSS are maturing into instruments that can deal with more complex problem-solving tasks while alleviating users from having to understand the technical details of a system.

Despite the continuing evolution of PSS, in practice many planning decisions are still ill-informed due to planners showing very critical, or even distrustful, attitudes towards computer assistance (e.g., Geertman, 2001). In this respect, Klosterman (2001, pp. 1) argued that:

“...the continuing failure of planners to use computers extensively for core planning functions results less from the limitations of their hardware and software than from a limited understanding of the proper role these tools should play in planning”.

Since user acceptance is a key issue in the successful deployment of any computer system (Yeh and Shi, 1999), this is a major hurdle implying that PSS developers should not just simply provide tools to planners but as well convince the intended users of the substantial contributions these tools can make to those typical planning activities such as forecasting, analysis, and evaluation. However, it is unreasonable to assume that it simply lacks good advertisement, even though the added value of currently available tools may already be obvious to developers. It is far more likely that PSS need further advancements

in order to be convincing towards its potential users. Actually, continued development of PSS is an ever-present must because planning practice requirements are not guaranteed to be invariable, computer capacities and capabilities tend to grow exponentially, and upcoming innovative technologies will deliver new means to establish sophisticated advancements.

1.3 AIM AND OBJECTIVES

The study reported in this thesis is motivated by the conviction that PSS ought to be intelligent in terms of both their problem-solving capability and operation in order to serve their intended purpose of facilitating the plan development process. This implies that they should be capable of providing full-range support within the complex decision-making process to be undertaken, while remaining highly understandable to users.

When it comes to bringing in intelligence into PSS regarding their problem-solving capability, it is most natural to primarily focus on the *models* component. Clearly, planners are likely to benefit when tools are provided that are accurate and smart with respect to handling issues like uncertainty and data imprecision. This will count even more when the tools constitute integrated toolsets with a scope that covers all phases of the process: from the definition of problems, via the generation of alternatives, to the selection of a best alternative. Nevertheless, planners need more than tools alone. Perhaps the most distinctive aspect about the profession of urban planners is the role they play as director (or manager) of a planning team that consists of land use specialists. During the plan development process there is a high frequency and intensity of knowledge exchange and interaction – cooperation, negotiation, consultation, and so on – within and around this team. Consequently, in planning decision-making as much a crucial role should be ascribed to these specialized knowledge sources as to supportive toolsets. The question is how this could and should be addressed within PSS.

The success of PSS will for a great deal depend on their usability and user acceptance. With the further expansion and enhancement of PSS toolsets, as well as with the addition of a collection of knowledge sources, however, comes the risk of increased difficulties for users to understand and work with the systems. Hence, the only road to success appears to be the development of a PSS framework that is flexible and intelligent enough to efficiently and effectively allow extensions, now and in future, while feeling most familiar and understandable to users. The team setting in which planning is

categorically practiced, and to which planners are fully accustomed, gives reason to believe that PSS might be able to fulfil both these requirements when their backbone is shaped according to such a factual human organizational structure. System extensions would then basically imply managing the human-like organization by means of updating, replacing or adding members with specified tasks and roles.

In search for techniques and concepts to structure a PSS framework as a human organization, multi-agent technology appears to be the evident answer. The purely anthropomorphic nature of this technology would be able to ensure a highly recognizable system operation, while its capabilities for knowledge representation and model enhancement would enable facilitating intelligent and sophisticated system functionality. Realizing its resourcefulness, the current use of multi-agent technology in the area of planning support remains surprisingly limited. Hence, the aim of the present study is to explore the opportunities of using multi-agent technology within the context of PSS in an effort to formulate and demonstrate its potential to serve as underlying principle, both structurally and intelligently. This is specified into two objectives. On the one hand, we intend to develop a conceptual PSS framework that fully exploits the anthropomorphic nature of multi-agent technology by means of a system structure that resembles the human organization in planning practice and allows for a flexible provision of tools and knowledge. On the other hand, we intend to broaden the scope of utilizing multi-agent technology for planning support purposes by means of developing an application for demonstrative purposes. As such, the study intends to make contributions in theoretical terms by exploring new concepts at both the system and component level, as well as in practical terms by taking user understandability as a starting point.

The study is conducted in the context of (local) land use planning, i.e., the part of urban planning that focuses on the development of land use plans, a process in which the urban planner needs to make decisions concerning the composition and configuration of land uses for a demarcated area. Given the study's exploratory purpose of providing principles and illustrations based on new concepts, the focus is on the process of planning rather than the contents of planning.

1.4 OUTLINE OF THE THESIS

To this effect, this thesis is organized into three parts, each consisting of two chapters. The first part is concerned with identifying the potential contributions of multi-agent

technology to PSS developments. First, Chapter 2 gathers from literature the requirements that PSS in general are expected to meet in order to improve both the quality of decision-support and the understandability to users. This is followed by a typological overview of PSS developments to give an impression of current foci and achievements, while topics for further improvement are identified. Second, Chapter 3 offers insight into the concepts of multi-agent technology and relates these to planning support in order to identify the aspects of PSS to which multi-agent concepts are likely to be able to contribute.

The second part of the thesis is focussed on the conceptual specification of a multi-agent PSS framework. As such, Chapter 4 merges the findings from the preceding chapters into the conceptual framework of the PSS named '*MASQUE*' that aims at the support of land use planning. This will explain at which system levels and for which purposes agents are considered necessary. Subsequently, Chapter 5 focuses on broadening the scope of utilizing multi-agent technology for planning support by specifying a multi-agent model for generating alternative plans. This will demonstrate how the technology can be applied for incorporating and synergistically utilizing expertise of multiple disciplines.

The third and final part of the thesis discusses the design, implementation and testing of a prototype application based on the model specified in Chapter 5. To this effect, Chapter 6 provides insight into the main design and implementation aspects of the application named '*MASQUE* Alternative Plan Generator'. Subsequently, Chapter 7 discusses the results of an extensive series of tests performed on this application in order to demonstrate the operation of the underlying model and its components.

Finally, Chapter 8 closes the thesis with a summary of major conclusions and a discussion of research findings to indicate the needs and possibilities for future research.

PART I
INVENTORY

2 PLANNING SUPPORT SYSTEMS: TOWARDS AN IDEAL

This chapter addresses the field of Planning Support Systems (PSS), starting with a brief introduction to explain what has been at the roots of this field and what has come to be known as the conceptual ideal of PSS. Then, the requirements for such systems are extracted from literature, followed by a mainly technological overview of PSS developments in recent years. Special attention is given to the increasing interests in intelligent solutions. This is finally subjected to a critical reflection to identify the challenges lying ahead in the quest for ideal PSS.

2.1 INTRODUCTION

Along with the rapid developments in computer technology, the use of computers in urban planning has become more widespread and versatile. Still, planners use computers primarily for general-purpose office functions such as processing documents, monitoring budgets and maintaining records (Klosterman, 2001). In addition, however, computers have also found application in planning practice to serve design purposes by means of Computer Aided Design (CAD) systems, which enable planners to use computer graphics for drawing and visualization tasks that before would have been done with pencil and paper. In recent years, these systems tend to evolve into management systems with functionalities for handling spatial data and mapmaking. As such, they are incorporating some of the functionalities of Geographical Information Systems (GIS), which enable storing, retrieving, manipulating, analysing and presenting geo-referenced data. Starting to find broad application in planning from the early 1990s, GIS have emerged as (Chapin, 2003, pp. 565):

“...the most powerful and important computer application ever to hit the profession of planning (...) and fundamentally altered the profession of planning, requiring planners to rethink their approaches to analyzing problems and compelling planning agencies to develop new ways of warehousing, updating, and making available the vast amounts of spatial data currently under development”.

The functionalities of GIS for mapmaking, spatial data management, and spatial analysis – which is the actual power of the technology – have turned it into the most

supportive off-the-shelf software available to planners today. Although GIS developers often promote their software as a panacea for planners, it alone cannot fulfil all planners' requirements (Yeh, 1999; Klosterman, 2001). Developed from a commercial standpoint, the software tends to focus on a wide area of application and, thus, offers only general-purpose spatial analysis capabilities. Notwithstanding the fact that the current generation of GIS does have an important place in more routine, managerial types of planning, this causes it to lack the means required for typical planning practices – such as formulating, analysing, forecasting and evaluating alternatives – that call for extensive computational resources and sophisticated simulation and modelling capabilities (Harris and Batty, 2001).³ Moreover, its coherence with other software is often insufficient, requiring planners to integrate the different outcomes manually, which often turns out to be an extremely cumbersome task (e.g., Bishop, 1998).

In fact, a vicious circle occurs as the lack of dedicated software – tools and/or systems – leads planners to be ignorant of their practicability, while the resulting lack of demand discourages software developers from producing suitable functionality (e.g., Harris and Batty, 2001). Hence, the provision of software that can ease and improve planners' decision-making primarily depends on non-profit and academic initiatives. Indeed, many contributions as such have been made over the last several decades. With the change in emphasis from information-based planning to knowledge-based planning in the 1980s, the concept of Decision Support Systems (DSS) arose, generating a substantial body of theory and a large number of applications (Klosterman, 2001). One decade later, the growing adoption of GIS in planning practice more or less specified this stream, and as a result, Spatial Decision Support Systems (SDSS) were introduced (Densham, 1991), which can be defined as interactive, computer-based systems designed to support a user or group of users in achieving a higher effectiveness of decision-making while solving a semi-structured spatial decision problem (Malczewski, 1999). However, as the practice of planning evolves over time, the support provided to users should change accordingly. For instance, in the light of *public participation* – one of the major concerns in current practice – also laypersons have become an important user group, requiring a different level and type of support than planning professionals. Moreover, with the prevailing notions of strategic and comprehensive planning, the problems to be supported by

³ Some dedicated GIS software does exist, such as TransCAD® (Caliper Corporation) that is designed specifically for transportation planning, combining GIS and transportation modelling capabilities.

systems have become progressively more complex.

Developments during the last decade have been focussed on answering the growing urge for more flexible, well-equipped and user-friendly systems to support different aspects of the planning process in an ever more comprehensive and intelligent way. The resulting models and techniques are commonly considered to form a new generation under the header of Planning Support Systems (PSS). Due to the broad spectrum that is inherent to planning, various definitions of PSS can be found in literature. Batty (1995) described PSS as a toolbox that should provide a wide variety of users with tools to explore, represent, analyse, visualize, predict, prescribe, design, implement, monitor and discuss planning issues. In more technical terms, Yeh (1999, pp. 883) defined a PSS as “a combination of computer-based methods and models that support planning functions (...) and comprise a whole suite of related information technologies that have different applications in different stages of planning”. Based on an inventory of current practice, Geertman and Stillwell (2004, pp. 292) concluded that PSS consist of “a wide diversity of geo-information tools that are dedicated to support public or private planning processes (or parts thereof) at any particular spatial scale and within a specific planning context”.

Although the definitions of PSS differ with concern to their formulation, they illustrate a considerable amount of consensus about the concept of PSS. Consolidating, PSS can be identified as dedicated collections of computer-based tools – models, methods, techniques – that support one or more user groups during the planning process with concern to performing (series of) tasks – across different stages, scales, contexts – by means of applying multiple technologies in an integrated manner. Based on this overall consensus, it is possible to define the basic structure of PSS. According to Bishop (1998), the conceptual ideal includes capabilities of a GIS, purpose-built decision support models or procedures, and realistic, real-time, interactive visualization of the impact of decisions. Klosterman (1999a) comes to the same three components, referring to them as information, models, and visualization. Merging the ideas from several researchers, Klosterman (2001, pp. 17) envisions the perfect PSS as:

“...a fully integrated, flexible, and user-friendly system that allows the user to (1) select the appropriate analysis or forecasting tool from an “intelligent dialog box” that helps the user identify the most appropriate methodologies and tools for dealing with a particular task; (2) link the appropriate analytic or projection model to the required local, regional, or national information stored – or accessed – through the PSS; (3) run the appropriate models to determine the implications of alternative

policy choices and different assumptions about the present and the future; and (4) instantaneously view the results graphically in the form of charts, maps, and interactive video/sound displays. The process would incorporate both qualitative and quantitative decision aids, as appropriate, facilitating voting, ranking, and group interaction.”

Despite the extensiveness of this perception, an addition is still required when realizing that the difficulties of planning are not only in the analysis or evaluation of a set of alternatives but also, and perhaps even more, in the generation of these alternatives. Execution of this task requires the ability to search for favourable solutions given multiple goals and objectives. Moreover, in the light of contemporary planning and the supposed role of PSS – i.e., planning as a co-production process in which PSS provide the infrastructure to reason together, i.e., to facilitate collective design, social interaction, interpersonal communication, and community debate that attempts to achieve collective goals and deal with common concerns (Klosterman, 2001) – all parties should be offered equal chances to play a role in every part of the planning process. As the various PSS user groups will differ in the extent to which they possess particular problem-solving abilities, this implies that PSS should be able to compensate for skills that are underdeveloped or even lacking. This is definitely the case for ‘creative’ tasks such as the generation of alternatives that makes people – even the skilful or professional – incline to decision-making that is largely based on subjective and intuitive considerations.

2.2 REQUIREMENTS

The degree to which PSS are successful in supporting users to efficiently and successfully deal with the complexity of tasks obviously depends on the extent to which they can fulfil requirements of usage and capabilities.

2.2.1 System usage

Due to the fact that PSS are meant as instruments for humans – either planners or the public – principles of good practice in human-computer interaction (HCI) need to be taken into account. Actually, this is a crucial factor in relation to system usability and system acceptance (Tobón and Haklay, 2003). If users do not like or understand how a system operates they might neglect or misuse it, whether or not the functionality offered

is highly appropriate. Owing to this importance, HCI has emerged as an active field of study. Hewett *et al.* (1996) provided a lucid impression of its contents. Besides the nature of HCI – which concerns the study and development of meta-models of HCI – they distinguished four interrelated aspects to the field. First, it includes the use and social context of computers, implying the embedding of computer systems in the social, organizational and work environment, from the viewpoint of intended applications, and the procedural strategies required to do this successfully. Second, it studies characteristics of human information processing, language, communication, interaction, and physical user characteristics. Third, attention is given to the variety of computer technologies that is used to support the interaction with humans, such as I/O devices, dialogue techniques, interface metaphors, and computer graphics. Fourth, it concerns the development process, including design approaches for human-computer dialogues, techniques and tools for implementing and evaluating them, and exemplar studies.

Geertman and Stillwell (2004) formulated several recommendations that address HCI issues in the context of PSS developments, among others that (i) a user interface is required that is “sensitive to the characteristics of the user, to the kind of information that it communicates to that user, and to the types of intended use that is made of the information provided” (pp. 306), and (ii) PSS should be appealing to their users, fulfilling their needs and desires. Although such recommendations sound rather obvious, truly following them may be hard when realizing that the usage of PSS commonly involves dealing with “the rich yet complex environment that GIS can offer” (Tobón and Haklay, 2003, pp. 404). The complexities of spatial data management and modelling easily encroach on the GUI of a system. There is a general need for more intelligent HCI solutions in order to reduce information overload and augment the cognitive limitations and rationality bounds of users (El-Najdawi and Stylianou 1993).

The majority of recent PSS applications appears to be rather promising in meeting HCI requirements. This is partly due to the shift in focus that PSS made from producing and implementing physical plans towards communicating plans for facilitating public participation (e.g., Batty, 2003). As citizens are laypersons when it comes to planning expertise, they are better kept unaware of the idiosyncratic complexities of PSS. In other words, when focussing on public participation PSS should simply restrict the access to more technical parts. Hiding their complexity, they can have a more simple and intuitive interface than PSS for serving planners ever could. This suggests the development of PSS that operate in different *user modes* in order to control how tools are offered to different

types of users. For instance, whereas a tool for simulation could be a ‘one-click’ action for citizens, planners could be given access to parameters that control the tool.

Ongoing improvements in computer graphics and innovations in GUI design have enabled the development of experimental PSS user interfaces. For instance, Hopkins *et al.* (2004) presented a “sketch-planning workbench” that allows the traditional activity of collaborative sketch planning to be conducted with the support of computers. Another example is the use of “multi-views” (van Maren and Moloney, 2000), which allows users to choose whether to work inside 2D or 3D views based on the notion that this choice not only depends on the kinds of task to perform, but also on individual preferences. Such innovative experiments in planning applications, however, are still quite rare when compared to the domain of architecture. This is most likely caused by the simple fact that GIS – as the core technology in planning – is essentially data-oriented as opposed to the graphics-oriented CAD tradition in architecture. It will therefore be very hard, not to say impossible, to completely release planners from the complexities involved in spatial data management and modelling. Nevertheless, it has been demonstrated that GIS interfaces can be made more intuitive when integrated with other technologies like Virtual Reality (e.g., van Maren and Moloney, 2000). The apparent synergy of HCI and GIS (Tobón and Haklay, 2003) brings good prospects of improvements in the usability of PSS.

2.2.2 System capabilities

Besides meeting requirements of system usage, PSS should as well conform to the functional needs and desires of their intended users, which implies that they must provide comprehensive sets of tools, methods and techniques that appropriately support the planning process. The concept of different user modes needs further extension in this respect, as it is not only a matter of *how* tools are offered but also of *which* tools are offered. Namely, citizens and planners may require completely different tools. According to the prevailing notion of PSS for communicating plans, citizens will generally need to be provided with tools that allow them to easily react to proposals of planners by means of voting (e.g., through choice experiments) or developing and submitting personal proposals, in written and/or graphical formats. Planners, on the other hand, will mostly need tools to generate and broadcast their own proposals, and tools to analyze and evaluate the data coming from experiments and proposals submitted by citizens. Besides from being provided with customized sets of tools (or models), planners would greatly

benefit from functionality to improve, develop and share models (Yeh and Qiao, 2004a).

As Geertman and Stillwell (2004) pointed out, PSS should be an integral part of the planning process and context. This implies that PSS should offer their users an integrated toolbox that is dedicated to supporting (a part of) the process. Consequently, PSS need to be structured according to the particularities of that (partial) process, which means that PSS should obey to contemporary *planning fundamentals* such as planning theories, conceptions, principles, customs and so forth. It also means that PSS must be sensitive to the local context, as “the field of PSS is marked by one-off applications, tailored to the specific contexts in question and in this sense represent responses to local conditions” (Batty, 2003, pp. vii).

In order to have PSS become fully interwoven with the supported process, it should be taken into account that planning issues often cross boundaries to several other disciplines. This interdisciplinary characteristic of planning concerns basically two levels. First, there are obvious interrelations between land uses when addressing planning issues like adjacency and accessibility. Second, at a higher and more abstract level there is a need to link “the spatial to the social, the environmental to the economic and so forth” (Geertman and Stillwell, 2004, pp. 306). This calls for PSS that provide their users with access to knowledge of various disciplines, as well as with tools to deduct or predict the possible consequences of particular spatial decisions, e.g., tools to gain insight into the involved costs and benefits in several respects.

This brings us to the two principal requirements for planning that devolve onto any PSS (Harris and Batty, 2001, pp. 47):

“First, since system optimization (which equates with the automatic generation of plans) is impossible, the search for good plans must be by way of an informed process of trial and error which generates alternatives and prepares them for testing. This is often called sketch planning.

Second, planning and policy making need extensive tools for tracing out the consequences of alternatives, since otherwise there is no way to compare alternatives on the basis of their costs and benefits, and no way to look for means of improving or replacing alternatives.”

Whereas the former requirement covers the activities that are usually referred to as problem definition, formulation of goals and objectives, and formulation of alternatives, the latter alludes to the activities of forecasting and evaluation. Together these two

requirements constitute the functional core part of PSS and, thus, they are the ruling guidelines based on which to judge ongoing PSS developments and identify needs for further improvements.

The need for and the use of alternatives in planning is a logical consequence of both the presence of under-constrained variables that give rise to many plausible values of dependent parameters (Tang, 1997), and the high degree of unpredictability (Xiang and Clarke, 2003). Alternatives provide the means to anticipate on different future paths of development for the exploratory purpose of identifying each path's inherent opportunities and threats. Thus, PSS that support modelling, displaying, exploring and evaluating alternatives (e.g., Liggett and Jepson, 1995b) will comply with an important need in urban planning.

Besides the specific requirements that can be formulated for the three separate PSS components – *information, models and visualization* (section 2.1) – there are several general requirements to meet as well that could be categorized under the header of system flexibility. For instance, PSS should allow “the introduction of new methods of simulation, new sources of data, new flows of work, and new measurement and presentation of outputs, (...) and be adaptable to a wide variety of situations, levels of information, size and type of area being planned, and styles of planning” (Harris and Batty, 2001, pp. 49). Additionally, PSS need to be flexible in offering many options to users in model selection, data selection, and the selection of different types of planning (Yeh and Qiao, 2004a).

In order to not let such a high degree of flexibility encroach on the understandability of users, PSS should provide guidance throughout the supported (partial) process by means of tools that offer insight into what has been done and how it was done, and what still needs to be done and how it (procedurally) can be done. Guidance may also be required to let the user, in the context of the supported (partial) process, successfully deal with the underlying technologies of the system and the tools and knowledge it incorporates.

2.3 DEVELOPMENTS

Traditionally, the development of PSS has been strongly linked to GIS that offer functionality for spatial data warehousing (*information*), general spatial analysis (*models*), and mapping (*visualization*). Consequently, it is not surprising that developments started

with extensions to contemporary GIS, as modelling tools were loosely coupled or embedded in existing GIS (Batty and Xie, 1994a/b). Realizing the general-purpose scope of GIS, resulting PSS obviously had much redundant functionality, hampering their usability. This clearly improved with the introduction of component-based software development (CBSD). As this new approach soon became mainstream, GIS components started to appear that enabled PSS developers to rapidly develop customized stand-alone systems (e.g., Klosterman, 1999b).

Obviously, any PSS needs to have GIS functionality but it cannot consist of that alone (Yeh, 1999; Klosterman, 2001). GIS might be considered as a natural part of PSS, neither more nor less. While the usefulness of its purely spatial (or geographical) context to planning is overly clear, it just as well lacks functionality that meets the specific requirements of planners. In fact, GIS per se fall short in operating as PSS in every sense, as the functionalities they offer regarding all three required components remain too general. Hence, additional tools, technologies and functionalities are needed.

Recent experiences in the field of PSS development are well documented with the complementary overviews of Brail and Klosterman (2001) and Geertman and Stillwell (2003). As Batty (2003) concluded from these two overviews, practice illustrates that the role of GIS is becoming less significant. The general reaction of PSS developers to the deficiencies of GIS has been to develop stand-alone applications that diminish GIS to merely a mapping technology. In the following, a general insight into PSS developments is provided – along the line of the *information, models, and visualization* components of PSS – to give a basic impression of the multitude of aspects that need to be addressed in the quest for ideal PSS, and of recent advances with respect to each component. Later comparison with the previously discussed PSS requirements will reveal the improvements that still remain.

2.3.1 Information

By nature, urban planning is an intense information-gathering activity with a need for information about demand and supply, about needs, wants, and provision (Yeh and Webster, 2004), while the volume of data that planners need is increasingly large. Often, these needs cannot be clearly described, as they are diverse and changing. Despite the fact that the supply of information has also increased rapidly, there is an ever-present discrepancy between available types of data and those required for particular intellectual

tasks (Harris and Batty, 2001).

Although GIS support the organization of information in certain ways – especially by offering functionality to combine diverse sources of geographic data through overlays – the current generation cannot (easily) accommodate the particular informational needs of planning, such as (Klosterman, 2001, pp. 16):

“... (1) information that is effectively “a-spatial” at a particular level of analysis (e.g., regional population and employment levels and trends for the analysis of a city and its components);(2) information over time (e.g., population, employment, and land use data for the past, present, and possible futures); and (3) measures of spatial interaction (e.g., the number of trips between zones).”

Due to such typical needs, planning practice would benefit most from “systems in which the nature of the problem and the way it is to be addressed form the rationale for spatial representation” (Harris and Batty, 2001, pp. 32). Regarding the first type of information mentioned, the main difficulty concerns the fact that planners ought to work with multiple sources of data with various resolution, detail level and representation style (Davis and Laender, 1999). As research of these aspects is still in progress (e.g., Weibel and Dutton, 1999), GIS that provide methods for spatial data generalization and multi-scale representations of data remain experimental. This problem closely relates to issues of data quality and availability of metadata (e.g., Thurston, 2002).

In planners’ decision-making the distribution of relevant phenomena in space and time plays a major role very frequently (Spaccapietra, 2001). Hence, there is a clear need for GIS functionality to deal with the complexity of aspects that, over time, change in terms of size/shape, location, or a-spatial properties. To date, however, systems are at best oriented towards problems in which data are recorded as snapshots in time (Harris and Batty, 2001). Many efforts are made to tackle the problem of representing both space and time in digital databases, which appears to be a rather fundamental issue, formulated as follows by Peuguet (2001, pp. 18):

“... the standard and traditional object/field dichotomy utilized in GIS data modelling (...) ignores the third element of what/where/when in also providing for a representational perspective that is time-based, needed in order to explicitly represent evolution of entities and locations, and the interrelationships of these through time.”

Hitherto, solutions have been mostly searched for in pairing GIS with other software. As such, some commercial space-time tools have been developed (e.g., the “4D GIS construction management” tools by Integral GIS Inc., USA; and the “4D tool for spatial planning processes” by Geodan, The Netherlands) by means of a loose coupling of GIS (specifically, ESRI® ArcGIS 3D Analyst) with scheduling software (Primavera® Project Planner or Microsoft® Project). These tools, however, should be considered as primarily time-related animation tools that do not address the actual problem, i.e., the traditional a-temporal idiosyncrasies of GIS. More fundamental solutions are evolving in areas such as dynamic environmental modelling (Karssenbergh and de Jong, 2005).

Perhaps the most complicated space-time phenomenon to deal with is travel demand. Consequently, many efforts have been – and still are – made to provide planners with predictive models to get a better understanding of the possible effects of their decisions (e.g., Arentze and Timmermans, 2000). The deficiency of current GIS becomes clear when realizing that its development took place totally independent of the changes in planners’ data requirements, formulated as follows by Shaw and Wang (2000, pp. 161):

“With the evolution of urban travel demand models from aggregate models to disaggregate models and from a trip-based paradigm to an activity-based paradigm, there is a growing need of managing disaggregate travel data with spatial and temporal components in a GIS environment”

In addition to the need for improvements with concern to the *management* of both spatial and a-spatial data within GIS environments, efforts are required to improve the *exchange* of these data. For long, this issue has remained at the background of discussions, but recently it has started to be addressed much more directly, due to the ever-increasing data volumes and the wide acceptance of Internet technology as means of communication. In planning practice, spatial data exchange is required rather frequently, occurring both between governmental layers in order to attune plans according to hierarchy, and between government and the community in order to facilitate public participation at different stages of the plan development process. The issue of spatial data exchange is not so much related to the operational or technical level – the current generation of GIS generally can input, or at least display, geographical data in a wide range of formats – as it is to the semantic level, which concerns how the data are perceived, recorded and modelled (e.g., Burrough and McDonnell, 1998). This implies that planning requires consensus on an *ontology*, which can be understood as an

exhaustive conceptualization of the domain represented in a typically hierarchical data structure containing all the relevant objects, relations, events, and processes within the domain (e.g., Fonseca *et al.*, 2000). In this respect, it is worthwhile noting the research of Hopkins *et al.* (2003) on a Planning Markup Language (PML), which has the aim to develop an open standard for representing the contents and meaning of plans and regulations through the use of XML (eXtensible Markup Language), a web language for information structuring and encoding and system-to-system messaging. XML is also the keystone of the Planning Analysis and Modeling Markup Language (PAMML) presented by Singh (2003). This language, or notation, is dedicated to making analyses used in the decision-making process more transparent to parties involved or interested in the process.

Recently, the importance of data exchange is starting to be recognized in practice too, as can be noticed from the development of standards. Governmental initiatives are emerging in order to facilitate the growing need for easy access and exchange of spatial data between stakeholders. A good example is the DURP initiative by the Dutch Ministry of Housing, Spatial Planning and the Environment (Ravi, 2000) that is aimed at the overall transformation to digitally exchangeable spatial plans. As part of this initiative a standard – referred to as the IMRO information model – has been defined to record and code spatial planning entities and their attributes.

2.3.2 Models

As stated, the urban planning profession is concerned with decision-making regarding issues with a high degree of complexity. In order to make qualitatively good decisions, planning practitioners require methodological skills to correctly retrieve information from available data and to draw reliable conclusions about uncertain aspects. Because these skills are often lacking, incompletely developed or simply not exercised, there is a large amount of ill-informed decisions being made across the whole spectrum of planning tasks. Hence, the necessity of decision support throughout the various phases of the planning process is overly evident. In reaction to this, a vast body of models and techniques has been developed over the last decades aimed at improving the quality of planning decisions, usually in particular stages of the process.

In the light of the ideal PSS, a context-sensitive *models* component is required that provide users with an integrated set of dedicated decision support tools, usually referred to as analytic or spatial models. Contemporary GIS lack such a customized ‘toolbox’.

Nevertheless, because of the fact that the majority of analysis in planning has a spatial context, PSS developers do consider GIS as a proper background for putting their dedicated tools into action. Consequently, much effort has been put into integrating GIS and planning support tools. Practically, this integration is accomplished either by embedding decision support functionality into a proprietary GIS or by embedding GIS functionality into a custom-made system. Batty and Densham (1996) described examples of both approaches. For long, the latter approach has been applied most often because it appeared to be difficult to incorporate analytic models into GIS (Yeh, 1999). Thanks to advances in the interoperability of GIS, however, the former approach has now become equally well applicable. In fact, the distinction between the two is becoming less significant over the years due to technological improvements on either side. Whereas proprietary GIS show increased flexibility and customizability, the speed and ease of developing custom-made systems has clearly improved with the release of software – such as ESRI® MapObjects and its successor ArcGIS Engine – that ease the development and embedding of GIS components.

For instance, Carsjens *et al.* (2003) discussed a tool for integrating environmental aspects into planning procedures that has been embedded in the proprietary ESRI® ArcView by means of the additional programming language Avenue for implementation. In the described test cases, the tool proved to be fast, efficient, and user-friendly. An excellent example of the other option – embedding GIS functionality into a custom-made system – is the software named “What if? PSS” (Klosterman, 1999b). This system is dedicated to land use planning, scenario-based, and using GIS data to support community-based processes of collaborative planning and collective decision-making. The included decision support tools are procedures for conducting land suitability analysis, projecting future land use demands, and allocating the projected demands to the most suitable locations. It clearly demonstrates how selected GIS-related decision support tools can be merged with the required GIS functionality into a standalone system. Considering the high level of customization that PSS generally require, it is most likely that this approach will gain more ground when the available GIS components become more and more versatile and flexible.

In recent years, however, the question of how to integrate decision support tools and GIS functionality has been turned into a background topic of discussion. Nowadays, most attention is paid to the question of how to adapt to the arriving new wave of models. This concerns especially the new generation of simulation models that brings about a huge

fundamental change in underlying principles and mechanisms. Traditional urban simulation models – most commonly devised as combinations of spatial interaction models, spatial choice models and simple functional statements – suffer from their aggregate approach and the multitude of subsequently needed simplifications and assumptions, which turns them into instruments that are weak concerning dynamics, details, usability and realism (Torrens, 2003). The recognition of these limitations has resulted in the gradual replacement of aggregate models by disaggregate models. Micro-simulation is now considered as perhaps the only practical way to study complex systems such as urban systems and, thus, the most suitable instrument to test the feasibility of decisions or predict their effects (Wu, 2002). Hence, the newest simulation models are bottom-up, as they operate at the scale of individuals and objects in the built environment. In a most natural way, they let the macroscopic urban phenomena and the spatial processes that shape them emerge from the microscopic behaviour of the individuals. As such, contemporary models are becoming more emergent or *generative* by nature (e.g., Epstein, 1999), and turn into tools that facilitate discussion instead of providing prescriptive remedies (Torrens, 2003).

While this revolutionary change in the very principle of urban modelling is taking place, its actual application in planning practice remains limited. To date, it is still unclear how the new generation of models should be integrated with GIS, as the idea of individuals and their interaction seems hard to incorporate (Batty and Longley, 2003). The *models* component of PSS is becoming much more demanding to the *information* component with which integration should be established. Perhaps even more problematic is the fact that, in the meantime, the *models* component will also need to conform to demands originating from its intended integration with the *visualization* component of PSS (see section 2.3.3). Whereas based on recent developments the *information* component could be expected to technically realize the 3D visualization demands, the theoretical extension of spatial analysis into 3D is likely to remain problematic:

“It is almost as if new theories of the 3D realm in cities and related environments are required before the analytic capabilities of GIS can be extended.” (Batty and Longley, 2003, pp. 429)

Regarding the contents of the *models* component, current PSS developments show a clear imbalance in the aspects of planning being addressed. As stated before, planning is not only about forecasting and evaluation but it also involves problem definition,

formulation of goals and objectives and generation of alternatives. Still, the overviews of current experiences in PSS development (Brail and Klosterman, 2001; Geertman and Stillwell, 2003) illustrate that the focus seems largely on enhancing public participation and forecasting based on what-if scenarios. Although there is a big challenge in facilitating these aspects of planning, the preceding stages of the plan development process should not remain without support. Perhaps it is the misconception that tools to support the early and more creative stages should and will be optimization tools that provide straightaway solutions. Given the large degree of uncertainty about the (interwoven) effects of decisions, however, it is practically impossible to optimize the urban system. Instead, the proper way to think of supporting, for instance, the generation of alternatives is to think of tools that give insight into opportunities and threats, identify solution spaces, provide quick scan functionalities, and so on, in order to feed planners with information that could help them formulate good alternatives. Examples that support the plausibility of successfully developing such tools are the modeling-to-generate-alternatives (MGA) technique based on mathematical programming (Brill *et al.*, 1990) – in the context of which it was found that providing users with a small number of distinctive alternatives is most effective for decision-making – and the Sketch Layout Model (SLM) based on a genetic algorithm to produce a set of alternative sketch maps (Feng and Lin, 1999).

2.3.3 Visualization

The intrinsic complexity of many planning issues makes it hard to interpret the effects of related decisions. Hence, good means of visualization are essential in the context of PSS. Actually, visualization is central to the current planning style of decentralization and participation (Batty and Longley, 2003). The type of visualization that PSS require will depend on both the intended user group(s) and the purpose(s) to serve. In early stages of the urban plan development process, for instance, planners themselves may get sufficient support from simple graphs or two-dimensional (2D) map representations. When the system is intended to support other user groups – anyone who may be less of an expert in interpreting the information – more natural, and therefore, more easily interpretable three-dimensional (3D) visualizations will be required. In other stages of the process, even planners may need realistic 3D visualization to assess the impacts of intended decisions.

Some conventional GIS do offer extensions to 3D, yet they face problems supplying

interaction with that additional dimension (e.g., Zlatanova, 2000). In order to let planners interact with the information, many researchers have suggested ways of extending GIS to support the third and/or fourth dimension by integrating GIS with Virtual Reality (VR). The simplest strategy in this respect is adding functionality to GIS for generating navigable three-dimensional representations from two-dimensional maps by extruding entities with the help of a height-attribute. Actually, because the third dimension is treated as an attribute of two-dimensional entities, this strategy only yields so-called 2.5D scenes. For producing actual 3D scenes, the databases have to be three-dimensional as well. Whether or not due to this fact the applications that have been developed strongly differ with respect to the functionality – i.e., level of interaction – they offer within their visualizations. Three categories of GIS/VR applications can be distinguished with increasing advances: from merely offering opportunities for navigation; via providing means for data retrieval and modification, to offering functionality for ‘full’ interaction by means of object manipulation and GIS analysis.

The first category of GIS/VR applications does not support 3D remodelling, i.e. making changes to objects while inside the 3D scene (e.g., Dodge and Jiang, 1997). There are no ways to interact with the objects in the scene, or stated differently, there is only a one-way link between the GIS and the VR with no means of feedback (Smith *et al.*, 1998). In order to view changes, one has to go back to the GIS, make the changes and generate a new 3D scene. Navigation is also the main objective of so-called ‘3D city models’ (e.g., Ennis and Lindsay, 2000), that combine 2D and 3D scenes to enable the virtual exploration of a city. Rather more advanced is the second category of applications that offers functionality for retrieving object data within the 3D scene (e.g., Köninger and Bartel, 1998), occasionally combined with functionality for modifying attribute data (e.g., Fujii *et al.*, 1995; Li *et al.*, 1999; Zlatanova, 2000). The most advanced category of applications to date provides ways for 3D remodelling and more. One such application is the ‘Urban Simulator’ reported on by Liggett and Jepson (1995a), which shows the Los Angeles area in high detail and supports working with scenarios for evaluating alternative designs and time-effects. The same functionality, albeit organized differently, can be found in the ‘CommunityViz’ software (Kwartler and Bernard, 2001), which facilitates 3D (re)modelling by offering its users – i.e., rural communities – a model library that contains typological objects. Finally, van Maren and Moloney (2000) and van Maren (2003) presented an application that provides three view modes – a 2D map view, a 3D bird’s eye view and a 3D street level view – and offers ways for manipulating and editing

GIS data as well as performing some basic GIS analyses in all these modes.

The fact that decision-making in urban planning concerns mostly geo-referenced objects implies that planners are dealing with spatial representations of data in two or more dimensions, and thus with design decisions. Various researchers (e.g., Blaser *et al.*, 2000; Harris, 2001; Ishii *et al.*, 2002; Hopkins *et al.*, 2004; de Vries *et al.*, 2005) have focussed on the development of ‘design support’ tools that enable, for instance, exploratory sketching in early stages of the process, integration of the various forms of representation used in urban design, or the interactive tuning of planning and design decisions.

2.4 INTELLIGENT PLANNING SUPPORT

As mentioned earlier, in the 1980s the emphasis in planning shifted from information to knowledge as the required keystone for decision-making. The resultant need for access to sets of analytic tools within user-friendly environments started a line of research that focussed on the development of Decision Support Systems (DSS), and more specifically Spatial DSS (SDSS). From the background of Management Information Systems, this line of research has primarily been drawn on data processing and/or operations research techniques to support users in their decision-making concerning unstructured or semi-structured problems by means of utilizing decision rules and models (‘model base’) coupled with a comprehensive database (e.g., Turban, 1988; Darlington, 2000). The overall objective of DSS is to improve the quality of decision-making by providing users with sets of previously separate tools (data and models) to form a unified whole that is more valuable than the sum of the parts (Malczewski, 1999). Generally, DSS require extensive user involvement in the sense that it is the user who needs to select appropriate models from the model base, indicate the data to use from the database, and interpret the system outputs as a basis for making a decision. Hence, the system is essentially a repository for models and data (‘toolbox’) and an environment to run models. The system outputs – e.g., optimal values or locations – will not provide users with solutions but quantitatively assist them in the search for solutions.

Although this can already be of great value to planners, it should be considered the main drawback of (S)DSS that *intelligence* is lacking in terms of reasoning capabilities that would enable solving problems on behalf of the user. The significance of this drawback becomes evident when realizing that decision-making in planning requires

capabilities to process both formal (e.g., legal requirements, planning procedures, and problem situations) and informal information (e.g., obtained through personal judgements, hunches, intuition, hearsay, and personal experiences), demanding support in both quantitative and qualitative terms (Han and Kim, 1990). Furthermore, one of the many complexities that planners experience is the fact that the information to be used is often incomplete, imprecise and dynamic, thus requiring human knowledge and expertise to manipulate and digest the information effectively and efficiently (Leung, 1997). According to Klosterman (2001), intelligent support implies computer systems that are able to deal with novel situations and new problems, to apply knowledge acquired from experience, and to use the power of reasoning effectively as a guide to behaviour. Because such functionality is not included in the concept of (S)DSS, the typical complexities of planning remain poorly supported.

The issue of providing users with intelligent computer systems has been the topic of a parallel line of research that set off decades ago from the background of Artificial Intelligence and materialized in the form of Expert Systems (ES), and more specifically Spatial ES (SES). The basic intention of ES is to *make* (tentative) decisions by means of a reasoning mechanism for propagating inferences over a knowledge base that contains an expert's knowledge for a particular problem domain (Turban, 1988). Human expertise is mimicked by capturing knowledge such as human experience, valuation, intuition and judgement (e.g., Leung, 1997; Darlington, 2000), which brings about the capability to solve (spatial) problems as well as or better than human experts, use expert knowledge in the form of rules or frames, and interact with users by transferring expertise and rendering advice or recommendations (Malczewski, 1999). However, whereas the problem area of DSS is usually broad and complex, ES are typically confined to narrow and well-defined problem domains and operate in highly isolated modes. The consequence is that they are most suitable for providing advice on repetitive problem areas such as diagnosis and malfunctions (e.g., Turban, 1988). Practically, the introversion of ES has been the main hindering factor for finding application in managerial decision support that requires much more negotiation and interaction with exogenous bodies of knowledge (Edwards, 1992). Theoretically, (S)ES have clear potentials to help out planners with various problems (Batty and Yeh, 1991). To date, however, only applications exist that address subfields – like problem formulation (George, 1995) – instead of providing solutions for the entire problem domain. Among a variety of plausible reasons (e.g., Han and Kim, 1990), this may be largely due to the fact that the domain of problems in planning is relatively large

and difficult to confine to boundaries, implying encoding more knowledge – perhaps even more than can be encoded – along various dimensions and disciplines, in order to enable decision-making (Arentze, 1999).

The anticipated contributions of DSS and ES to the field of planning are clearly complementary (Table 2-1). In order to create more powerful and useful computer systems, the idea of integrating these technologies into systems that could yield synergy has developed into a new line of research since the late 1980s. Commonly, this integration is considered to be a matter of using ES techniques within a DSS framework for enhancing the modelling capabilities of the system or improving the intelligence of the system in various components including data management, model management and user interface (Arentze, 1999). Resulting applications have been given various names, such as intelligent DSS (IDSS), intelligent support systems, expert DSS (EDSS), expert support system (ESS), and knowledge-based DSS (KBDSS) (e.g., El-Najdawi and Stylianou 1993). Albeit less common due to the wide scope required for planning support, it is also possible to integrate DSS into conventional ES, which would mean recognizing the fact that experts often use quantitative models to support their experience, intuition, or rules-of-thumb (Han and Kim, 1990).

From the viewpoint of spatial decision-making, Malczewski (1999) described a generic framework for a Spatial Expert Support System (SESS) in which an ES component is linked (or integrated) to each of the three basic components of an SDSS, i.e., a geographical database management system (GDBMS), a model-based management system (MBMS), and a dialogue generation and management system (DGMS). As opposed to this system architecture that advocates enhancing each of the SDSS components with dedicated ES techniques, Leung (1997) suggested one central ES shell to operate as the core or brain of SDSS, providing linkages – interfaces – with the various SDSS components. In addition, the necessity of knowledge acquisition modules is

Table 2-1. Synergetic contributions of DSS and ES to integrated systems (source: Turban, 1988)

DSS contribution	ES contribution
<ul style="list-style-type: none"> • Experience in data collection • Experience in implementation • Personalized advice to users to match their decision styles • Quantitative, mathematical, and computational reasoning 	<ul style="list-style-type: none"> • Intelligent advice (faster and cheaper than human) to DSS or its user • Explanatory capabilities • Computerization of decision-making process • Qualitative analysis (e.g., analogical reasoning, pattern recognition, and content analysis)

explicitly stressed within this system architecture. As such, the ES shell directs control and information flows, while providing facilities to represent and store domain specific knowledge acquired from human experts and learning examples, and to contain metaknowledge for inference control, systems and user interface, and external communication. Albeit in conceptual terms, El-Najdawi and Stylianou (1993) proposed a model for DSS-ES integration that basically stands midway between these two architectures, as the enhanced DSS components are linked through a *blackboard*, implying that communications between the components is arranged through a shared database. Several other expert components are suggested as part of this system architecture, among which a *knowledge cache* that, by means of a collection of intelligent agents, represents the knowledge of specialists in the organization. These agents are intended to be available to the various system components through the blackboard, and to help the user come to a better understanding of the structure and formulation of a problem at hand if necessary. Additionally, the agents are assumed to act as expert critics by analyzing the user's decision-making process – including the choice of models and databases, interpretation of results and conclusions – and providing suggestions.

In general, consensus appears to exist over the fact that DSS-ES integrations require multiple experts – either in terms of components or interfaces – in order to have the resulting system operate intelligently. But there are more possible ways to adopt the notion of multi-experts than only for interconnecting DSS components. In this respect, the collection of intelligent agents as suggested by El-Najdawi and Stylianou (1993) is noteworthy, as it advocates the use of a multi-expert concept for the inclusion of a repository of multidisciplinary knowledge that can support both system components and users. Some prototype systems exist that embody this type of concept, like the Intelligent CAD System (ICADS) developed in the area of building design and physics (Pohl *et al.*, 2000) that incorporates multiple experts in domains such as natural and artificial lightning, noise control, structural system selection, climatic determinants, and energy conservation. The system assists architects in the development, analysis, and evaluation of solutions during the early design process.

The repository concept could be expanded with an extra dimension, when adopting the way in which the notion of multi-experts is commonly used in the field of pattern recognition (e.g., Cordella *et al.*, 1999): in order to improve the reliability of system output (decision), several compensatory experts are assigned to solve the same problem under the assumption that, by suitably combining the results of these experts according to

a certain rule (combining rule), the performance obtained can be better than that of a single expert. In general terms, this approach is resembled by some recent applications in the field of urban planning. In Ligtenberg *et al.*, (2001) and Ligtenberg *et al.* (2004) simulation studies were described in which different stakeholders with various objectives are represented to perform the same task (e.g., determining the preferred location for new urbanization), after which the varying outcomes are subjected to a voting procedure in order to select the outcome for implementation.

2.5 CONCLUSIONS AND DISCUSSION

Many of the issues that ought to be addressed for meeting the requirements of the supposed ideal PSS are receiving scientific attention and dedication. Generally, the developments proceed steadily and promising paths are being explored. A large amount of the ongoing research, however, does not originate from the notion of ideal PSS because many topics are fundamental and require solutions from fields of research such as GIS, visualization and user interface techniques. Hence, advances in PSS development are largely dependent on the advances made with regard to a combination of technologies. At the same time, the reality of planning practice makes it mandatory to deliver PSS as one-off applications in response to the characteristics of the typical problem and/or situation they should address. In many cases, the aim is to support a part of the planning process (i.e., a certain range of decisions), requiring only 'partial' solutions. For instance, when the supported part requires decision-making based on insights into 2D aspects, the system should focus on optimizing the provision of that particular support; more complex 3D or 4D models would be redundant. The dissimilarity between systems can also be a result of differences in the intended user groups or the objectives regarding system usage, causing variation in, for instance, the provided set of models or the degree to which such models are integrated. Overall, PSS developments remain largely uncoordinated, which causes disorientation from the goal that should bind them in order to ever reach ideal PSS.

In order to streamline developments it is crucial to pay attention to the conceptual framework of PSS. Following from the situation described above, efforts must be made to develop a framework that is both generic in providing the key building blocks needed for any PSS and flexible in accounting for the singularities of specific cases. Important to bear in mind here is that, despite their indispensability regarding system functionality, the required technological components should not drive system structure but serve only

supportive purposes. This is especially important in the light of GIS that, for long, has been ascribed a core position within PSS, constituting the *information* component. Recent developments, however, show that a re-evaluation is taking place that turns GIS into a serving instead of ruling component focussed on provides data management and mapping functionality. This widely accepted repositioning of GIS to the background of PSS – like the *visualization* component – forms a perfect starting point to reconsider the framework of PSS in the light of their ideal.

Obviously, shifting GIS to the background implies that an alternative way needs to be found to structure and organize PSS in general terms. Since the particularities of planning are basically assimilated by the *models* component, it is practically self-evident to consider this as the component that ought to be most central and directive within the framework of PSS. In its operation as core component, support should be conjointly provided by the *information* and *visualization* components, while the user interface should take care of optimizing the system appearance and usability. For defining a suitable and viable PSS framework, however, it is obviously not simply a matter of reassessing the roles of components and rearranging them. Essentially, it requires attention for all aspects that determine the operation of PSS as coherent assemblages of components and as instruments to users.

Experiences show that the complexity and diversity of problems in areas such as planning can only be appropriately addressed by means of systems that demonstrate *intelligence* stretched out over several dimensions. For instance, intelligence is required for system flexibility – in terms of extensibility and adaptability – as it is certain that successive changes will occur in requirements coming from planning practice as well as in the technological and methodological instruments delivered by industry and science. Subsequently, the variety and complexity of models and data to be bundled by PSS demand intelligence along two dimensions. First, it is needed to ensure flawless information and control flows of models and data, implying a smooth integration of system components, ideally yielding synergy. Second, intelligent solutions are needed with concern to the user interface. PSS should not only demonstrate a consistent sensitivity to user characteristics, information types and information usage by providing various operation modes, they should also remain comprehensible to users irrespective of the depth and width of their functionality, which requires user assistance that is intelligent in terms of avoiding information overload and augmenting the limitation in skills of users. Moreover, user assistance should be anticipatory, corrective, explanatory, guiding, and so

on, in a non-obtrusive manner.

It is manifest that PSS will not only benefit from intelligence that optimizes their operation but just as well from intelligence that improves the support of decision-making to users. Research that has been conducted with concern to DSS-ES integration illustrates that the centrally positioned *models* component is eligible for intelligent sophistication in a dual manner. On the one hand, PSS would greatly benefit from intelligent solutions regarding model management and model construction, as the selection of appropriate models and the development and incorporation of new models are often difficult and time-consuming tasks for users. On the other hand, the use of generally every model would strongly improve when intelligence is present to assist with input preparation (e.g., use of subjective knowledge for problem formulation) and output interpretation (e.g., use of judgemental and explanatory capabilities).

Considering the argued central position of the *models* component within the PSS framework, the question of what aspect to use for structuring the framework could be traced back to the question of what should basically structure the *models* component. As stated before, models within PSS address specific stages of the planning process and, most ideally, jointly cover the whole process from problem identification until the choice after evaluation. So to say, the planning process forms the actual blueprint of the *models* component and, thus, it should do so for the framework of PSS as well. At least, this holds in general terms, as it will depend on the exact purpose of each process how it will be structured in detail and what range of models will need to be provided. The requirements for the *information* and *visualization* components follow from this, alike the needs regarding the user interface.

Planning decision-making generates highly frequent needs for expertise from a variety of disciplines. Although the knowledge contained by the *models* component of PSS can be of various types and concern various disciplines, it is traditionally encapsulated in the mechanisms of models. This makes it hard at best to retrieve any specified piece of knowledge. Moreover, the knowledge provided altogether in PSS often exhibits missing parts. In other words, the required expertise is usually hidden and incomplete, complicating both its use and its management. Accordingly, there are good reasons to expand the framework of PSS with a component that explicitly incorporates the specialized knowledge from the various disciplines involved in planning. Since the need for expertise originates from within the planning process, the component in question should obviously form a duality with the process-related *models* component. In search for

ideal PSS, the operation of this *knowledge* component should go beyond that of a plain repository and be intelligently connected with every single part of the framework. Namely, synergy lies ahead when knowledge is utilized for the operation of other components (to generate data, to function as model input or to be a source for user assistance), while at the same moment the other components form the very sources for knowledge retrieval.

The success of PSS deployment will highly depend on the degree to which the required dimensions of intelligence can be realized. PSS will not be considered useful, and will definitely not convince users of their potential contributions to the decision-making process, as long as they lack the intelligence required to decide on the use of appropriate types of knowledge and the right kind of information to solve planning problems. Nor they will, when they overwhelm users by releasing too much of their inherent complexities. Basically, the incorporation of intelligence in PSS requires the use of methods and techniques from the areas of artificial intelligence and knowledge engineering. As a branch of artificial intelligence, multi-agent technology offers promising concepts and techniques that give it the potential to improve PSS with concern to both system usage and system capabilities. For instance, it perfectly matches the strong demand of planners for disaggregate models, while also providing the means to realize the desired *knowledge* component. At the same time, it has the potential to serve the purpose of organizationally structuring PSS in a way that facilitates all kinds of internal control flows and still remain most natural to human users. The next chapter will provide an introduction to the area of multi-agent technology and explain how this technology is believed to play a crucial role in the quest for ideal PSS.

3 MULTI-AGENT TECHNOLOGY AND PLANNING SUPPORT

This chapter introduces the concepts of multi-agent technology and lists possible types of applications. As such, it identifies the opportunities these concepts bring along for propelling Planning Support Systems (PSS) developments, and outlines how a layered agent organization can be used in relation to the urban system.

3.1 INTRODUCTION

The complexity of issues that software systems are supposed to address seems to be ever increasing. Particularly in the last decade it has become clear that traditional concepts of computer science are inadequate to deal with this trend efficiently and effectively. Consequently, computer science has started to adopt concepts originating from other sciences, such as organizational and social science (Zambonelli and Van Dyke Parunak, 2003, pp. 22):

“..., as it is recognized that the behaviour of a large scale software system can be assimilated more appropriately to a human organization aimed at reaching a global organizational goal, or to a society in which the overall global behaviour derives from the self-interested intentional behaviour of its individual members, than to a logical or mechanical system.”

Or, as Jennings (2001, pp. 38) states:

“When designing software, the most powerful abstractions are those that minimize the semantic gap between the units of analysis that are intuitively used to conceptualize the problem and the constructs present in the solution paradigm.”

It is this compulsory shift in software engineering that has made multi-agent technology to become one of the most important and appealing technologies emerging in computer science today. It is considered a new paradigm for analyzing, designing, and implementing software systems (e.g., Sycara, 1998). According to Zambonelli and Van Dyke Parunak (2003) the change can even be called a revolution as it brings qualitative changes to the characteristics of software systems, leading to four main system properties:

- *Situatedness*: software components execute in the context of an environment, can influence it, and can be influenced by it;
- *Openness*: software systems are subject to decentralized management and can dynamically change their structure;
- *Locality in control*: the components of software systems represents autonomous and proactive loci of control;
- *Locality in interactions*: software components interact accordingly to local (geographical or logical) patterns.

To this effect, multi-agent technology offers the tools, techniques, and metaphors to design and implement systems that are open, complex and/or ubiquitous (Jennings and Wooldridge, 1998). The fact that, above all, it provides a high-level view on systems by means of a natural view of the world gives rise to a new generation of application structures with a high degree of anthropomorphy and, hence, understandability by the users (Cuenca and Ossowski, 1999). The naturalness and ease with which a variety of applications can be characterized in terms of agents has convinced many researchers and developers of its potential. The domains in which it is finding application are extremely diverse (e.g., Nwana, 1996; Jennings *et al.*, 1998), including workflow management, network management, air-traffic control, business process re-engineering, data mining, information retrieval/management, e-commerce, education, personal digital assistants, e-mail, digital libraries, command and control, smart databases, scheduling/diary management, and more.

3.1.1 Agents

Surprisingly, there is no real consensus about the core question of exactly what an agent is (Jennings and Wooldridge, 1998). Besides the wide applicability of the agent concept, this should be attributed to a lack of standards, which is a common problem for every new technology (Janca and Gilbert, 1998). Nevertheless, it is possible to identify certain agent characteristics that are commonly considered as central to an agent. To start with the general structure of an agent, it is rather undisputed that its *anatomy* is constituted of three major components (e.g., Cuenca and Ossowski, 1999):

- A *perception* subsystem that allows the agent to be situated in the environment by data acquisition and in the agent society by perceiving agent messages.

- An *intelligence* subsystem that manages the different aspects of information processing as well as individual and social problem solving.
- An *action* subsystem that enacts the intentions produced by the intelligence subsystem, displaying messages to the user, or sending messages to other agents.

In these terms, an agent may still appear not any different from a conventional input-processing-output cycle at first sight, but there are two additional and generally agreed upon aspects that determine its singularity. First, multi-agent technology promotes the concept of *intentional* intelligence, which is the capability of an agent to behave autonomously so as to achieve a given goal (Zambonelli and Van Dyke Parunak, 2003). This *autonomy* refers to the principle that an agent can operate without the direct intervention of humans or other agents, having control over its own actions and internal state (Jennings and Wooldridge, 1998), and being directed by a set of tendencies in the form of individual objectives (Ferber, 1999). Still, an agent's autonomous action principally takes place within the scope of the task(s) that it is delegated to perform by initially the developer and later the user (Janca and Gilbert, 1998). Humans thus remain having control by setting the constraints under which an agent operates. Second, an agent is supposed to demonstrate *flexibility* in three respects (Jennings and Wooldridge, 1998). It should have the ability to perceive its environment and respond to changes (*responsiveness*), the ability to exhibit opportunistic, goal-directed behaviour and take initiative (*proactiveness*), and the ability to interact with others (user or agent) to complete their tasks and to help others with theirs (*sociability*).

A popular approach in formalizing agent-based systems is based on practical reasoning theory and characterizes an agent as a rational decision-maker (Jennings and Wooldridge, 1998), i.e., as an entity whose internal state can be expressed in terms of mental constructs or attitudes such as beliefs, desires, and intentions (BDI), representing respectively the information, motivational and deliberative states of that agent (Rao and Georgeff, 1995). An agent's decision-making is then characterized in terms of those constructs. Commonly, *bounded rationality* is assumed to realistically say that a single agent will have limited capacity in the sense of its knowledge, its computing resources and its perspective (Sycara, 1998). Hence, agents do not have global information and they do not have infinite computational power, instead they typically use a limited set of relatively simple rules based on local information (Epstein, 1999). In section 3.2 it will

become clear that the actual knowledge and capabilities of agents are fully dependent on the application they are designed for, i.e., the purpose they are meant to serve. Hence, agents tend to materialize in many forms that only share some abstract principles, making both their appearance and operation very distinctive.

3.1.2 Multi-agent systems

When adopting an agent-oriented view, it becomes apparent that most of the issues to be addressed by software systems require or involve multiple agents to represent, for instance, the decentralized nature of problems, the multiple loci of control, the multiple perspectives or the competing interests (Jennings, 2001). Hence, *Multi-Agent Systems* (MAS) – systems designed and implemented to inhabit societies of agents – will better reflect reality when considering the micro-simulation of processes in the surrounding world. Moreover, societies of agents have greater problem solving strengths than agents operating in isolation through mechanisms of *social interaction*. In single agent systems the interaction of an agent is restricted to that with its *physical* environment: it carries out actions that affect the physical environment, which reciprocally, influences its actual choice of actions. In MAS this is supplemented with a *social* environment, and for agents the purpose of interaction comes down to achieving individual objectives while managing the dependencies that arise from all agents' actions upon the physical environment and that influence future decision-making.

Social interaction can take on different forms, dependent on the specific issue that the system addresses. In cases where the agents have similar (higher-level) goals or common problems, for instance, interaction is usually described in terms of (Jennings *et al.*, 1998; Huhns and Stephens, 1999):

- *Cooperation*: working together towards a common aim.
- *Coordination*: organizing problem-solving activity in such a way that harmful interactions are avoided or beneficial interactions are exploited.
- *Negotiation*: coming to an agreement that is acceptable to all the parties involved.

For the design and implementation of MAS, the issue of interaction generally implies “determining who does what, when, by what means, in what way and with whom; to solving, in other words, the different sub-problems which make up collaboration by task distribution, coordination of actions and resolution of conflicts” (Ferber, 1999, pp. 52).

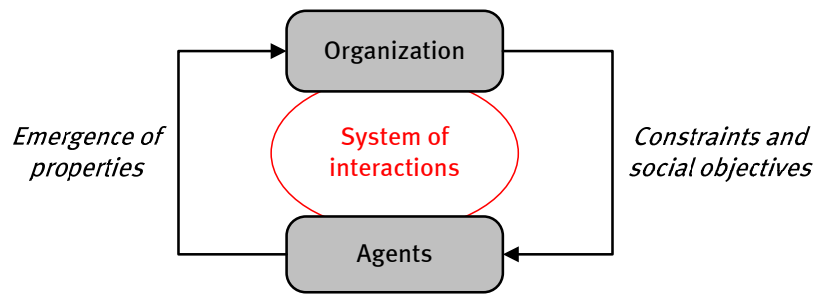


Figure 3-1. The mindset of a multi-agent system (source: Ferber, 1999)

Interactions within MAS are typically founded in some *organizational context* that defines the nature of the relationships between the agents (Jennings, 2001). For instance, agents can act in a flat organization as peers working together in a team, or in a more hierarchical way as managers and operators. Similar to real (human) organizations, MAS can be organized in groups and subgroups with possible overlapping memberships. Moreover, they are not necessarily static but can be subject to evolving relationships, especially within the frame of realistic, open systems (or organizations). Hence, the design and implementation of MAS require the definition of roles, behaviour expectations, and authority relations, or – in terms of structure – the definition of the pattern of information and control relations that exist among agents and the distribution of problem-solving capabilities among them (Sycara, 1998). In summary, the three key concepts defining MAS are agents, interactions and organizations (Figure 3-1).

3.2 MULTI-AGENT APPLICATIONS

As stated earlier, multi-agent technology is finding various applications across many domains. Generally, several categories of applications can be distinguished (e.g., Ferber, 1999). For the purpose of planning decision support three categories are of main interest (e.g., Ferrand, 1996):

- *User assistance*: having purely computing (i.e., virtual) agents, with either different or similar skills, cooperate or coordinate in order to accomplish tasks on behalf of users or to help users in doing so.
- *Micro-simulation*: analyzing the properties of theoretical models of the surrounding world in an attempt to explain or forecast collective phenomena emerging from the interaction of individual behaviours of virtual agents.

- *Software design*: using agents to develop computing systems capable of evolving through the interaction, adaptation and reproduction of virtual agents.

The characteristic that all three categories have in common is the notion of having a collection of agents who together achieve a set of tasks or goals in a largely non-deterministic environment, while each agent represents a specific set of problem solving knowledge and skills. The intention of MAS is to coordinate the skills, knowledge, plans and experience of different agents to pursue a common high-level system goal. Complexity and diversity is handled by *modularity*, i.e., the physical and/or mental distribution of problem solving knowledge and capabilities. The way in which this distribution is arranged depends on the types of problems to solve. Together with the exact knowledge and capabilities of agents, this aspect determines the main differences between the three categories of application.

3.2.1 Agents for user assistance

When serving the purpose of user assistance, agents intend to solve problems on behalf of the user. Because such problems are likely to be too complex for any agent to perform in isolation, the total expertise required to solve the problem is usually distributed among multiple agents, each specialized – in terms of their representation and problem-solving paradigm – at solving a particular aspect of it. In order to solve the complete problem, the agents must cooperate to ensure that interdependencies are properly managed. This situation can thus be denoted as “cooperation between specialists” (Ferber, 1999, pp. 32). The decomposition of the overall problem into a number of smaller and simpler components – easier to develop and maintain – allows each agent to employ the most appropriate paradigm for solving its particular problem, rather than being forced to adopt a common uniform approach that represents a compromise for the entire system, but which is not optimal for any of its subparts (Jennings and Wooldridge, 1998).

Because of the strong emphasis on cooperation (with other agents) these agents are often denoted as *collaborative agents*, while having the typical characteristics of being static, large, and coarse-grained (Nwana and Ndumu, 1998). The way in which they communicate with the user can be either direct (‘face to face’) or through intermediate agents that forward user tasks. These intermediates – usually denoted as *interface agents* – represent another type of user assisting agents. They offer active assistance in the user

interface by providing alternative views to a problem, context-sensitive suggestions and critics, learning and predicting the needs of the user, and/or automating repetitive or semi-repetitive procedures (Angehrn and Dutta, 1998; Lieberman, 1998). Their primary concern is the interaction between system (component) and user, and frequently include an animated interface tool that can engage the user in an interaction, e.g., aimed at eliciting and solving user goals (Hendler, 1999). In order to improve or ease the interaction, interface agents tend to hide the complexity of difficult tasks, while they can also train and teach the user, or help different users collaborate (Maes, 1994). An interesting aspect to the notion of interface agents is that they can be applied to ‘wrap’ valuable legacy software for purposes of enhancing its use or incorporating it into larger systems (Lieberman, 1998; Jennings, 2001). The most important aspect of interface agents, however, is their learning capability as this determines their competence and gain of trust from users. This emphasis on learning, together with their common operation as individuals, clearly distinguishes them from collaborative agents.

3.2.2 Agents for micro-simulation

A different approach to the use of multi-agent technology is to have agents represent members of a population inhabiting an environment for the purpose of studying how individual (i.e., micro) behaviours generate aggregate (i.e., macro) regularities from the bottom up (e.g., Epstein, 1999). This brings about very natural instruments to anticipate on trends in the environment through monitoring and early warning as well as to predict and value the short-term and long-term consequences of implementing certain policy measures – i.e., modifications in environmental conditions – in attempts to either encourage or discourage those trends. Or, as Ferber (1999, pp. 36) stated:

“MAS bring a radically new solution to the very concept of modelling and simulation in environmental sciences, by offering the possibility of directly representing individuals, their behaviour and their interactions (...) it is thus possible to represent a phenomenon as the fruit of the interactions of an assembly of agents with their own operational autonomy.”

As opposed to collaborative agents (section 3.2.1), who interact because of sharing a common higher-level goal, agents in a micro-simulation interact because of sharing the same environment. To take a car traffic simulation as an example, motorists share the road network as the environment in which they behave. Besides the flexibility of his or

her planned activity schedule and the characteristics of the network, the decisions of an agent (motorist) regarding use of the road network will be based on his or her (aggregated) expectations regarding the use of the network by others. So to say, the agents will be primarily driven by explicit personal goals rather than common goals (that could still exist implicitly). The interaction mechanism underlying micro-simulations is thus based on mainly competitive instead of cooperative (or collaborative) considerations. Furthermore, agents in micro-simulations mostly appear in much larger numbers than collaborative agents do, while they are also smaller sized in the sense of the number of rules used to capture their behaviour.

3.2.3 Agents for software design

Both in appearance and operation, agents do not necessarily need to be noticeable to the user; they can also fulfil important tasks at the background to serve the purpose of software design. Whereas user assistance agents solve problems that originate from the user and micro-simulation agents solve their own individual problems, agents for software design facilitate or optimise the operation and use of the system in which they reside. Two general applications of these agents are distinguishable. On the one hand, there are software design agents that facilitate the interaction between, and the operation of, other agents within MAS. Similar to how interface agents (section 3.2.1) play the role of intermediates between user and system (or other agents), so-called *middle agents* (e.g., Sycara, 1998) are entities to which other agents advertise their capabilities in order to help any arbitrary agent in finding agents with certain capabilities. Other frequently used names for this type of agents are facilitators, mediators, brokers, or yellow pages agents (Flores-Mendez, 1999). Middle agents are especially useful in open systems where organizations (section 3.1.2) tend to vary over time. Similar tasks are performed by *control agents*, who primarily provide control services to other agents to help them function together, and *translation agents*, who remove problems of incompatibilities between agents or systems with different data standards (Hendler, 1999).

On the other hand, there are software design agents that operate at a level of higher abstraction, serving the purpose of coupling MAS with various kinds of software components and information sources. Multi-agent technology offers a flexible way to interconnect components and to have these components interact effectively. *Agent wrappers* can be built around existing components to ease their use and to enable

interoperability with other components (e.g. Lieberman, 1998). Furthermore, multi-agent technology can provide solutions to efficiently use distributed information sources (Sycara, 1998), which implies the use of *mobile agents* (Nwana and Ndumu, 1998), i.e., agents that are capable of roaming wide area networks.

3.3 Multi-agents and planning support

According to Jennings and Wooldridge (1998), multi-agent technology promises to solve problems with respect to open, complex, and ubiquitous computing systems. Planning Support Systems (PSS) are demanded to operate in a domain that is characterized by a multiplicity of complexities (section 1.1.3). At the same time, PSS ought to be open for the incorporation of new models, tools and techniques when these are developed external to the system. Besides this, the broad span of urban planning – across stages, scales, contexts, disciplines, and so on – advocates functionality to provide dedicated sets of tools for the specific project at hand (section 2.2.2). Finally, the tasks to be performed by the user – the urban planner – are numerous and extensive and ask for more intelligent and cooperative software. Through the techniques and metaphors it offers, multi-agent technology promises to be the versatile means with which problems along all these frontiers can be addressed.

3.3.1 Opportunities

For GIS in general, the profile of users is rapidly changing from ‘computer specialists’ to the ‘application-oriented users’ with little or no familiarity with the idiosyncrasies of computer systems (Spaccapietra, 2001). Hence, there is a great demand for friendlier and simpler human-computer interfaces that support application-oriented interactions. Given the fact that the user groups of PSS – planners and a variety of stakeholders – are all application-oriented, this demand also holds for any PSS that seeks successful deployment. Consequently, PSS interfaces ought to be flexible in order to account for the typical interactions with casual users that can be rather abrupt and unstructured. As described before, multi-agent technology provides the means – *interface agents* – to change the system from a dumb receptor of task descriptions to an entity that cooperate with the user to achieve their goal (Jennings and Wooldridge, 1998). For instance, Tang *et al.* (2001) developed an agent-based GIS that includes an interface agent to

interactively assist users with forming geographic queries. Such interface agents can be structurally provided by means of an organization (section 3.1.2), enabling them to cope with the possibly unstructured user requests.

Klosterman (2001) claimed that the difficulty of realizing the ideal PSS is not so much the development of the required analytic and display modules as is the integration of all these modules, and especially the development of standards for data storage, access and interchange. Some research studies can be found that illustrate the potential of multi-agent technology to serve such software design and operation issues. For instance, Sengupta and Bennett (2003) presented agents that emulate the behaviour of GIS analysts to retrieve spatial data and analytical models from the Internet, and to transform such data in order to meet the input requirements of models through the use of GIS software. Tang *et al.* (2001) addressed the integration of heterogeneous data sources by means of a broker-agent architecture (section 3.2.3). Rodrigues *et al.* (1997) developed a multi-agent concept for a system that is able to dynamically evolve as the requirements and data associated with the system change. The incorporated agents are capable of reasoning over representations of space, while intended to give support with locating and retrieving spatial information in large networks, to facilitate GIS user interfaces, to implement improved spatial tasks, and to create interfaces between GIS and specific software packages (see also Rodrigues and Raper, 1999). These studies give good reasons to believe that multi-agent technology can be the guiding principle for implementing and following standards for data management and thus for increasing PSS interoperability.

With concern to the core functions of PSS – providing the means for both generating and evaluating alternative plans (section 2.2.2) – it is obvious that planning support tools need to have a geographical context in order to deal with aspects of location, adjacency and accessibility. Successful attempts have been made with concern to developing tools that integrate multi-agents and GIS. However, the attention has been mostly restricted to the development of agent-based micro-simulation tools that visualize and analyse time-evolving processes in an urban environment by using agents to model the behaviour of individuals, e.g. regarding shopping (Koch, 2000), recreation (Bishop & Gimblett, 2000) or patterns of daily activities.⁴ The agents are submerged into a virtual urban environment and interact with each other and their environment. The use of multi-agents for micro-

⁴ See Parker *et al.* (2003) for an extensive review of agent-based micro-simulation tools for land use and land cover change.

simulation purposes has already resulted in valuable and promising tools for predicting and evaluating the effects of different policy scenarios and plan alternatives as part of the *Choice* phase in urban plan development (section 1.1.2).

Concurrently, from literature it seems to be far less evident to use multi-agent technology for decision support in the preceding *Design* phase of urban plan development in which the alternatives are generated. Such a design-oriented application would opt for an agent representation of the various specialists involved in that phase, implying a fundamentally different approach to the implementation and operation of agents. Whereas in micro-simulations agents are situated *inside* the environment and act *within* that environment, in applications where they represent specialists they will actually be situated *outside* the environment and only act *upon* it (cf., a group of experts gathering around a map representation of the environment). Despite the lack of attempts to utilize multi-agent technology for this designing part of urban plan development, it would be entirely complementary to micro-simulation tools, providing the means to quickly generate sets of alternative plans, while offering insight into the effects of decision-makers' trade-offs during the process. It brings about opportunities to shed light on the problem-solving side of planning as well, in addition to the study of urban phenomena.

Some examples do exist that support the idea that multi-agent technology is equally suitable for supporting group decision-making processes like alternative plan generation. In Ligtenberg *et al.* (2001) and Ligtenberg *et al.* (2004) an application, leaning on cellular automata techniques, is discussed in which agents are applied to simulate multi-actor spatial planning. In this application agents have the same overall objective – locating space for urbanization – but as they represent different interest groups – e.g., planning authorities, farmers organizations and environmentalists – they have different preferences. Based on a set of predefined scenarios the agents negotiate and vote in a stepwise fashion to develop plans. Zamenopoulos and Alexiou (2003) presented a tool concept in which plans are generated by a group of distributed knowledge sources (humans or agents), mainly focussing on how to arrange it as an interactive process of continuous learning and adaptation. Ferrand (1996) applied multi-agent concepts to multi-actor spatial planning in three different applications and within the social and political context of the process as it appears in France. The first application uses the agent concept to represent physical objects (e.g., electricity masts) that solve location problems together; the second enables negotiation between distributed (human) actors by attaching an assisting agent to each of them that communicate with each other; the third simulates a spatial negotiation process

between actors that are implemented as agents.

Given the vast amount of tools that is likely to be necessary to have PSS facilitate the whole process of urban plan development, functionality is required to assist users in selecting appropriate tools and to manage the collections of tools. Yeh and Qiao (2004a, 2004b) proposed a knowledge-based PSS following a component-based software development approach. Their system provides assistance to users and model developers to build new models or to select predefined models from an existing model library for their problems. Multi-agent technology is used to handle model knowledge, to facilitate knowledge-based model selection and incremental model development, and to incorporate models and knowledge rules in a problem-solving process.

3.3.2 Multi-agents and the urban system

In an exemplary effort to define a suitable and comprehensive PSS structure, Hopkins (1999) suggested to build on elements of both geographic modelling and planning according to an object-oriented approach and, as such, to distinguish actors, activities, flows, investments, facilities, regulations, rights, issues, forces, opportunities and constraints. This apt suggestion acknowledges the fact that the urban environment is the result of the actions and reactions of various parties that attempt to achieve particular personal goals within the boundaries set by the institutional context and cultural and political norms. For example, in order to achieve particular desires individuals and households will try to find a house that provides them maximum utility, within constraints set by availability, awareness, affordability, and so on. In reaction to this latent demand, the housing industry will try to offer a housing stock which satisfies this demand, subject to the limitations set by the government in terms of, for example, housing norms, building permits, and land use plans. The role of planning authorities in this regard is to coordinate this process; to set particular goals, norms and guidelines approved through some democratic process; and to try to improve the quality and performance of the system by taking a more active role in stimulating particular developments or even taking part in investments themselves. Because this process is public in essence, planning authorities are, at least officially, guided by a set of high-level goals or principles, which reflect how political parties cope with religious, economic, social and cultural issues pertinent for their country, region and city.

The actual evolution of the urban environment is determined by the relative strength

of these various influences and the outcome of the multiplicity of individuals' actions and reactions at the aggregate level. Some change can be best viewed as being monotonic. For example, the aging of housing stock is such a process. Other change comes about as the balance of uncoordinated actions of individuals. For example, the traffic situation during morning commute hours is the result of the individual actions of all those going to work. Yet, other change is more complex in that it is the result of action-reaction chains. For example, changes in the retailing structure can be viewed as the result of actions of household, related to actions of retailers, which in turn act or react to actions of their competitors and changes in aggregate demand. In case of planning authorities, change is actively planned, and often many decisions are coordinated, negotiated, and so on.

Evidently, PSS need to comply with the quintessence of planning, which is equally well in the underlying complex network of interdependencies and interactions between various autonomous parties involved in the process as in the physical environment. Given the fact that the actual strengths of multi-agent technology are its anthropomorphic nature and complex problem solving capabilities, it is reasonable to expect that PSS will meet this fundamental requirement when multi-agents form the rationale of the overall framework. From the previously described settings of urban plan development, the following distinction in main layers of agents is deducible:

- the urban planning team
- individuals and households
- firms and organizations
- the natural environment
- the physical environment
- the institutional context
- the political environment

The 'urban planning team', composed of land use specialists, fulfils a central role in the urban plan development process from the very moment that it is triggered by newly defined higher-level system goals coming from the 'institutional context' and 'political environment'. The team then initiates and manages the process stage of *alternative plan generation* through a democratic process in which 'individuals and households' as well as 'firms and organizations' take part. The 'natural environment', together with the 'physical environment', determines the situational opportunities and constraints that are inputs to the alternative plan generation procedure.

In the subsequent process stage of *alternative plan evaluation*, the performance of each alternative plan is judged by means of simulating the evolution of the urban environment as a product of the interactions between all the parties involved. In this micro-simulation setting ‘individuals and households’ as well as ‘firms and organizations’ are the main players, who interact with each other and the environment. Their choices and decisions affect the conditions of the ‘natural environment’ and ‘physical environment’, and vice versa. In this evolutionary process the ‘urban planning team’ can have a guiding role by stimulating certain developments and preventing others, which will provide the system user with insights into the necessity of guidance and the effectiveness of different types or degrees of guidance. The team’s role concerns relatively short-term guidance in comparison with the long-term monitoring role of the ‘institutional context’ and the ‘political environment’.

3.4 CONCLUSIONS AND DISCUSSION

To date, the attempts to design or implement PSS from a multi-agent perspective have been largely restricted to micro-simulation applications. The discussion above, however, indicates that multi-agent technology can be utilized for various purposes at various levels of PSS, supported by the fact that many researchers in other areas have become convinced of the contributions it can make to the capabilities of computer systems. On the one hand, agents have the clear potential to simplify our use of PSS by hiding complexities, by offering guidance and assistance, or by taking over tedious tasks. On the other hand, they can enhance the systems by means of their ability to tackle complex problems through modularity and flexibility. It is undisputable that PSS will benefit from the fact that a multi-agent representation allows one to bring together the knowledge, methods, models and software components that have been developed over the last decade. Moreover, its capabilities to provide an intelligent formal framework allows for easier management of PSS, as new knowledge, models, cases and so on can be more easily added to update the system.

Along with the promises of multi-agent technology, however, also comes potential danger (Norman, 1994). This especially concerns the risk of users losing the sense of control, which comes in a close relation with users’ trust in delegating tasks to agents (Jennings and Wooldridge, 1998). In urban planning practice it is a well-known problem that planners are rather distrustful towards computational support (e.g., Geertman, 2001).

The multifunctional problem-solving power of multi-agent technology in conjunction with the overly anthropomorphic nature might be able to turn planners to accepting agent-based PSS, under the strict condition that their sense of control is clearly maintained. It is probably best to let users have continuous control over the system, i.e., to allow them to intervene at any time, to decline agents' suggestions, to overrule their decisions, and so on. This consideration should be taken into account already in the stage of designing the system.

In order to see the implications of taking multi-agents as the guiding principle for PSS, the following chapter blends the supposed contributions of the technology to the field of planning support into a comprehensive conceptual framework for a multi-agent planning support system that, as the focus of this research, is aimed at supporting land use planning.

PART II
PRINCIPLES

4 FRAMEWORK FOR A MULTI-AGENT PLANNING SUPPORT SYSTEM

This chapter illustrates the implications of fully exploiting the opportunities of multi-agent technology within the context of PSS. While focussing on application in land use planning, it introduces a multi-agent concept that could operate as the structuring principle for a PSS framework. This is followed by a further elaboration on the types of agents defined within this concept. Special attention is given to the agent-based structure of the knowledge component to be added to the framework.

4.1 INTRODUCTION

The land use plan is the main instrument in Dutch urban planning (section 1.1.1) and provides regulations, in both quantitative and qualitative terms, for the types of permissible land uses in a strictly bounded area. It can be developed in various degrees of detail, ranging from a plan that provides a general land use allocation for zones to a plan that gives additional regulations about the internal design of these zones. Although land use plans actually appear rather simple in their final state, their development takes place by means of a decision-making process that is full of complexities. Also at the local scale planners need to consult different sources of knowledge – i.e., experts – and apply various decision support models – i.e., tools – during the process in order to ensure that well-informed decisions are made that result in high quality plans. Hence, PSS that intend to support the development of land use plans ought to provide access to and offer assistance in the application of these essential resources. Moreover, they have to do so through a most understandable user interface.

The discussion in the previous chapter has given us three main reasons to believe that multi-agent technology has the potential to help PSS in fulfilling these requirements. First, the technology presents itself as a generic means for software engineering that can ease the development and management of PSS through *modularity* at different levels, from system components to agent components. Second, it provides the desired opportunities for intelligent planning support by means of the *autonomy* and *flexibility* of agents. Third, its inherent *anthropomorphy* can greatly contribute to the usability of PSS.

The following sections address the conceptual framework of a PSS named ‘*MASQUE*’ – Multi-Agent System for supporting the *Quest for Urban Excellence* – that

intends to fully exploit the versatile potential of multi-agent technology with regard to serving planning support purposes.⁵ The system focuses on land use planning and applies multi-agent concepts to deal with issues of program design, user assistance and micro-simulation. The research presented in this thesis covers the first two purposes and forms the basis of continuing research on both the sophistication of agents' reasoning mechanisms (Ma *et al.*, 2004), and the inclusion of agent-based micro-simulation for forecasting and evaluation purposes (Devisch *et al.*, 2004).

4.2 *MASQUE* SYSTEM ARCHITECTURE

Generally, the ideal system architecture of PSS is believed to consist of three internal components – referred to as *information*, *models*, and *visualization* – that users access through a user interface (Figure 4-1-a). This architecture has two major drawbacks, however. On the one hand, the successful application of PSS following this architecture highly depends on the users' ability to oversee the complete functionality of the system and comprehend how this functionality is organizationally structured. It is a common fact that valuable functionality of computer applications often remains unused because users are unaware of its availability or not fully understand its operation within the context of the system. Although training could partially compensate, the risk of computer systems not being used to their full capacity is likely to take serious forms when it comes to systems that (need to) offer a wide range of functionality, such as PSS. On the other hand, the architecture does not (explicitly) accommodate for the knowledge that intelligent planning support requires. As stated earlier, this knowledge is at best partially contained within the mechanisms of the *models* component, remaining irretrievable for users.

The *MASQUE* system architecture (Figure 4-1-b) rules out these drawbacks by having a multi-agent concept form the system's structural backbone and introducing an agent-based *knowledge* component to address planners' need for expertise on a variety of disciplines.⁶ Furthermore, based on planners' need for GIS functionality in both 2D and 3D, the *information* and *visualization* components of *MASQUE* are considered to form

⁵ The *MASQUE* framework does not yet exist entirely, as implementation has been restricted to the multi-agent model for alternative plan generation specified in the next chapter. Nevertheless, the descriptions provided in this chapter are in present tense for simplicity of language.

⁶ Further discussion follows in sections 4.3 and 4.4.

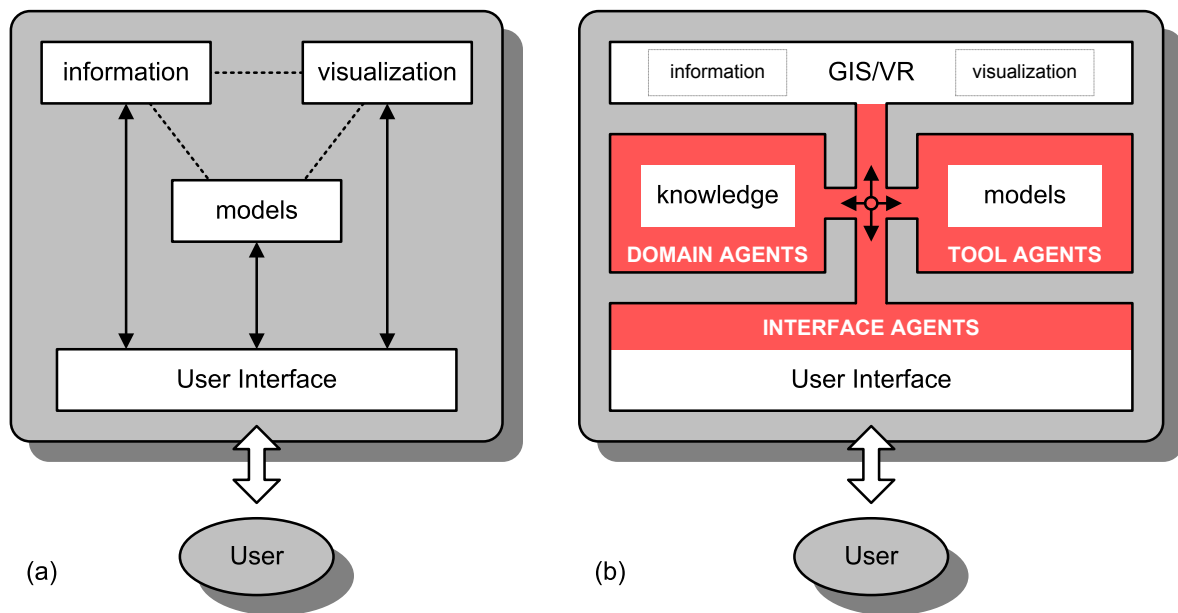


Figure 4-1. System architecture: (a) PSS in general and (b) *MASQUE*

an integrated GIS/VR component that, ideally, allows for handling temporal data, both disaggregate and aggregate data, as well as data of various resolutions, detail levels and representation styles.

For the purpose of contextualizing the system to the plan development process that it is meant to support, *MASQUE* assumes a generally applicable process structure (Figure 4-2) as can be derived from Simon (1977) and Rutledge (1971). This iterative process structure distinguishes three phases:

- An *Intelligence* phase in which the need for a plan is identified because an urban area requires expansion (new developments) or improvement (redevelopments). Subsequently, the goals and objectives are formulated and an inventory of the site is made, which are activities that feed each other until a provisional equilibrium is reached.
- A *Design* phase in which both the inventoried site and the formulated goals and objectives are analyzed for the purpose of generating a set of distinctive alternative plans, of which each plan indicates a feasible way to achieve the goals and objectives.
- A *Choice* phase in which the impacts of the alternatives are assessed by means of a forecasting procedure, followed by an evaluation aimed at selecting the plan that most efficiently and effectively achieves the goals and objectives.

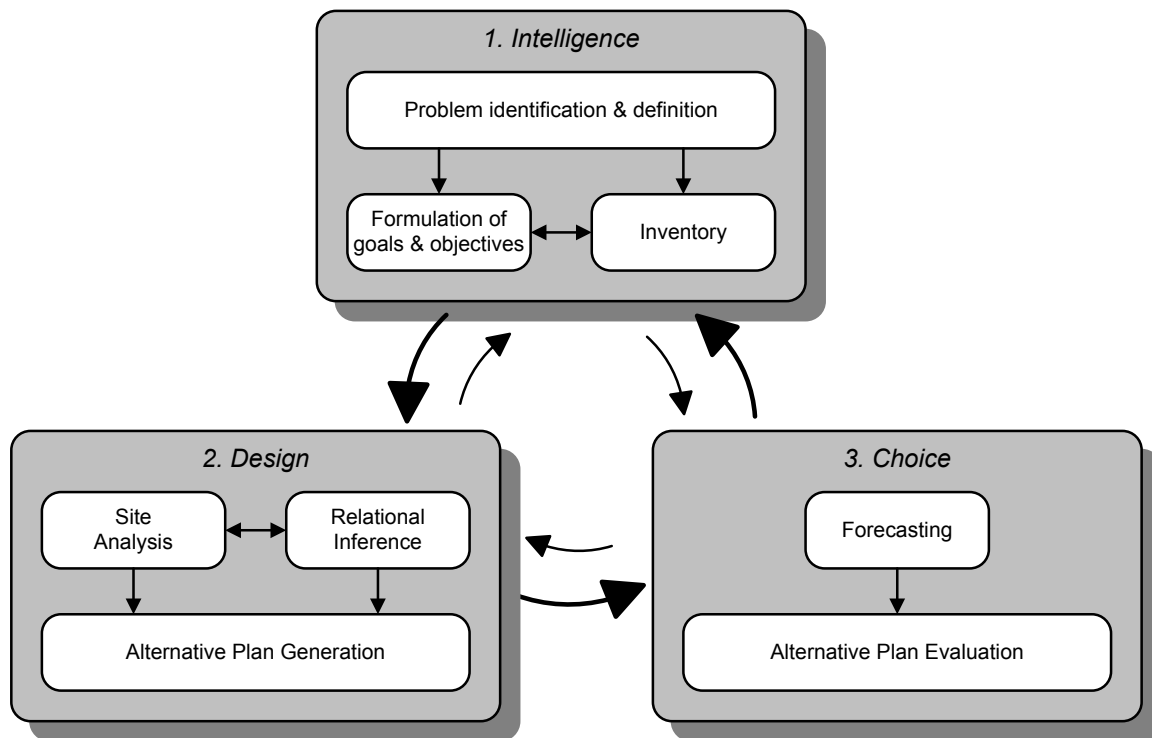


Figure 4-2. *MASQUE*: plan development process

Regarding the first phase, *MASQUE* provides functionality to make inventories of a site, i.e., tools to input both spatial and a-spatial data about the study area and its surroundings in order to build up project databases. During the process both users and agents can query these databases for information that is required to perform analysis. For the formulation of goals and objectives, the system offers support to bring together and tune all goals and objectives concerning a study area and its surroundings, stemming from exogenous sources (e.g., parties involved and hierarchically enforcing plans) and/or from other process phases. The support is aimed at extracting an overall set of goals and objectives – further referred to as a *plan program* – that reflects the requirements that each plan will have to meet. Because the goals and objectives for a project coming from different sources can be overlapping or conflicting, functionality is offered for defining different plan program scenarios.

The second phase is supported by means of tools to elaborate on the information from the inventory and plan program, and tools for subsequently generating alternative plans. In the former case, this implies that geographical analysis tools (e.g., for overlay analysis) are provided to assist users in determining the opportunities and constraints of the study area and its surroundings based on the project databases. This analysis focuses on aspects concerning the *absolute* location, which reflects the suitability and feasibility

of the land related to different uses. Additionally, assistance is provided to infer the required spatial relationships between land uses from the plan program under study, i.e., to extract any requirements regarding the mutual configuration of land uses. Relevant in this case are aspects of *relative* location, which reflects the functional relationships of land uses and the lines of travel between them. The subsequent generation of alternative plans is supported by means of a multi-agent tool, which will be discussed in detail in the following chapters. This tool deploys the domain agents that are embedded in the system's knowledge component.

In the third phase, *MASQUE* assists users in forecasting the effects of each alternative plan that has been generated. Agent-based micro-simulation tools enable studying the long-term performance of proposed environments by having a synthetic population virtually 'live' inside them. As such, users can get insight into the evolution of environments under the influence of different possible autonomous developments (i.e., scenarios of, for example, socioeconomic changes in the population or (stimulated) changes in travel behaviour). Multi-criteria evaluation techniques are offered to determine which alternative plan is performing best. If at any moment during the process problems occur (e.g., low plan performances during evaluation), the iterative nature of the process allows returning to any of the previous plan development activities to make refinements.

In essence, the process is a complex network of decision-making tasks that, in practice, are performed by (groups of) people with various expertise and skills. It is based on this notion that the system's agent organization is structured and mapped onto the process. There are agents active in the system that guide and assist users throughout (a part of) the process. At appropriate times, when certain tasks have to be performed, they introduce other agents that are able to perform those tasks, either individually or in groups, on behalf of users.

4.3 AGENT TYPOLOGY

The conceptual framework of *MASQUE* distinguishes three types of agents, each dedicated to serve a specific purpose in system operation (Figure 4-1). As the agents are linked to specific system components, data flows in the system take the form of agent communication, while the interaction between agent types enables a smooth integration of system functionality. Furthermore, the consistent definition of agents as human-like specialists in a particular field improves users' understandability.

4.3.1 Agents for human-computer interaction

The usability of PSS depends highly on the technologies they incorporate. For instance, including GIS technology will make a system more advanced but, as it brings in rather specialized functionality, the system will also become more demanding towards users' geographical skills. The same holds for adding Virtual Reality technology to PSS, which would complicate system usage because of requiring skills, for instance, to use devices for navigation in virtual environments. On the other hand, the usability of PSS is strongly influenced by the particularities and scope of the supported planning phase(s). The more complex and diverse the supported problems are, the more functionality is required and the more skills users need to have. Hence, the apparent risk is that the amount and complexity of system functionality reaches a level at which the system becomes too hard to work with for the intended user groups. In order to reduce the experienced complexity of a system and, thus, to improve its usability, attention must be paid to the organization of the system, including its user interface. As mentioned before, the idiosyncratic complexities of planning problems opt for user guidance, which should be more intensive as users lack more of the required knowledge and/or skills concerning the system's underlying technologies, its field of application, the process it supports, or the tools and expertise it offers.

Since humans feel most natural and accustomed to interaction with other humans rather than machines, the anthropomorphic nature of multi-agent technology makes it pre-eminently suitable for improving the user interface of PSS by means of intelligent software agents that have a human-like way of interaction and support users in dealing with both system and context. In the *MASQUE* framework, **Interface Agents** (Table 4-1) monitor the behaviour of users with respect to the system's underlying technologies and field of application. Consequently, these agents gain insight into users' abilities to deal with these issues, which enables them to react on occurring problems by releasing appropriate suggestions, warnings, ideas, etc. Like their name indicates, interface agents are part of the system's user interface, implying that they are implemented as visible companions or assistants to users.

In total, five of them are defined within the envisioned framework of which two are constantly active from start to end when using the system: (i) the '*system guide*' who is concerned with offering system-wide guidance regarding technology-related issues and promotes the use of the provided functionalities by calling them to the attention of users

Table 4-1. *MASQUE* Interface Agents

Agent	System guide	Process guide	Phase guides
Quantity	One	One	One per planning phase (total 3)
Activation	System start-up	System start-up	Start of phase
Deactivation	System exit	System exit	End of phase
Knowledge	System organization; System functionality	Plan development process	Phase-related procedures
Input	Request of user/agent; Signals of user progress	Request of user/agent; Signals of user progress	Request of user/agent; Signals of user progress
Capabilities	Monitor user in system; Promote functionality; Refer to any other agent	Monitor user in process; Scenario management (create, adjust, remove); Refer to <i>phase guides</i> for phase-related details	Monitor user in phase; Explain purpose and use of phase-related operations; Refer to <i>tool managers</i> and <i>domain managers</i>
Output	Explanation of system; Introduction of agents	Explanation of process; Introduction of agents	Explanation of phase; Introduction of agents

and explaining their purpose and usage; and (ii) the ‘*process guide*’ who provides support regarding the organizational issues of the plan development process and manages project-related information, such as scenarios (i.e., alternative plan programs) that are developed for a project. The remaining three interface agents are named ‘*phase guides*’ and engaged in offering user guidance within a specific planning phase (intelligence, design, choice). A *phase guide* is operational from the moment the user enters the corresponding phase (on own initiative or guided there by the *process guide*) until he or she leaves that phase. A *phase guide* documents the decisions that are made as part of the activities within the phase, and forwards this information to the *process guide* in order to have that agent update the project information. Whether on their own initiative or on the user’s behalf, *phase guides* can contact both *Tool Agents* (section 4.3.2) and *Domain Agents* (section 4.3.3), who can provide them with information regarding available (kinds of) tools and discipline-related expertise, respectively.

4.3.2 Agents for model use and management

The use of planning support tools in a system generally assumes that the user knows

which tools are available, what each tool can be used for, and how the different tools relate to one another. Being faced with a problem that needs some analysis to be solved, the user needs to select the (most) appropriate tool, whether or not after searching the system’s manual or help function. When applying a tool, support is received from a help wizard at best. In this very common way of PSS operation, choosing the right tool may not be easy for users, as the number of available tools can be large. Similarly, the application of tools may be hampered by the methodological complexities underpinning many of them.

MASQUE offers support in model use and management by means of providing a hierarchically two-layered organization of **Tool Agents** (Table 4-2) that is embedded in its *models* component. The top layer consists of three agents (*tool managers*) – each one linked to a planning phase – that have knowledge about the tools that can be applied to perform the necessary tasks within the corresponding phase, including generally applicable tools for tasks such as summary or spatial statistics. Hence, a *tool manager* assists in searching and organizing tools, which includes their addition, construction, customization, and evaluation [cf., *model management* (Yeh and Qiao, 2004a, 2004b)]. In other words, these agents give support for expanding or modifying the system’s collection of tools. When none of the available tools can directly solve the problem at hand, the

Table 4-2. *MASQUE* Tool Agents

Agent	Tool manager	Tool operator
Quantity	One per planning phase (total 3)	One per tool
Activation	Start of phase	Call by tool manager
Deactivation	End of phase	End of task
Knowledge	Availability, general purpose, construction and operation of tools for use in planning phase	Tool idiosyncrasies (specific use of the tool)
Input	Request of user/agent	Request of <i>tool manager</i>
Capabilities	Search tools to match request; Tool management (add, create, customize, evaluate)	Promote tool characteristics; Assist with tool application; Detect errors in tool application
Output	Search results (ranked list of tools); Refer to <i>tool operator(s)</i>	Explanation of tool; Tool output (maps, tables, graphs, etc.)

agents can provide support in constructing new tools by re-using, adjusting and re-assembling components of existing tools to meet the requirements.

Underneath the layer of *tool managers*, a second layer is distinguished that consists of agents named '*tool operators*' who are responsible for separate tools. Technically, these agents take the form of tool "wrappers" (e.g., Sycara, 1998), which means they form shells around tools – either legacy or newly developed – and thus ease the interaction between users and those tools. A *tool operator* knows the characteristics of his or her tool (purpose, possibilities, limitations, input requirements, processing aspects, output options, and so on). This information is open to *tool managers* in order to support their searches for (components of) tools, whether or not on behalf of the user. A *tool operator* becomes operational when his or her tool is selected for actual use. The agent is then introduced to the user and will offer full support on applying the tool within the context of the current planning phase, while anticipating on processing errors. The availability of a tool and its *tool operator* in a particular planning phase depends on the applicability of that tool in that phase, which is determined at the time of registering the tool to the system. When tools are added to the system, or when new tools are assembled from existing components, new *tool operators* are created.

Since the *models* component of *MASQUE* is intended as a repository in which all kinds of tools to support the decision-making in urban planning are brought together in an integrated fashion, the system is also supposed to contain agent-based micro-simulation models (e.g. Devisch *et al.*, 2004). The collections (or populations) of agents that will inhabit such tools could be considered to form an additional third layer of *Tool Agents*. This level of detail, however, is outside the scope of the current research.

4.3.3 Agents for use of specialized knowledge

During the plan development process, the urban planner is dealing with a wide range of disciplines and will often run into problems that ask for specific expertise with regard to such disciplines. The planner will then have to look for persons having the required expertise, and ask them for help, advice, an opinion, or a contribution. Since many problems in land use planning refer to multiple disciplines, intensive and time-consuming interaction between several experts is required to find proper solutions. Each expert will have to contribute to the solution of the entire problem by solving a certain piece of it, while ensuring that his or her own solution is not conflicting with the solutions of others.

Table 4-3. IMRO land use classification

Land use (or domain)	Description
Housing	Residential areas
Business	Working areas (e.g. offices, agriculture, industry)
Transportation	Traffic and transportation facilities (e.g. roads, parking)
Services	Medical, social, educational, shopping, cultural, and sport facilities
Green space	Green areas
Recreation	Areas for leisure (e.g. day recreation, theme park, holiday resort)
Technical infrastructure	Pipelines, cables, and radial connections
Hydraulic constructions	Constructions for water management

This complex group activity involving participants with heterogeneous skills is often referred to as *collaborative design* (Edmonds *et al.*, 1994).

In order to answer the need for specialized knowledge in the plan development process, *MASQUE* introduces *Domain Agents* who are embedded in the system's *knowledge* component (section 4.4). These agents are specialists on particular land uses, i.e., they have knowledge about aspects of location, quantity, quality and implementation of their land use. In total, eight domains or land uses are distinguished (Table 4-3), following the IMRO data model (Ravi, 2000) for exchanging plan data between urban planning agencies in the Netherlands.

Two types of *Domain Agents* are distinguished (Table 4-4). First, there are '*domain managers*' who are members of the 'urban planning team' (section 3.3.2) that is headed by the user who plays the role of urban planner. Within the team a *domain manager's* general purpose is to give land use specific support to the user throughout the plan development process. This implies that, besides providing solutions to instant, clearly specified and partial problems, the *domain managers* as a team can independently develop plans if the user, for instance, would like to have a quick and firsthand impression of the solution space or the opportunities and constraints for a project. If the team receives such a request, every *domain manager* will use his or her capability to initiate the development of plan proposals from a personal viewpoint, whether or not requesting other *domain managers* to contribute. Furthermore, a *domain manager* takes part in the subsequent evaluation of plan proposals.

Second, for every domain one or more '*domain consultants*' are distinguished who are also specialized in the corresponding land use. In contrast to *domain managers*, however, they are not part of the urban planning team but represent the consultancy firms

Table 4-4. *MASQUE* Domain Agents

Agent	Domain manager	Domain consultant
Quantity	One per land use (total 8)	One or more per land use
Activation	Call by user or agent	Call by user or agent
Deactivation	End of task	End of task
Knowledge	Aspects of location, quantity, quality and implementation of a land use	Aspects of location, quantity, quality and implementation of a land use
Input	Request of user or agent	Request of user or <i>domain manager</i>
Capabilities	Process queries; Initiate plan proposals; Contribute to plan proposals; Evaluate plan proposals	Contribute to plan proposals
Output	Answer to query; Partial plan proposals; Full plan proposals; Plan performance scores	Partial plan proposals

that are often involved in the plan development process. Only on request, *domain consultants* provide the user, or a *domain manager*, with land use specific contributions that take the form of partial (i.e., thematic) plan proposals. When two or more consultants are available for a domain, the client – the user or a *domain manager* – chooses the best solutions from those returned. Although consultants in the same domain can all perform the same tasks, essentially they apply their own design and decision principles, i.e. rules they defined based on what they learned from previous projects. Most of these principles will be location-dependent conditional rules and can differ fundamentally or show very subtle differences when only some parameter settings are dissimilar.

The involvement of *Domain Agents* – in particular *domain managers* – in planning phases depends on the activities to be conducted. Within the *Intelligence* phase, these agents can become needed for giving advice on *inventory* matters, like definition of categories when storing information about certain (aspects of) land uses, or identification of different kinds of problems in the current urban environment. Concerning the *plan program* they can assist with interpreting and encoding goals and objectives, as well as with identifying fundamental (in)compatibilities between goals and objectives that are referring to different (aspects of) land uses. During the *Design* phase, the same agents can

be consulted for *relationship inference*, making use of their knowledge about aspects concerning the *relative* location of land uses. As such, they can assist with identifying synergies and conflicts occurring between land uses, addressing issues of adjacency and accessibility. For performing *site analysis* the agents can offer support by means of their knowledge about aspects concerning *absolute* location of land uses; they can assist in determining the suitability of certain land with respect to their land uses. Their role in generating alternative plans has already been mentioned. Finally, when the process is in its *Choice* phase, they may share their knowledge about the quantitative and qualitative aspects of land uses for determining the physical feasibility of establishing certain relationships in an alternative plan, which may result in the agents giving advice to users with respect to forecasting and evaluation, e.g., to reconsider the relative importance of land use relationships.

4.4 *MASQUE* KNOWLEDGE COMPONENT

In comparison with contemporary PSS, the framework of *MASQUE* incorporates an additional *knowledge* component that interconnects the system to the plan development process by means of a group of *Domain Agents* (section 4.3.3) that assist and advice the user in a most recognizable way as their specification closely resembles that of the human actors involved in planning practice. Practically, the interconnection between system and process implies that these agents are working according to the same three-phased process structure of plan development as the user. However, the component is not only meant to contextualize the system, it is just as well to serve the purpose of providing users with intelligent support and solutions. As indicated, the very core of planning is formed by the formulation of alternatives – at the level of goals and objectives as well as at the level of plans – and the subsequent procedure of evaluation through forecasting. The *MASQUE knowledge* component is facilitating these elemental planning tasks by means of utilizing the multi-agent concepts of modularity, autonomy, flexibility and anthropomorphy to construct a team of human-like experts – *Domain Agents* – that enables incorporating and synergistically utilizing knowledge of multiple disciplines in planning decision-making in a straightforward and gradual fashion.

As described, the user plays the role of urban planner and is in charge of the urban planning team that consists of software agents, to be precise, *domain managers* (section 4.3.3). This role gives users the authority to define preconditions in advance, to enforce

preferences, and to overrule team members (i.e., agents) for reasons of common goods. The team as a whole is responsible for providing solutions to instant partial problems and for generating sets of alternative plans based on a situation and plan program given. In the latter case, all team members will take the opportunity to initiate plan proposals, which are plans suggested for the study area that take into account the demands for all land uses. An initiating *domain manager* has three options available for integrating the demands for a land use other than his or her own into a plan proposal (Figure 4-3):

- Applying system-provided standards and rules-of-thumb related to the other land use and personally perform the task. This has the advantage that the initiator can make the plan proposal most suitable regarding his or her own goals and objectives. However, it also brings the risk that, at the time of evaluation, the team member responsible for the other land use will judge the completed proposal to perform poorly.
- Delegating the task to a *domain consultant* (section 4.3.3) related to the other land use who will work within given guidelines. This guarantees that the allocation of the land use in question is at least feasible from the perspective of that domain, while the concessions to be made by the initiator are likely to be moderate. However, the final performance of the plan proposal can still be relatively low, when the enforced guidelines have asked for large concessions to the other land use.
- Cooperating with the team member who is an expert in the other land

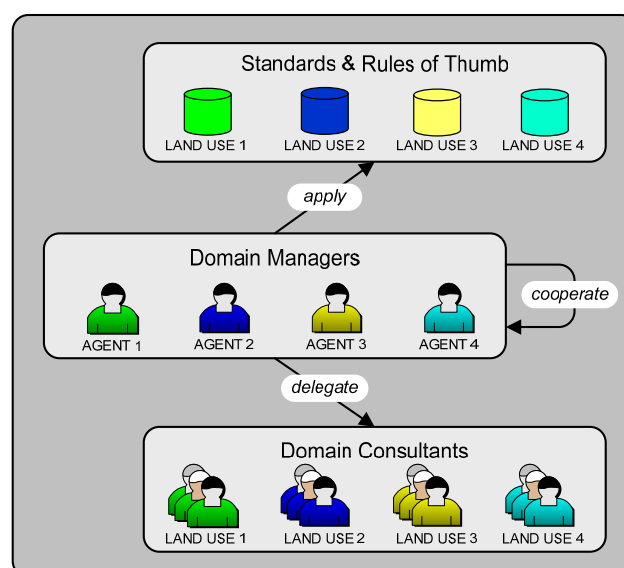


Figure 4-3. *MASQUE*: knowledge component

use, which is the safest way of acting with concern to the plan proposal's final performance. However, as it requires obtaining the commitment of the other agent, it may demand more or larger concessions to be made by the initiator with concern to his or her own land use.

The option an initiating agent will choose depends on the extensiveness and complexity of the tasks per land use, as formulated in the *plan program*, and that agent's beliefs about the necessity to cooperate with others in the urban planning team. Considering the collaborative nature of the urban plan development process, however, all *domain managers* will show a preference for cooperation and a willingness to accept a lower personal plan performance in the interests of reaching a joint solution or establishing a higher overall plan performance. This means that agents always first try the option of cooperation. Only in cases that this approach is not working out successfully, they will turn to another option. The user has the same three options available for developing alternative plans within the system.

4.5 CONCLUSIONS

This chapter has introduced a conceptual framework for a PSS that is believed to address the complications experienced in urban plan development by means of using multi-agents as a structuring system backbone. The utilization of this technology is expected to make the framework intelligent and flexible in terms of operation, functionality, usability, extensibility and adaptability.

In the previous section, we have started to focus on the PSS *knowledge* component, which is considered necessary within the context of ideal PSS (section 2.5), and we have suggested an agent-based structure for this component. The remainder of this thesis shall persist in focussing on the *knowledge* component and, in particular, on the cooperation between the agents labelled as *domain managers* who operate within it. Along this line, the following chapters will specify a multi-agent model for alternative plan generation, discuss the design and implementation of that model into an operational application, and report on the results of sensitivity analyses performed on the model.

5 A MULTI-AGENT MODEL FOR ALTERNATIVE PLAN GENERATION

This chapter presents a multi-agent model for supporting planners in generating plan alternatives, which is one of the core activities in planning and, consequently, one of the core parts of the MASQUE Planning Support System. First, some organizational issues are discussed. Then, a further specification of the agents operating within the model is provided. Finally, the agents' interaction is formalized in terms of two basic protocols.

5.1 INTRODUCTION

In the previous chapters, the opportunities for enhancing Planning Support Systems (PSS) by means of multi-agent technology were discussed and conceptually specified for the MASQUE system. Acknowledging the necessity of developing PSS that serve the elemental planning activities of generating alternatives and evaluating them, MASQUE includes an agent-based *knowledge* component that embeds a community of *domain managers* – i.e., agents who are specialized in specific land uses – and directly links the system to the plan development process. In the remainder of this thesis, our attention is fully focussed on the operation and communication of these particular agents in order to examine the possibility and viability of using multi-agent technology to support the generation of plan alternatives, in addition to its mainstream application in the context of forecasting and evaluation tools. Hence, an instrument has been developed that allows planners to have the *domain managers* cooperatively generate alternative plans through an incremental process that reflects the group decision-making in urban plan development practice. This chapter specifies the agent-based model underlying the instrument.

5.2 CONTEXT

As mentioned, the MASQUE framework intends to support the development of land use plans. Based on the notion of an agent as a computational equivalent of a human expert, *domain managers* are presented as experts in specific disciplines involved in land use planning, such as housing, transportation, services, and green space (Table 4-3). Collectively, they form the 'urban planning team' (section 3.3.2) and provide users access

to the multidisciplinary expertise required in the urban plan development process. Users can commission the agents to cooperatively generate a set of alternative plans based on a given situation and given preconditions (e.g., regarding clustering, adjacency, and accessibility), while following a certain interaction protocol (section 5.4). Cooperation between the agents implies that they act truthfully towards each other and take into account both personal and common interests in their decision-making.

5.2.1 Organization

The urban planning team is operating according to the plan development process that underpins *MASQUE* (section 4.2). Assuming a given *inventory* and *plan program*, the *domain managers* start by individually performing analysis – identifying opportunities and threats in the study area and listing the interrelationships with other land uses (and thus agents) – and take part in the cooperative plan development process (Figure 5-1). Besides being comprehensive, this process should result in a compact but distinctive set of alternative plans (e.g., Brill et al., 1990). Hence, a *viewpoint approach* is promoted in which every agent initiates and leads the development of a *plan proposal* for which he or she requests others to contribute by expressing their claims for land units based on personal and/or collective interests. This implies that an agent can play two roles:

- The agent acts as *initiator* and decides upon how to integrate the various contributions into a proposal.
- The agent acts as *participant* and contributes on request to a proposal that is initiated by another agent.

When an initiator has drawn up his or her plan proposal, all agents in the team verify whether it is acceptable and feasible from their personal standpoint. If so, the plan proposal receives the status of *alternative plan*, which completes the alternative plan generation procedure. In the successive *Choice* phase of the process (section 4.2), the impacts of each alternative are forecasted and assessed over various periods of time before the team starts the concluding alternative plan evaluation procedure that is aimed at selecting the best plan.

5.2.2 Assumptions

The *domain managers* need input to be able to perceive, reason and act. Besides the input

they will receive from each other during the process, they use information from both the *inventory* and the *plan program* that are assumed to have been prepared in advance. The former source provides them with information regarding the environment, i.e., characteristics of the study area and surrounding areas in their current state. For an agent this information forms the basis on which he or she determines the opportunities and constraints of the study area, which relates to basically two aspects. On the one hand, the physical characteristics of the study area (e.g., soil condition, slope and vegetation) will influence the suitability of land parcels for accommodating that agent's land use. On the other hand, this same suitability is influenced by the existing land uses in the study area's surroundings, as these could cause synergies or conflicts with the agent's own land use. Consequently, *MASQUE* will offer tools to bring together all available geographical and non-geographical data in project databases. User support is provided by means of both *Interface Agents* (section 4.3.1), who can help with identifying needed system functionalities for data management, and *Tool Agents* (section 4.3.2), who can assist with

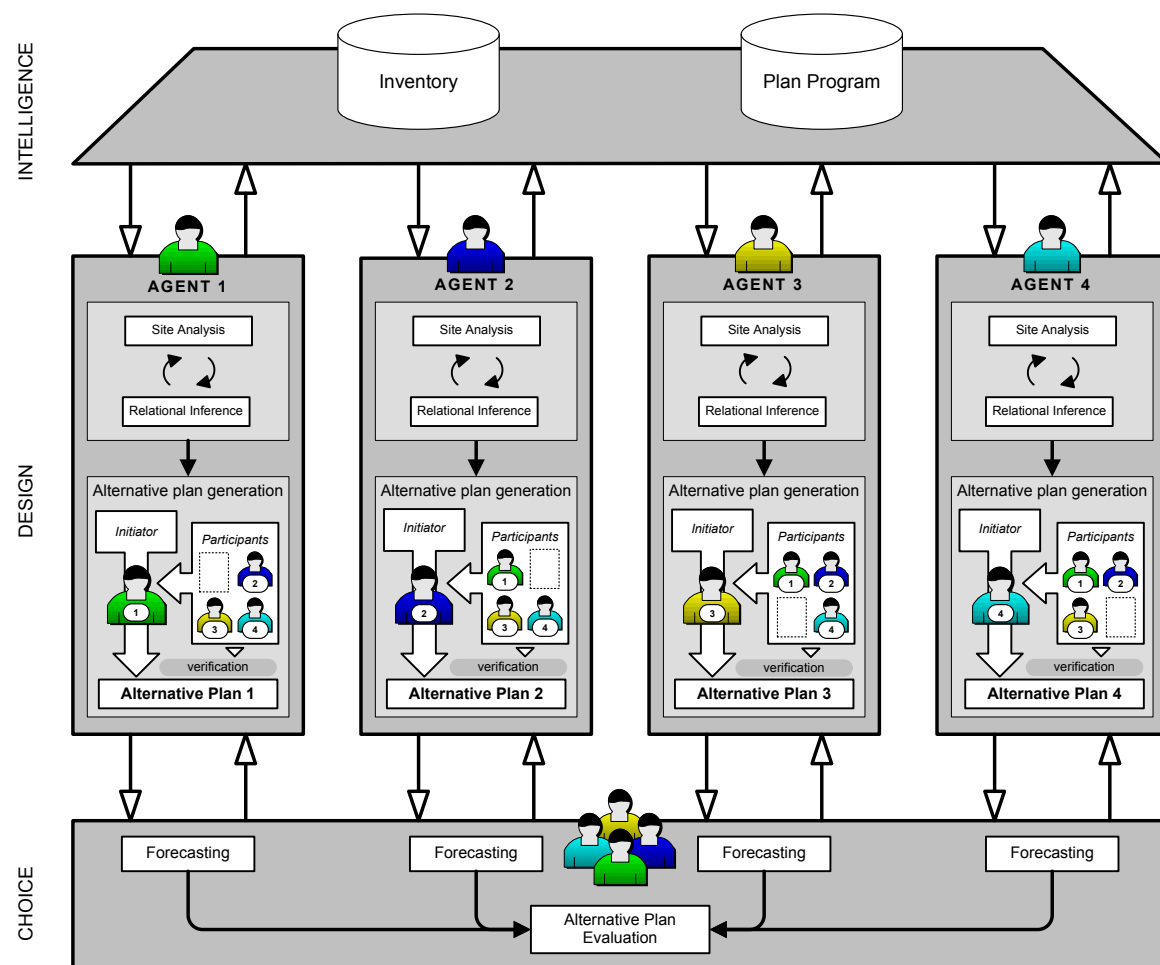


Figure 5-1. Cooperation of *domain managers* in the plan development process

the actual use of provided instruments, such as for data conversion or integration.

The *plan program* is the source from which agents can retrieve information regarding the project, i.e., requirements for the land uses that need to be allocated in the study area and the goals and objectives that have been formulated. Agents use this information to formulate their personal goals, objectives, and strategies by extracting all the preconditions related to their land use and filtering out possible inconsistencies. The preconditions may originate from different exogenous parties and sources:

- Higher governmental layers may impose preconditions onto the developments in the study area by means of larger-scale plans, i.e., regional plans and structure plans (section 1.1.1).
- The project originator – usually the municipality in question – may define preferences or requirements based on local desires or needs.
- The user of the system – in the role of urban planner – may choose to define additional preconditions.

MASQUE provides tools to support the integration of preconditions into one or more consistent set(s) of goals and objectives, and to store these in project input files (i.e., as plan program scenarios). Two kinds of preconditions are distinguished here. On the one hand, there are *general* preconditions that every agent should take into account and relate to aspects such as sustainability and equity. On the other hand, *specific* preconditions are defined that refer to one particular domain – criteria related to aspects of location, quantity, quality, and realization of a land use – and need to be accounted for by agents specialized in that domain.

5.3 AGENT SPECIFICATION

As a starting point for model development, a discretization of the environment in space is assumed by means of a *grid* representation of the study area and its surrounding areas. Consequently, the land units are equally sized square *cells* with individual attributes. This assumption is believed to be merely practical, having no implications for the applicability of the concept when working with vector representations of environments. Additionally, it is assumed that both the *inventory* and the *plan program* have already been prepared for the agents to start the alternative plan generation procedure. With respect to the former information source, it is assumed that all data requirements of agents are fully met,

whereas for the latter the provided information is assumed to explicitly specify for each land use the area that needs to be claimed by the corresponding agent, where the sum of all listed areas equals the total size of the study area. In conjunction with the assumption of a grid representation, this means that an agent's *task size* is expressed in terms of the numbers of cells that needs to be claimed for his or her land use.

Generally, *domain managers* are to claim cells in the study area for their land use based on the information they can extract from the environment and/or receive from each other. Any claim an agent makes will need to conform to the exogenous criteria that are found in the project's *plan program*, i.e., the given task size for the agent and the preconditions addressing his or her land use. Concurrently, an agent is expected to pursue his or her personal goals and objectives – endogenous criteria – as well. In view of the fact that such goals and objectives will be (partly) project-specific, it is clear that the first thing an agent should do when getting involved in a plan development project is to specify his or her personal goals and objectives for that project. This requires every agent to individually analyse the available information about the project and the environment to identify the inherent opportunities and constraints based on his or her expert knowledge.

The decision of an agent whether or not to claim a cell is considered to be based on the total number of cells the agent needs and the suitability of that cell in the agent's eyes relative to the suitability of other cells. The *relative suitability* of a cell is operationalized as the *expected utility* of that cell compared with that of other cells, expressed as an index within a [0,100] range. Every agent privately determines the expected utility for each cell by running a decision model that consists of an expandable set of rules to judge cells based on both their physical and spatial attributes. Each decision rule addresses one specific attribute to measure the partial (or 'single-attribute') expected utility of a cell based on that attribute, which contributes to the total (or 'multi-attribute') expected utility of that cell measured by the agent's decision model.

The more land uses there are to allocate in the study area, the more likely competition over cells occurs. During the course of developing a plan proposal, it is unclear to an agent what requirements and/or desires other agents will try to meet, how persistent they will be in claiming particular cells, and what compromises they will be willing to make. Consequently, agents are faced with uncertainty of information in their decision-making, albeit in a decreasing fashion when the plan proposal is gradually drawn up. The uncertainty during the development of a plan proposal forces an agent to apply his or her utility functions in a probabilistically manner. If information would be certain

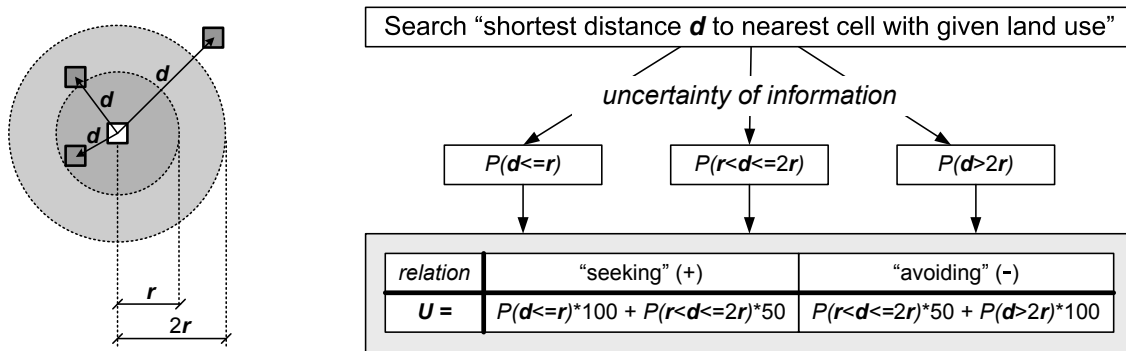


Figure 5-2. Utility function to determine the expected utility (U) of a cell based on accessibility

and an agent would, for instance, judge the *accessibility* to a land use l , the result of searching the shortest distance to the nearest cell with that land use could be directly translated into terms of utility by means of a continuous function of which the course would depend on whether or not the agent appreciates nearness to that land use. Nevertheless, due to the fact that the information is uncertain, it is necessary to predefine an exhaustive set of mutually exclusive states and determine the expected utility value for each observed state of the attribute:

- In the case of *accessibility* (Figure 5-2) this implies predefining a certain number of concentric ‘zones’ around cells and preset the utilities for the cases in which each zone would indeed accommodate the nearest cell with land use l (e.g., in outward order {100, 50, 0} for nearness-seeking and {0, 50, 100} for nearness-avoiding). When the rule is applied, an agent first needs to determine the probabilities that land use l will be found in these zones based on the available information, for example $\{P(d \leq r), P(r < d \leq 2r), P(d > 2r)\}$. Then, the agent can calculate the expected utility, U , by using these probabilities as weights in a weighted summation of the preset zone-base utilities. For example, when accessibility is desired with preset utilities of {100, 50, 0}, and the agent finds the probabilities to be {0.4, 0.2, 0.4}, the expected utility will be $(0.4 * 100) + (0.2 * 50) = 50$.
- In the case an agent considers the *adjacency* of a land use l to contribute to the expected utility of cells, the required decision rule can take a simpler form (Figure 5-3). For instance, judgments can be made based on the highest probability of finding the land use in a cell’s direct

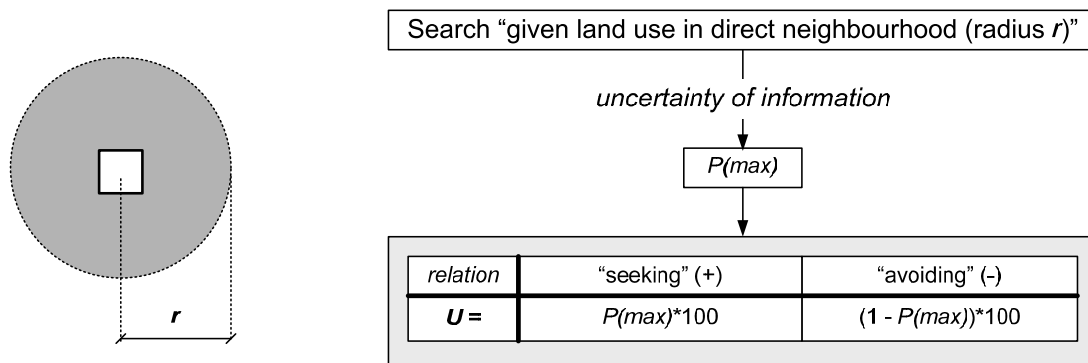


Figure 5-3. Utility function to determine the expected utility (U) of a cell based on adjacency

neighbourhood, i.e., within a specified (short) distance of the cell.⁷ This requires presetting the utility for actually finding the land use in this direct neighbourhood (or not) – e.g., 100 – based on whether the land use’s adjacency is appreciated (or not). Subsequently, the agent can calculate the expected utility, U , by means of determining the highest probability of finding the land use (or not) and using this as a weight to the corresponding preset utility of finding it (or not).

- For an agent who is responsible for a land use that requires a catchment area, the size of the *population* within a given radius will be one of the main factors contributing to the expected utility of cells. Hence, the agent could, for instance, judge the expected population in the area around the cell, N , and compare that to the population required to ensure a fail-safe operation, N_{\min} . Using a ratio $f = N/N_{\min}$ to make this comparison, the expected utility will decrease more severely as the ratio becomes smaller, e.g., according to a quadratic equation (Figure 5-4).

Owing to this uncertainty in decision-making, agents are confined to act based on their *beliefs* about what claims would be most attractive to them, given their *beliefs* about

⁷ Regarding the formulation of decision rules related to adjacency, it is important to avoid dependency on the resolution used for representing space in a grid. This means that these rules require (i) expressing the search radius in distance (instead of in number of cells, e.g., Von Neumann or Moore neighbourhoods) and (ii) defining states for which probabilities can be measured without combinatorial calculations over cells (e.g., ‘the probability of finding land use 1 in the direct neighbourhood’ would imply measuring the probability of finding the land use in one or more cells in that neighbourhood, which is highly sensitive to the number of cells; instead, it is better to use statistical indicators such as ‘maximum’ or ‘average’).

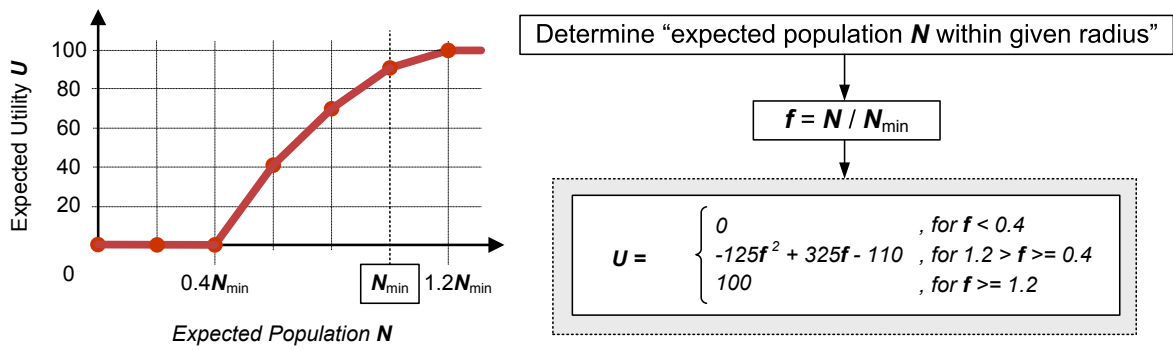


Figure 5-4. Utility function to determine the expected utility (U) of a cell based on population

what would be the most probable outcome of the plan proposal considering all interests known at a certain moment in time. In addition, the need to balance one’s own interests (i.e., one’s personal requirements and desires) with general interests (i.e., a joint solution) provokes an agent to act strategically when expressing his or her claims.

In the role of initiator, an agent collects claims from all agents (including his or her own claims) and determines what assignments of cells to suggest to the different agents (i.e., the assignments for the different land uses). This implies that the initiator, while reasoning from his or her own viewpoint and interests, needs to have the capability to resolve (believed) conflicts of interests between the different parties. As this requires another balancing of interests, an initiator’s action will be strategic as well. The probabilistic nature of agents’ reasoning and action gives natural grounds for an implementation based on probability theory. Figure 5-5 outlines the reasoning and interaction components used by agents when playing the two possible roles.

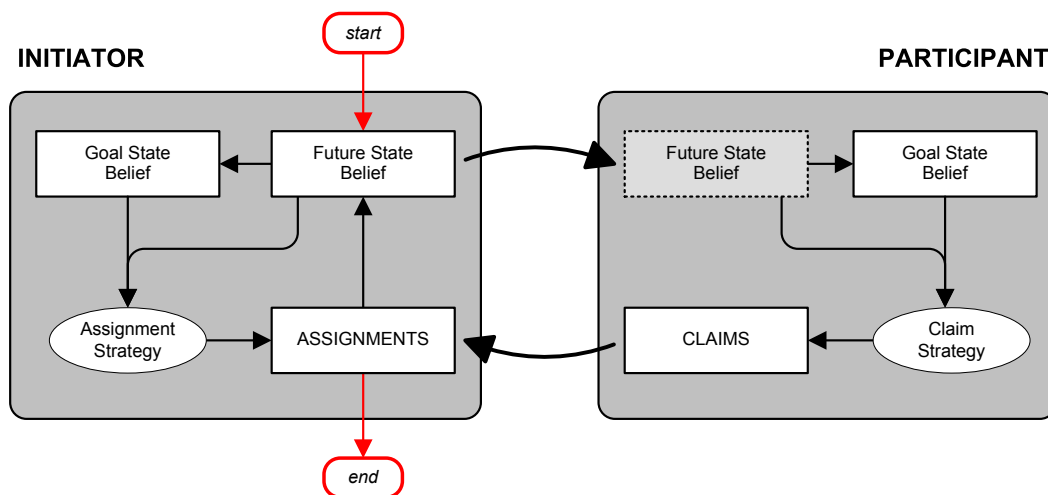


Figure 5-5. Model building blocks: components for reasoning and interaction

5.3.1 Beliefs

The model distinguishes two types of beliefs, as mentioned above, namely an agent's belief about what claims would be most attractive to the land use in which he or she is specialized and his or her belief about what would be the most probable outcome of the plan proposal considering all interests known. Both types of beliefs are formed based on the available (uncertain) information at a given moment in time during the process.

5.3.1.1 Goal state belief

The belief an agent has about what allocation of claims would be most attractive to the land use he or she represents is referred to as that agent's *goal state belief*. This personal belief addresses all cells in the study area, consisting of I rows and J columns of cells, and indicates for every cell the probability that this agent x will claim it, as seen from the viewpoint of that agent at time t :

$$Bel\{GS\}_x^t = \begin{bmatrix} P(C_{1,1,x}^{g,t} = 1) & \dots & P(C_{1,J,x}^{g,t} = 1) \\ \vdots & \ddots & \vdots \\ P(C_{I,1,x}^{g,t} = 1) & \dots & P(C_{I,J,x}^{g,t} = 1) \end{bmatrix}, \quad \forall i = 1, \dots, I, j = 1, \dots, J, \quad (5-1)$$

where $P(C_{i,j,x}^{g,t} = 1)$ is the probability of cell (i, j) being claimed ($C=1$) by agent x at time t , with index g indicating it concerns the agent's goal state. Obviously, the probability of not claiming ($C=0$) the cell is: $P(C_{i,j,x}^{g,t} = 0) = 1 - P(C_{i,j,x}^{g,t} = 1)$.

An agent constructs his or her goal state belief in three consecutive steps. First, the agent applies a personal *decision model* to determine the expected utility, U , of each cell based on the available information. While at the start of the process this information is limited to the attribute data found in the available information sources, during the process more accurate information becomes available when the initiator makes assignment decisions. The agent computes:

$$U_{i,j,x}^t = f_x^t \left[Z, L^t, P(L^t) \right] \quad (5-2)$$

where $U_{i,j,x}^t$ is the expected utility of cell (i, j) as believed by agent x at time t , taking a value in the range of $[0,100]$. This value is a function of vectors of relevant cell attributes:

- Physical characteristics, Z (e.g., soil condition, slope, vegetation).
- The location relative to *current* land uses, L^c , i.e., to cells in the surroundings of the study area for which land uses are given.
- The location relative to *future* land uses, $P(L^f)$, i.e., to other cells in the study area for which land uses are to be determined.

Important to note is that the last category concerns probabilistic data, $P(L^f)$, that follow from the belief of agent x at time t concerning the most probable outcome of the plan proposal (section 5.3.1.2). During the process, however, increasingly more decisions will be made and the uncertainty will steadily reduce. Thus, the beliefs will gradually become more factual.

Second, the agent converts the expected utility, U , of each cell into a preliminary probability, P^* , that expresses the basic chance that the agent claims the cell. It is assumed that this conversion is represented by a symmetric S-shaped curve, which means that there is an inflection point linked to the level of expected utility that the agent uses to distinguish between more attractive cells and less attractive cells. As the utility decreases from this point, the probability of claiming a cell will quickly decrease and smoothly approach its minimum ($P^* = 0$). Oppositely, as the utility increases from the inflection point, the probability will quickly increase and smoothly approach its maximum ($P^* = 1$). The following equation is proposed to represent the conversion:

$$P^*(C_{i,j,x}^{g,t} = 1) = \left\{ 1 + \exp \left[\beta (\alpha - U_{i,j,x}^t) \right] \right\}^{-1} \quad (5-3)$$

where α is a parameter that indicates the level of expected utility linked to the curve's inflection point and β is a parameter that controls the slope of the curve at that point. The curve expresses the notion of an agent discriminating between cells he or she wants to set as a target and other cells. The parameters α and β define the settings for this discrimination. It seems reasonable to suggest that the position of the curve's inflection point, as controlled by parameter α , is proportionally related to the average expected utility of all cells in the study area at time t , \bar{U}^t , implying that a higher average will result in a horizontal shift of the curve to the right-hand side (Figure 5-6). Furthermore, as the percentage of cells having an expected utility near the average increases, the discrimination will require a higher accuracy, implying that the slope of the S-shaped

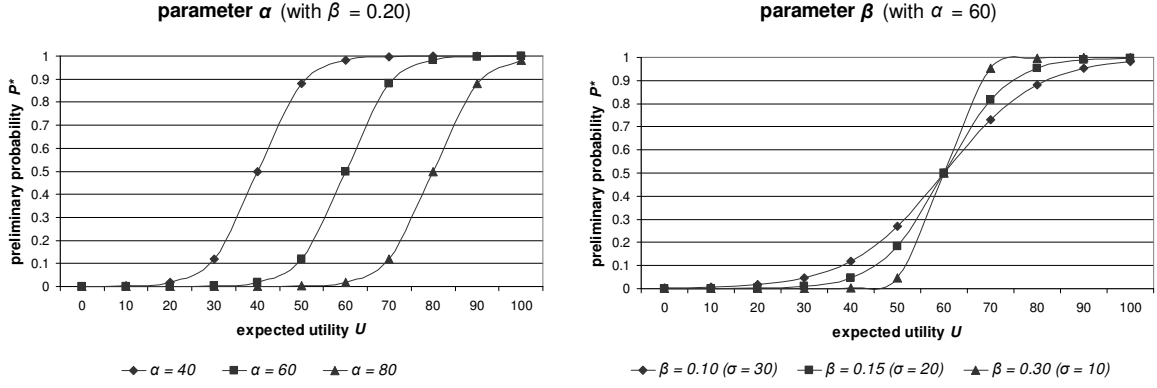


Figure 5-6. Sensitivity of parameters α and β that define the S-shaped curve used for converting the expected utility (U) of cells into a preliminary probability (P^*)

curve should show steeper, corresponding to a higher value of β . Hence, parameter β is considered to be inversely proportional to the standard deviation of the whole set of cells in the study area at time t , σ^t , meaning that a lower standard deviation will make the slope of the curve steeper (Figure 5-6). Based on sensitivity tests the following default settings for the two parameters are assumed:

$$\alpha = \bar{U}^t, \quad \beta = \frac{3}{\sigma^t} \quad (5-4)$$

Third and last, the agent takes into account the desired *level of clustering* for his or her land use and identifies those pieces of land that together form the best allocation. For this, the agent converts the preliminary probability P^* for each cell into a final probability P by making the latter conditional on the probabilities of claiming the adjacent cells in its eight-cell neighbourhood, $H_{i,j}$, that consists of cells that share a side (denoted by \perp) and cells that share a corner (denoted by \angle) with the centre cell. For agent x this conversion is implemented as follows:

$$\begin{aligned} P\left(C_{i,j,x}^{g,t} = 1 \mid \overline{C_{i,j,x}^{g,t}} = c, \overline{C_{H_{i,j,\perp},x}^{g,t}} = c_{\perp}, \overline{C_{H_{i,j,\angle},x}^{g,t}} = c_{\angle}\right) \\ = (1-\phi)C_{i,j,x}^{g,t} + \phi \left[\gamma \overline{C_{H_{i,j,\perp},x}^{g,t}} + (1-\gamma) \overline{C_{H_{i,j,\angle},x}^{g,t}} \right], \quad \forall c, c_{\perp}, c_{\angle} \in \{0,1\} \end{aligned} \quad (5-5)$$

where ϕ is a parameter in the range $[0,1]$ indicating the intensity of clustering that agent x desires, and γ is a parameter in the range $[0,1]$ setting the value that agent x gives to the average preliminary probability of claiming the neighbours at the side of the centre cell,

$\overline{C_{H_{i,j,\perp},x}^{*,g,t}}$, relative to the value given to that of claiming its corner neighbours, $\overline{C_{H_{i,j,\angle},x}^{*,g,t}}$.

Table 5-1 illustrates the proposed method for including clustering in the format of a conditional probability table (CPT). As an example, a desired intensity of clustering (ϕ) of 0.8 and a distinction between side and corner neighbours (γ) of 0.75 are assumed. The table shows how the final probability P of an agent x claiming a cell (i, j) (see final column) is conditional on the set of claims this agent makes for that cell and the average claims for that cell's side and corner neighbours (see first three columns). Every row in the table represents a *state* that is a combination of possible deterministic claims [i.e., cells are claimed (1) or not (0)] and linked to a final probability P . Every row can be read as an *if-then* rule. When applying the CPT, the following probabilities are the inputs:

- The preliminary probability of claiming the cell, $P(C_{i,j,x}^{*,g,t} = 1)$.
- The average preliminary probability of claiming the cell's side neighbours, $P(C_{H_{i,j,\perp},x}^{*,g,t} = 1)$.
- The average preliminary probability of claiming the cell's corner neighbours, $P(C_{H_{i,j,\angle},x}^{*,g,t} = 1)$.

For every state, these input probabilities [i.e., $P(C^* = 1)$ if the state implies claiming the corresponding cell(s); and $1 - P(C^* = 1) = P(C^* = 0)$ if it implies not claiming the cell(s)] are multiplied to get the state's *occurrence probability*. This probability multiplied with the state's final probability will give the state-based *partial final probability*. For example, when the input probabilities are {0.3, 0.6, 0.5}, respectively, the first state will

Table 5-1. CPT for including clustering for cell (i, j) at time t
(example with $\phi = .80$, $\gamma = .75$)

$C_{i,j,x}^{*,g,t}$	$\overline{C_{H_{i,j,\perp},x}^{*,g,t}}$	$\overline{C_{H_{i,j,\angle},x}^{*,g,t}}$	$P(C_{i,j,x}^{*,g,t} = 1)$
1	1	1	1
1	1	0	0.8
1	0	1	0.4
1	0	0	0.2
0	1	1	0.8
0	1	0	0.6
0	0	1	0.2
0	0	0	0

lead to a partial final probability of 0.09 (i.e., $0.3 \times 0.6 \times 0.5 \times 1$), the second state to 0.072 [i.e., $0.3 \times 0.6 \times (1-0.5) \times 0.8$], the third state to 0.024 [i.e., $0.3 \times (1-0.6) \times 0.5 \times 0.4$], and so on. All states' partial final probabilities sum up to the searched for final probability for the centre cell, i.e., after including clustering. In this example, the relatively high average preliminary probabilities for the neighbouring cells (0.6 and 0.5) will cause the agent to decide to increase his or her probability of claiming the centre cell, which had a preliminary value of $P(C_{i,j,x}^{*,g,t} = 1) = 0.3$, to, finally, $P(C_{i,j,x}^{g,t} = 1) = 0.09 + 0.072 + 0.024 + \dots = 0.52$. The probabilities for all study area cells together make up the agent's goal state belief.⁸

5.3.1.2 Future state belief

An agent's claims are determined based on his or her belief about the most probable outcome of the plan proposal. This belief, referred to as the *future state belief*, gives insight into the prevailing expectations regarding the future state of the study area and can be considered as the belief an agent has about the direction in which the plan proposal is heading under the supervision of the initiating agent. While the plan proposal is under development the expectations are changing in response to the effects of the initiator's assignment decisions. Given the cooperative nature of the plan development process, the future state belief is considered to be a collective belief that is constructed and iteratively updated by the initiator of the plan proposal, who forwards it to participants in order to have them determine their claims given the presented prevailing expectations. The future state belief addresses all cells in a study area, consisting of I rows and J columns of cells, and gives for every cell the probability distribution for the candidate land uses as perceived by initiator y at time t :⁹

$$Bel\{SAF\}_y^t = \begin{bmatrix} P(L_{1,1,y}^t) & \dots & P(L_{1,J,y}^t) \\ \vdots & \ddots & \vdots \\ P(L_{I,1,y}^t) & \dots & P(L_{I,J,y}^t) \end{bmatrix} \quad (5-6)$$

⁸ When it is considered to be included in agents' claiming behaviour (section 6.3.2), the resulting values P are scaled in order to make their sum for the whole study area equal to the agent's task size.

⁹ SAF stands for Study Area Future, highlighting the fact that a future state belief concerns the whole set of land uses, as opposed to a goal state belief that only addresses one land use.

where $P(L_{i,j,y}^t)$ is the probability distribution of cell (i, j) across a set L of n candidate land uses as perceived by initiator y at time t :

$$P(L_{i,j,y}^t) = \{P(A_{i,j,1,y}^t = 1), P(A_{i,j,2,y}^t = 1), \dots, P(A_{i,j,n,y}^t = 1)\} \quad (5-7)$$

where $P(A_{i,j,x,y}^t = 1)$ is the probability of initiator y assigning cell (i, j) to land use (or agent) x . For initialization ($t = 0$) of $Bel\{SAF\}$, the initiator uses the predefined agent task sizes (i.e., the required numbers of cells per land use) to get a probability distribution across land uses for every cell in the study area. During the plan generation process ($t > 0$), initiator y updates the belief by applying a real-time defined function f_y that expresses the belief to be conditional on the set of all received claims, including the initiator's own:

$$Bel\{SAF\}_y^t = f_y(Claim\{SA\}_1, Claim\{SA\}_2, \dots, Claim\{SA\}_n), \quad \forall t > 0 \quad (5-8)$$

where $Claim\{SA\}_x$ are the claims for the study area (SA) received from agent x (section 5.3.2.1). In order to perform a belief update, the initiator needs to balance the different inputs or interests. Based on his or her personal requirements and/or preferences for adjacency and accessibility to other land uses, and given general requirements to ensure that common interests are taken into account, the initiator defines the importance of each input (i.e., each participant's claims). The belief update takes place cell-by-cell and can be implemented by means of a discrete choice model, e.g., a logit model that has the advantage of encouraging distinction between assignments while avoiding values of zero or one. This characteristic is very suitable to reflect the notion of an initiator who is decisive in making assignments but is reserved in doing so with certainty (even if all the claims were expressed with certainty) in consideration of the unpredictability of process developments. Consequently, an initiator keeps all options open until the claims are stabilizing for the whole study area, and only then, act with certainty in the assignment of a cell (section 5.3.2.2). For updating the future state belief, a logit model would take the following form:

$$P(A_{i,j,x,y}^t = 1|G) = \exp(V_{i,j,x,y}^t) / \sum_1^n \exp(V_{i,j,n,y}^t) \quad (5-9)$$

where G is a vector consisting of n inputs: $V_{i,j,x,y}^t$ is the claim for cell (i, j) from agent x at

time t as weighted by initiator y :

$$V_{i,j,x,y}^t = w_{y,x} C_{i,j,x}^{s,t}, \quad \forall C_{i,j,x}^{s,t} \in \{0,1\} \quad (5-10)$$

where $w_{y,x}$ reflects the importance that initiator y gives to the input regarding land use x , ranging from “no importance” ($w_{y,x} = 0$) to “high importance” ($w_{y,x} = 5$). $C_{i,j,x}^{s,t}$ is the claim made for cell (i, j) by agent x at time t , with index s indicating that it concerns the agent’s strategic claims (section 5.3.2.1).

Table 5-2 gives an example of the logit-based belief update in the format of a CPT, where agent 1, as initiator, considers a set of three inputs $\{C_1, C_2, C_3\}$, consisting of his or her own claims and the claims of two others, agents 2 and 3, and processes these by means of a personally defined weight set $\{w_{1,1} = 5, w_{1,2} = 3, w_{1,3} = 1\}$. The table illustrates the above-mentioned reservation an initiator applies: in the cases that only one agent is claiming the cell (states 1-0-0, 0-1-0, and 0-0-1), the assignment probability for that agent becomes dominant (to a degree that depends on the applied weight set) but not absolutely certain. Furthermore, in the case where none of the agents claim the cell (state 0-0-0), the assignments will be undecided in the sense that every agent will have equal chance of still seeing the cell to be assigned to his or her land use. Obviously, alternative ways to update the future state belief by means of different types of models are possible.

Application of the CPT works in an identical manner as before (section 5.3.1.1). For example, when the inputs – *claim probabilities* – are $\{0.2, 0.6, 0.8\}$, respectively, the *occurrence probability* of the first state is 0.096 (i.e., $0.2 \times 0.6 \times 0.8$), which leads to *partial final probabilities* of 0.083 (i.e., 0.096×0.867), 0.011 (i.e., 0.096×0.117), and 0.002 (i.e., 0.096×0.016) for the agents 1, 2 and 3, respectively. The second state’s

Table 5-2. CPT for a logit-based update of $Bel\{SAF\}$ for cell (i, j) at time t
(example with $w_{1,1} = 5, w_{1,2} = 3, w_{1,3} = 1$)

C_1^s	C_2^s	C_3^s	$Bel\{SAF\}_y$		
			$P(A_{1,y} = 1)$	$P(A_{2,y} = 1)$	$P(A_{3,y} = 1)$
1	1	1	0.867	0.117	0.016
1	1	0	0.876	0.118	0.006
1	0	1	0.976	0.007	0.018
1	0	0	0.987	0.007	0.007
0	1	1	0.042	0.844	0.114
0	1	0	0.045	0.909	0.045
0	0	1	0.212	0.212	0.576
0	0	0	0.333	0.333	0.333

occurrence probability is 0.024 [i.e., $0.2 \times 0.6 \times (1-0.8)$], giving partial final probabilities of 0.021 (i.e., 0.024×0.876), 0.003 (i.e., 0.024×0.118), and 1.4E-04 (i.e., 0.024×0.006) for the respective agents, and so on. By summing the partial final probabilities per agent over all states, the *assignment probability* for each agent is found. Hence, for the given example the outputs are {0.279, 0.502, 0.220}, which indicates that the assignments are strongly influenced by the claims of agents whose contributions are considered most important, even when these claims have a relative low probability. This implies that, as long as there is a chance that such an agent will claim a particular cell, the initiator will ensure that his or her assignments for that cell keep that option open. Evidently, this tendency weakens when the initiator's weighing of claims becomes more levelled. For example, when the initiator's weighs all claims equally, the inputs {0.2, 0.6, 0.8} will be processed into assignments that are more proportional: {0.234, 0.349, 0.417}.

After having determined his or her future state belief, the initiating agent has to scale the outcomes by means of an Iterative Proportional Fitting (IPF) procedure in order to satisfy two fundamental constraints regarding the whole study area. On the one hand, the sum of all assignments for a land use x should equal the task size TS_x for that land use:

$$\sum_{i=1}^I \sum_{j=1}^J P(A_{i,j,x,y}^t = 1) = TS_x \quad (5-11)$$

On the other hand, the sum of all assignments for a cell (i, j) should equal one:

$$\sum_{x=1}^n P(A_{i,j,x,y}^t = 1) = 1 \quad (5-12)$$

where n is the number of candidate land uses.

5.3.2 Strategic actions

Agents express their claims when they, as participants, are requested to do so by the initiator of the plan proposal under development. As initiator, an agent processes all claims, including his or her own, and communicates the resulting assignments to the participants, who then assess these assignments to determine their (change in) satisfaction and act accordingly. The only means that agents have to execute these tasks are their prevailing beliefs, which provokes them to act strategically.

5.3.2.1 Claims

When an agent is requested to express his or her claims, he or she is asked to indicate, for each cell, whether he or she wants to claim that cell. Hence, the claims for a study area (SA), consisting of I rows and J columns of cells, as decided and expressed by the participating agent x at time t , are:

$$Claim\{SA\}_x^t = \begin{bmatrix} P(C_{1,1,x}^{s,t} = 1) & \dots & P(C_{1,J,x}^{s,t} = 1) \\ \vdots & \ddots & \vdots \\ P(C_{I,1,x}^{s,t} = 1) & \dots & P(C_{I,J,x}^{s,t} = 1) \end{bmatrix}, \quad \forall i=1\dots I, j=1\dots J \quad (5-13)$$

where $P(C_{i,j,x}^{s,t} = 1)$ is the strategic claim C^s for cell (i, j) , as expressed by agent x at time t . The agent's strategy relates to his or her chosen way of combining his or her personal goal state belief ("what is most attractive?") with the collective future state belief ("what is most likely?"). A participant comes to this decision in two steps. First, the agent updates his or her goal state belief by means of reapplying his or her decision model based on the probabilistic information contained in the future state belief that has been updated and forwarded by the initiator. Second, the agent decides how to combine the two beliefs. Obviously, the claims can be based simply on only one of the beliefs, resulting in either a most *opportunistic* or a most *compliant* strategy. Besides following such a pure strategy, the agent can choose a mixed strategy that takes both beliefs into account. The strategic decision of combining beliefs in order to define claims is assumed to take the form of a weighted summation that operates on a cell-by-cell basis:

$$P(C_{i,j,x}^{s,t} = 1 | C_{i,j,x}^{g,t} = c, A_{i,j,x,y}^t = a) = \theta C_{i,j,x}^{g,t} + (1-\theta) A_{i,j,x,y}^t, \quad \forall c, a \in \{0,1\} \quad (5-14)$$

where θ is a parameter $[0,1]$ indicating the level of opportunism applied by agent x , i.e., the extent to which he or she pursues his or her (updated) personal goal state belief, $C_{i,j,x}^{g,t}$, as opposed to the current collective future state belief, $A_{i,j,x,y}^t$, for any cell.¹⁰ As with the goal state belief (section 5.3.1.1), the formulation of claims is followed by a clustering inclusion procedure and an optional scaling procedure.

¹⁰ Because of constructing the future state belief personally, an initiator will act in full compliance ($\theta = 0$) with it when defining his or her own claims.

The θ parameter is considered to express the fundamental part of an agent's *claim strategy*. During the process, however, the agent will get an impression of the effectiveness of this strategy, and accordingly, could refine his or her strategy by means of *claim accentuation* that allows him or her to stress claims [i.e., to increase the claims for cells that belong to his or her *best set* (i.e., the k cells for which his or her claims are highest, with k equal to the agent's task size), and to decrease the claims for all cells that do not]. The extent to which an agent x accentuates his or her claim for a cell (i, j) is assumed to be calculated as follows:

$$C_{i,j,x}^{new} = \begin{cases} \min\{C_{i,j,x}^{old} (1 + f_{c,x}), 1\}, & \text{for } (i, j) \in \text{'best set for agent } x\text{'} \\ C_{i,j,x}^{old} (1 - f_{c,x}) & , \text{ for } (i, j) \notin \text{'best set for agent } x\text{'} \end{cases} \quad (5-15)$$

where $f_{c,x}$ is a claim accentuation factor with a range of [0,1] and taking a value of:

$$f_{c,x} = \begin{cases} f_{c,x}^d (1 - \theta) & , \text{ for } \Delta S_x < 0 \\ f_{c,x}^d & , \text{ for } \Delta S_x = 0 \\ \min\{f_{c,x}^d (1 + \theta), 1\} & , \text{ for } \Delta S_x > 0 \end{cases} \quad (5-16)$$

where $f_{c,x}^d$ is the user-defined default (d) value that indicates the basic level of claim accentuation that ought to be applied by the agent. ΔS_x is the change in satisfaction (section 5.3.2.3) that the agent measures by comparing the current assignments with the previous ones. An increase in satisfaction encourages the agent to accentuate his or her claims, while a decrease causes discouragement. The agent's basic claim attitude, as reflected by the θ parameter (equation 5-14), determines the degree of encouragement or discouragement based on the assumption that an agent will feel more strongly encouraged (or discouraged) when he or she is more opportunistic by default (i.e., when θ is higher). When considered to be part of agents' claiming behaviour (section 6.3.2), an agent scales the claims after accentuation to have their sum equal his or her task size.

5.3.2.2 Assignments

The initiator of a plan proposal processes all claims into assignments, which means that this agent determines for each cell the probabilities of the different land uses to be

assigned to that cell. Hence, the assignments for a study area (SA), consisting of I rows and J columns of cells, as determined and expressed by initiator y at time t are:

$$\text{Assignment}\{\text{SA}\}_y^t = \begin{bmatrix} P(L_{1,1,y}^t) & \dots & P(L_{1,J,y}^t) \\ \vdots & \ddots & \vdots \\ P(L_{I,1,y}^t) & \dots & P(L_{I,J,y}^t) \end{bmatrix} \quad (5-17)$$

where $P(L_{i,j,y}^t)$ is the probability distribution for cell (i, j) as expressed by initiator y at time t across a set L of n candidate land uses (cf. equation 5-7). The construction of assignments is identical to updating a future state belief (cf. section 5.3.1.2), using weights for every land use as parameters in the procedure. The initiator acts strategically regarding the extent to which he or she will attempt to lead the plan proposal development in a personally desired direction and/or to accelerate the process. Two strategic instruments are available to the initiator.

The weights used to construct the assignments are initialised at the start of the process based on the relative importance of each land use to the initiator. During the process, the initiator has the possibility to apply *weight set adjustment* based on whether or not cells are identified as belonging to agents' best sets. The adjustment made by initiator y for land use x and for a cell (i, j) is decided upon as follows:

$$w_{y,x} = \begin{cases} w_{y,x}^d + \left(\frac{s^+ * m^- * f_{a,y}}{n-1} \right), & \text{for } (i, j) \in \text{'best set for agent } x\text{'} \\ w_{y,x}^d - \left(\frac{s^- * m^+ * f_{a,y}}{n-1} \right), & \text{for } (i, j) \notin \text{'best set for agent } x\text{'} \end{cases} \quad (5-18)$$

where $w_{y,x}^d$ is the default (d) weight used by initiator y for land use x . Parameter s indicates the available adjustment space for either an increase (+) of the weight up to the maximum ($w = 5$) or a decrease (–) down to the minimum ($w = 0$), whereas m is the number of times cell (i, j) belongs (+) or not belongs (–) to agents' best sets, given n agents (or claims). The user-defined factor $f_{a,y}$ sets the intensity of adjustment that ought to be applied by initiator y . After having determined the assignments based on an adjusted weight set, the initiator scales the outcomes by performing an IPF procedure (section 5.3.1.2).

A second strategic instrument is *cell finalization* by means of which the initiator can permanently assign a cell to a land use when the assignments are not sufficiently different from the previous assignments. This is considered to be the case when, for each land use, the sum of square differences across all cells in the study area is not larger than a user-defined critical minimum ΔA_{min} . The agent then searches the cell with the most certain assignment (i.e., the highest probability across all cells and all land uses) and sets that cell's assignment to a value of 1 for the agent (or land use) who had most change of receiving it and to a value of 0 for all other agents. Owing to the fact that finalizations are permanent, it will cause agents directly to shift their interests to other cells. It will also have an indirect effect, as it will influence the subsequent expected utility calculations for the surrounding cells. The resulting changes in claims will lead to changes in assignments, and vice versa, until the next finalization is made. In addition to the critical minimum, the cell finalization method has a second user-defined parameter, namely the iteration limit, i_{lim} , which can directly accelerate the process. Based on this parameter the initiator will enforce cell finalization after the indicated number of iterations, irrespective of the fact the assignments are sufficiently different or not. An IPF procedure (section 5.3.1.2) is used to scale the assignments after performing cell finalization.

5.3.2.3 Assessment of assignments

Having expressed his or her claims to the initiator, a participant will, in return, receive the assignments suggested by the initiator. The participant now compares the claims made ("what was asked for") and the assignments proposed ("what is offered in return") in order to judge to what extent the claims are met by the assignments, or in other words, to determine his or her *satisfaction*. A measure to express this satisfaction needs to meet two requirements. First, it should be a generic measure and therefore be independent of the number of cells in the study area. Second, it should take into account the fact that a high probability of being assigned a cell is valued more when it concerns a cell that was claimed strong than when it concerns a weakly claimed cell. Hence, the satisfaction can be determined as follows:

$$S_x^t = \frac{\sum_{i=1}^I \sum_{j=1}^J [P(C_{i,j,x}^{s,t} = 1)P(A_{i,j,x,y}^t = 1)]}{\sum_{i=1}^I \sum_{j=1}^J P(C_{i,j,x}^{s,t} = 1)} \quad (5-19)$$

where $P(C_{i,j,x}^{s,t} = 1)$ is the probability that agent x will claim cell (i, j) , and $P(A_{i,j,x,y}^t = 1)$ is the probability that this cell will be assigned to agent (land use) x . The probabilities related to claims operate as weights on the probabilities related to assignments, resulting in a value between 0 and 1.

A participating agent compares his or her satisfaction with the satisfaction in the previous round in order to detect whether the assignments converge (i.e., satisfaction is increasing) or diverge (i.e., satisfaction is decreasing) relative to his or her claims. The participant's subsequent action of determining adjusted claims is influenced accordingly, as the agent will be respectively encouraged or discouraged (section 5.3.2.1). Since a successful cooperation is of common interest to the agents, rejection of assignments will only occur when there are no other options left. This implies that, when assignments are not yet fully finalized, agents will continue their cooperation, i.e., they either accept the assignments given at that time when being sufficiently satisfied, or they retry by adjusting their claims when they are not¹¹. At the very end of the process – when the suggested assignments only show finalized cells – the possibility of a retrial is no longer available and agents will have to choose between acceptance and rejection. Implementation of this decision requires having a satisfaction threshold S_{min} for every agent, in which the difference from 1 indicates the flexibility of an agent. The calculated satisfaction S_x is reflected against the threshold: when $S_x \geq S_{min}$, the agent will accept the assignments and, if in the last round, the plan proposal as such; when $S_x < S_{min}$ he or she will retry by submitting adjusted claims, or, if in the last round, reject the plan proposal.

5.3.3 Selection of alternative plans

When all agents have initiated a plan proposal, the team verifies the resulting set in order to determine which plan proposals meet every agent's firsthand critical requirements (i.e., requirements of feasibility) and can thus be rightfully given the status of alternative plans. For verifying a plan proposal an agent applies the same personal decision model that he or she also applied during the generation of that plan proposal. This means that the verification takes place based on agents' critical requirements regarding a minimum

¹¹ In the case of acceptance, the process continues with the initiator advancing the assignments towards a complete land use allocation. In the case of retry, the participant will reply to the initiator with adjusted claims, based on which the latter will then redefine its assignments.

expected utility, and that the results are also expressed in terms of expected utility within a range of [0,100]. As every agent has a right of veto, only a plan proposal that meets all agents' critical requirements should be recognized as an alternative plan.

5.4 INTERACTION PROTOCOLS

The alternative plan generation procedure is triggered when the user of the system commissions the team of *domain managers* to explore the development options for the study area, as described by the *inventory* data and given the preconditions documented in the *plan program*. For every land use that is mentioned in the *plan program*, a *domain manager* is called into action to represent that land use pro-actively in a process of searching for feasible alternative land use configurations. Here, pro-activeness of agents refers to their ability to initiate the development of plan proposals and contact each other when contributions are required. Besides the truthfulness and pursuance of both personal and common interests mentioned before, the envisioned cooperative team operation has several manifestations:

- Participating agents make contributions to a plan proposal whenever requested by the initiator of that plan proposal.
- Every agent intends to work towards collectively accepted solutions either as initiator or as participant.
- All agents determine their initial goal state beliefs based on the given *plan program* and submit these beliefs to a repository from which each agent can initiate the development of a plan proposal.¹²

In the role of initiator an agent continuously updates the future state belief and provides this as given to the other agents when requesting for their contributions. Starting the development of a new plan proposal, the initiating agent retrieves the various initial goal state beliefs from the repository and constructs his or her first version of the collectively shared future state belief. This is the starting point of the cooperative process. From here, different *interaction protocols* can be distinguished for organizing the procedure in which the initiator incrementally draws up the plan proposal from the various claims he or she receives from the involved participants. It is for the way of

¹² The submission of goal state beliefs is restricted to the initialization. In subsequent interactions, agents personally update these beliefs and only privately use them to determine their claims.

requesting and processing claims that the initiator can apply two basically distinctive protocols, i.e. either simultaneously (section 5.4.1) or sequentially (section 5.4.2), which are likely to cause differences in plan outcomes as trade-offs between interests are made in a different setting and at different moments.

In general terms, convergence is established by organizing the cooperation between initiator and participants as an iterative process in which both consider adjustment of their strategies, and thus their decisions, in order to try to find acceptable solutions. Moreover, the incremental finalization of cells (section 5.3.2.2) brings an entropy-minimizing principle into the model that ensures a stepwise decrease in the uncertainty of both claims and assignments.

5.4.1 Simultaneous contributions

The interaction protocol based on simultaneous contributions implies that the initiator requests and processes all claims for the different land uses at once. Consequently, all participants are involved in the process of shaping the plan proposal from start to end. The relative importance the initiator gives to the various land uses – reasoning from his or her own land use – is solely embedded in its assignment strategy that is open for real-time adjustment whenever new claims are received. The process is as follows (Figure 5-7):

Step 1: The initiator sends the current future state belief, $Bel\{SAF\}$, to all involved participants at once, requesting for their claims.

Step 2: Every participant updates his or her personal goal state belief, $Bel\{GS\}$, decides about his or her *claim strategy* on how to combine this belief with the given $Bel\{SAF\}$, and sends the resulting *claims* back to the initiator.

Step 3: Having received the claims of all participants, the initiator adds them to his or her own claims, judges the whole set of claims to determine his or her *assignment strategy*, constructs the *assignments* – probabilistic until fully finalized – for the whole study area and for all land uses, and suggests these to the participants.

Step 4: Every participant assesses the suggested assignments by determining his or her satisfaction. When sufficiently satisfied, a participant will express his or her *acceptance* to the initiator, who then proceeds according to step 5. When not sufficiently satisfied, the reply to the initiator depends on the prevailing stage of the process:

4.1: When the suggested assignments were not fully finalized, the

participant will use the opportunity to *retry* in an attempt to have the initiator adjust the assignments to make them sufficiently satisfactory. For this the agent will redefine his or her claims by returning to step 2 and using the suggested assignments – which show the most recent intentions of the initiator and as such provide the most probable future state – to update his or her goal state belief and to reconsider his or her claim strategy.¹³

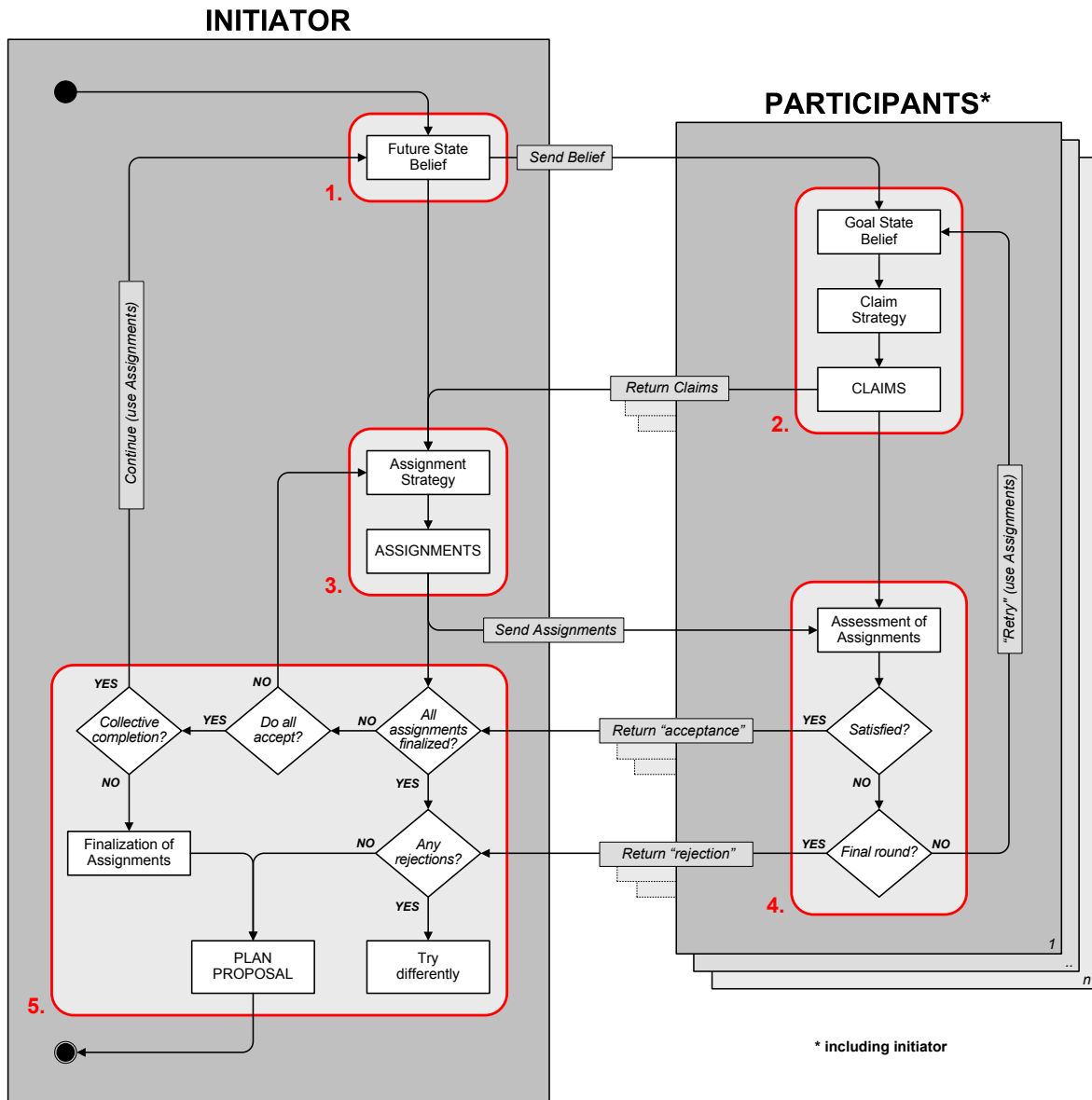


Figure 5-7. Interaction protocol: simultaneous contributions

¹³ In reaction to redefined *claims*, the initiator redefines his or her *assignments* in order to increase the chance that the participant will be sufficiently satisfied.

4.2: If the suggested assignments were already fully finalized, the process is in its last round and the opportunity to retry no longer exists for the participant. Consequently, the fact of not being sufficiently satisfied leaves the participant no other option than to return his or her *rejection* to the initiator.

Step 5: The initiator acts in accordance to the messages – acceptance, retrial or rejection – that he or she receives from the participants and the current stage of the process. As long as the suggested assignments were not fully finalized, the initiator will not receive any notices of rejections (see step 4.1). At worst, the initiator will receive new claims from all participants and no notices of acceptance, while the best case would be that all participants accept the assignments. Consequently, the initiator checks whether or not this best case is occurring. If not, he or she redefines the assignments taking into account the redefined claims he or she received, which means returning to step 3. If all participants did accept the non-finalized assignments, the initiator advances the process by using either one of the following two options in an attempt to *complete* the plan proposal:

5.1: *Collective completion:* the initiator uses the accepted (but still probabilistic) assignments as an updated $Bel\{SAF\}$ and returns to step 1 to continue the process in full cooperation with the involved participants.

5.2: *Individual completion:* the initiator takes the all-participants-acceptance as been given the freedom to finalize the remaining cells according to his or her personal judgement, i.e., without any further involvement of participants. In this case the plan proposal will be completed, although it may bring more difficulty to successfully pass the subsequent verification and receive the status of alternative plan (section 5.3.3).

In the case that fully finalized assignments were suggested, the initiator checks whether all participants returned notices of acceptance. If so, the development of his or her plan proposal has been successfully finished. If not, the attempted retrials of one or more participants failed, resulting in one or more compelled rejections to the final assignments (see step 4.2). In that case, the initiator will need to try *alternative ways* to obtain a plan proposal, for instance:

- Negotiate cell swaps, i.e. exchanging assigned cells between participants in order to have each party be sufficiently satisfied, which implies that

one or more accepting agents agree with a lower (but still acceptable) satisfaction if that can make rejecting agents to accept.

- Rerun the process, either based on adjusted settings such as in the assignment parameters (default weight set, intensity of iterative weight adjustments, and so on; section 5.3.2.2) or based on another protocol.
- Apply another method to make assignments for one or more land uses by calling in the help of *domain consultants* or using standards and rules-of-thumb, instead of in full cooperation with the responsible participants (section 4.4).

5.4.2 Sequential contributions

A distinctively different interaction protocol would be one that is based on sequential contributions, implying the generation of plan proposals by having initiators request and process the claims for the different land uses one by one, instead of all at the same time. An initiator then intends to reach agreement over the assignments with a selected participant before turning to the next. In this setting, participants only become involved in the process at the time the initiator requests for their subsequent contributions. Hence, the relative importance the initiator gives to the various land uses is not only embedded in its assignment strategy but also, and more obviously, in the order in which he or she calls upon the participants.

In comparison with the interaction protocol of simultaneous contributions, the process differs regarding two aspects (Figure 5-8). First, there is an obvious difference in step 1, when the initiator uses the current future state belief, $Bel\{SAF\}$, to value the importance of the different remaining contributions from the viewpoint of his or her own land use in order to determine the *order-of-turns* to apply in the process. He or she then sends the $Bel\{SAF\}$ to the *participant-at-turn*, requesting that agent's claims. Second, acting in accordance to the current stage of the process in step 5 implies that the initiator checks whether or not there are other participants remaining when the assignments for the participant-at-turn are completely finalized. When there are, the initiator goes back to step 1 in order to call upon the next participant, with the assignments forming the new $Bel\{SAF\}$. In the case of having received a notice of rejection from the participant-at-turn while still other participants remain, the rejecting agent will be put in a 'suspended mode' and contacted after reaching agreement with a following participant. This is according to

the notion that agreement over the assignments can still be reached at that later stage, when the rejecting participant becomes aware that it might be necessary to lower its satisfaction threshold for the sake of common interests. At the moment that the last suggested assignments were completely finalized for all land uses, there will be no participants remaining and the initiator will see whether they all accepted. If so, the plan proposal has been drawn up successfully. If not, the initiator has to find other ways to acquire a completed plan proposal (section 5.4.1).

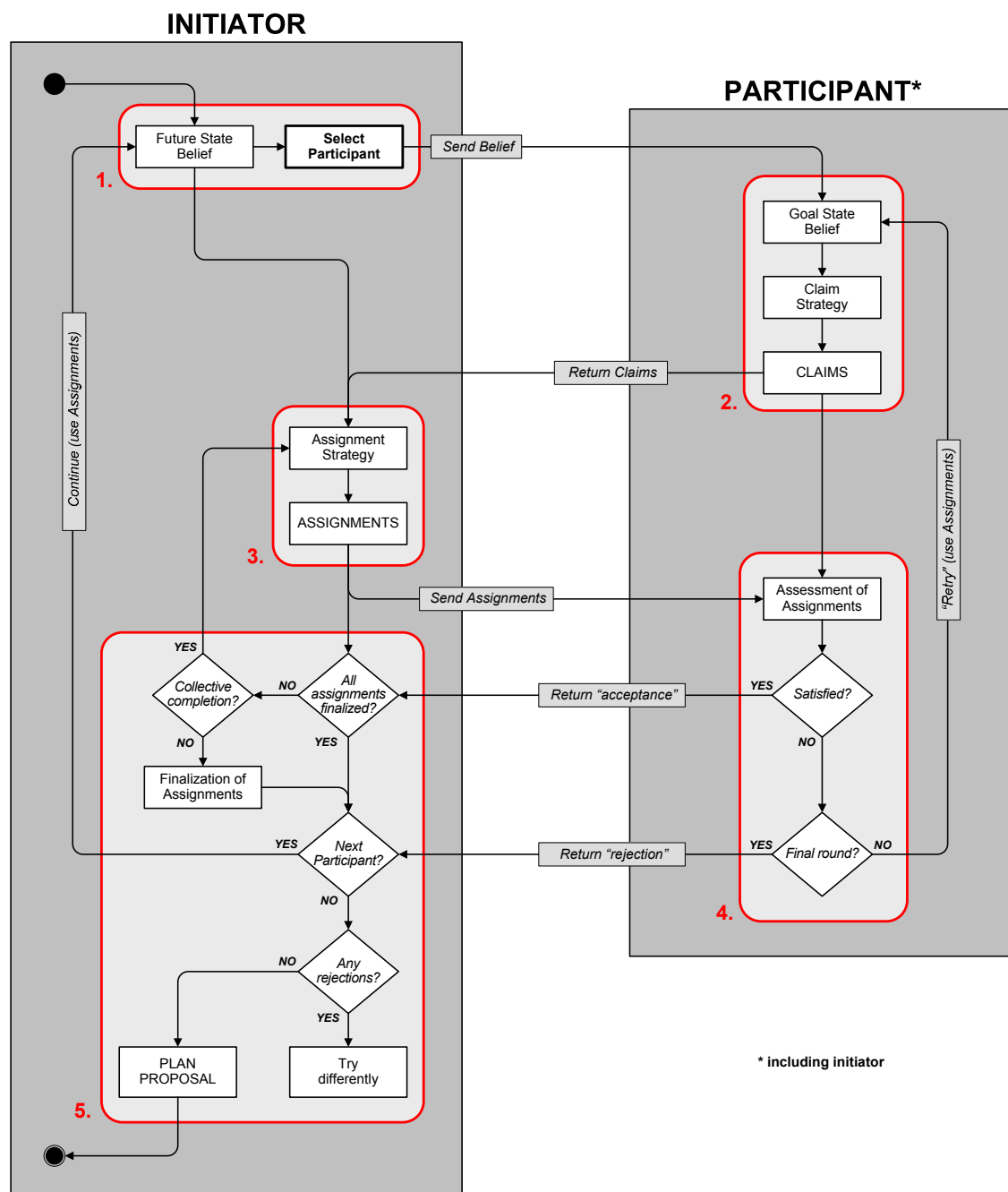


Figure 5-8. Interaction protocol: sequential contributions

5.5 CONCLUSIONS

Whereas multi-agent technology has already been widely acknowledged as a powerful tool for simulating space-time phenomena that emerge from the behaviours of individuals, households, firms, organizations, and so on, it is rather surprising to observe that attempts to further exploit the technology within the context of PSS remain limited in number even though the potential applications seem rather evident (section 3.2). Recalling the two principal requirements for PSS (section 2.2.2), there is a fundamental and instant need for finding ways in which the generation of alternative plans can be supported equally well as the subsequent evaluation of these plans. Furthermore, the crucial role of specialized knowledge in planning decision-making asks for flexible and intelligent means to incorporate expertise over a range of disciplines. An appropriate solution may be found in the utilization multi-agent technology that, through its concepts of anthropomorphy and modularity, makes it possible to represent the individual experts inside an urban planning team, give direct access to the specialized knowledge of these experts and provide the opportunity to simulate the interactive decision-making within an urban planning team for both generating and evaluating alternative plans. The addition of this 'layer of agents' (section 3.3.2) would bring PSS an important step closer to a complete representation of the urban system and, thus, provide planning practitioners with more powerful support in performing their professional tasks.

This chapter has specified a multi-agent model dedicated to the generation of alternative plans in order to draw attention to the opportunity of having multi-agent technology support this more creative *Design* phase of planning as well, supplementary to its apparently most obvious application in the *Choice* phase to serve forecasting and evaluation purposes (section 4.2). Without a doubt, variations on the presented model can come to one's mind, as well as further enhancements. Considering the novelty of this type of model, however, it can be considered equally important to allow demonstration and testing of the model in order to highlight the characteristics that make it different from conventional multi-agent applications in planning. For that reason, the next chapter will discuss the design and implementation of the model into a prototype application, after which an extensive series of tests is conducted to examine the properties of the model.

PART III
ILLUSTRATIONS

6 DESIGN AND IMPLEMENTATION OF *MASQUE* Alternative Plan Generator

This chapter describes the design and implementation of a prototype application for alternative plan generation based on the multi-agent model discussed in the previous chapter. First, a brief contemplation is provided on the question whether or not to use existing multi-agent shells for building this application. Then, the design of the instrument is described in terms of its structural components, followed by an explanation of how to operate the instrument.

6.1 INTRODUCTION

In order to demonstrate and test our multi-agent model for alternative plan generation (Chapter 5), it must be turned into an operational planning support instrument. This requires translation of the conceptual specifications into a logical and structural design, followed by the implementation of that design into a prototype application. Before that, however, it should be assessed whether there are tools available that could support these activities.

A large number of toolkits, shells, platforms, and so on, is currently available to ease the development of multi-agent applications. This illustrates the fact that multi-agent technology is being considered as a key solution in response to the ever-increasing demand for more advanced and complex software systems. It goes without saying that the rationale of these development tools is strongly influenced by the nature of the industries that generate the highest demands and have large budgets for research and development. As a consequence, the majority of available multi-agent development tools is customized to meet the typical requirements of multi-agent solutions searched for in areas such as distributed computer technology, e-commerce, and mobile telecommunication. Hence, they largely concentrate on the ability of agents to travel across information and communication networks to perform tasks on behalf of users – generally, retrieving and delivering information – and their ability to interact with each other over these networks according to specific protocols. Definitely, many tools demonstrate noteworthy aspects (e.g., Ricordel and Demazeau, 2000; Mangina, 2002), such as a highly GUI-driven approach, code generating functionality and process-wide support of building distributed multi-agent systems. Yet, the scope of each single tool tends to be narrow and brings

several kinds of limitations. Using it means designing agents according to a specification that closely matches that of the platform in order not to be condemned to changing unsupported source code. Additionally, an analogue in focus is required. When agent mobility is not an issue, such as in our planning support instrument, the use of platforms that specifically and mainly address this feature is far from evident. Moreover, due to the intended generality of the tools, the specification of agent properties and behaviour remains highly abstract. Finally, the majority of available tools are in fact still prototypes that serve experimental purposes and are under continuous development, which usually comes along with a lack of documentation and user support.

Besides the main focus on network-related agent mobility and interaction, other aspects and applications are also targeted. In the context of planning support, and urban plan development in particular, most interesting are the toolkits that focus on agent-based simulation, such as Swarm and Repast (e.g., Tobias and Hofmann, 2004). As mentioned earlier, agent-based simulation is concerned with modelling the behaviour of (large) populations of individuals who interact while moving around in a shared environment. The toolkits offer building blocks that can be used for creating instruments that provide insight into spatial-temporal phenomena, ranging from pedestrian behaviour (e.g., Schelhorn *et al.*, 1999) to urban sprawl (e.g., Rand *et al.*, 2002). Their support of simulating large numbers of relatively small-sized ('reactive') agents makes them highly suitable to serve the purpose of forecasting the effects of alternative plans or policies. Far less, however, they meet the requirements for more elaborated reasoning and non-moving action of agents such as in our proposed planning support instrument.

The discrepancies between the concepts of our model and the rationale of available multi-agent development tools are plentiful and, thus, both the usefulness and applicability of such tools can be questioned. Hence, it was decided to design and build the intended instrument without the use of proprietary multi-agent tools. The important advantage it brings us is that design and implementation can be in complete conformance with the semantics of our conceptual model.

6.2 DESIGN

The conceptual elements defined in our model – agents, mental constructs, cells, message passing, and so on – make an object-oriented foundation self-evident. Accordingly, the structural design chosen for our prototype application consists of three intertwined parts:

the environment that agents are dealing with, the team organization they form, and the mental constructs they use.

6.2.1 Environment

In the process of developing an urban plan, the environment that agents are dealing with is a raster representation of the study area and its surroundings. Figure 6-1 shows the static structure of the environment, i.e., the set of object classes that defines the environment and enables interaction with it. For each class the diagram indicates the *fields* that define the properties of objects of that class and the *methods* that define the operations that can be performed on them.

The environment is formed by a `Raster` that has a certain width and height and consists of one or more square cells of a certain size. Each `Cell` belongs to one particular raster in which it occupies a geographical location that is uniquely identified by a combination of column number and row number. Because the environment contains both the study area and its surrounding areas, a cell has a field that indicates whether or not it belongs to the study area. The data describing the characteristics of a cell (e.g., land use) are stored in a list of one or more attributes. An `Attribute` is defined by a combination of a name and a value.

The `Raster` methods provide agents with the tools to retrieve information about sizes: the width and height of the raster, the width and height of the study area, and the size of cells (i.e., resolution). Furthermore, the methods `getStudyAreaXshift()` and `getStudyAreaYshift()` enable agents to retrieve the position of the study area within the raster. The `Cell` methods give access to the position (column and row

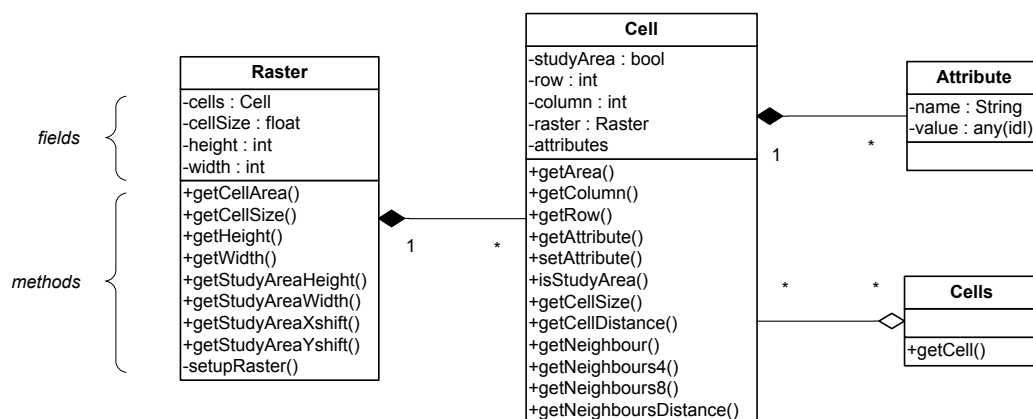


Figure 6-1. Static structure of the environment (UML notation)

numbers) of any cell within the raster (or study area), allow storing and retrieving cell attributes, and get the shortest distance to any other cell. Additionally, methods are provided to identify neighbouring cells: `getNeighbour()` returns the cell that is located at the given relative coordinates; `getNeighbours4()` gives the four adjacent cells with which a cell shares a border (i.e., the *Von Neumann* neighbourhood); `getNeighbours8()` identifies the eight adjacent cells with which either a border or a corner is shared (i.e., the *Moore* neighbourhood); and `getNeighboursDistance()` provides an agent with all the cells that are located within a given radius. The last three methods mentioned return a `Cells` object (i.e., a set of cells), as their search results always concern more than one cell. Its `getCell()` method can be used to retrieve any cell from the set.

6.2.2 Agent organization

The agents are organized in a team that is dedicated to creating a set of alternative plans. Figure 6-2 displays the static structure of the agent organization. When the user wants a set of alternative plans to be developed, a `Team` is formed (or initialized) and handed the required information to generate alternative plans: an area description (i.e., a `Raster`) and the tasks defined for each land use or agent (i.e., a `PlanProgram`). Based on the latter information the required agents are created in the `LandUseAgentFactory` and added to the team as members. Every agent in the team is a `LandUseAgent`, who represents one of the land uses listed in the `LandUse` enumeration, whose claiming and assigning behaviour is captured by sets of parameters and weights, and whose decision rules to judge the expected utilities of cells are contained in a decision model. A team can have one or more plansets, where each `PlanSet` belongs to one specific team and can consist of one or more plans. A `Plan` is a raster representation of the environment showing the land use configuration for the study area within its surroundings, as incrementally drawn up by the agent that initiated its development.

The aforementioned team formation is executed by means of the `initialize()` method of the `Team` class, which makes the call to the `LandUseAgentFactory` to create the required agents and add them to the team's list of members. The team's `run()` method starts the alternative plan generation process by systematically appointing each agent as initiator once. When an initiator has completed his or her plan, the plan will be

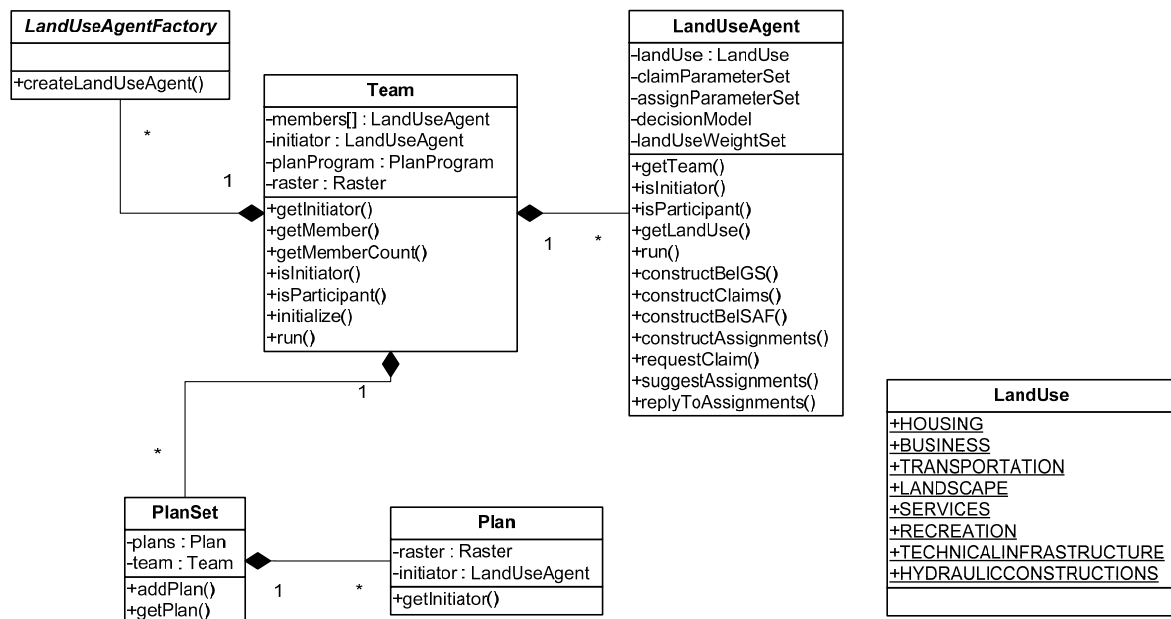


Figure 6-2. Static structure of the agent organization (UML notation)

handed to the team and stored in the current planset. The `team` class also provides methods to enable the identification of agents by their roles during the process. The `LandUseAgent` class, having similar methods for identification, provides a `run()` method that describes the interaction protocol according to which an agent, in the role of initiator, will organize the plan development. Furthermore, the class contains methods with which an agent forms his or her mental constructs (goal state beliefs, future state beliefs, claims and assignments), and methods for communicating claims and assignments to other agents.

6.2.3 Mental constructs

The mental constructs agents use for reasoning and acting are categorized into two groups (Figure 6-3). On the one hand, there are three constructs that are specializations of a `ValueGrid`, a data structure in which an agent stores floating point values that are results of calculations: either expected utilities (ranging between 0 and 100) or probabilities (ranging between 0 and 1). The size of a `ValueGrid` is equal to that of the raster representing the environment (section 6.2.1), while the values it contains always concern the land use that the agent represents (section 6.2.2). On the other hand, there are two mental constructs an agent uses when playing the role of initiator that are specializations of a `LandUseGrid`, which is a set of one or more `valueGrid` objects, where the size of the set is equal to the number of land uses (or agents) involved in the

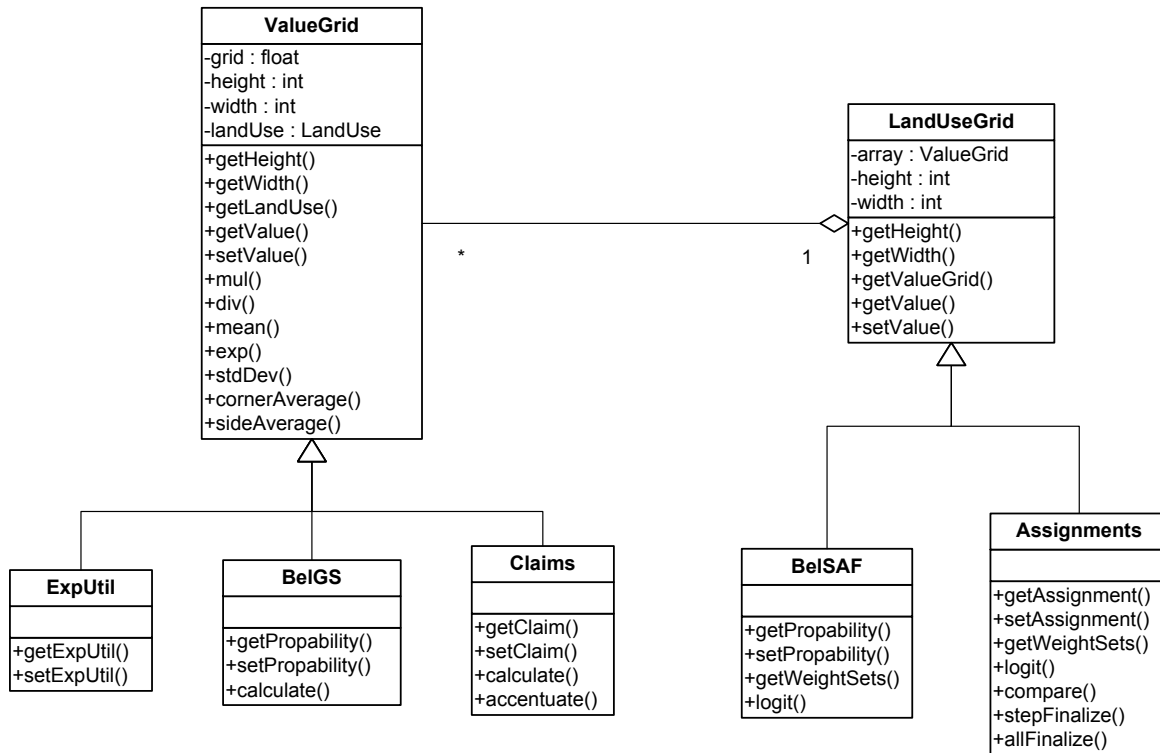


Figure 6-3. Static structure of the mental constructs (UML notation)

process. The data stored in a `LandUseGrid` are cell-based probability distributions across the contained `ValueGrid` objects. This implies that all the values concerning the same cell together always sum to one.

Any type of `ValueGrid` provides agents with methods to retrieve and store data and to perform various mathematical operations on its values, including methods to find the average values of neighbouring cells at the corners and sides of a specified cell. Moreover, both the specialized `BelGS` and `Claims` classes have a `calculate()` method with which an agent converts inputs (expected utilities and beliefs, respectively) into claim probabilities, while accounting for his or her desired level of clustering. `Claims` objects also provide an `accentuate()` method that allows an agent to stress (or unstress) his or her claims when feeling encouraged (or discouraged) to do so.

Given the fact that a `LandUseGrid` consists of `ValueGrid` objects, an initiating agent simply uses the latter's methods to store and retrieve data. In addition, he or she can also retrieve a complete `ValueGrid` from a `LandUseGrid`. The specialized `BelSAF` and `Assignments` classes both provide `getWeightSets()` and `logit()` methods. With the former method an initiator can interpret the combined characteristics of the received claims to (optionally) adjust his or her weight set for these claims on a cell-by-cell basis. The `logit()` method represents the approach that initiators take to determine

their assignment probabilities for the study area, given the received claims and their (default or adjusted) weight sets. In the iterative process of claiming and assigning, initiators will finalize cells – i.e., permanently assign cells to land uses – whenever the process stabilizes (section 5.3.2.2). To determine whether a cell finalization is required initiators use the `compare()` method, while for performing the actual finalization they use the `stepFinalize()` method. The `allFinalize()` method is applied when individual completion of a plan is chosen.

6.3 IMPLEMENTATION

Given the design as discussed above, our multi-agent model for alternative plan generation has been implemented in a Java application, using Borland® JBuilder™. The prototype instrument, named ‘*MASQUE* Alternative Plan Generator’, allows for having four agents – specialists regarding housing, business, green space and services – generate alternative plans for rasterized environments with study area surroundings that are configurations of one or more of the same land uses. The interaction between agents takes place according to the protocol of simultaneous contributions (section 5.4.1).

The user interface of *MASQUE* Alternative Plan Generator consists of the following four components (Figure 6-4):

- ‘Menu bar’: provides access to the various functionalities.
- ‘I/O frame’: enables managing program input and output.

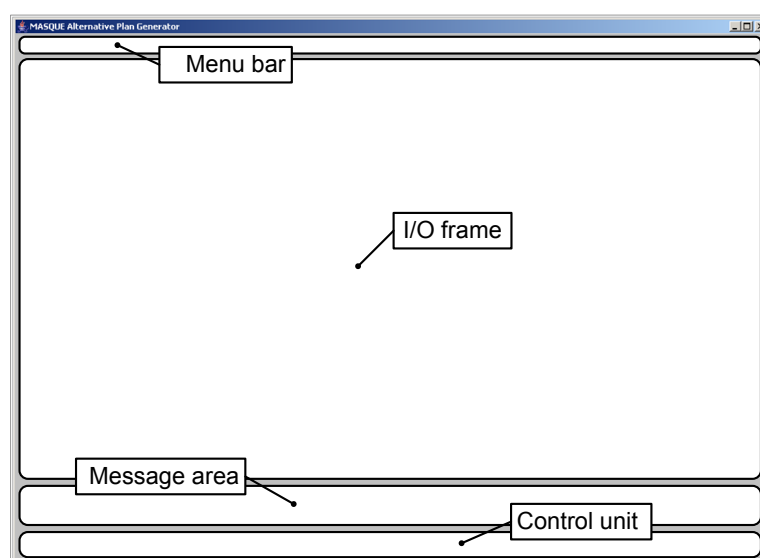


Figure 6-4. *MASQUE* Alternative Plan Generator: GUI components

- ‘Message area’: displays information about parameters and user actions, such as opening and saving files.
- ‘Control unit’: gives direct access to running the program.

Below it is explained how users can operate the instrument, starting with a concise description of the file format required as model input. Then, it is clarified how to set and adjust the parameters and rules that define the behaviour and decision-making of agents. Finally, it is described how the underlying model can be executed and what functionality for examining results is provided.

6.3.1 File input

The only operation that is really required to run the model and, thus, to let the team of agents generate a set of alternative plans, is opening a ‘project input file’ by means of selecting the option “File/Open project...” in the menu bar or by clicking the “Open project...” button located on the left-hand side of the control unit (Figure 6-5). When a file is opened, the filename will be displayed next to the button.¹⁴ Simultaneously, the “GENERATE PLANS” button on the right-hand side of the control unit will be enabled, implying that the alternative plan generation could already be executed based on default settings.

Technically, a project input file is a tab delimited text file with a format that shows great similarity with the *ASCII ArcGrid* format for representing a raster map as a text file. The file needs to contain (i) an *area description* that gives the sizes of the area and the land uses currently located in the surroundings of the study area, and (ii) a *plan program* that states the number of cells to be claimed for each land use by corresponding agents. Figure 6-6 displays an example for a case in which the study area measures 1500 meters by 1500 meters and the surroundings form a belt of 600 meters wide around the study



Figure 6-5. Program control unit: opening a project input file (example)

¹⁴ It is important to note that the folder from which the project input file is uploaded will be used for storing various output files as well. This implies that different projects, and also runs of the same project under different settings, should be organized in separate folders.

<i>Area size and resolution</i>	ncols	9							
	nrows	9							
	cellsize	300							
<i>Land use data</i>	rasterinit								
		1	1	1	1	4	4	4	4
		1	1	1	1	1	4	4	4
		5	5	0	0	0	0	0	4
		1	1	0	0	0	0	0	4
		1	1	0	0	0	0	0	1
		1	1	0	0	0	0	0	1
		1	1	1	1	1	1	2	1
		1	1	1	1	1	1	2	1
<i>Task sizes</i>	task_H	13							
	task_B	3							
	task_G	7							
	task_S	2							

Figure 6-6. Format of a project input file (example)

area, making the total environment to measure 2700 meters by 2700 meters. Three blocks of information are to be distinguished in a project input file. In its first three lines, the file gives the area size and resolution to apply. In given example, the environment is divided into 81 cells (9 columns by 9 rows), each with a size of 300 meters by 300 meters. Then, in a number of lines that is equal to the number of specified rows, the land uses are provided for each cell in the environment, using IMRO land use codes (Ravi, 2000). The cells making up the study area are given an initial value of zero, indicating that their land use is ‘to be determined’. The last block consists of the plan program information, indicating which agents are to be involved in the process and what number of cells they each have to claim (their ‘task size’). The sum of these task sizes ought to be equal to the number of zero values in the ‘land use data’ block (here, 25 cells).

6.3.2 Behaviour of agents

The settings and outputs of the model are organized in the I/O frame of the application. This frame consists of three tabs of which the first one – labelled ‘Behaviour’ (Figure 6-7) – provides the fields to set all the necessary parameters (in rows) for defining the individual behaviour of agents (in columns). For a brief explanation of the meaning of a parameter and its permitted value, users can click on the question mark icon near that parameter’s label. The explanation is then displayed in the program’s message area.

When starting the program, the behaviour tab is automatically loaded with default parameter values, which can be restored at any time by means of the menu option “File/Restore Defaults/Behaviour Parameters”.

Two sets of behavioural parameters are distinguished. The first set (on top half of tab; Figure 6-7) consists of parameters that define the behaviour of agents with respect to claiming cells for their land use (section 5.3.2.1):

- ‘Clustering intensity’: the level of clustering an agent desires (0 = none; 1 = maximum).
- ‘Neighbour distinction’: the relative weight an agent uses for neighbouring cells at the sides and corners of a cell when including his or her desired level of clustering [e.g., ‘0.6’ indicates that for side neighbours a weight of 0.6 is applied and a weight of 0.4 (i.e., $1 - 0.6$) for corner neighbours].
- ‘Opportunism’: the degree of opportunism an agent applies (0 = fully compliant; 1 = fully opportunistic). When an agent plays the role of initiator, this parameter is automatically set to zero for that agent.

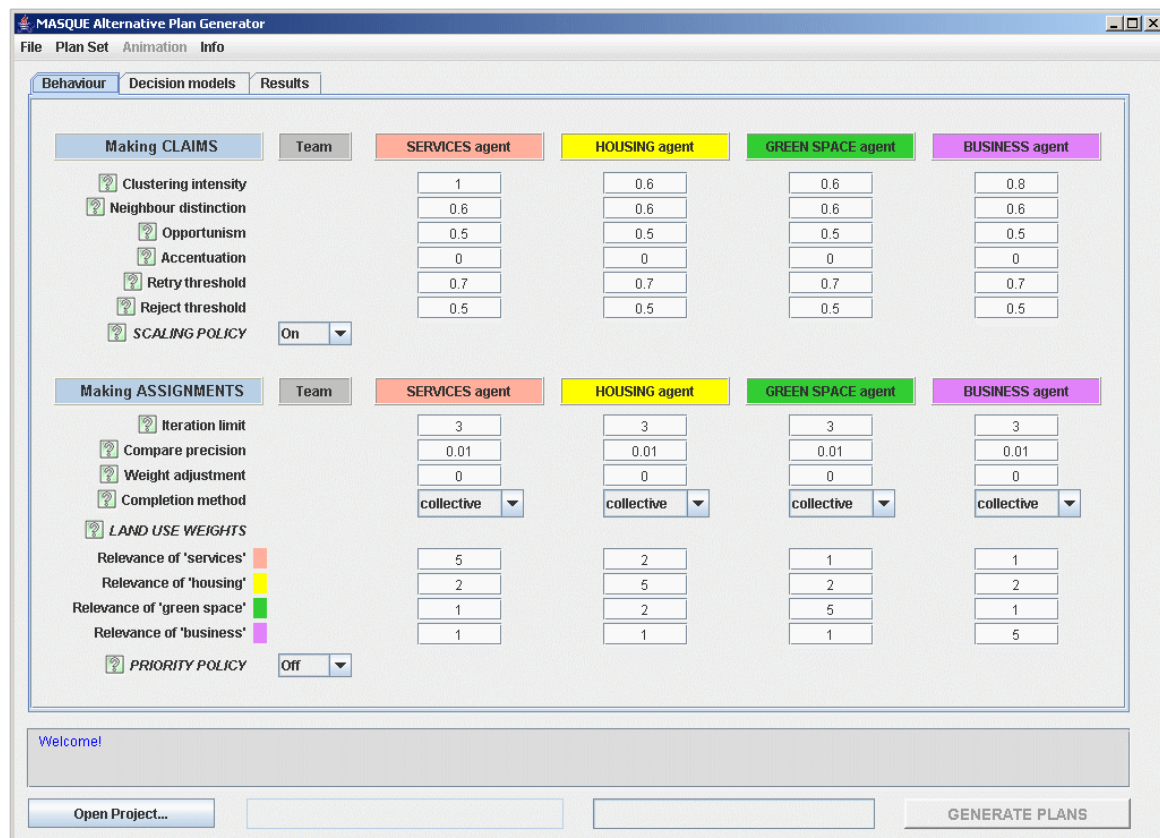


Figure 6-7. Setting the behaviour of agents (incl. default values)

- ‘Accentuation’: the extent to which an agent stresses its claims (0 = none; 1 = maximum).
- ‘Retry threshold’: an agent’s satisfaction threshold for deciding whether to reply with ‘accept’ or ‘retry’ in response to assignments as long as these are not completely finalized (0 = minimum; 1 = maximum).
- ‘Reject threshold’: an agent’s satisfaction threshold for deciding whether to reply with ‘accept’ or ‘reject’ in response to assignments that are completely finalized and thus have reached their final state (0 = minimum; 1 = maximum). Logically, this threshold is to be set lower than or equal to the retry threshold.

All these claim-related parameters can be adjusted for every agent separately. This is in contrast to the ‘scaling policy’ parameter that is to be set for the whole team, indicating whether or not agents should scale their claims to have their claims sum up to their task size. When this policy is operative, an agent’s claims should be interpreted as a unified set addressing the whole study area. When inoperative, the claims remain independent of an agent’s task size and claims should be interpreted on a cell-by-cell basis.

The second set of parameters (see bottom half of tab; Figure 6-7) describes the behaviour of agents when they play the role of initiator and assign cells to land uses based on the claims they receive from other agents (cf. section 5.3.2.2):

- ‘Iteration limit’: the maximum number of subsequent rounds an initiator allows before enforcing the finalization of a (next) cell (any integer number larger than zero).
- ‘Compare precision’: the precision an initiator applies for comparing new assignments with previous ones. When the difference is smaller than this value, a next cell finalization will be made (any floating point number larger than zero).
- ‘Weight adjustment’: the degree of weight adjustment an initiator applies when renewing assignments based on whether or not cells belong to how many agents’ best sets. (0 = none; 1 = maximum).
- ‘Completion method’: the method an initiator applies when everyone accepts the assignments while cells remain to be permanently assigned to a land use. The choice is whether (‘collective’) or not (‘individual’) to involve the other agents in completing the plan.

- ‘Land use weights’: a series of weights, denoted as ‘Relevance of ...(land use)’ parameters, that indicate the default weight an initiator gives to the specified land use (0 = minimum; 5 = maximum). By default, the relevance of an initiator’s own land use is set to maximum.

With the ‘weight adjustment’ set to zero (= default), an initiator will always use his or her default land use weights for making assignments during the development of a plan proposal, implying that he or she will apply the same weights for every cell in every round. When the weight adjustment is set to a non-zero value, however, the entered default weights will function as a base from which the initiating agent will vary the applied weights per cell and per round based on a combined interpretation of the claims.

Finally, a ‘priority policy’ parameter related to making assignments is set at the level of the whole team of agents. This parameter indicates whether or not initiators are enabled to let the first cell finalization concern their own land use. When priority policy is applied, initiators themselves make explicit claims – i.e., they accentuate their claims to the maximum – until the first finalization has been performed, setting an initial direction for the development of their plan proposal.

6.3.3 Decision-making of agents

The second tab in the *MASQUE* Alternative Plan Generator’s I/O frame, labelled ‘Decision models’, is dedicated to the formulation of each agent’s private decision model with which he or she performs land suitability analysis in order to determine the expected utility per cell. As in the case of behavioural parameters (section 6.3.2), the default settings for decision models are automatically loaded when the program is started. The ‘Decision models’ tab organizes its settings in a second-order tabbed window, of which each tab describes one agent’s decision model. Figure 6-8 shows the settings for the services agent. The developed prototype makes a distinction between four categories of rules that an agent can include in his or her land suitability analysis, regarding:

- ‘Accessibility’: to determine the expected utility of a cell based on the shortest distance to a specified land use. The arguments used for these rules are a land use, a search radius and an objective function.
- ‘Adjacency’: to determine the expected utility of a cell based on the presence of a specified land use in the direct neighbourhood of that cell.

SERVICES agent - decision rules

Accessibility-based Utility

	weight	radius [m]	objective
Housing	2	400	seek (+)
Business	0	0	n/a
Green space	0	0	n/a
Services	2	400	avoid (-)

Soil-based Utility

weight: 0

Adjacency-based Utility

	weight	objective
Housing	2	seek (+)
Business	0	n/a
Green space	1	seek (+)
Services	2	seek (+)

Population-based Utility

weight: 2

radius [m]: 600

population threshold: 6000

density [dwelling/ha]: 30

household size: 2.3

Figure 6-8. Setting the decision model of the services agent (incl. default values)

The required arguments are a land use and an objective function.

- ‘Soil condition’: to determine the expected utility of a cell based on soil-related characteristics of that piece of land. For this aspect the expected utilities are (optionally) predefined in the project input file.
- ‘Population’: to determine the expected utility of a cell based on the population size within a given radius (‘catchment area’). The arguments needed are a search radius, a population threshold, and averages of housing density and household size.

The way in which the various rules constitute the agent’s decision model is formulated by setting a relative weight for each rule (Figure 6-8). Hence, the outcome of the decision model – an overall expected utility for each cell that forms a measure with which that cell can be compared to others in terms of relative attractiveness – is a weighted average of the expected utilities resulting from applying the separate rules to that cell.

6.4 RUNNING THE APPLICATION

After having opened a project input file, and (optionally) having adjusted the behaviour and decision-making of agents to meet ones requirements, the program is ready to generate a set of alternative plans. The underlying model is executed by clicking the



Figure 6-9. Program control unit: generating plans

‘GENERATE PLANS’ button that is located on the right-hand side of the control unit (Figure 6-9). As a response, the team of agents will be formed and agents will be systematically appointed as initiator, making each of them responsible for generating an alternative plan from a personal viewpoint. A progress bar in the control unit indicates the status of the process. The actual time that the team needs for generating a set of plans will depend on both project-related aspects (e.g., number of agents, size and resolution of the environment and precision and number of iterations used by initiators) and computer-related aspects (e.g., processor speed and available memory capacity).

When the process has been completed, the resulting set of alternative plans is displayed in the third tab of the program’s I/O frame, labelled ‘Results’. To illustrate, Figure 6-10 shows the output for a case where the study area measured 1500 meters by 1500 meters and a resolution (i.e., the height and width of a single cell) of 300 meter was applied. For each plan, the following information is provided for further examination:

- The initiator of the plan (shown in the header).
- A map representation of the plan’s final state.
- An overview of each agent’s evaluation of that final plan state.
- An animation of the incremental development of the plan (accessible through an “Animate” button).

The score with which an agent evaluates a plan is the result of that agent running his or her decision model based on the plan’s final state (as shown in the corresponding map). Thus, the score is expressed in terms of expected utility and takes a value between 0 and 100. In the illustrated case, the services agent, for instance, evaluates his or her self-initiated plan with a weighted overall score of 91. This score is a weighted summation of subscores¹⁵ that, as displayed, indicate the agent assessed the plan with a score of 100 for accessibility (‘acc’), 90 for adjacency (‘adj’) and 76 for population (‘pop’).¹⁶

For each initiator, two output files are automatically created during the process: one in which the claims from every agent for each cell in each round are recorded, and

¹⁵ Rules concerning soil-based utility (‘soi’) were not included in the example displayed here.

¹⁶ Chapter 7 will give further details about the interpretation of results.

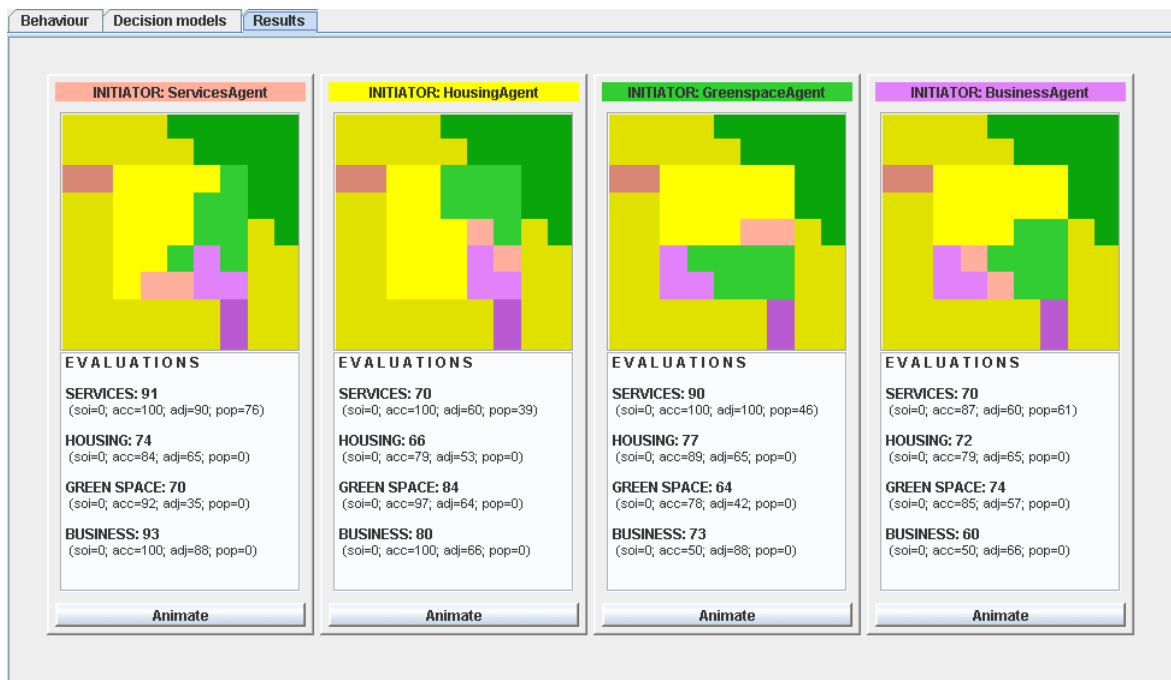


Figure 6-10. Generated set of alternative plans (example)

another in which the same is done for the assignments. Both these files are tab-delimited text files that have a header indicating the initiator it concerns and the type of data recorded. The files have the study area cells organized in columns and the probabilities (representing either claims or assignments) in rows for the subsequent rounds and involved agents. Figure 6-11 shows a part of the assignments data recorded by the services agent as an example. The first column indicates the round (where ‘-1’ denotes the initialization of assignments based on the proportional task sizes of agents), the second column the land use code of agents, while the third and following columns contain the assignment probabilities per cell for each round and each agent.

A click on the “Animate” button of a plan (Figure 6-10) causes a new window to appear in which the stepwise evolution of both claims and assignments concerning that plan can be studied (Figure 6-12), using the controls on the bottom right corner of the window to go forward and backward, either manually (‘step’) or automatically (‘auto’). Several animation options for displaying assignments and claims are provided in a drop-down list. The options available for *assignments* enable showing for every round:

- ‘Highest values’: for every cell the land use is displayed that has the highest assignment probability.
- ‘Finalized cells’: the cells are displayed that have been permanently assigned to a land use (probability = 1.0).

ServicesAgent is initiator - assignments.									
X	ID	0	0	0	0	0	1	1	1
Y	ID	0	1	2	3	4	0	1	1
-1	2	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12
-1	4	0.28	0.28	0.28	0.28	0.28	0.28	0.28	0.28
-1	1	0.52	0.52	0.52	0.52	0.52	0.52	0.52	0.52
-1	5	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08
0	2	0.077706	0.093692	0.100259	0.120086	0.13323	0.084267	0.09451	
0	4	0.219981	0.25654	0.255783	0.32339	0.365234	0.232297	0.235974	
0	1	0.656866	0.593235	0.585757	0.464746	0.398765	0.642288	0.615536	
0	5	0.045446	0.056533	0.058201	0.091777	0.102771	0.041148	0.05398	
1	2	0	0.103605	0.111831	0.123205	0.128576	0.100004	0.106336	
1	4	0	0.263074	0.276843	0.308253	0.325384	0.255549	0.262185	
1	1	1	0.567696	0.535648	0.48337	0.454734	0.580081	0.563492	
1	5	0	0.065625	0.075678	0.085173	0.091306	0.064366	0.067987	
2	2	0	0.108939	0.118905	0.128411	0.133484	0	0.11164	
2	4	0	0.264609	0.294241	0.320226	0.334524	0	0.27057	
2	1	1	0.556318	0.505963	0.463156	0.440096	1	0.544686	
2	5	0	0.070134	0.080891	0.088207	0.091897	0	0.073104	
3	2	0	0	0.124425	0.133134	0.137903	0	0.114633	
3	4	0	0	0.309701	0.334695	0.347196	0	0.270595	
3	1	1	1	0.480424	0.440474	0.420059	1	0.541397	
3	5	0	0	0.08545	0.091696	0.094842	0	0.073375	
4	2	0	0	0.124382	0.137475	0.142331	0	0	
4	4	0	0	0.30022	0.347877	0.362091	0	0	

Figure 6-11. Output of assignments data (example: plan by services agent)

- ‘Focus on land use’: for every cell the assignment probability is displayed for a specific land use based on a gradient-colour scheme to indicate five degrees of probability, {0.0-0.2, 0.2-0.4, 0.4-0.6, 0.6-0.8, 0.8-1.0}, from light to dark.

These options can be combined with one of the animation options provided for *claims*, which enable showing for every round:

- ‘Best sets’: the cells are displayed that belong to an agent’s *n* highest claim probabilities, where *n* equals the agent’s task size.



Figure 6-12. Single plan animation (example: plan by services agent)
 Note: claims show best sets, while consequent assignments show finalized cells

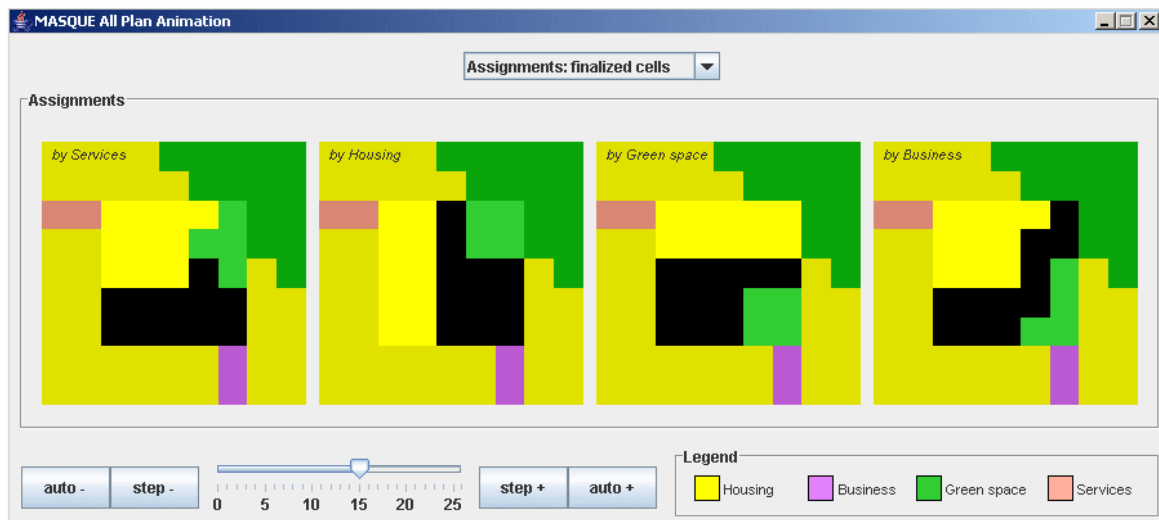


Figure 6-13. All plan animation (example)
Note: assignments show finalized cells for each plan

- ‘Focus on land use’: for every cell the claim probability is displayed, using a gradient-colour scheme for five degrees of probability (see above) from light to dark.

The animations are prepared in real time by using the corresponding output files to determine the requested data for display (either one or more land uses or gradient-colours of a land use). A supplementary type of animation is available through the menu “Animation/All plans”, which will open a new window (Figure 6-13) in which the incremental development of all alternative plans in a set can be displayed simultaneously, showing assignment-related data based on either ‘highest values’ or ‘finalized cells’.

6.5 CONCLUSIONS

This chapter discussed the design and implementation of a prototype application based on the earlier specified multi-agent model for alternative plan generation. The application aims at providing users with the possibility of quickly developing small sets of alternative plans for purposes such as exploring the solution space of a project or its inherent opportunities and threats, and visualizing the effects of interdependencies between land uses (or agents). Hence, it intends to let planners gain information that can be helpful in their search for good alternatives.

In relation to the conceptual framework of *MASQUE* (Chapter 4) and the context of this thesis, the presented application will allow us to examine the operation of domain

agents according to the interaction protocol of simultaneous contributions (section 5.4.1). It should be noted that the implementation has been purely driven by the need to demonstrate the principles of (i) representing the experts within an urban planning team in terms of agents and (ii) simulating the cooperative decision-making in the team and, thus, to draw attention to the particularities of such a concept. This brings along two important implications with respect to its practical use. First, the application is simplified in the sense that only small sets of simple rules are used to implement agents' decision models. Second, the application has the form of a stand-alone tool that does not accommodate the other required types of agents – i.e., interface agents and tool agents – promoted earlier (sections 4.3.1 and 4.3.2). Hence, to let the application operate as a component in the overall system framework and be applied in practice, it will be necessary to upgrade the contained knowledge and arrange the agents' interaction with other types of agents. Furthermore, the application in its current state gives users direct access to all parameters that describe the behaviour and decision-making of agents, which might not be optimal for its usability and understandability to planning practitioners. At this stage of development, however, the application in its current setup fits our requirements as modellers and developers because it allows us to easily test the operation of the underlying multi-agent model in detail. The next chapter reports on this testing by means of a systematic series of analyses.

7 TESTING THE MODEL: SENSITIVITY ANALYSIS

This chapter describes a systematic series of tests performed on the MASQUE Alternative Plan Generator to determine whether the underlying multi-agent model is working as anticipated and whether the incorporated parameters and rules – defining respectively the behaviour and decision-making of agents – have the intended effects. As such, this will demonstrate how a multi-agent concept can serve more planning support purposes than micro-simulation alone.

7.1 INTRODUCTION

Numerical simulation can be considered as the most appropriate means – and perhaps the only available means – to test and explore the properties of models that describe generative phenomena (e.g., Epstein, 1999; Wu, 2002). For our model, this implies analysing its sensitivity by investigating the effects of systematically changing the settings of parameters and decision rules – i.e., sweeping parameter spaces – with the purpose to understand the relationships inherent to the model and the relation between model input and output. The developed application provides the means to do so and to address essential questions, such as:

- Can the model generate alternatives that are distinctive and useful?
- Is the model adequately sensitive to the characteristics of a project, i.e., the study area's surroundings and the plan program?
- Is the model sufficiently insensitive to aspects to which it should be, in particular the resolution used to represent the environment in a grid?
- What are the specific effects of agents' decision rules?
- What are the consequences when agents adjust their claiming behaviour?
- What will change when initiators behave differently with concern to determining their assignments?

Given the fact that this research serves *exploratory* purposes – in that it investigates the use of a multi-agent concept for generating alternative plans – rather than *explanatory* purposes, the required sensitivity analysis can be performed on a hypothetical case. This, however, is under the condition that the case is not oversimplified in terms of, for

instance, interdependencies and possible conflicts of interest between agents. Conversely, a certain degree of simplicity is in fact required to enable tracking the consequences of changing parameter settings on the performance of plans, from the viewpoint of both individual agents and the team. Because the same holds for the level of detail embedded in agents' decision rules, a limited number of simple rules per agent is desired.¹⁷

The testing of our model is performed by means of a 'benchmark', i.e., a specified case that functions as a basis to which all results are consistently compared in terms of plan outcomes (i.e., their spatial patterns), plan evolution (i.e., their incremental development), and plan performance (i.e., the evaluation scores calculated by agents).

7.2 BENCHMARK TEST CASE

With the *MASQUE* Alternative Plan Generator in mind, defining a benchmark implies setting up a project – i.e., creating an environment, defining an urban planning team consisting of agents, and formulating the task of each agent – as well as defining the behaviour and decision-making of every agent in the team.

7.2.1 Environment and team

An environment is considered that consists of a study area measuring 128 hectares and surroundings that form a belt of at least 600 meters wide around the study area (Figure 7-1). The assumption of a rasterized space (section 5.3) implies that the environment consists of equally sized square land parcels ('cells') of which number and size depend on the resolution chosen. The default resolution is set to 200 meters, meaning that each cell measures 200 x 200 meters, the study area consists of 32 cells and the surroundings is at least 3 cells wide. The data describing cell properties remains restricted to the attribute "land use". The figure depicts how four land uses – housing, business, green space and services – are located around the study area. Obviously, for cells located in the study area itself the land use is still undetermined at the start.

To allow for multiple interdependencies among land uses, the urban planning team includes four agents¹⁸, each appointed to claim land for a specific land use: an housing

¹⁷ The degree to which individual behaviour of agents actually reflects the decision-making of the represented human experts is outside the scope of this research.

¹⁸ These are all the agents implemented in the prototype application (section 6.3).

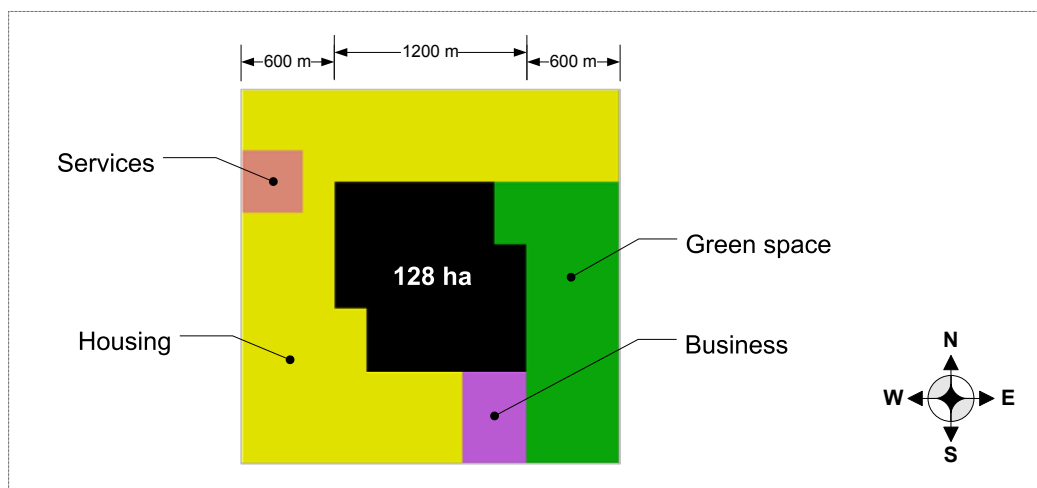


Figure 7-1. Benchmark test case: study area (in black) and its surroundings

agent (to claim 68 ha for housing), a business agent (16 ha), a green space agent (36 ha) and a services agent (8 ha). Proportionally, these land use demands approximate the average plan composition in Dutch planning practice.

7.2.2 Agent behaviour

In the role of initiator, agents apply sets of land use weights, ranging from *no importance* (= 0) to *high importance* (= 5), to process the claims expressed for the various land uses. The default weight sets (Table 7-1) are based on the assumption that initiators value their own contributions as most important; the remaining weights are intuitively set.

In the process of generating alternative plans, agents demonstrate behaviour for making claims (including assessing assignments) and making assignments that can differ individually. Each of these categories of behaviour is captured by a set of parameters of which purpose and operation have been described in Chapter 5.

Table 7-1. Benchmark test case: land use weight sets of initiators (default settings)

Initiator	Claims for land uses			
	Services	Housing	Green space	Business
Services agent, S	5	2	1	1
Housing agent, H	2	5	2	2
Green space agent, G	1	2	5	1
Business agent, B	1	1	1	5

The behavioural settings for the benchmark are displayed in Table 7-2. The clustering intensity parameter (ϕ) is varied per agent, reaching maximum (= 1) for the services agent and set lowest (= 0.6) for both housing agent and green space agent. All other behavioural parameters are assumed to be equal for all agents in the team:

- As participants, they all make a modest distinction between side and corner neighbours as part of clustering inclusion ($\gamma = 0.6$). Additionally, they choose a mixed claim strategy, adding equal value to their two beliefs ($\theta = 0.5$), while not accentuating their claims ($f_c = 0$). When assessing assignments, they use the same thresholds to make accept/retry and accept/reject decisions (0.7 and 0.5, respectively). Finally, the team-defined scaling policy is operative, having all agents make claims dependent on their task sizes.
- As initiators, all agents perform cell finalizations by using the same iteration limit ($i_{lim} = 3$) and critical minimum ($\Delta A_{min} = 0.01$), while none of them uses weight set adjustment ($f_a = 0$) when making assignments. Furthermore, the method chosen for completing a plan is collective for all agents, while the priority policy is not operative.

Table 7-2. Benchmark test case: agents' behaviour (default settings)

Agent	Claiming*						Assigning**		
	Clustering		Strategy		Assessing		Iteration limit	Compare precision	Weight adjustment
	Intensity	Neighbour distinction	Degree of opportunism	Claim accentuation	Accept/retry threshold	Accept/reject threshold			
ϕ	γ	θ	f_c	S_{min}^{rtr}	S_{min}^{rjt}	i_{lim}	ΔA_{min}	f_a	
Services agent, S	1	0.6	0.5	0	0.7	0.5	3	0.01	0
Housing agent, H	0.6	0.6	0.5	0	0.7	0.5	3	0.01	0
Green space agent, G	0.6	0.6	0.5	0	0.7	0.5	3	0.01	0
Business agent, B	0.8	0.6	0.5	0	0.7	0.5	3	0.01	0

* Scaling policy = 'on'

** Completion method = 'collective'; Priority policy = 'off'

7.2.3 Agent decision-making

The agents use personal decision models consisting of rules derived from Dutch planning standards (in Dutch: ‘planologische kengetallen’) as much as possible (Table 7-3). The benchmark assumes that each agent uses at least one rule related to accessibility and one related to adjacency (section 5.3), where ‘accessibility’ refers to the effect of having a specific land use located at a certain distance, while ‘adjacency’ concerns the effect having a specified land use located in the direct neighbourhood.¹⁹

For example, the housing agent looks at accessibility of green space based on a primary radius of 400 meters. This means (Figure 5-2) that he or she will determine (i) the probability of finding green space within this primary radius, (ii) the probability of finding it within a distance of twice the primary radius, and (iii) the probability of finding it at even a longer distance (or not at all). Because the agent considers accessibility of

Table 7-3. Benchmark test case: agents’ decision models (default settings)

Agent	Accessibility				Adjacency			Population			
	Weight	Land use	Objective	Radius (m)	Weight	Land use	Objective	Weight	Radius (m)	Minimum population	Average density and household size
Services agent, S	2	H	+	400	2	H	+	2	600	6000	{30, 2.3}
	2	S	-	400	1	G	+				
					2	S	+				
Housing agent, H	1	B	+	800	2	B	-				
	1	G	+	400	2	G	+				
	2	S	+	400							
Green space agent, G	2	H	+	800	1	B	-				
	1	G	+	400	1	S	+				
Business agent, B	2	B	+	600	2	B	+				
					1	G	+				

¹⁹ The notion of ‘direct neighbourhood’ is used to prevent that adjacency rules are affected by the resolution applied (section 5.3). It is considered to be the area within a radius that is half of the smallest radius used in the entire collection of (default) agent decision rules. Hence, here it is the area within a 200 meters radius.

green space as a positive contribution to his or her expected utility, the higher the probability of finding green space nearby a cell, the higher the expected utility of that cell. Simultaneously, the housing agent judges adjacency of, for instance, business as negative, which means (Figure 5-3) that finding business in the direct neighbourhood of a cell will lower the expected utility of that cell. Besides accessibility and adjacency rules, the services agent uses a population-based rule. With this rule he or she judges to what extent the (expected) population within a radius of 600 meters around a cell meets the required minimum of 6000 inhabitants (assuming an average housing density of 30 dwellings per hectare and an average household size of 2.3 persons). The better the expected population proportionally relates to the required minimum population, the higher the expected utility (Figure 5-4). The rules comprising an agent's decision model are applied to every cell in the study area and the total expected utility of a cell is calculated as the weighted average of these rules' outcomes for that cell.

7.2.4 Benchmark output

Running the *MASQUE* Alternative Plan Generator for the benchmark reveals that the discussed settings lead to some overlaps between agents' interests. Figure 7-2 shows the initial best set of cells for each agent, dependent on the number of cells an agent requires. These are the claims before initiators make their first assignments and, thus, they form the situation from which every alternative plan will have the potential to develop differently.

The resulting set of alternative plans demonstrates a clear distinctiveness between plans regarding both land use configuration and performance (Figure 7-3). This already indicates that the viewpoint approach taken in the underlying model – i.e., having agents initiate their own plans and incrementally draw up these plan from their personal viewpoint based on each other's contributions – is noticeably effective. Interesting to note, actually, is that initiators do not (necessarily) evaluate their own plans best. This relates to the fact that even initiators are facing uncertainty regarding the plan outcome due to the cooperative nature of the development process. Hence, plans are not simply one-sided optimizations of initiators' interests but development tracks along which initiators constantly take others' interests into account as well. For any agent, participation in the development of plans initiated by others can thus be considered equally rewarding as personally initiating a plan.

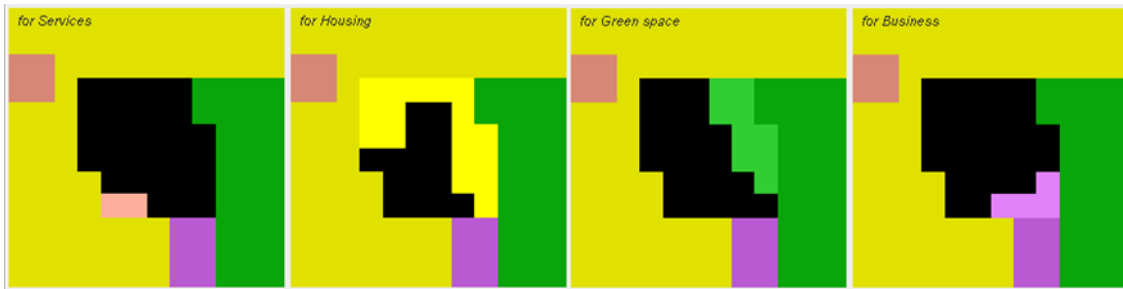


Figure 7-2. Benchmark test case: initial best sets of agents [from left to right, services agent (2 cells), housing agent (17 cells), green space agent (9 cells) and business agent (4 cells)]

When interpreting the plans' land use configurations, some aspects appear that are unanticipated from the viewpoint of single agents. In particular, the fact that services can end up at a location that is near to the existing services area in the northwest of the surroundings contradicts with one's expectation, especially when this also occurs in that agent's own plan. Apart from the abovementioned equality of initiators and participants, it suggests that, within the services agent's decision model, the rule expressing a preference for avoiding nearness to other services areas might be lacking the power to ensure a more intuitive location of the land use in every plan. Furthermore, it might indicate a procedural inequality related to differences in task size between agents.²⁰

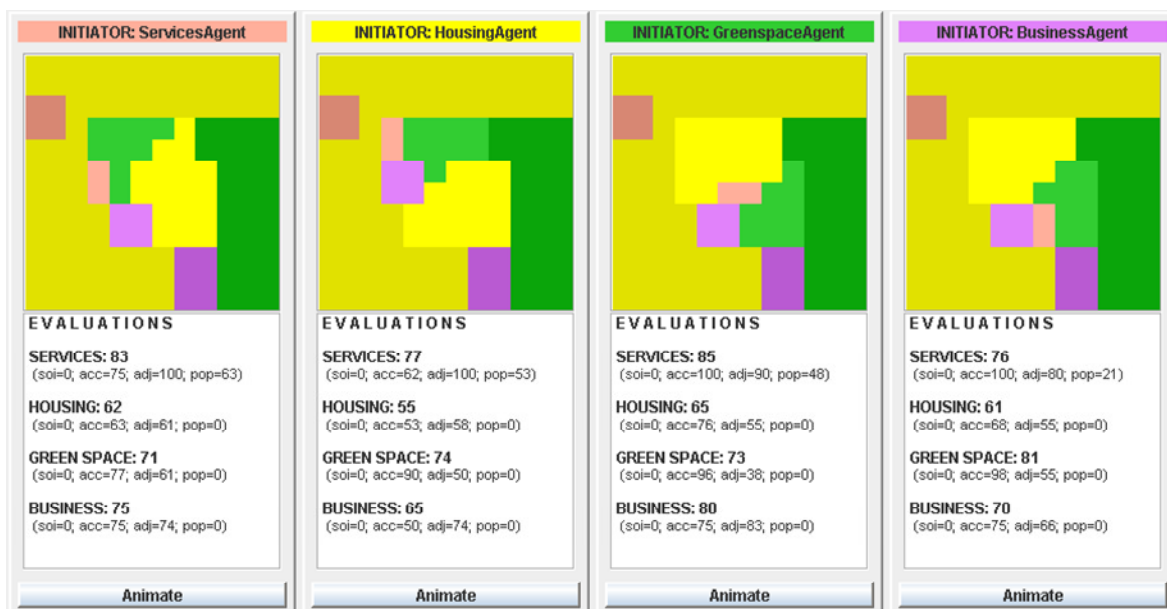


Figure 7-3. Benchmark test case: plan outcomes and evaluations
Note: see plan headers for colour coding

²⁰ The test results discussed in subsequent sections will provide more insight into such and other matters.

7.3 ANALYSIS OF CASE CHARACTERISTICS

The main purpose of the model is to enable generating a distinctive set of alternative plans for any given project in a consistent way. Therefore, it needs to respond to the particular nature of a project, i.e., the given land use configuration in the surroundings of the study area and the tasks listed in the plan program. At the same time, the choice of resolution – a consequence of representing the environment by means of a grid – should not substantially influence the results or their interpretation.

7.3.1 Surroundings

For testing the model’s ability to respond to variations in the land use configuration of the study area’s surroundings, two cases C-1 and C-2 are defined that slightly differ from the



Figure 7-4. Results of testing model sensitivity to changes in surroundings

benchmark regarding their surroundings. Case C-1 assumes the concentrated area of services is located at the north side of the study area (instead of at the west side), while case C-2 assumes the presence of a second area of green space at the west side. The resulting sets of alternative plans (Figure 7-4) illustrate that the model is indeed sensitive to the land use configuration of the surroundings.

Changing the services area's location to the north side (case C-1) causes all plans to change considerably. Two aspects in particular are noteworthy. First, the outcomes of the plans initiated by the services agent and the housing agent become identical, although they do follow a different track of incremental development. Second, housing is assigned to the north section of the study area in every plan. In the benchmark this only held for two plans; in the other cases, housing was found mainly in the southeast corner of the study area. The incremental development of plans reveals that, due to the assumed change in location of services, the start position from which housing develops is always found along the northern border, as the representing agent finds the cells there no longer only score well for the accessibility of green space but also for the accessibility of services. With the housing agent preferring the north part of the study area, green space is found in the southeast corner except for the plan initiated by the representing agent personally. It turns out that, at the moment this agent finalizes the first cell for his or her own land use, it is a cell in the southwest corner of the study area for which the assignments probability for the agent's own land use stands out most clear against those for all other agents' land uses.²¹ Being developed from this corner, a physical connection with the existing green space at the east side of the study area appears impossible in the remainder of the process. This is partly because the course of actions enables the business agent to persist in focusing on the southeast corner, which indicates that small-task agents do not need to be at an disadvantage compared to large-task agents, despite the fact that the interplay between the latter agents sets the main direction of plan evolution while the former agents have more nuance influence until, at a later stage of the process, they start receiving finalizations as well (section 7.3.2).

Assuming additional green space in the surroundings (case C-2) changes all plans compared to the benchmark. A similar tendency in the development of housing occurs

²¹ This relative judgement of assignment probabilities is the general criterion used by an initiator to determine which cell to finalize. It implies that the agent selects the cell about which he or she is most certain regarding any of the land uses.

here. The representing agent shifts his or her focus to the cells in the northwest corner that perform well for accessibility of both green space and services. Because the emphasis of housing is slightly more on the west side of the study area, the green space agent is able to retain his or her attention to the southeast corner and connect to the existing green space on the east side of the study area. Since all agents apply objectives for accessibility and/or adjacency of green space, it is obvious that the increased amount of green space in the surroundings, in general, enables agents to meet these objectives more easily. As a result, the majority of evaluations indicate a better plan performance than the benchmark.

7.3.2 Plan program

As mentioned, the relative difference in the task size of agents is of primary importance to the evolution of plans. With the first series of finalizations consistently concerning the land uses of large-task agents (i.e., housing and green space) a course of development starts in which, only at a later stage, the land uses of small-task agents (i.e., services and business) also become candidates for finalizations.²² Although this is a plausible course of events, the latter category of agents should not be at a disadvantage in terms of plan performance. To examine this aspect, two cases C-3 and C-4 are tested in which, relative to the benchmark, the proportional differences between agents' tasks are moderated and eliminated, respectively (Figure 7-5).

The test results reveal that, indeed, the moment at which all land uses have experienced one or more finalizations occurs earlier when agents' tasks become more

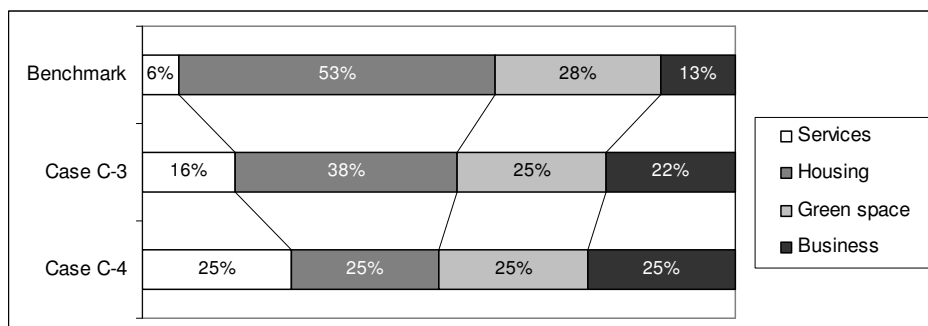


Figure 7-5. Settings for testing model sensitivity to changes in plan program

²² Evidently, this also depends on the proportion between the number of cells an agent needs and the number of cells he or she finds suitable (or attractive). The more narrow the choice is, the more focussed the claims will be.

Table 7-4. Stages at which all land uses have experienced one or more finalizations

	Plan's initiator			
	Services agent, S	Housing agent, H	Green space agent, G	Business agent, B
Benchmark	26 (S)*	28 (S)	29 (S)	29 (S)
Case C-3	17 (S)	18 (S)	18 (S)	19 (S)
Case C-4	9 (G)	10 (G)	9 (H)	10 (G)

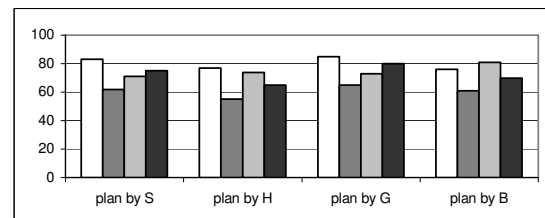
* (..) indicates the last land use to receive its first finalization.

equal-sized (Table 7-4). In the benchmark this moment was reached at a late stage: in both the plans initiated by the green space agent and the business agent only five vacant cells remain when services receives its first cell. For the cases C-3 and C-4 this is clearly advancing to a stage that is around halfway and at one-third of the process, respectively. Interesting to note is that, when all tasks are equalled (case C-4), services is no longer the land use that receives its first finalization latest, like it was in all other instances.

Regarding plan performance one might expect that the smaller the task of an agent and, thus, the later this agent receives his or her first finalization, the more this agent's

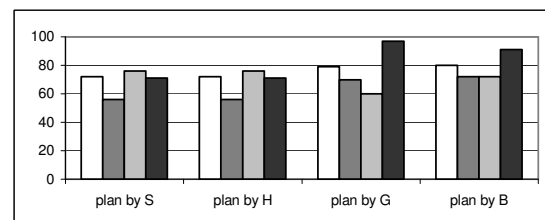
Benchmark

	plan by S	plan by H	plan by G	plan by B
Shopping agent, S	83	77	85	76
Housing agent, H	62	55	65	61
Green space agent, G	71	74	73	81
Business agent, B	75	65	80	70



Case C-3

	plan by S	plan by H	plan by G	plan by B
Shopping agent, S	72	72	79	80
Housing agent, H	56	56	70	72
Green space agent, G	76	76	60	72
Business agent, B	71	71	97	91



Case C-4

	plan by S	plan by H	plan by G	plan by B
Shopping agent, S	73	74	73	72
Housing agent, H	74	71	69	77
Green space agent, G	73	67	77	72
Business agent, B	87	72	90	90

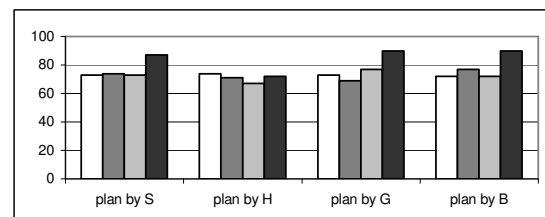


Figure 7-6. Results of testing model sensitivity to changes in plan program: plan evaluations

evaluation is bound to lag behind those of others. However, the evaluations of the three resulting plansets indicate that the effects of proportional changes in agent tasks are rather diverse (Figure 7-6). The anticipated disadvantage of a small task seems to only hold for the business agent. The evaluations by the services agent reveal an opposite tendency alike the housing agent, while the green space agent's evaluations vary too much to draw a conclusion. Hence, there is no apparent advantage or disadvantage in having a relatively large or small task. When an agent's land use will extend over a large part of the study area, the overall (average) performance is more or less secured as the performances of single cells are levelling out. In contrast, when an agent's land use requires only a small set of cells, there is proportionally more chance of finding cells that score well for adjacency of other land uses, especially when it concerns land uses of large-task agents.

Considering the evaluations for the whole team of agents, plan performances tend to balance as agents' tasks are more balanced, i.e., the evaluations of agents become more similar. Only the business agent's evaluations contrast this tendency as they become more distinctive. A look at the agent's decision model, however, explains this. The business agent is mainly interested in the accessibility and adjacency of (other) business, which implies a preference for a location in the southeast corner of the study area. In the benchmark plans, however, this corner is already assigned to either housing or green space at the moment the business agent becomes a candidate for finalizations. Due to increasing the agent's task size in the cases C-3 and C-4, he or she starts receiving finalizations when the preferred corner location is still vacant. Comparably, the other agents have less-focussed interests since they apply more diverse rules. Evidently, it is the ease with which an agent is able to meet his or her objectives within the particular context of a project – i.e., surroundings and plan program – that determines the sensitivity of that agent's evaluation.

7.3.3 Resolution

Due to the fact that cells form the unit of reasoning and decision-making in the model, increasing the resolution for representing the environment means that agents will require more time for their belief updates and, thus, that the model's execution time will increase. Running the *MASQUE* Alternative Plan Generator reveals that the relationship between model execution time and the number of cells resembles a power function. Whereas running the benchmark with a 200-meter cell resolution (i.e., cells measuring 200x200

m²; an environment of in total 144 cells of which 32 constituting the study area) takes four seconds, a 100-meter cell resolution (environment = 576 cells; study area = 128 cells) requires around three and a half minutes to complete, while a 50-meter cell resolution (environment = 2304 cells; study area = 512 cells) lifts the execution time to approximately five and a half hours.²³

Execution time is not the only aspect that is dependent on the resolution. Because plans are drawn up cell-by-cell, the resolution has consequences for the procedure as well: the higher the resolution, the higher the number of cells and the larger the number of sequential rounds over which plans are constructed. Since agents update their beliefs and express their claims in every round, the question may be raised to which extent plans remain consistent when the resolution is changed. The aforementioned three cell resolutions – with cells measuring 200x200 m², 100x100 m² and 50x50 m² respectively – were studied to examine this aspect. As for the resulting spatial patterns (Figure 7-7), the plans show a reasonable degree of consistency, especially when concentrating on the relative positioning of housing and green space. This is substantiated by the Kappa (*K*) statistic – an indicator for a cell-based map comparison that ranges from –1 (maximum disagreement) to +1 (maximum agreement) – calculated for each transition²⁴. Of in total eight transitions – i.e., increases of a plan's resolution either from 200m to 100m or from 100m to 50m – there are five for which the broad outlines of the spatial pattern are maintained (*K* > .5).

Appendices 7-A, 7-B, 7-C and 7-D provide the possibility to compare the evolution of each plan for the three resolutions based on a series of snapshots. It can be concluded that consistency in a plan's spatial pattern over resolutions is attributable to consistency in that plan's evolution from the very start of the process. The longer the consistency in plan evolution lasts, the more consistent the resulting spatial pattern will remain. Good to notice is that the model provides a certain tolerance to differences in the location of finalizations: as long as such differences remain within certain geographical boundaries, only minor differences in a plan's spatial pattern are to be expected. This shows most clearly from the 100-meter and 50-meter resolution developments of the plan initiated by the services agent (Appendix 7-A): although there are several differences in the order in

²³ Clocking times mentioned are for running the model on a standard laptop computer with Intel® Pentium® M Processor (1.4 GHz) and 512 MB working memory.

²⁴ Kappa was calculated with GEONAMICA® Map Comparison Kit, version 2 (Hagen-Zanker, et al., 2005).

which various parts of the study area are permanently assigned to land uses, the final plan remains nearly the same ($K = .912$).

Yet, it is in fact only correct to speak of tolerance with concern to the location of large-task agents' land uses (housing and green space). The land uses of small-task agents (services and business) show considerable sensitivity to changes in the evolution due to the simple reason they receive finalizations from a later stage of the process. Then their opportunities depend on (i) which parts of the study area are still left vacant by the interplay of housing and green space, and (ii) whether these parts are sufficiently spacious to fit their land uses (dependent on their desired degree of clustering). For instance, a comparison of the 200-meter and 100-meter resolution developments of the plan initiated by the housing agent (Appendix 7-B) shows that, once services becomes a candidate for finalizations, the vacant parts of the study area are different. Under a 100-meter resolution

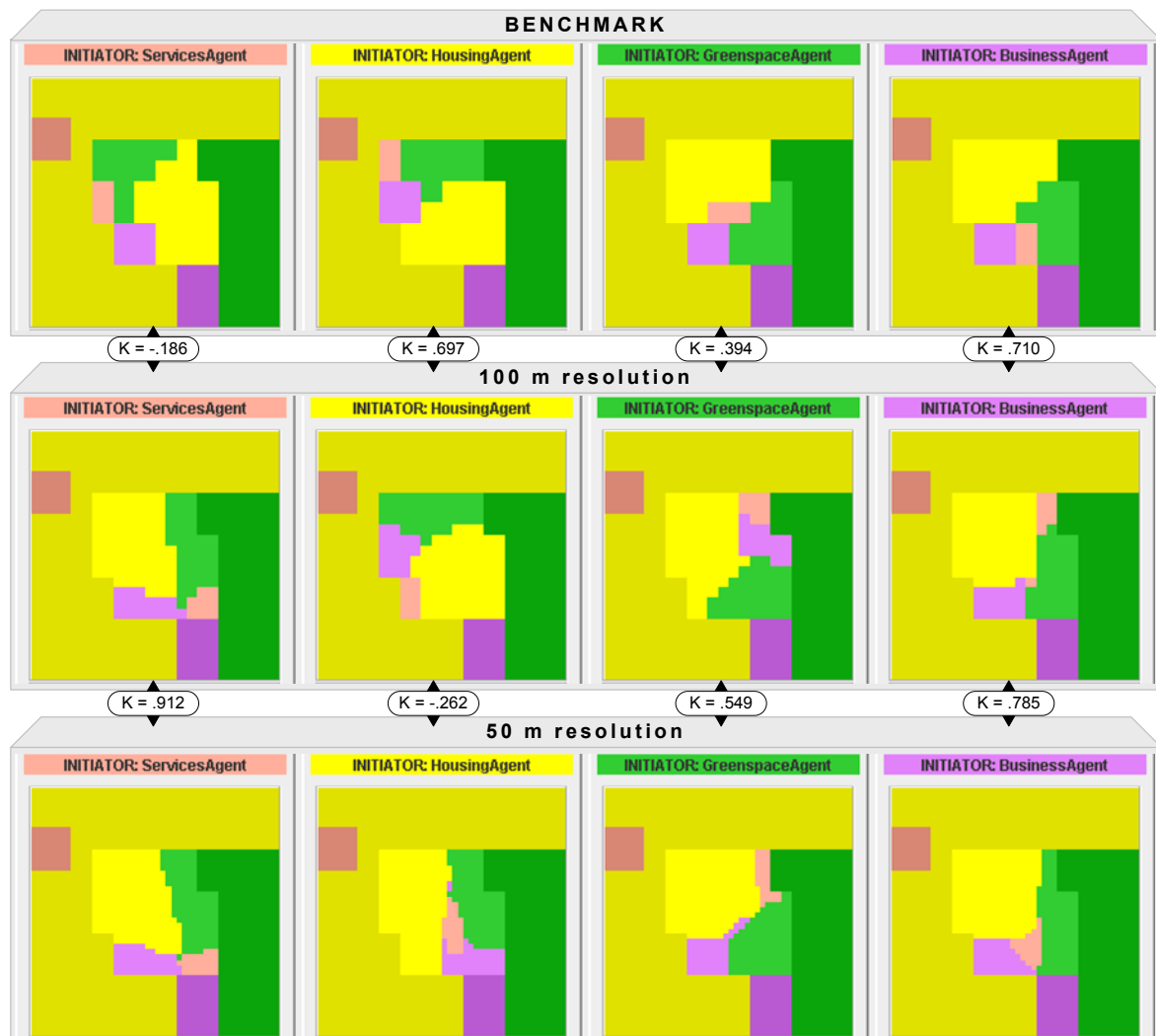


Figure 7-7. Results of testing model sensitivity to changes in resolution: plansets and Kappa statistic

a slight counter-clockwise rotation of the parts assigned to housing and green space has taken place at that time, which causes the southwest corner to be vacant instead of the northwest corner. The 100-meter resolution developments of the plans initiated by the green space agent (Appendix 7-C) and the business agent (Appendix 7-D) illustrate the consequences of a lack of sufficient continuous space for a land use. In comparison with the evolution of these plans under 200-meter and 50-meter resolutions, it can be observed that the business agent and the services agent, respectively, encounter a lack of space in the southwest and south-central parts of the study area, respectively, once they may receive finalizations.

Because the plan development process is modelled as a chain of actions and reactions, finalizations influence agents' preferences and decisions. Hence, a change in (the order of) finalizations can cause a change in plan evolution and, possibly, a change in spatial pattern. Moreover, the earlier in the process a change in finalizations occurs, the more differences in a plan's final state may be expected. The test results indicate that two transitions bring about a serious dissimilarity in spatial pattern (Figure 7-7): the 200-100 transition of the plan initiated by the services agent ($K = -.186$) and the 100-50 transition of the plan initiated by the housing agent ($K = -.262$). A look at these plans' evolution (Appendices 7-A and 7-B) learns that, because of changing the resolution, the first finalization concerns a cell at a completely different location. Logically, the larger a shift in a plan's starting point of development – in both cases, it changes from the east-central part of the study area to the northwest corner – the more likely the plan evolution will deviate and the more likely the final plan exhibits differences in spatial pattern. An investigation of the finalization procedure gives insight into the occurring shifts. In general terms, the cell that is selected for finalization is the one for which the assignments are most certain, i.e., the cell for which the assigning probability for any single agent – i.e., the probability of that agent receiving the cell – is most distinctive from the assigning probabilities for all other agents together.

Figure 7-8 depicts the 200-100 transition of the plan initiated by the services agent and gives the assignment probabilities based on which the initiator selects the first cell to finalize. The 200-meter resolution points out the cell at location (2,5) as the one with most certainty about its assignment: the probability of 0.58 for housing is the largest value found among all cells and all agents. With a 100-meter resolution, however, it is cell (0,0) for which certainty is highest ($P = 0.61$) instead of one of the four cells that constitute the 200-meter cell (2,5) selected before. Considering 100-meter cells in clusters

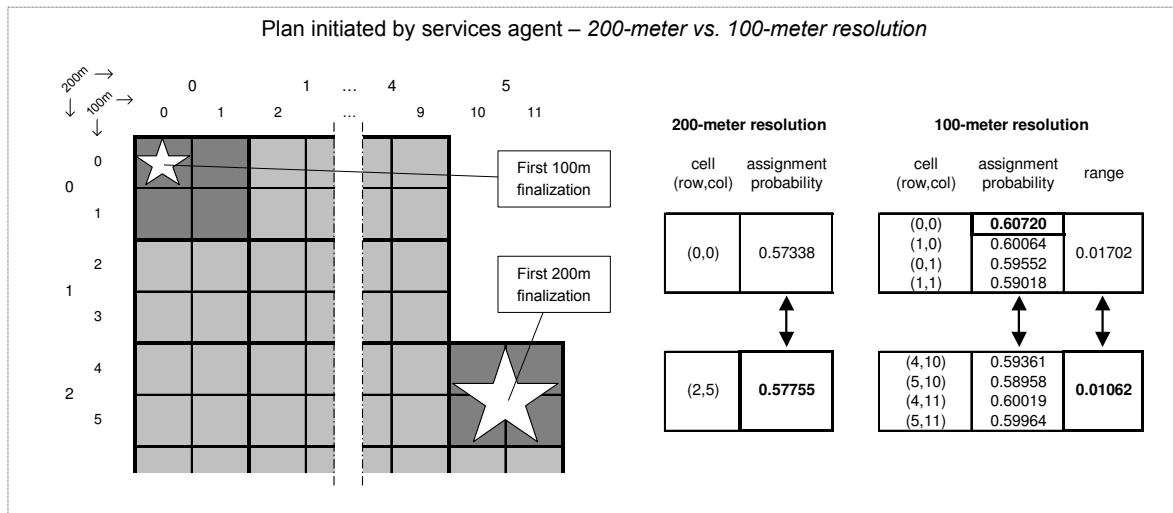


Figure 7-8. Shift in location of first finalization due to a change in resolution

of four (i.e., 200-meter clusters) and comparing their distributions of values, reveals that the range of cluster (2,5) is smaller than that of cluster (0,0). The fact that this larger coherence of values in the former cluster can also be considered as an indicator of higher certainty, could explain why the 200-meter resolution resulted in the selection of cell (2,5). An identical situation takes place for the 100-50 transition of the plan initiated by the housing agent.

With respect to plan performance, it seems reasonable to expect that the consequences of finalizations are more drastic in cases of low resolution, since the pieces of land being processed are larger and fewer rounds will be available for attempts to reduce possible negative effects. The evaluation results (Figure 7-9), however, do not

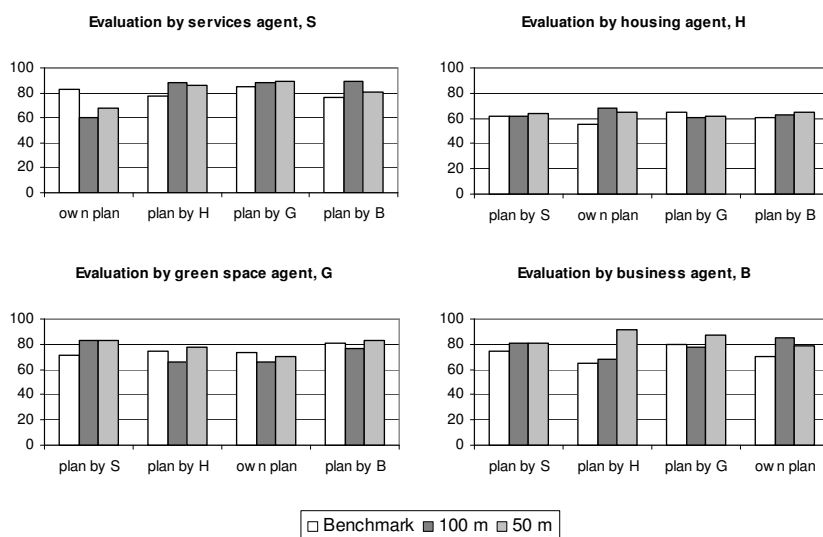


Figure 7-9. Results of testing model sensitivity to changes in resolution: plan evaluations

give any reason to assume that there is a relation between performance score and resolution: finalizations appear to be equally influential in higher resolution cases as in lower resolution cases. On the contrary, the expected disadvantage of a low resolution may not be revealed because it is counterbalanced by the disadvantage a high resolution will have when occurring negative effects cannot be reduced but instead accumulate over a large number of remaining rounds.

7.4 ANALYSIS OF DECISION MODELS

The *MASQUE* Alternative Plan Generator implements three types of utility functions that agents can use as part of their decision models. In order to examine whether each of these types of rules operate as expected, and to assess their impact on the generated plans, one or more rules per type were investigated, while distinguishing between large-task agents (i.e., housing and green space) and small-task agents (i.e., services and business).

7.4.1 Accessibility-based utility

In the benchmark, the two large-task agents both apply multiple rules related to accessibility. The green space agent, for instance, uses a rule that expresses a preference for accessibility of other green space areas (primary radius = 400 meters). The agent succeeds in satisfying this rule fairly well (Figure 7-3): three plans show a physical connection with the green space area at the west side of the study area, whereas in the plan initiated by the services agent the minimum distance between green spaces inside and outside the study area is still limited to 200 meters. Reversion of this rule's objective – i.e., assuming the agent attempts to avoid accessibility of other green space – leads to a set of plans that change in accordance with what one might expect (e.g., Figure 7-10): the planned green space is completely detached from the existing green space, while the distance between the two areas has become 400 meters or more. Exclusion of the rule – i.e., assuming the agent does neither seek nor avoid accessibility to other green space – also leads to a set of plans in which connections between planned and existing green space are missing. This tendency of all remaining rules – i.e., rules used by the green space agent and all other agents in the team – to avoid accessibility between green space areas implies that the connections established in the benchmark were indeed attributable to the green space agent.



Figure 7-10. Results of testing the green space agent's rule for accessibility of other green space (example: plan initiated by the housing agent)

Also the two small-task agents judge one or more aspects of accessibility. For instance, the business agent uses a rule that expresses his or her preference for accessibility of other business areas (primary radius = 600 m). In the benchmark, every plan except for the one initiated by the housing agent has business assigned to the southwest corner of the study area, which brings an expected utility of 75 for accessibility to the existing business area located at the south/southeast side. Reversion of the rule's objective – i.e., assuming nearness of other business areas is not appreciated – results in the relocation of business in three plans. Whereas the plan initiated by the green space agent does not change, the other plans all have the planned area for business move northwards (e.g., Figure 7-11), which gives an expected utility of at least 50 for the rule in question. Excluding the rule gives the same outcomes as reversing its objective, except for the business agent's own plan that remains the same as in the benchmark. Apparently,

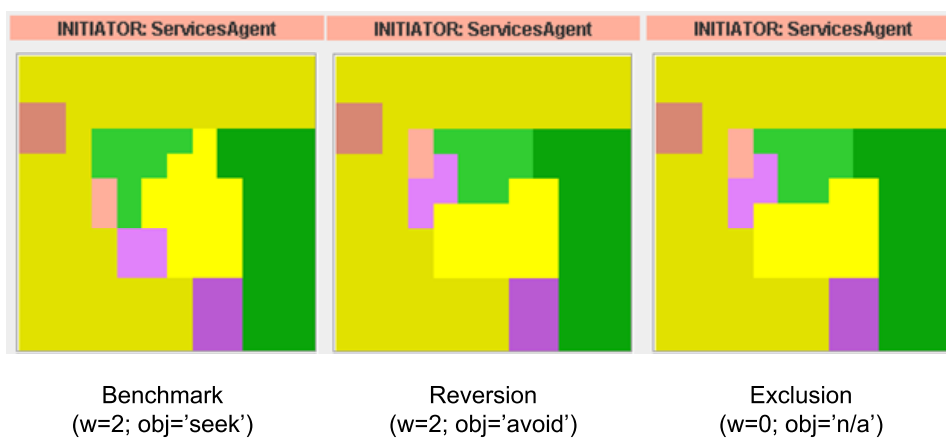


Figure 7-11. Results of testing the business agent's rule for accessibility of other business (example: plan initiated by the services agent)

playing the role of initiator makes an important difference in what this agent could achieve given the typical characteristics of the project illustrated here (section 7.2).

7.4.2 Adjacency-based utility

Both large-task agents apply two rules related to adjacency. For instance, the housing agent uses one that expresses his or her preference to avoid adjacency of business. The set of plans that resulted from the benchmark shows that it is difficult for the housing agent to fully meet this requirement: the planned housing area shares borders with business over a distance of 200 meters at best. The rule is effective, however, as reversion of its objective – i.e., assuming the housing agent prefers adjacency of business – illustrates that for every plan housing consistently starts developing from the southeast corner to be adjacent to the existing business area in the surroundings (e.g., Figure 7-12), which leads to a total bordering distance of at least 800 meters. Apparently, in the benchmark plans it is difficult for the housing agent to establish a complete detachment from business due to the large area that his or her land use requires, whether or not in combination with the interactive effects of other agents' behaviour within the particular benchmark settings.

Exclusion of the rule leads to plan outcomes that, based on judging the adjacency between housing and business, can be categorized in between those resulting from the benchmark and the plans after reversion. With concern to the plans' overall spatial pattern, however, most similarity is found with the benchmark outcomes. Interesting to note is that, judging from its final state, the plan personally initiated by the housing agent remains unaffected by both reversion and exclusion of the rule. Nonetheless, its evolution shows a difference that is in line with what might be expected from these adjustments. Whereas in the benchmark's plan housing started developing from the east-central part of the study area, reversing the rule causes housing to start from the southeast corner that is adjacent to the existing business area, while exclusion causes the first finalization for housing to take place at an intermediate location. Nevertheless, the consequences of this difference for the other agents remain moderate, as they are not triggered to adjust their claims to such an extent that it causes permanent divergences in plan evolution and subsequent changes in plan outcomes.

Adjacency rules are also included in the decision models of small-task agents. For instance, the business agent uses one that shows a preference for the adjacency of green space. The plans resulting from the benchmark show that this requirement is only

limitedly satisfied, as at best two of the four cells that constitute the planned business area are indeed adjacent to green space, while in the plan personally initiated by the business agent it is remarkable to not find any adjacency. Neither reversion nor exclusion of the rule brings changes to the resulting set of plans, although minor differences can be found in the plans' evolution.

This lack of effects, however, appears to be partially attributable to the resolution since a 100-meter resolution does reveal differences for the plan initiated by the housing agent and, especially, for the plan initiated by the green space agent. In the latter plan it can be clearly seen that exclusion of the rule results in a spatial pattern that stands midway between the benchmark plan and the plan after reversion (Figure 7-13). Yet, the fact that with this higher resolution there are still two plans that, despite evolutionary differences, remain exactly the same in their final appearances makes it clear that the effect of the rule is fairly weak. The cause of this lies largely in the fact that the business agent has a small task size and adds relatively low weight to the adjacency rule. In the given circumstances, the business agent is most interested in the southeast corner of the study area for connecting to existing business (Figure 7-2), but in nearly all cases (one of) the large-task agents also become focussed on that area. When the first finalization for business is to be made, the southeast corner is already assigned to another land use and the business agent is forced to shift his or her focus to parts of the study area for which adjacency to green space is not guaranteed. Due to the low weighing of the rule, however, the (possible) loss of utility is limited and reversion or exclusion of the rule is hardly felt by the agent. In section 7.5.2 it will be discussed how claim accentuation could be a means for small-task agents to advance the moment from which they start receiving finalizations, which would make the effects of their decision rules easier noticeable.

7.4.3 Population-based utility

The only agent that applies a population-based utility function is the services agent, who determines the expected population size within a specified distance based on the future belief regarding the assignments of involved cells to housing, which is interpreted in terms of expected utility. The effects of this population rule on plans and their performance were tested by means of examining the consequences of adjusting the rule's weight in the agent's decision model.

As such, though, the only observable effect is the logical change in evaluation scores

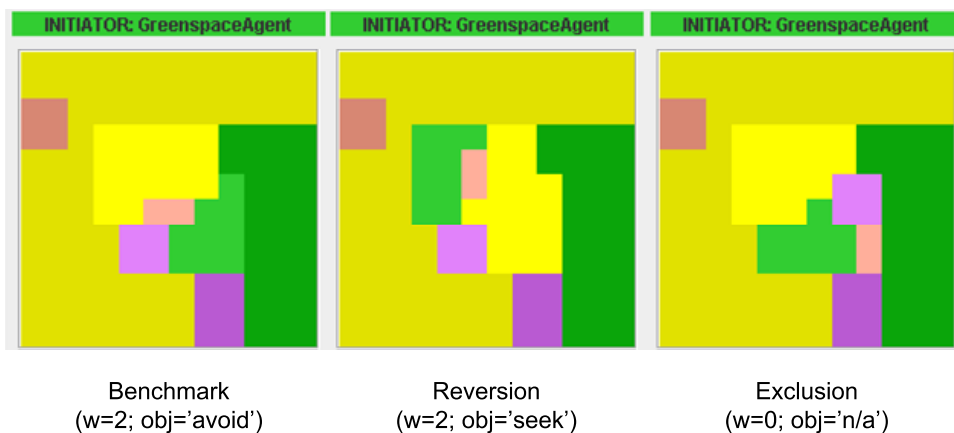


Figure 7-12. Results of testing the housing agent's rule for adjacency of business (example: plan initiated by the green space agent)

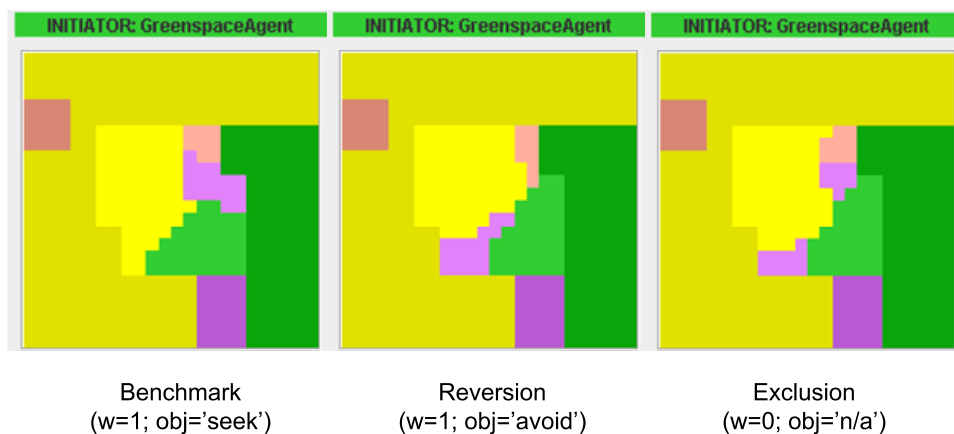


Figure 7-13. Results of testing the business agent's rule for adjacency of green space (example: plan initiated by the green space agent with a 100-m resolution)

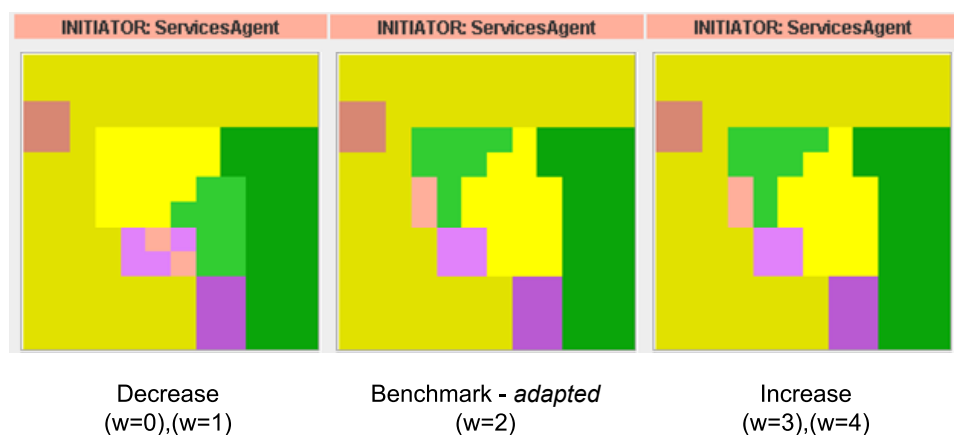


Figure 7-14. Results of testing the services agent's rule for population (example: plan initiated by the services agent with adapted benchmark – see text)

calculated by the services agent: the few occurring differences in plan evolution are trivial and do not change the resulting spatial patterns. Evidently, the aforementioned disadvantage of small-task agents plays an important role here as well: the services agent requires only two cells and, at the time he or she starts receiving finalizations, the number of vacant cells is limited. Another aspect of influence, however, is the fact that the services agent applies two more rules that also imply a judgement of housing, namely one for accessibility and one for adjacency. Although operating in different ways, the three rules in question do correlate, since what they judge or measure is not entirely mutually exclusive. For instance, finding a neighbouring cell assigned to housing brings along expected utility from the viewpoint of not only adjacency but also accessibility and population. This, together with the fact that the overall expected utility is the weighted average of the expected utility per category, causes adjustments to the population rule to be more or less compensated for by the other two rules. Hence, proper testing of the rule requires filtering out its effect by excluding the housing-related rules for accessibility and adjacency from the services agent's decision model.

Based on this temporary adaptation – giving identical spatial patterns as the actual benchmark – the effect of the services agent's population rule becomes observable, albeit only the plan personally initiated by this agent undergoes changes in spatial pattern (Figure 7-14). Whereas the plan's spatial pattern from the (adapted) benchmark is maintained when the weight is increased, it changes dramatically when the weight is decreased or set to a zero value. The reason of this change is a difference in the initial claims of the services agent. While in the (adapted) benchmark the services agent was initially focussed on the southwest corner, lowering the importance of population causes the agent to find cells of interest in the northeast corner of the study area as well. This causes the services agent, as initiator, to make the first finalization – i.e., for housing – for a cell in the northwest corner instead of one in the east-central part of the study area. Since the plan from the (adapted) benchmark had the northwest corner assigned to green space, this change in starting location already implies a permanent divergence in plan evolution and, thus, leads to a different plan outcome.

7.5 ANALYSIS OF CLAIMING BEHAVIOUR

An agent determines the relative attractiveness of all cells in the study area by using his or her decision model, which combines rules that measure various cell characteristics. The

outcomes of an agent's decision model form the basis on which he or she formulates claims. How the outcomes of a decision model are translated into claims depends on an agent's claiming behaviour, which is captured by a small set of parameters. These parameters express an agent's desire of clustering, his or her claiming strategy and whether or not claims are to be scaled. Below, the main parameters are analyzed across their range for every agent to examine the effects in terms of the plan performance measured by the agent in question (i.e., what are the effects for the agent personally?), the initiators of plans (i.e., what are the consequences for the agents that make the assignment decisions?) and the team (i.e., what happens to the overall performance of plans?).

7.5.1 Clustering

Clustering refers to an agent's behaviour of taking into account the relative attractiveness of neighbouring cells dependent on that agent's desire to have his or her land use be planned as a geographically continuous area. The parameter ϕ defines an agent's desired intensity of clustering, i.e., the degree to which the probability of claiming a cell is affected by that of claiming its neighbouring cells. In the benchmark the clustering intensity was assumed to vary per agent based on intuitive considerations regarding the represented land uses. Although the parameter was only set to maximum for the services agent, the resulting plans already showed continuous areas for all land uses.

Figure 7-15 depicts the various plan performances measured (see above) as results of testing the parameter for large-task agents. In the case of the housing agent (see top row) it appears that, in general, the plan performance for that agent personally tends to improve when he or she pursues a lower clustering intensity. This is obvious when realizing that, the more continuous the area of housing will be, the more cells will give lower scores for adjacency and accessibility of other land uses, simply because they are surrounded by other housing cells. From the viewpoint of other agents in their role of initiator, however, especially the small-task agents measure a better performance when the housing agent's clustering intensity is increased. Defining the plan performance for the team as the summation of performances measured by every single agent for that plan, the plan initiated by the housing agent personally performs best for the team when that agent tries to achieve a low clustering intensity, while the plans initiated by the small-task agents appear to benefit most when the housing agent sets the clustering intensity to an extreme value (either 0 or 1), and the plan initiated by the green space agent is evaluated

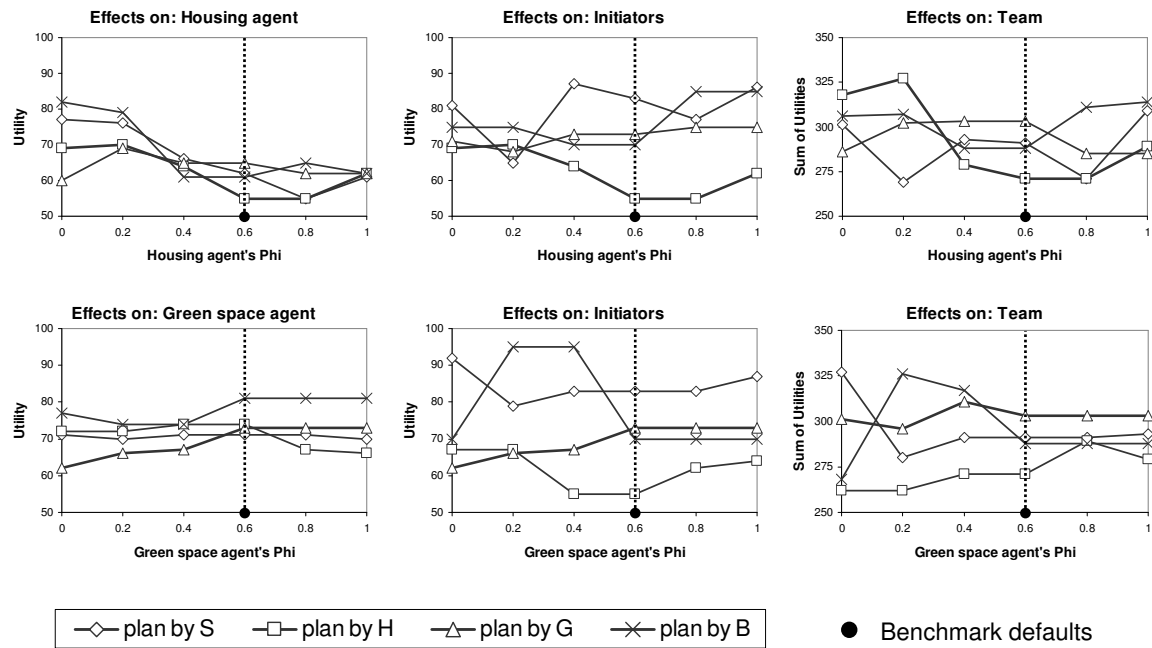


Figure 7-15. Results of testing the clustering intensity (parameter ϕ , 'phi') for large-task agents: housing agent (top row) and green space agent (bottom row)

best with an intermediate value. The results of testing the clustering intensity for the green space agent (see bottom row) illustrate that this agent is less sensitive than the housing agent. Among all plans, it is especially the plan initiated by the business agent for which changing the green space agent's clustering intensity has consequences for the initiator that also carry over onto the team's evaluation.

Figure 7-16 displays the results of performing identical tests for the small-task agents. It clearly shows that changing the clustering parameter for either the services agent or the business agent has effects too, albeit in a more moderate fashion than doing so for large-task agents. This is in line with what might be expected since small-task agents can only influence the plan development process from the background until they become candidates for finalizations, which is at a relatively late stage compared to large-task agents. Looking at the results, the services agent (see top row) can establish a slightly better performance for the plan he or she personally initiates, and for all plans together, when trying to achieve a slightly lower clustering intensity (0.6 to 0.8) than in the benchmark. However, especially for the housing agent as initiator it would be more beneficial when the services agent sets his or her clustering intensity to a lower value (0.2 to 0.4). Nonetheless, the team's evaluation of the housing agent's plan shows that these contrast preferences compensate each other and, thus, that the services agent's clustering intensity hardly matters from the team's perspective. The results for the business agent

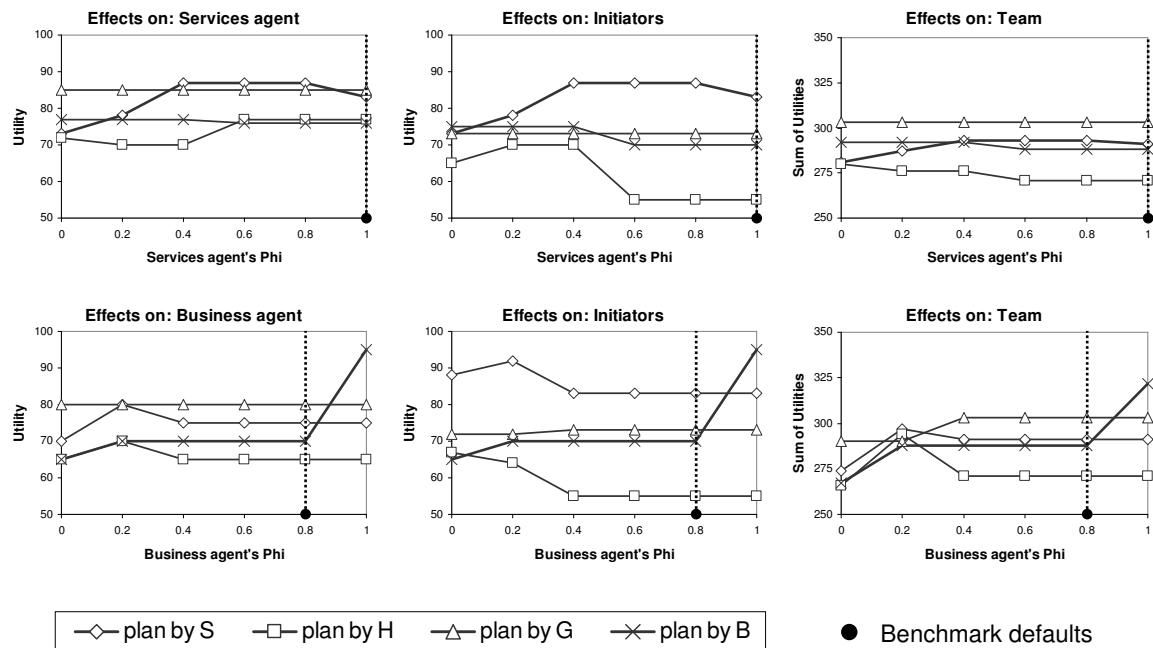


Figure 7-16. Results of testing the clustering intensity (parameter ϕ , 'phi') for small-task agents: services agent (top row) and business agent (bottom row)

(see bottom row) are different, as this agent could considerably improve the performance of his or her own plan by choosing a maximum clustering intensity, which not affects the performances of other plans, neither for initiators nor for the team. Setting the parameter to a value of 0.2 gives an interesting effect as well: the plans by the housing agent and the services agent improve from the viewpoint of both the business agent and their initiators, while the team evaluates all plans nearly equal despite their distinct spatial patterns.

The parameter γ works in conjunction with the clustering intensity and provides the possibility to set the relative importance of neighbouring cells concerning their location at either a side or a corner of a cell. The higher the value is set, the more importance is given to border-sharing neighbours as opposed to corner-sharing neighbours. Although the parameter has a range from 0 to 1, values below 0.5 are actually irrational since they imply valuing land uses at corners more than those at borders. As it is questionable whether this parameter should be an instrument for determining the most optimal claiming behaviour of agents, it is not further discussed here.

7.5.2 Claiming strategy

Primarily, the claiming strategy of an agent concerns the degree of opportunism that this agent applies when determining his or her claims, i.e., to which extent this agent follows

his or her (personal) goal state belief as opposed to the (collective) future state belief. Additionally, an agent has the option of claim accentuation, which makes claims more expressive by means of increasing claims for cells that belong to his or her best set of cells and decreasing the claims for cells that do not.

7.5.2.1 Opportunism

An agent's degree of opportunism is set by means of the θ parameter. In the benchmark, all agents defined their claims by balancing their beliefs ($\theta = 0.5$). When being the initiator of a plan, an agent takes care of constructing and updating the future state belief. It is realistic to assume that an initiator will act in full compliance with this belief when defining his or her claims. Consequently, the parameter automatically takes a value of zero for initiators.²⁵ Sensitivity analysis can reveal the effects of changing an agent's degree of opportunism on the plan performances for that agent, for initiators of plans and for the team.²⁶

Figure 7-17 illustrates the results of testing the parameter for large-task agents. In the case of the housing agent (see top row), changing the degree of opportunism moderately affects the performances for this agent personally with respect to two plans. Interestingly, a low degree of opportunism – or, a high degree of compliance – makes the plan initiated by the business agent the best performing plan for the housing agent, the initiator of that plan and the team. The plan initiated by the services agent, instead, shows that the housing agent would benefit most from applying a high degree of opportunism, while actually the team reaches best results with a contrary setting in which the housing agent acts in full compliance with the future state belief provided by the services agent. The results of testing the parameter for the green space agent (see bottom row) show that the plans initiated by the housing agent and the services agent undergo changes. Most remarkable when increasing the agent's degree of opportunism are the consequences for the plan initiated by the services agent, as a parameter value of 0.8 seems to cause a serious – but occasional – decrease in performance for the initiator and, subsequently, the team. This phenomenon seems to be more or less coincidental, as the plan evolution

²⁵ This also explains why it is not necessarily beneficial to play the role of initiator (section 7.2.4).

²⁶ As the zero-value for initiators is automatically applied, changes in parameter value for a particular agent will be overruled for the plan initiated by that agent. Hence, the performances related to the plan initiated by a tested agent remain unaffected.

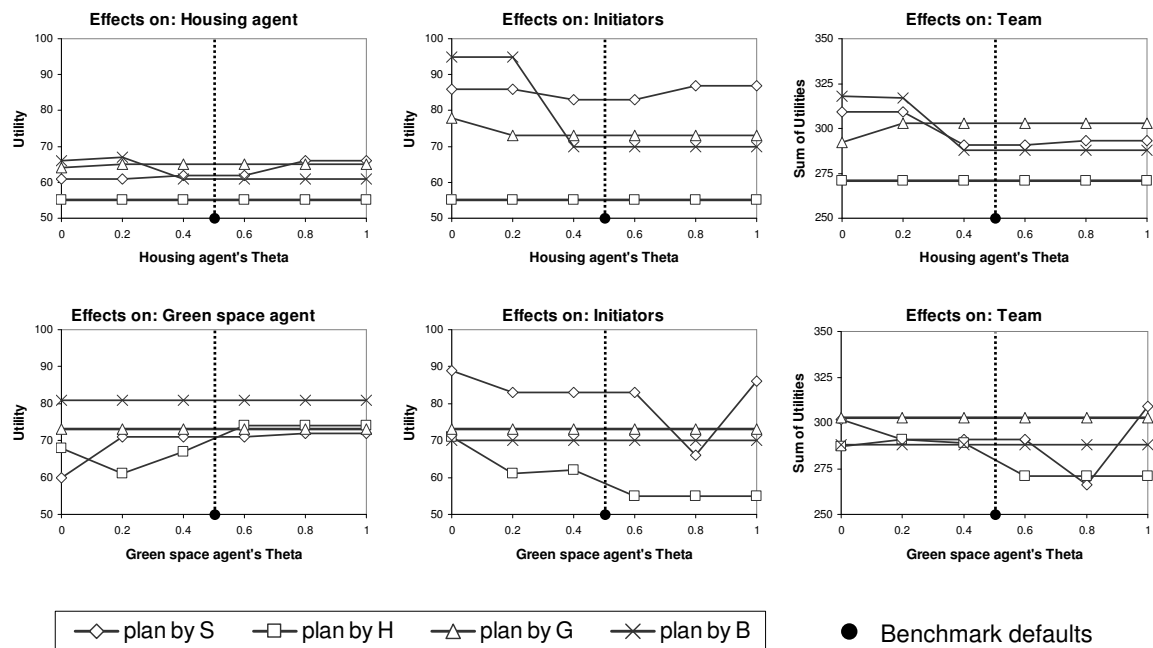


Figure 7-17. Results of testing the opportunism (parameter θ , 'theta') for large-task agents: housing agent (top row) and green space agent (bottom row)

changes course in a way that in the end turns out to be disadvantageous for the initiating agent. The consequences of decreasing the green space agent's degree of opportunism are most felt for the plan initiated by the housing agent, as the performance of this plan tends to drop for the green space agent personally, while it improves from the viewpoints of both its initiator and the team.

Figure 7-18 illustrates that the degree of opportunism for small-task agents is hardly effective: whereas changing the parameter for the business agent (see bottom row) only causes small differences in the performances related to the plan initiated by the services agent when extreme values (i.e., either 0 or 1) are tested, varying the services agent's degree of opportunism (see top row) does not cause any changes in plan outcomes and, thus, plan performances. This lack of effects, however, is entirely in line with what might be expected when realizing that the degree of opportunism is an instrument to set an agent's persistency in claiming what is considered to be most attractive (with the involved risk of overplaying one's hand), rather than an instrument – such as claim accentuation – with which an agent could advance the moment of receiving finalizations.

7.5.2.2 Accentuation

In general, an agent with a small task size will not have cells permanently assigned to his

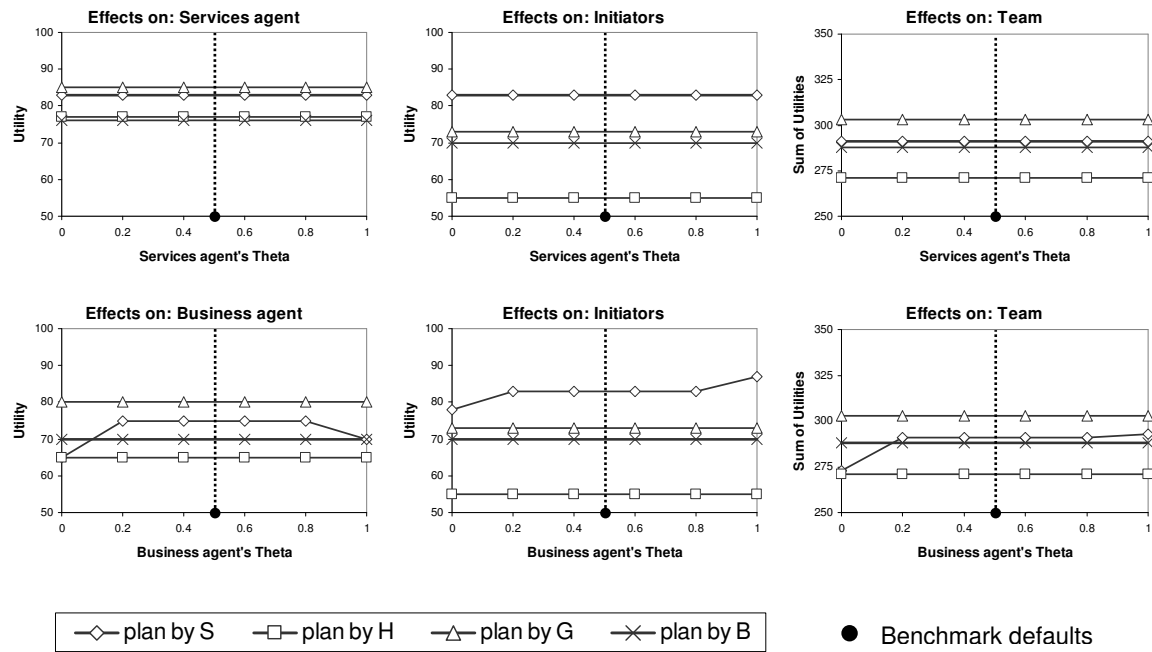


Figure 7-18. Results of testing the opportunism (parameter θ , 'theta') for small-task agents: services agent (top row) and business agent (bottom row)

or her land use until the process reaches a relatively late stage. The main causes of this are (i) the proportion of vacant cells versus required cells and (ii) the variation in suitability of vacant cells, as determined by the agent. When there is a large surplus of suitable vacant cells, the agent's claims will not be clearly focussed on particular cells. In contrast, agents who are faced with a smaller surplus of suitable cells will be more expressive in their claims and, consequently, they will get priority with the permanent assignment of cells. Although this situation is not necessarily at a disadvantage to small-task agents, as the certainty about neighbouring cells increases over time, they can attempt to advance the moment of becoming candidate for cell finalizations by using the optional strategic instrument of claim accentuation. The instrument – available for every agent – requires setting a factor f_c that indicates the basic level of accentuation an agent should apply, which takes any value within a [0,1] range (section 5.3.2.1).

Figure 7-19 displays the results of testing the claim accentuation instrument for large-task agents. In the case of the housing agent (see top row), using the instrument leads to three (out of four) plans that perform better for this agent as the accentuation factor is increased up to a value of 0.8; the only exception is the plan initiated by the services agent of which the performance for the housing agent starts decreasing when the factor is set to 0.4 or higher. From the viewpoint of other agents as initiator, however,

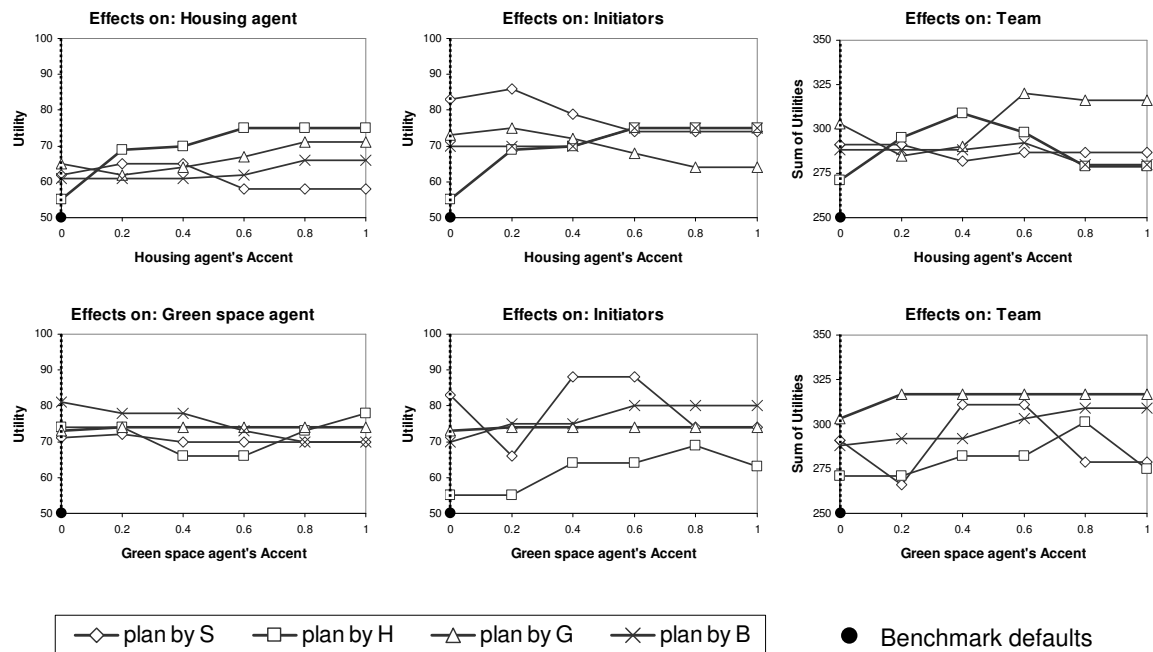


Figure 7-19. Results of testing the claim accentuation (factor f_c) for large-task agents: housing agent (top row) and green space agent (bottom row)

increasing the housing agent's factor tends to cause a slow but steady decrease in performance for both the services agent and the green space agent; the only initiator that experiences a slight benefit is the business agent. Regarding the evaluations by the team, changes in the housing agent's factor have most effect on the agent's own plan and the plan initiated by the green space agent. In the latter case, the decrease in performance experienced by the initiator is overcompensated by the increase of performance that others measure. The results for the green space agent (see bottom row) show that the agent personally does not notice much change in performance when his or her claim accentuation factor is increased, as opposed to the consequences for other agents in their role of plan initiators. Whereas the plans initiated by the business agent and the housing agent tend to perform better (for both initiator and team) with an increase of the factor up to a value of 0.8, the plan initiated by the service agent reveals varying results across the tested range of the parameter.

In comparison, the claim accentuation instrument appears to lead to more distinctive results when small-task agents apply it (Figure 7-20). This is understandable when realizing that, without accentuating their claims, large-task agents already receive finalizations relatively early in the process, before others do. Hence, the consequences of increasing their accentuation will be rather limited compared to the changes in

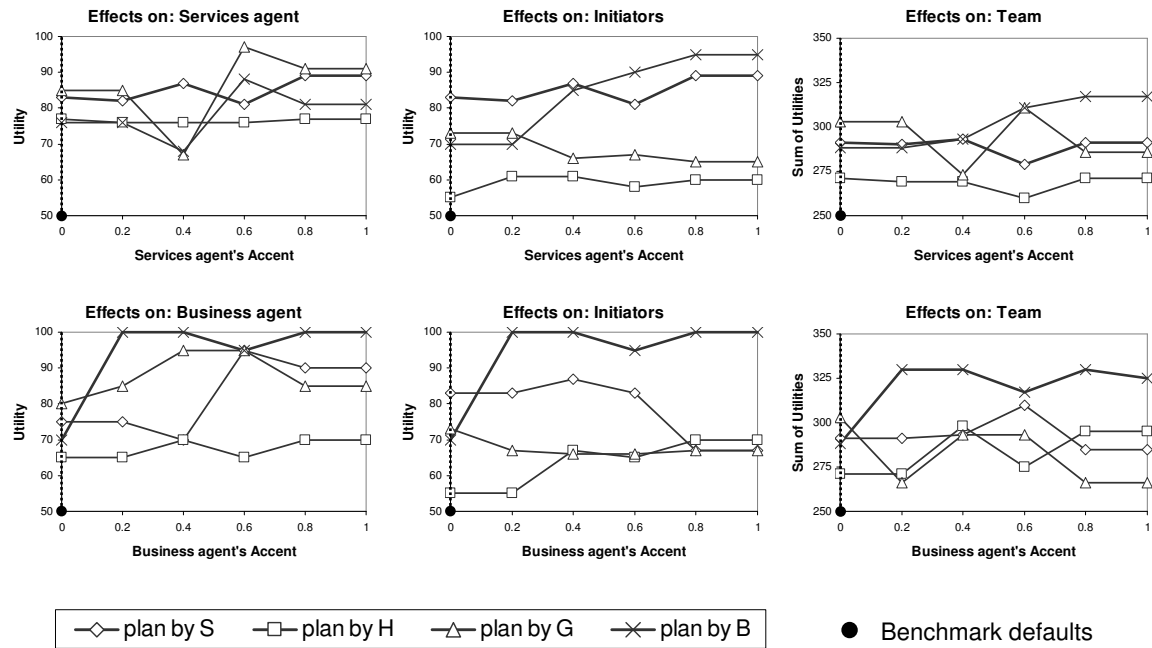


Figure 7-20. Results of testing the claim accentuation (factor f_c) for small-task agents: services agent (top row) and business agent (bottom row)

finalizations – i.e., changes in the order of both cells and land uses – that are bound to occur when increasing the accentuation by small-task agents. In the case of the services agent (see top row), the plan that improves most from all possible viewpoints is the plan initiated by the business agent, whereas the performances related to the green space agent’s plan shows most fluctuations, which is mainly attributable to the consequences felt by the service agent personally. The results for the business agent (see bottom row) illustrate that this agent is able to reach (nearly) maximum performance for his or her own plan, which also positively affects the team’s evaluation. The situation in which all plans together are evaluated best by the team is when the business agent’s accentuation factor is set to a value of 0.4.

7.5.3 Scaling policy

In the benchmark, agents are set to scale their claims, i.e., to have the claims over all cells in the study area sum up to their task sizes. As such, the claims expressed by an agent can be interpreted as the unified claim this agent makes for the whole study area. This, however, is not a necessity with concern to procedural correctness or process outcomes, because of the fact that initiators always ensure that their assignments are scaled to make the sum of claims per land use equal to the task size of that land use. Consequently,

during the course of plan development the claims of an agent will automatically converge to sum up to his or her task size. Disabling the claim scaling policy would make the claims of agents independent of their task sizes, enabling small-task agents to claim cells as easily or strong as large-task agents, especially at the early stage of the process.²⁷

Figure 7-21 depicts the differences in plan outcomes when turning the claim scaling policy off. In the benchmark there were two plans – initiated by the services agent and the housing agent – for which the first finalization (regarding housing) was made for a cell located in the east-central part of the study area, despite the facts that (i) this cell also belonged to the green space agent’s best set and (ii) many other cells were available that only belonged to the housing agent’s best set. This situation is a typical consequence of having agents scale their claims and is not only affecting small-task agents. Even the green space agent, who has the second largest task size, was not capable of claiming the east-central cell sufficiently strong to have the initiator of either plan select a different cell and/or land use for the first finalization. When inactivating the scaling policy, however, the claim from the green space agent for the east-central cell gains (relative) strength and

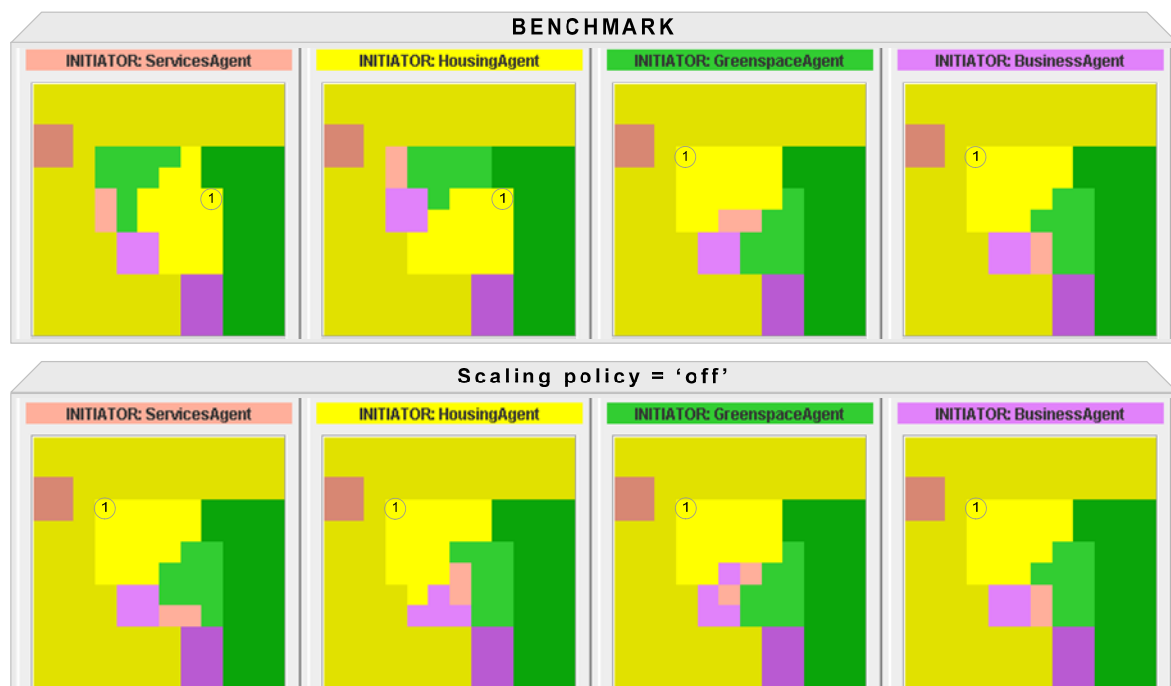


Figure 7-21. Results of testing the team-based claim scaling policy
 Note: ① indicates the location of each plan’s first finalization

²⁷ During the course of the process, this equality gradually disappears again due to the automatic convergence of claims, which is part of the assignment procedure.

causes the first finalization in all plans to concern the cell located in the northwest corner of the study area, leaving the east-central cell vacant. An important consequence seems to be that the generated alternatives become less distinctive. In the tested case, the initial advantage for small-task agents does not take effect. Actually, most disadvantage is experienced by the services agent, who in the benchmark found the plan initiated by the green space agent and his or her own plan to perform well (85 and 82, respectively); due to inactivating the scaling policy these scores drop (62 and 74, respectively).

7.6 ANALYSIS OF ASSIGNING BEHAVIOUR

In the role of initiator, an agent takes charge of developing a plan by means of an iterative process in which he or she requests all agents in the team for their claims in order to integrate these claims, and his or her personal claims, into assignments that are suggested to all agents. Based on this information, agents decide whether or not to update their claims after which these are sent to the initiator of the plan again, who updates the assignments, and so on. The initiator's assignments are fully probabilistic when the process starts²⁸, but they evolve to become fully deterministic at the end of the process by means of incremental cell finalizations decided upon by the initiator. The way in which an agent fulfils the role of initiator is defined by a small set of parameters. Tests were performed to examine the effects of changing an initiator's initial weight set, using the instrument of weight adjustment and applying a priority policy.²⁹

7.6.1 Initial weight sets

The main component of an agent's assigning behaviour is the weight set that defines the relative importance this agent gives to the various land uses from the viewpoint of his or her own land use. Based on this weight set, an agent integrates the various claims into assignments. Although the *MASQUE* Alternative Plan Generator includes an option to make agents' weight sets reactive to the process (section 7.6.2), the benchmark assumed weight sets to operate in a static manner. Obviously, an initiator's weight set influences

²⁸ This is irrespective of the fact that claims could be expressed deterministically.

²⁹ The iteration limit and compare precision are not discussed here because tests indicate that the process converges sufficiently smooth, leaving the mechanism of enforced finalizations unused. Furthermore, the completion method is left out of the discussion due to its fairly modest effect.

both plan evolution and plan outcome, even when the initial weight sets are sustained throughout the process. To examine this influence, two cases are tested in which, relative to the benchmark, agents' weight sets are moderated (i.e., weights of 5 (maximum) are lowered to 3) and equalized (i.e., all weights in a set are set to 1).

The plansets resulting from moderating and equalizing agents' weight sets (Figure 7-22) illustrate that changes occur when initiators are softening their a priori distinction between claims for different land uses. The moderation tested, actually, still causes only a change in the plan initiated by the housing agent. Interesting to note is that the effect of moderation on this plan is identical to the effect of disabling agents' claim scaling policy (section 7.5.3): a different cell is selected for the first finalization (i.e., regarding housing). However, the explanation here slightly differs, as the claim that the green space



Figure 7-22. Results of testing the initial weight sets of initiators: plansets

agent makes for the east-central cell remains the same. By lowering the relative importance of his or her own claims, it is the initiator – i.e., the housing agent – who now considers claims for other land uses proportionally stronger than before. This causes the certainty about the assignment of the east-central cell to decrease to such an extent that it becomes smaller than the certainty about assigning the cell in the northwest corner of the study area. Consequently, the initiator selects the latter cell for the first finalization, which changes the course of plan evolution completely.

Equalization of agents' weight sets has analogous effects on the other three plans. Whereas in the plan initiated by the services agent it is again the first finalization (regarding housing) that is the primary reason of the plan changing, the divergence for the plans initiated by the green space agent and the business agent becomes a fact at the moment that these agents are about to finalize a cell for their own land use or shortly before, respectively.

7.6.2 Weight adjustment

As mentioned, an initiator's weight set can be made responsive to the process, i.e., in every round the weight set can be adjusted per cell based on the claims made for that cell at that moment (section 5.3.2.2). This instrument, serving the purpose of process guidance and acceleration, requires setting a factor f_a that indicates the intensity of adjustment an initiator should apply, expressed as a value within a [0,1] range.

A look at the test results (Figure 7-23) reveals that, except for plans initiated by the business agent, the use of weight adjustment has varying consequences for the plan performances measured by individual agents. From the viewpoint of initiators, the performances seem rather stable although an exception should be made for the green space agent's plan, since this agent does experience a decrease in performance over the whole tested range. Actually, this is the plan that is affected most, which is felt by all agents, especially the small-task agents. Regarding the team's evaluations, the effects of using weight adjustment are modest. The only two changes worth mentioning are (i) the increased performance of the services agent's plan when the factor is set to a value of 0.2 or larger, and (ii) the decrease occurring for the plan initiated by the green space agent when setting the factor to a value larger than 0.8. Both these changes for the team are mainly due to the changes experienced by small-task agents.

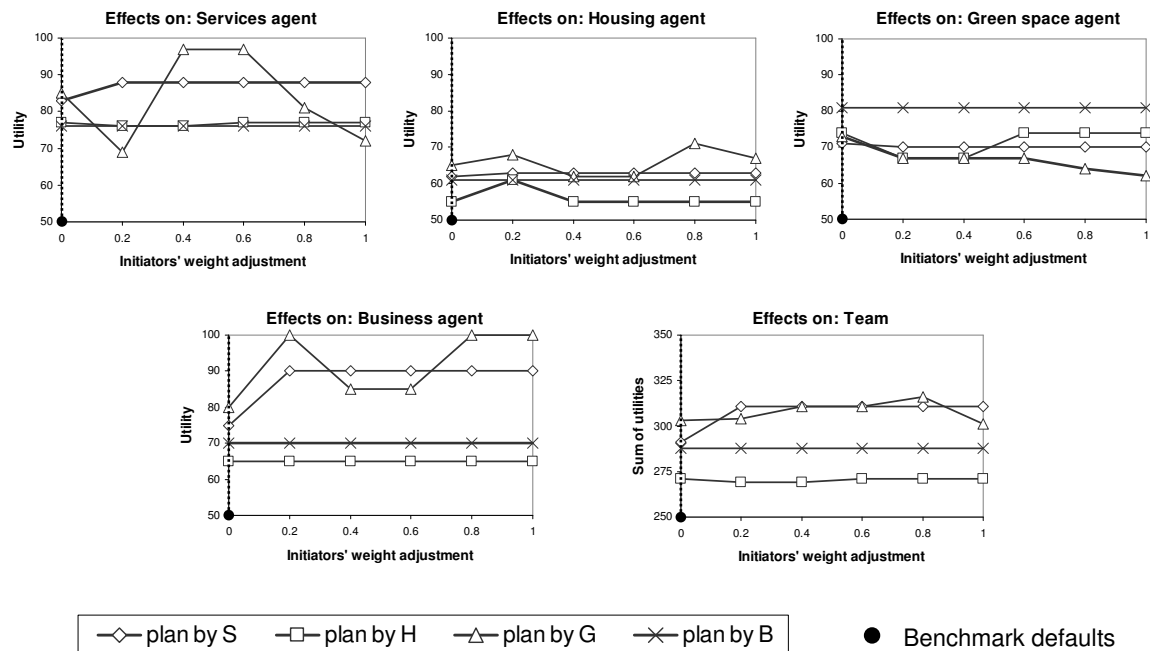


Figure 7-23. Results of testing the weight adjustment (factor f_a) used by initiators

7.6.3 Priority policy

The *MASQUE* Alternative Plan Generator provides a team-level ‘priority policy’ option with which all initiators at once can be instructed to give priority to their own land use when deciding upon the first finalization of their plans in order to set the initial direction of the plan’s development. When the option is enabled, an initiator uses the claim accentuation instrument (section 7.5.2.2) to stress his or her claims to the maximum and, thus, to make his or her claims most explicit in the first phase of the plan development process. As soon as the first finalization has been made, the initiator’s claim accentuation takes the user-defined value again. Although enabling the priority policy option does not imply that the first finalization will concern the initiator’s own land use in all circumstances – e.g., participating agents can accentuate their claims as well – influence on the process is to be expected.

In the benchmark, the priority policy was assumed not operative. The results of testing the effects of enabling the option (Figure 7-24) show that changes in both plan evolution and outcome take place that could be expected. In all four plans the initiator succeeds in having the first finalized cell concern his or her own land use, which has a clear and logical effect on the direction of plan evolution. Particularly interesting are the benefits for small-task initiators, who are able to have their land use located in the

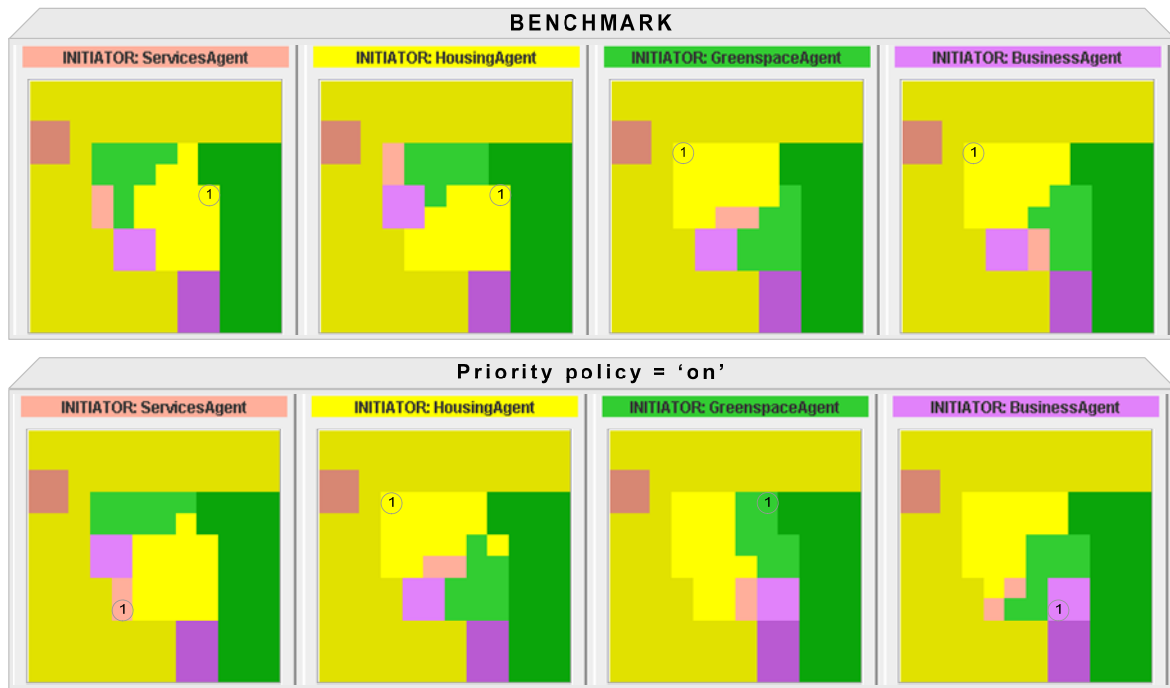


Figure 7-24. Results of testing the team-based priority policy for initiators
 Note: ① indicates the location of each plan's first finalization

(initially) most preferred part of the study area, which secures a certain level of performance based on relationships with existing land uses in the surroundings of the study area. Consequently, enactment of a priority policy has a general tendency to improve the performance of plans from the viewpoint of their initiators relative to the performance these agents measure for plans in which they participated.

7.7 CONCLUSIONS

The series of tests discussed in this chapter have verified the operation of agents within a cooperative process aimed at generating alternative plans. The findings surely show promising in the sense that the applied viewpoint approach and related distinction in agent roles (section 5.2.1) appear to be an effective means to let the process result in sets of alternatives that are both generally distinctive in their spatial patterns and useful in the insight they can provide into the interplay between agents (or land uses). Furthermore, the tests reveal that the *MASQUE* Alternative Plan Generator reacts well to changes in case characteristics. Overall, the effects of differences in surroundings or plan program are evident and, most important, explainable from the agents' decision-making. Regrettably, however, the model falls a bit short in its robustness related to the applied resolution, as

the results reveal a certain degree of arbitrariness that traces back to the start of the process when a cell needs to be selected for the first finalization.

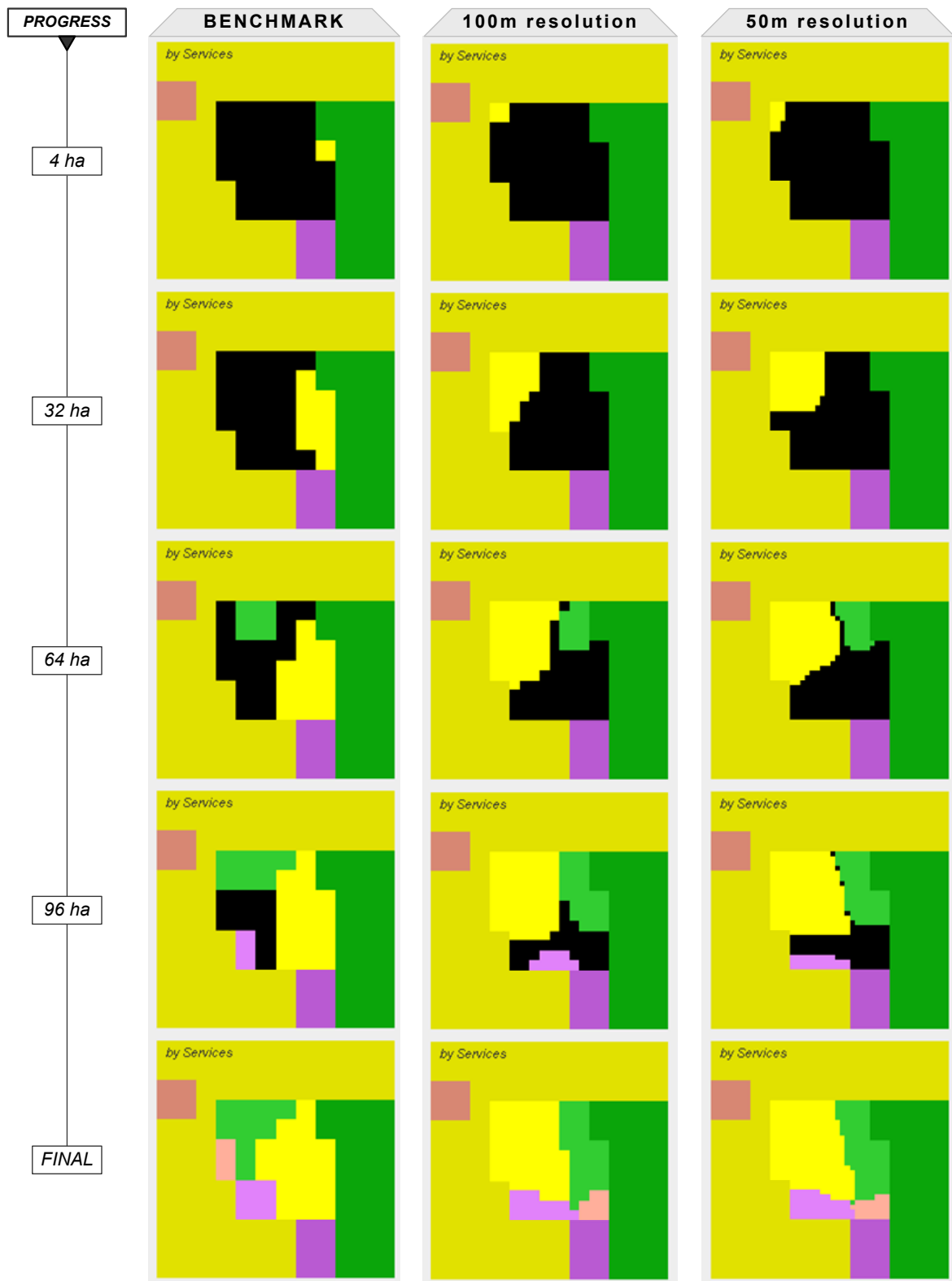
The tested types of decision rules all appear effective for both large-task agents and small-task agents, albeit that the magnitude of their effect, logically, depends on the actual aspect that is measured by the rule, the weight given to the rule in agent's decision model, and case characteristics. Although this was substantiated by means of only simple decision rules, it forms a good foundation for following efforts aimed at the sophistication of agents' knowledge and reasoning. Whereas now there may still be doubt about the degree of realism for some plan outcomes, this is likely to improve when agents become more advanced representations of human experts.

Finally, regarding the behaviour of agents – for both claiming and assigning – the tested parameters have noticeable and mostly explainable effects. Examination of these effects reveals that it might be most difficult to find a proper setting for the clustering intensity an agent should pursue, as the effects for the individual agent and those for the team are found to be rather opposite. For the claiming strategy, it seems that a low level of opportunism and an intermediate degree of accentuation have most tendency to foster both individual and collective interests, while enacting a claim scaling policy might be best. With concern to the assigning behaviour of agents, it appears likely that the quality of generated plans improves when reducing the differences made between land uses in agents' initial weight sets, while adjustment of weight sets during the process seems to be only beneficial to small-task agents. The most effective parameter related to assigning behaviour, however, is the priority policy that revealed promising results for the tested case. Given their clear differences in effect, we can conclude that the behavioural parameters in the model are generally operating in the anticipated complementary fashion.

The model specified in Chapter 5 has only been partially implemented (section 6.3). Consequently, several aspects still remain for testing and study, such as the effects of applying different interaction protocols (section 5.4) or the effects of using different (e.g., more detailed) land use classifications. The next and last chapter will formulate the apparent needs for further research, as part of the overall evaluation of the research project.

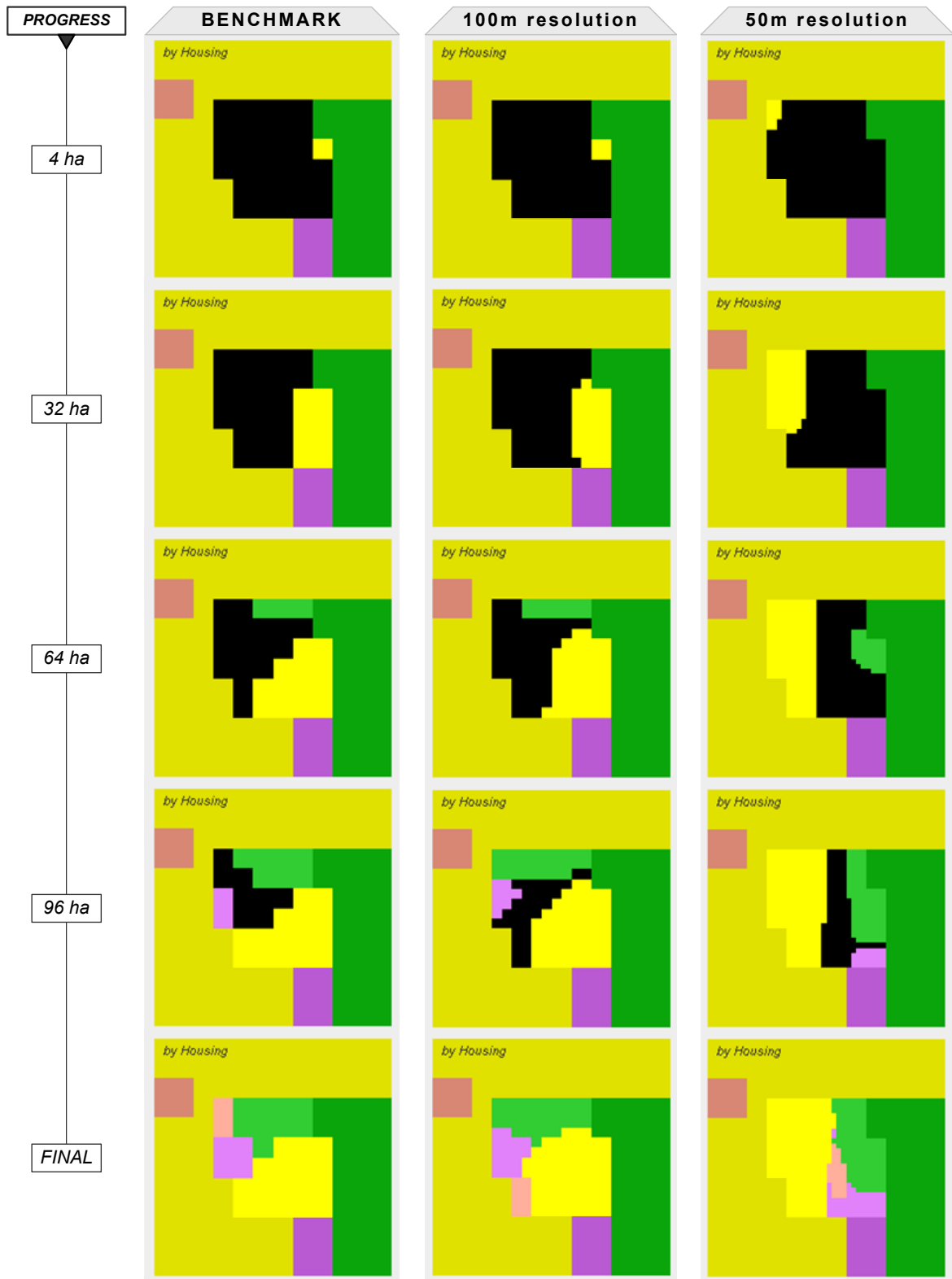
Appendix 7-A

Evolution of the plan initiated by the services agent



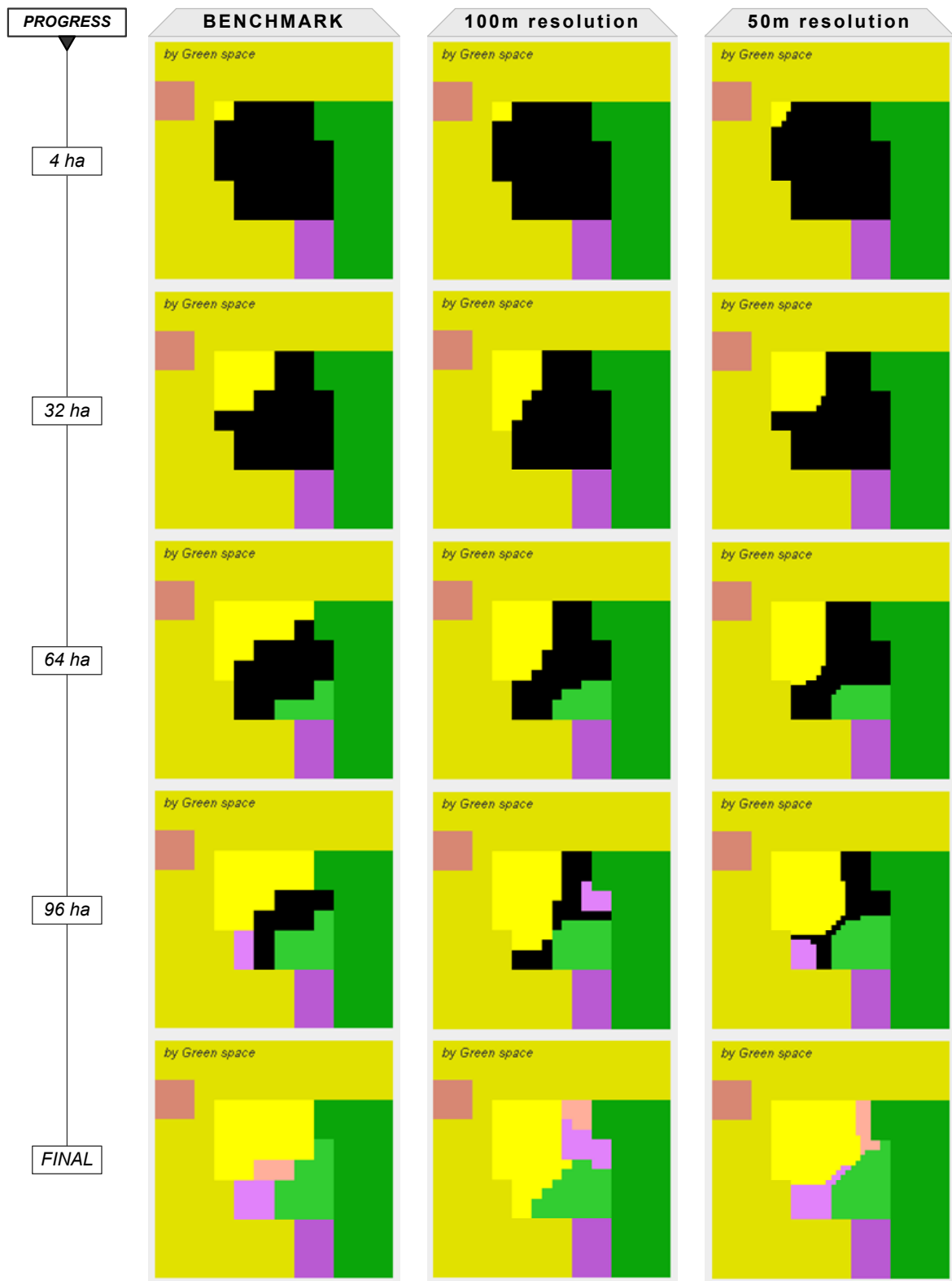
Appendix 7-B

Evolution of the plan initiated by the housing agent



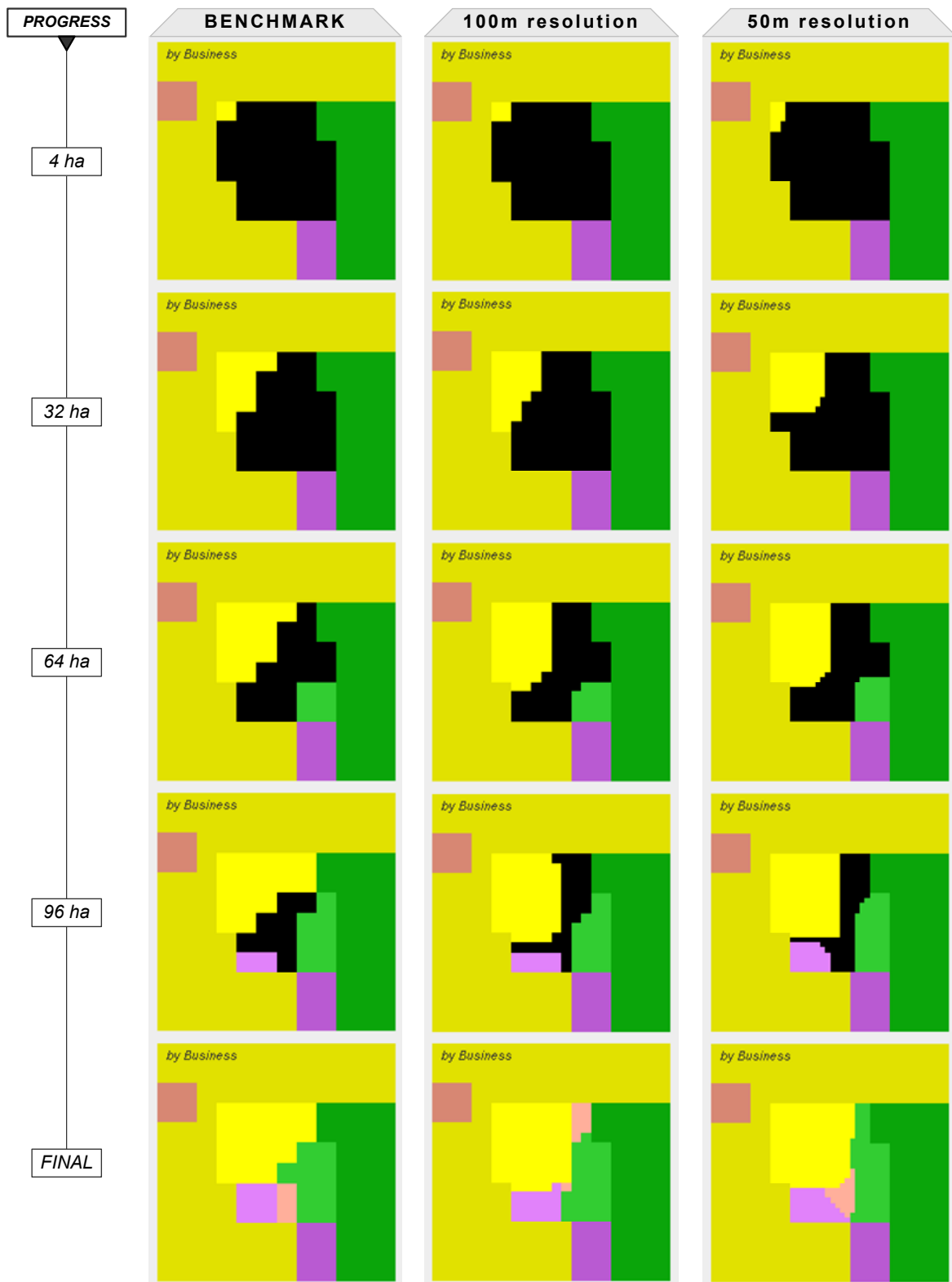
Appendix 7-C

Evolution of the plan initiated by the green space agent



Appendix 7-D

Evolution of the plan initiated by the business agent



8 CONCLUSIONS AND DISCUSSION

This last chapter summarizes the major conclusions that can be derived from the three parts identified in this thesis, discusses the believed potentials of multi-agent based PSS, and formulates possible directions of future research.

8.1 INTRODUCTION

In response to the persistent need for more advanced and more intelligent decision support in planning, the goal of this research project has been to explore the opportunities of using multi-agent technology within the context of Planning Support Systems (PSS) to a wider extent than the current state-of-the-art. To that effect, a three-stage study has been conducted. First, an inventory has been made of ongoing PSS developments against the background of the supposed ideal PSS, which was supplemented with an investigation of the opportunities that multi-agent technology is offering in this perspective. Second, conceptual principles for structuring a PSS framework have been developed, based on a multi-agent concept that fully exploits the planning support potentials of the technology. This was further expounded with the specification of a multi-agent model that addressed one of the core tasks in planning: generating alternative plans. Third, a prototype application of this model has been designed and implemented, followed by a series of sensitivity tests to illustrate the versatility and viability of multi-agent concepts for planning support.

8.2 CONCLUSIONS FROM PART I

Given the difficulties that planners are facing, the ideal PSS is commonly assumed to consist of three components: an *information* component to provide capabilities of a GIS, a *models* component to incorporate purpose-built decision support models or procedures, and a *visualization* component for realistic, real-time, interactive presentation of the impact of decisions. Together, these components require contributions from a wide range of research disciplines. Many advances have already been made in the last decades, but also many issues still remain to be properly addressed, such as the accommodation of temporal and disaggregate data, the provision of 3D spatial analysis, and the inclusion of design support tools. However, it goes without saying that the most problematic part in

the quest for ideal PSS is to have a comprehensive system address the various issues in an integrated fashion. This is especially true when realizing that, among many other complicating factors, planning has a location-specific nature, demands for data are diverse and changing, and decision support tools evolve owing to advances in both technologies and methodologies. Hence, the ultimate call is for a framework that is not only comprehensive but also flexible in terms of extensibility and adaptability.

In the last decade, it is recognized that the only appropriate answer to this call is the development of systems that provide intelligent solutions, in particular for the facilitation of information and control flows, the use of uncertain or fuzzy data, and knowledge acquisition and management. Along with the moderating role of GIS in the context of PSS witnessed in current developments, it appears that course is set for a system framework in which the collection of tools (or *models*) forms the central component that incorporates knowledge of various types and concerning various disciplines, while the *information* and *visualization* components move to the background to play serving roles. Yet, two main concerns rise from this envisioned framework. First, the encapsulation of knowledge in the mechanisms of the *models* component is very likely to hamper the framework's flexibility with respect to retrieving, updating and adding knowledge. Disentangling knowledge and models to form two separate but interlinked components appears to be an obvious improvement. Second, the system understandability to users and, thus, the system usability is put at risk when the framework is structured too flexible and user guidance and assistance is limited or lacking.

As a branch of artificial intelligence, multi-agent technology offers promising concepts and techniques that give it the potential to improve PSS with concern to both system usage and system capabilities. Its anthropomorphic nature allows designing PSS as organizations of computational equivalents of humans ('agents'), which is a highly recognizable abstraction to planners – who work with teams of specialists in many disciplines that all make unique contributions to the process – as opposed to the logical or mechanical abstractions underlying traditional systems. Within this abstraction, complexity and diversity are handled by modularity, i.e., agents are assigned specific problem solving knowledge and skills, while the interaction between agents is the mechanism that brings synergy to the system's overall problem solving capabilities. The key opportunities of utilizing multi-agent technology in PSS developments extend over three dimensions. First, it provides the means to improve user assistance by having agents accomplish tasks on behalf of users and offer guidance through procedures and processes.

Second, it offers the possibility of directly representing individuals, their behaviour and their interactions, which not only answers the strong demand of planners for disaggregate models that describe the urban system, but also the need of explicitly capturing the knowledge of various specialists in the system. Third, it can serve the purpose of software design in terms of interconnecting components (either existing or new), organizationally structuring a system and having it evolve and adapt to users. Consequently, there is a wide range of opportunities, from the improvement of user-system interaction, via the enhancement of a system's modelling capabilities and the representation of multi-actor decision-making, up to the optimization of components and system structure.

8.3 CONCLUSIONS FROM PART II

In an effort to envision how multi-agent technology can serve as the guiding principle for structuring ideal PSS, the identified opportunities provided by the technology to the field of planning support were blended into a comprehensive conceptual framework for a multi-agent planning support system named *MASQUE* that, as the focus of this research, aims at supporting land use planning.

Regarding architecture, two main changes are to be considered crucial for improving the usability of PSS and to enhance the functionalities. First, the traditional trinity of equal-level components needs a rearrangement that (i) merges the *information* and *visualization* components into a supportive integrated GIS/VR component, (ii) centres the *models* component together with a newly added *knowledge* component, and (iii) incorporates different types of agents to facilitate the interplay of components and the interaction between user and system. Second, a vital condition is the contextualization of PSS to the plan development process, in which the actual linkage between system and a generally formulated phased process is established by means of the agent organization.

The implementation of this rearranged and contextualized PSS framework requires an agent organization consisting of three types of agents. First, a group of 'interface agents' needs to be deployed for monitoring the user's behaviour with respect to the system's structure, functionalities and underlying plan development process. These agents can gain insight into the user's abilities to deal with these issues, which enables them to react on occurring problems by releasing appropriate suggestions, warnings, ideas, and so on. Second, a group of 'tool agents' needs to be embedded in the system's *models* component, organized according to the identified process phases. These agents can assist

users with respect to organizing the system's collection of tools (i.e., adding, constructing, customizing, and evaluating tools) and help users or other agents with selecting and using specific tools. Third, a group of 'domain agents' needs to be assembled to form the system's *knowledge* component and represent the variety of land use specialists involved in planning. Having knowledge about a variety of aspects regarding a particular land use, these agents can offer assistance throughout the process, from problem identification to the generation and evaluation of alternatives.

Given the advocated necessity of adding a *knowledge* component to the conceptual PSS framework, and the concurrent need to demonstrate the suitability and potential of multi-agent concepts for planning support in ways that add up to the current mainstream of agent-based micro-simulation tools, our focus has been directed at the operation of agents within the *knowledge* component. A model was specified that describe how these agents – who, referred to as *domain managers*, are specialists in particular land uses – can cooperatively generate sets of alternative plans through an increasingly informed search process in which plan proposals are incrementally drawn up. The suggested process was based on a 'viewpoint approach' that, from a given area description ('*inventory*') and a list of land use demands ('*plan program*'), lets every agent (i) initiate and lead the development of a plan proposal and, on request, (ii) participate in others' proposals. Agents express their claims for raster cells (i.e., land units) by determining the *relative suitability* of available cells in terms of *expected utility*, based on information extracted from the environment and/or received from each other. Initiating agents collect claims, process them into assignments, and suggest these assignments to participating agents. Since land uses are commonly interdependent, agents influence each other through their claims, which implies their decision-making takes place under conditions of uncertainty. Hence, in a probabilistic manner, they act based on their *beliefs* about what claims are most attractive to them, given their *beliefs* about the most likely outcome of the plan proposal considering all interests known at a certain moment. Moreover, the need to balance personal interests with general interests (i.e., a joint solution) makes the formulation of both claims (i.e., a balancing of beliefs) and assignments (i.e., a balancing of claims) into *strategic actions*. The behaviours of agents for making claims and assignments were captured by small sets of parameters, whereas their decision-making was assumed to be based on individual decision models consisting of expandable sets of rules – taking the form of utility functions – to judge the relative suitability of cells by means of a weighted measurement of both physical and spatial attributes.

8.4 CONCLUSIONS FROM PART III

To demonstrate that multi-agent technology is more resourceful than merely to serve as an instrument to develop micro-simulation tools, the specified model for alternative plan generation was implemented in a prototype application: the *MASQUE* Alternative Plan Generator. The usefulness and applicability of available toolkits, shells, platforms, and so on, to ease the development of multi-agent applications were found questionable in this respect, given the clear discrepancies between the rationale of these tools and the concepts of our model. Consequently, it was decided to design and build the application solely based on the object-oriented programming language Java, using the software development environment of Borland® JBuilder™.

The structural design chosen for our application consisted of three intertwined parts: the environment that agents are dealing with, the team organization they form, and the mental constructs they use. The consequent implementation was aimed at having a team of four agents – specialists regarding housing, business, green space and services – generate alternative plans for rasterized environments with study area surroundings that are configurations of one or more of the same land uses. The resulting application required only limited input in the form of a ‘project input file’ that gives the necessary area description (i.e., its dimensions and existing land uses) and plan program (i.e., the demand per land use); default settings for both behavioural parameters and decision rules are automatically uploaded. The settings and outputs of the model were organized in the I/O frame of the application that consists of three tabs. The first tab – labelled ‘Behaviour’ – contains fields to set all the necessary parameters for defining the individual behaviour of agents. The second tab – labelled ‘Decision models’ – provides the means to formulate each agent’s personal decision model with which he or she performs land suitability analysis in order to determine the expected utility per cell. After running the model, the third tab – labelled ‘Results’ – displays the set of alternative plans (including the performances measured by each agent). Animation functionalities were added to enable examining and comparing the incremental development of plans.

The demonstration of the suggested multi-agent model for alternative plan generation was given the form of a comprehensive series of sensitivity analysis tests, implying an investigation of the effects of systematically changing the model’s settings of behavioural parameters and decision rules. This has provided insight into the relationships inherent to the model and the relation between model input and output. The findings

showed promising in the sense that the mechanism of the model appeared to be effective in letting the process result in sets of alternatives that are both distinctive and useful. Furthermore, the tests revealed that the *MASQUE* Alternative Plan Generator reacts well to changes in case characteristics (i.e., in surroundings and plan program) as the effects are evident and explainable. However, the model was less robust as wished for concerning changes in the applied resolution. Then again, all tested types of decision rules showed effective, logically with a varying magnitude. Also changes in the behaviour of agents – for both claiming and assigning - had noticeable and mostly explainable effects. Given their clear differences in (magnitude of) effect, we could conclude that the behavioural parameters are operating in the anticipated complementary fashion.

8.5 DISCUSSION

The study reported in this thesis has identified the main applications of multi-agent technology that are expected to enable the development of more intelligent and flexible PSS, in terms of its operation, functionality, usability, extensibility and adaptability. To illustrate the implications of using a multi-agent concept as the guiding principle for PSS, a conceptual framework has been formulated in terms of required components and agent types. Given the reality of planners being faced with a variety of complications and the consequent need for more advanced and more intelligent decision support, this investigation and subsequent formulation of principles can be regarded as having been a purposeful and meaningful endeavour.

Current PSS developments show that the application of multi-agent concepts remains restricted to the development of agent-based micro-simulation tools for forecasting and evaluating alternative plans. We have argued that, in an entirely complementary way, multi-agent concepts can equally well be used for the preceding task of generating alternative plans, which has been exemplified by means of the development of a prototype application concerning a multi-agent model for alternative plan generation. The series of tests performed on this application has given insight into the details and, thus, differences of using multi-agents for another planning support purpose. In this respect, the illustrative part of our study can be considered as having been both original and valuable.

The developed application provides users with the possibility of quickly developing small sets of alternative plans for purposes of exploring the solution space of a project or

its inherent opportunities and threats, visualizing the effects of interdependencies between land uses (or agents), and so on. As such, planners can gain information that can help them in their search for good alternatives. However, there are a few points to bear in mind that originate from the fact that, above all, the current implementation has been driven by the abovementioned need for demonstrating the versatility of multi-agent concepts. As a consequence, users can adjust all behavioural parameters and decision rules without restrictions. Whereas this is more of a necessity for us (i.e., modellers and developers), it is not optimal in the long term with respect to the usability and understandability for end users. Moreover, this primary intention to illustrate new principles has made the application's design, implementation and testing to focus on examining the mechanisms of the underlying multi-agent model in order to clarify its conceptual differences with the customary use of multi-agents for planning support, as opposed to validating the model and delivering a ready-to-use instrument. For this, additional efforts will be required.

In overall conclusion, the question remains whether the academic community of PSS developers will consider it worthwhile, necessary or even desirable to conform to a general framework such as the one suggested in this study, irrespective of the question whether the advantages are evident. Rationally, the location-specific nature of planning could be regarded as the key factor that conflicts with the notion of a generally applicable framework and, thus, only permits the development of one-off applications. To consider this as an accomplished fact, however, is not in line with the advances being made with regard to component-based software engineering and development and the role that multi-agent technology can play in this respect. This study has made an attempt to bring this to mind, in an effort to facilitate the development of PSS that are ideal from the viewpoint of both developers and users.

8.6 POSSIBLE DIRECTIONS OF FUTURE RESEARCH

Regarding the developed *MASQUE* Alternative Plan Generator prototype, an obvious subject for future research is improvement of the current implementation. In this respect, the most obvious aspect to nominate for further investigation is the sensitivity to resolution. Although the results might already improve by using the model in a repeated series of runs with increasing resolutions (i.e., having the output of one run form the input of the next), a study is desired to find out to which extent this is a consequence of the interaction protocol that is currently applied, and whether this could be improved by

means of small adjustments or by defining a different protocol. In the latter case, conducting a survey among planning practitioners could be a good starting point.

Although derived from planning standards as much as possible, the rules used to form agents' decision models are fairly simple in the current implementation. To enhance the knowledge of agents and, thus, to turn them into better matching representations of human specialists, it is desirable to consult such persons involved in planning practice and capture their knowledge in rules. In addition, there is a need to incorporate other agents in the model, especially one to take care of transportation. This agent will in fact form the biggest challenge, as it needs to be determined how the assumed rasterization of space can accommodate this network-oriented land use. Once the team is expanded and more sophisticated knowledge is contained, validation of the model could be performed on the basis of having both human planners and specialists take part in a role-playing game for which the same aims are set as for the agents in the model.

Then, further efforts are required to gain insight into the interactive effects of the behavioural parameters in the model. In this light, there is also the aspect of finding the right settings for each agent in a given project. From the viewpoint of usability, this should not be a user task but a task of agents themselves. To realize this, the agents need to be equipped with learning capabilities that, mainly based on case-based reasoning techniques, should enable them to interpret the success of their strategic behaviour in terms relative to project characteristics, i.e., the typical aspects of the study area, its surroundings and the plan program. Since the learning of agents through case-based reasoning can as well serve the purpose of feeding, updating and extending their decision models, this aspect can be considered as a fundamental issue for future research.

Finally, concerning the suggested framework for multi-agent PSS, any attempt to build an operational system based on the introduced concepts is likely to bring up many detailed design and implementation questions that are hard, not to say impossible, to foresee at this stage. The novelty of the envisioned types of agents, however, is not in their operation, knowledge, capabilities or interaction but in their application in the context of PSS. Hence, experiences with multi-agent applications reported on in other fields of research will be valuable sources to build on. Even so, problems occurring due to the application in planning cannot be ruled out in advance. Once such problems have been identified, more topics for further research will become clear.

BIBLIOGRAPHY

- Arentze, T.A. (1999), *A Spatial Decision Support System for the Planning of Retail and Service Facilities*, Ph.D. Thesis, Eindhoven University of Technology, Eindhoven, The Netherlands.
- Arentze, T.A. and H.J.P. Timmermans (2000), *ALBATROSS, A Learning Based Transportation Oriented Simulation System*, European Institute of Retailing and Services Studies, The Hague, The Netherlands.
- Angehrn, A.A. and S. Dutta (1998), "Case-based decision support", *Communications of the ACM*, 41(5), pp. 157-165.
- Batty, M. (1989), "Urban modelling and planning: reflections, retrodictions, and prescriptions", in: B. MacMillan (ed.), *Remodelling Geography*, Basil Blackwell, Oxford, pp. 147-169.
- Batty, M. (1995), "Planning support systems and the new logic of computation", *Regional Development Dialogue*, 16(1), pp. 1-17.
- Batty, M. (2003), "Planning Support Systems: technologies that are driving planning" (preface), in: S. Geertman and J. Stillwell (eds.), *Planning Support Systems in Practice*, Springer-Verlag, Berlin, Germany, pp. v-viii.
- Batty, M. and P.J. Densham (1996), "Decision support, GIS, and urban planning", *Systema Terra*, V(1), pp. 72-76.
- Batty, M. and P.A. Longley (2003), "Researching the future of GIScience", in: P.A. Longley and M. Batty (eds.), *Advanced Spatial Analysis: The CASA book of GIS*, ESRI Press, Redlands, California, USA, pp. 427-435.
- Batty, M. and Y. Xie (1994a), "Modeling inside GIS: part 1: model structures, exploratory spatial analysis", *International Journal of Geographic Information Systems*, 8(3), pp. 291-307.
- Batty, M. and Y. Xie (1994b), "Modeling inside GIS: part 2: selecting and calibrating urban models using ARC/INFO", *International Journal of Geographic Information Systems*, 8(5), pp. 451-470.
- Batty, M. and Y. Xie (1994c), "From cells to cities", *Environment and Planning B: Planning and Design*, 21, pp. s31-s48.
- Batty, M. and A.G.O. Yeh (1991), "The promise of expert systems for urban planning", *Computers, Environment and Urban Systems*, 15(3), pp. 101-108.
- Blaser, A.D., M. Sester and M.J. Egenhofer (2000), "Visualization in an early stage of the problem-solving process in GIS", *Computers & Geosciences*, 26(1), pp. 57-66.
- Bishop, I.D. (1998), "Planning support: hardware and software in search of a system", *Computers, Environment and Urban Systems*, 22(3), pp. 189-202.
- Bishop, I.D. and H.R. Gimblett (2000), "Management of recreational areas: GIS, autonomous agents, and virtual reality", *Environment and Planning B: Planning and Design*, 27(3), pp. 423-435.
- Brail, R.K. and R.E. Klosterman (eds.) (2001), *Planning Support Systems, Integrating*

geographic information systems, models, and visualization tools, ESRI Press, Redlands, California, USA.

- Brill Jr., E.D., J.M. Flach, L.D. Hopkins and S. Ranjithan (1990), "MGA: a decision support system for complex, incompletely defined problems", *IEEE: Transactions on Systems, Man and Cybernetics*, 20(4), pp. 745-757.
- Burrough, P.A. and R.A. McDonnell (1998), *Principles of Geographical Information Systems*, Oxford University Press Inc., New York, USA.
- Carsjens, G.J., R.J.A. van Lammeren and A. Ligtenberg (2003), "STEPP: a strategic tool for integrating environmental aspects into planning procedures", in: S. Geertman and J. Stillwell (eds.), *Planning Support Systems in Practice*, Springer-Verlag, Berlin, Germany, pp. 139-154.
- Chapin, T.S. (2003), "Revolutionizing the core: GIS in the planning curriculum", *Environment and Planning B: Planning and Design*, 30(4), pp. 565-573.
- Cordella, L.P. , P. Foggia, C. Sansone, F. Tortorella and M. Vento (1999), "Reliability parameters to improve combination strategies in multi-expert systems", *Pattern Analysis & Applications*, 2(3), pp. 205-214.
- Cuena, J. and S. Ossowski (1999), "Distributed models for decision support", in: G. Weiss (ed.), *Multiagent Systems, A Modern Approach to Distributed Artificial Intelligence*, MIT Press, London, UK, pp. 459-504.
- Darlington, K. (2000), *The Essence of Expert Systems*, Prentice Hall, Harlow, UK.
- Davis Jr., C.A. and A.H.F. Laender (1999), "Multiple representations in GIS: materialization through map generalization, geometric, and spatial analysis operations", in: C.B. Medeiros (ed.), *Proceedings of the 7th International Symposium on Advances in Geographic Information Systems (ACM-GIS '99)*, Kansas City, Missouri, USA, November 2-6, pp. 60-65.
- Densham, P. (1991), "Spatial Decision Support Systems", in: D.J. Maguire, M.F. Goodchild and D.W. Rhind (eds.), *Geographical Information Systems: Principles and Applications*, Longmans, London, UK, pp. 403-412.
- Devisch, O., H.J.P. Timmermans, T.A. Arentze and A.W.J. Borgers (2004), "Towards a generic multi-agent engine for the simulation of spatial behavioural processes", in: J.P. Van Leeuwen and H.J.P. Timmermans (eds.), *Recent Advances in Design and Decision Support Systems in Architecture and Urban Planning*, Kluwer Academic Publishers, Dordrecht, The Netherlands, pp. 145-160.
- Dodge, M. and B. Jiang (1997), "Geographical Information Systems for urban design: providing new tools and digital data for urban designers", paper presented at the Learning Spaces Conference, De Montfort University, Milton Keynes, UK, November 26, http://www.casa.ucl.ac.uk/newvenue/learning_spaces.pdf (last accessed: October 18, 2005).
- Edmonds, E.A., L. Candy, R. Jones and B. Soufi (1994), "Support for collaborative design: agents and emergence", *Communications of the ACM*, 37(7), pp. 41-47.
- Edwards, J.S. (1992), "Expert systems in management and administration – are they really different from Decision Support Systems?", *European Journal of Operational Research*, 61(1-2), pp. 114-121.

-
- El-Najdawi, M.K. and A.C. Stylianou (1993), "Expert Support Systems: Integrating AI Technologies", *Communications of the ACM*, 36(12), pp. 55-65/103.
- Ennis, G. and M. Lindsay (2000), "VRML possibilities: the evolution of the Glasgow model", *IEEE Multimedia*, 7(2), pp. 48-51.
- Epstein, J.M. (1999), "Agent-based computational models and generative social science", *Complexity*, 4(5), pp. 41-60.
- Feng, C.M. and J.J. Lin (1999), "Using a genetic algorithm to generate alternative sketch maps for urban planning", *Computers, Environment and Urban Systems*, 23(2), pp. 91-108.
- Ferber, J. (1999), *Multi-Agent Systems, An Introduction to Distributed Artificial Intelligence*, Addison-Wesley, Harlow, UK.
- Ferrand, N. (1996), "Modelling and supporting multi-actor spatial planning using multi-agents systems", in: *Proceedings of the Third NCGIA Conference on Integrating GIS and Environmental Modelling*, Santa Fe, New Mexico, USA, January 21-25, http://www.ncgia.ucsb.edu/conf/SANTA_FE_CD-ROM/sf_papers/ferrand_nils/santafe.html (last accessed: October 18, 2005).
- Flores-Mendez, R.A. (1999), "Towards a standardization of multi-agent system frameworks", *ACM Crossroads*, 5(4), <http://www.acm.org/crossroads/xrds5-4/multiagent.html> (last accessed: October 18, 2005).
- Fonseca, F.T., M.J. Egenhofer, C.A. Davis Jr. and K.A.V. Borges (2000), "Ontologies and knowledge sharing in urban GIS", *Computers, Environment and Urban Systems*, 24(3), pp. 251-271.
- Fujii, T., K. Imamura, T. Yasuda, S. Yokoi and J. Toriwaki (1995), "A virtual scene simulation system for city planning", in: R.A. Earnshaw and J.A. Vince (eds.), *Computer Graphics: Developments in Virtual Environments*, Academic Press, London, UK.
- Geertman, S. (2002), "Participatory planning and GIS: a PSS to bridge the gap", *Environment and Planning B: Planning and Design*, 29(1), pp. 21-35.
- Geertman, S. and J. Stillwell (eds.) (2003), *Planning Support Systems in Practice*, Springer-Verlag, Berlin, Germany.
- Geertman, S. and J. Stillwell (2004), "Planning support systems: an inventory of current practice", *Computers, Environment and Urban Systems*, 28(4), pp. 291-310.
- George, R.V. (1995), "Expert support for problem formulation in urban planning", *Environment and Planning B: Planning and Design*, 22(2), pp. 175-190.
- Hagen-Zanker, A., G. Engelen, J. Hurkens, R. Vanhout and I. Uljee (2005), *Map Comparison Kit 2: User Manual*, publication of the Research Institute for Knowledge Systems (RIKS bv), Maastricht, The Netherlands.
- Han, S.Y. and T.J. Kim (1990), "Intelligent urban information systems: review and Prospects", in: T.J. Kim, L.L. Wiggins, and J.R. Wright (eds.), *Expert Systems: applications to urban planning*, Springer-Verlag, New York, USA, pp. 241-264.
- Harris, B. (2001), "Sketch planning: systematic methods in planning and its support", in: R.K. Brail and R.E. Klosterman (eds.), *Planning Support Systems: Integrating geographic information systems, models, and visualization tools*, ESRI Press,

-
- Redlands, California, USA, pp. 59-80.
- Harris, B. and M. Batty (2001), "Locational models, geographic information, and Planning Support Systems", in: R.K. Brail and R.E. Klosterman (eds.), *Planning Support Systems: Integrating geographic information systems, models, and visualization tools*, ESRI Press, Redlands, California, USA, pp. 25-57.
- Hendler, J. (1999), "Making sense out of agents", *IEEE Intelligent Systems*, 14(2), pp. 32-37.
- Hewett, T.T., R. Baecker, S. Card, T. Carey, J. Gasen, M. Mantei, G. Perlman, G. Strong and W. Verplank (1996), *ACM SIGCHI Curricula for Human-Computer Interaction*, report of the ACM Special Interest Group on Computer-Human Interaction, Curriculum Development Group, <http://sigchi.org/cdg/> (last accessed: October 18, 2005).
- Hopkins, L.D. (1999), "Structure of a planning support system for urban development", *Environment and Planning B: Planning and Design*, 26(3), pp. 333-343.
- Hopkins, L.D., N. Kaza and R.V. George (2003), "Planning Markup Language: representing the meanings of plans and regulations", paper presented at the AESOP/ACSP Joint Conference, Leuven, Belgium, July 8-12, <http://www.rehearsal.uiuc.edu/projects/PML/reports/ascppaper2003.pdf> (last accessed: October 18, 2005).
- Hopkins, L.D., R. Ramanathan and R.V. George (2004), "Interface for a sketch-planning workbench", *Computers, Environment and Urban Systems*, 28(6), pp. 653-666.
- Huhns, M.N. and L.M. Stephens (1999), "Multiagent systems and societies of agents", in: G. Weiss (ed.), *Multiagent Systems, A Modern Approach to Distributed Artificial Intelligence*, MIT Press, London, UK, pp. 79-120.
- Ishii, H., J. Underkoffler, D. Chak, B. Piper, E. Ben-Joseph, L. Yeung and Z. Kanji (2002), "Augmented urban planning workbench: overlaying drawings, physical models and digital simulation", in: *Proceedings of the First IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2002)*, Darmstadt, Germany, September 30 - October 1, pp.203-211.
- Janca, P.C. and D. Gilbert (1998), "Practical design of intelligent agent systems", in: N.R. Jennings and M.J. Wooldridge (eds.), *Agent Technology: Foundations, Applications, and Markets*, Springer-Verlag, Berlin, Germany, pp. 73-89.
- Jennings, N.R. (2001), "An agent-based approach for building complex software systems", *Communications of the ACM*, 44(4), pp. 35-41.
- Jennings, N.R., K. Sycara and M.J. Wooldridge (1998), "A roadmap of agent research and development", *Autonomous Agents and Multi-Agent Systems*, 1(1), pp.7-38.
- Jennings, N.R. and M.J. Wooldridge (1998), "Applications of intelligent agents", in: N.R. Jennings and M.J. Wooldridge (eds.), *Agent Technology: Foundations, Applications, and Markets*, Springer-Verlag, Berlin, Germany, pp. 3-28.
- Karssenber, D. and K. de Jong (2005), "Dynamic environmental modelling in GIS, part 1: modelling in three spatial dimensions", *International Journal of Geographical Information Science*, 19(5), pp. 559-579.
- Klosterman, R.E. (1999a), "New perspectives on planning support systems" (guest

-
- editorial), *Environment and Planning B: Planning and Design*, 26(3), pp. 317-320.
- Klosterman, R.E. (1999b), "The What if? collaborative planning support system". *Environment and Planning B: Planning and Design*, 26(3), pp. 393-408.
- Klosterman, R.E. (2001), "Planning Support Systems: a new perspective on computer-aided planning", in: R.K. Brail and R.E. Klosterman (eds.), *Planning Support Systems: Integrating geographic information systems, models, and visualization tools*, ESRI Press, Redlands, California, USA, pp. 1-23.
- Koch, A. (2000), "Linking multi agent systems and GIS, modeling and simulating spatial interactions", paper presented at the 29th International Geographical Congress, Seoul, South Korea, August 14-18, <http://www.rwth-aachen.de/geo/Ww/deutsch/MultiAgentsKoch.pdf> (last accessed: October 18, 2005).
- Königer, A. and S. Bartel (1998), "3D-GIS for urban purposes", *GeoInformatica*, 2(1), pp. 79-103.
- Kwartler, M. and R.N. Bernard (2001), "CommunityViz: an integrated planning support system", in: R.K. Brail and R.E. Klosterman (eds.), *Planning Support Systems: Integrating geographic information systems, models, and visualization tools*, ESRI Press, Redlands, California, USA, pp. 285-308.
- Leung, Y. (1997), *Intelligent Spatial Decision Support Systems*, Springer-Verlag, Berlin, Germany.
- Lieberman, H. (1998), "Integrating user interface agents with conventional applications", *Knowledge-Based Systems*, 11(1), pp. 15-23.
- Li, Q., W. Shi and B. Yang (1999), "3D city modelling based on an integrated data model", in: *Proceedings of the Geoinformatics'99 Conference: Geoinformatics and Socioinformatics*, Ann Arbor, Michigan, July 19-21, http://www.umich.edu/~iinet/chinadata/geoim99/Proceedings/li_qingquan.pdf (last accessed: October 18, 2005).
- Liggett, R.S. and W.H. Jepson (1995a), "An integrated environment for urban simulation", *Environment and Planning B: Planning and Design*, 22(3), pp. 291-302.
- Liggett, R.S. and W.H. Jepson (1995b), "Use of real-time visual simulation technology for urban planning/design decision making", in: *Proceedings of the Fourth International Conference on Computers in Urban Planning and Urban Management*, Melbourne, Australia, July 11-14, pp. 51-63.
- Ligtenberg, A., A.K. Bregt and R. van Lammeren (2001), "Multi-actor-based land use modelling: spatial planning using agents", *Landscape and Urban Planning*, 56(1-2), pp. 21-33.
- Ligtenberg, A., M. Wachowicz, A.K. Bregt, A. Beulens and D.L. Kettenis (2002), "Multi-agent systems modelling for actor based spatial planning: the human factor", *Proceedings Geo-Informatiedag Nederland 2002*, Ede, The Netherlands, February 15, pp. 39-44.
- Ligtenberg, A., M. Wachowicz, A.K. Bregt, A. Beulens and D.L. Kettenis (2004), "A design and application of a multi-agent system for simulation of multi-actor spatial planning", *Journal of Environmental Management*, 72(1-2), pp. 43-55.

-
- Lynch, K. and G. Hack (1984), *Site Planning*, MIT Press, London, UK.
- Ma, L., T.A. Arentze, A.W.J. Borgers and H.J.P. Timmermans (2004), "Using bayesian decision networks for knowledge representation under conditions of uncertainty in multi-agent land-use simulation models", in: J.P. Van Leeuwen and H.J.P. Timmermans (eds.), *Recent Advances in Design and Decision Support Systems in Architecture and Urban Planning*, Kluwer Academic Publishers, Dordrecht, The Netherlands, pp. 129-144.
- Maes, P. (1994), "Agents that reduce work and information overload", *Communications of the ACM*, 37(7), pp. 31-40.
- Malczewski, J. (1999), *GIS and multicriteria decision analysis*, John Wiley & Sons, Inc., New York, USA.
- Mangina, E. (2002), *Review of Software Products for Multi-Agent Systems*, report of AgentLink II, the European Network of Excellence for Agent-Based Computing, <http://www.agentlink.org/admin/docs/2002/2002-47.pdf> (last accessed: October 18, 2005).
- Maren, G. van (2003), "Key to virtual insight: a 3D GIS and virtual reality system", in: S. Geertman and J. Stillwell (eds.), *Planning Support Systems in Practice*, Springer-Verlag, Berlin, Germany, pp. 193-204.
- Maren, G. van, and J. Moloney (2000), "A virtual interactive urban planning system: Karma-vI", in: H.J.P. Timmermans and B. de Vries (eds.), *Proceedings of the 5th International Conference on Design and Decision Support Systems in Architecture and Urban Planning*, Nijkerk, The Netherlands, August 22-25, pp. 196-203.
- Norman, D.A. (1994), "How might people interact with agents", *Communications of the ACM*, 37(7), pp. 68-71.
- Nwana, H.S. (1996), "Software agents: an overview", *Knowledge Engineering Review*, 11(3), pp. 205-244.
- Nwana, H. and D. Ndumu (1998), "A brief introduction to software agent technology", in: N.R. Jennings and M.J. Wooldridge (eds.), *Agent Technology: Foundations, Applications, and Markets*, Springer-Verlag, Heidelberg, Germany, pp. 29-49.
- Parker, D.C., S.M. Manson, M.A. Janssen, M.J. Hoffmann and P. Deadman (2003), "Multi-agent system for the simulation of land-use and land-cover change: a review", *Annals of the Association of American Geographers*, 93(2), pp. 314-337.
- Peuguet, D.J. (2001), "Making space for time: issues in space-time data representation", *GeoInformatica*, 5(1), pp. 11-32.
- Pohl, J., A. Chapman and K.J. Pohl (2000), "Computer-aided design systems for the 21st century: some design guidelines", in: H.J.P. Timmermans and B. de Vries (eds.), *Proceedings of the 5th International Conference on Design and Decision Support Systems in Architecture and Urban Planning*, Nijkerk, The Netherlands, August 22-25, pp. 307-324.
- Rand, W., M. Zellner, S.E. Page, R. Riolo, D.G. Brown and L.E. Fernandez (2002), "The complex interaction of agents and environments: an example in urban sprawl", in: *Proceedings of the Agent 2002 Conference on Social Agents: Ecology, Exchange, and Evolution*, Chicago, USA, October 11-12, pp. 149-161.

-
- Rao, A.S. and M.P. Georgeff (1995), "BDI agents: from theory to practice", in: *Proceedings of the 1st International Conference on Multi-Agent Systems (ICMAS-95)*, San Francisco, USA, June 12-14, pp. 312-319.
- Ravi (2000), *Data Model for Spatial Planning: IMRO* (in Dutch), publication 00-06 of Ravi, Amersfoort, The Netherlands.
- Ricordel, P.M. and Y. Demazeau (2000), "From analysis to deployment: a multi-agent platform survey", in: A. Omicini, R. Tolksdorf and F. Zambonelli (eds.), *Engineering Societies in the Agents World: First International Workshop, ESAW 2000, Berlin, Germany*, Lecture Notes in Computer Science, Vol. 1972, Springer-Verlag, Berlin, Germany, pp. 93-105.
- Rodrigues, A., C. Grueau, J. Raper and N. Neves (1997), "Research on spatial agents", in: *Proceedings of the Third Joint European Conference and Exhibition on Geographical Information (JEC-GI'97)*, Vienna, Austria, April 16-18, pp. 68-78.
- Rodrigues, A. and J. Raper (1999), "Defining spatial agents", in: J. Raper and A. Câmara (eds.), *Spatial Multimedia and Virtual Reality*, Research Monographs Series, Taylor & Francis, London, UK, pp. 111-129.
- Rutledge, A.J. (1971), *Anatomy of a Park : The Essentials of Recreation Area Planning and Design*, McGraw-Hill, London, UK.
- Schelhorn, T., D. O'Sullivan, M. Haklay and M. Thurstain-Goodwin (1999), "STREETS: an agent-based pedestrian model", CASA working paper series, paper 9, Centre for Advanced Spatial Analysis, University College London, London, UK, <http://www.casa.ucl.ac.uk/streets.pdf> (last accessed: October 18, 2005).
- Sengupta, R.R. and D.A. Bennett (2003), "Agent-based modelling environment for spatial decision support", *International Journal of Geographical Information Science*, 17(2), pp. 157-180.
- Shaw, S.L. and D. Wang (2000), "Handling disaggregate spatiotemporal travel data in GIS", *GeoInformatica*, 4(2), pp. 161-178.
- Simon, H.A. (1977), *The New Science of Management Decision*, Prentice-Hall, Englewood Cliffs, New Jersey, USA.
- Singh, R.R. (2003), "Sketching the city: a GIS-based approach", *Environment and Planning B: Planning and Design*, 26(3), pp. 455-468.
- Smith, A., M. Dodge and S. Doyle (1998), "Visual communication in urban planning and urban design", CASA working paper series, paper 2, Centre for Advanced Spatial Analysis, University College London, London, UK, <http://www.casa.ucl.ac.uk/urbanplan.pdf> (last accessed: October 18, 2005).
- Spaccapietra, S. (2001), "Spatio-temporal data models and languages" (editorial), *GeoInformatica* 5(1), pp. 5-9.
- Sycara, K.P. (1998), "Multiagent systems", *AI Magazine*, 19(2), pp. 79-92.
- Tang, M.X. (1997), "A knowledge-based architecture for intelligent design support", *Knowledge Engineering Review*, 12(4), pp. 387-406.
- Tang, C., L.D. Xu and S. Feng (2001), "An agent-based geographical information system", *Knowledge-Based Systems*, 14(5-6), pp. 233-242.
- Thurston, J. (2002), "Is GIS Working?: spinning wheels vs. moving ahead", *GIS Vision*

-
- Magazine*, May 2002, <http://www.gisvisionmag.com/vision.php?article=200205%2Ffeature2.html> (last accessed: October 18, 2005).
- Tobias, R. and C. Hofmann (2004), "Evaluation of free Java-libraries for social-scientific agent based simulation", *Journal of Artificial Societies and Social Simulation*, 7(1), <http://jasss.soc.surrey.ac.uk/7/1/6.html> (last accessed: October 18, 2005).
- Tobón, C. and M. Haklay (2003), "Human-computer interaction and usability evaluation in GIScience", in: P.A. Longley and M. Batty (eds.), *Advanced Spatial Analysis: The CASA book of GIS*, ESRI Press, Redlands, California, USA, pp. 391-405.
- Torrens, P.M. (2003), "Cellular automata and multi-agent systems as planning support Tools", in: S. Geertman and J. Stillwell (eds.), *Planning Support Systems in Practice*, Springer-Verlag, Berlin, Germany, pp. 205-222.
- Turban, E. (1988), *Decision Support and Expert Systems: Managerial Perspectives*, Macmillan Publishing Company, New York, USA.
- Tweed, C. (1998), "Supporting argumentation practices in urban planning and design", *Computers, Environment and Urban Systems*, 22(4), pp. 351-363.
- Vries, B. de, V. Tabak and H. Achten (2005), "Interactive urban design using integrated planning requirements control", *Automation in Construction*, 14(2), pp. 207-213.
- Weibel, R. and G. Dutton (1999), "Generalizing spatial data and dealing with multiple representations", in: P.A. Longley, M.F. Goodchild, D.J. Maquire and D.W. Rhind (eds.), *Geographical Information Systems, Principles, Techniques, Applications, and Management*, John Wiley & Sons, Inc., New York, USA, pp. 125-155.
- Wu, F. (2002), "Complexity and urban simulation: towards a computational laboratory", *Geography Research Forum*, 22, pp. 22-40.
- Xiang, W.N. and K.C. Clarke (2003), "The use of scenarios in land use planning", *Environment and Planning B: Planning and Design*, 30(6), pp. 885-909.
- Yeh, A.G.O. (1999), "Urban planning and GIS", in: P.A. Longley, M.F. Goodchild, D.J. Maquire and D.W. Rhind (eds.), *Geographical Information Systems, Principles, Techniques, Applications, and Management*, John Wiley & Sons, Inc., New York, USA, pp. 877-888.
- Yeh, A.G.O. and X. Shi (1999), "Applying case-based reasoning to urban planning: a new planning-support system tool", *Environment and Planning B: Planning and Design*, 26(1), pp. 101-115.
- Yeh, A.G.O. and J.J. Qiao (2004a), "Component-based approach in the development of a knowledge-based planning support system (KBPSS), Part 1: the architecture of KBPSS", *Environment and Planning B: Planning and Design*, 31(4), pp. 517-537.
- Yeh, A.G.O. and J.J. Qiao (2004b), "Component-based approach in the development of a knowledge-based planning support system (KBPSS), Part 2: the model and knowledge management systems of KBPSS", *Environment and Planning B: Planning and Design*, 31(5), pp. 647-671.
- Yeh, A.G.O. and C. Webster (2004), "Planning, government, information, and the Internet" (guest editorial), *Environment and Planning B: Planning and Design*, 31(2), pp. 163-165.

-
- Zambonelli, F. and H. Van Dyke Parunak (2003), "Signs of a revolution in computer science and software engineering", in: P. Petta, R. Tolksdorf and F. Zambonelli (eds.), *ESAW, engineering societies in the agents world: international workshop, 3rd, Madrid, Spain, September 16-17, 2002: revised papers*, Lecture Notes in Artificial Intelligence 2577, Springer-Verlag, Berlin, Germany, pp. 13-28.
- Zamenopoulos, T. and K. Alexiou (2003), "Structuring the plan design process as a coordination problem: the paradigm of distributed learning control coordination", in: P.A. Longley and M. Batty (eds.), *Advanced Spatial Analysis: The CASA book of GIS*, ESRI Press, Redlands, California, USA, pp. 407-426.
- Zlatanova, S. (2000), *3D GIS for Urban Development*, Ph.D. Thesis, Institute for Computer Graphics and Vision (ICGV), Graz University of Technology and International Institute for Aerospace Survey and Earth Sciences (ITC), Enschede, The Netherlands.

AUTHOR INDEX

- Achten, 29
Alexiou, 47
Angehrn, 42
Arentze, 23, 31, 56, 63
Baecker, 17
Bartel, 28
Batty, 4-5, 14-15, 17, 19-22, 25-27, 30
Ben-Joseph, 29
Bennett, 46
Bernard, 28
Beulens, 4, 33, 47
Bishop, 14-15, 46
Blaser, 29
Borges, 24
Borgers, 56, 63
Brail, 21, 27
Bregt, 4-5, 33, 47
Brill, 27, 70
Brown, 100
Burrough, 23
Candy, 63
Card, 17
Carey, 17
Carsjens, 25
Chak, 29
Chapin, 13
Chapman, 32
Clarke, 20
Cordella, 32
Cuena, 38
Darlington, 29-30
Davis, 22, 24
Deadman, 46
Demazeau, 99
Densham, 5, 14, 25
Devisch, 56, 63
Dodge, 28
Doyle, 28
Dutta, 42
Dutton, 22
Edmonds, 63
Edwards, 30
Egenhofer, 24, 29
El-Najdawi, 17, 31-32
Engelen, 129
Ennis, 28
Epstein, 26, 39, 43, 117
Feng, C.M., 27
Feng, S., 45-46
Ferber, 39-43
Fernandez, 100
Ferrand, 4, 41, 47
Flach, 27, 70
Flores-Mendez, 44
Foggia, 32
Fonseca, 24
Fujii, 28
Gasen, 17
Geertman, 6, 15, 17, 19, 21, 27, 50
George, 18, 24, 29-30
Georgeff, 39
Gilbert, 38-39
Gimblett, 46
Grueau, 46
Hack, 1
Hagen-Zanker, 129
Haklay, 16-18, 100
Han, 30-31
Harris, 14, 19-20, 22, 29
Hendler, 43, 44

Hewett, 17
Hoffmann, 46
Hofmann, 100
Hopkins, 18, 24, 27, 29, 48, 70
Huhns, 40
Hurkens, 129
Imamura, 28
Ishii, 29
Janca, 38-39
Janssen, 46
Jennings, 37-40, 42-43, 45, 50
Jepson, 20, 28
Jiang, 28
Jones, 63
Jong, de, 23
Kanji, 29
Karssenberg, 23
Kaza, 24
Kettenis, 4, 33, 47
Kim, 30-31
Klosterman, 2, 6, 13-16, 21-22, 25, 27, 30, 46
Koch, 46
Köninger, 28
Kwartler, 28
Laender, 22
Lammeren, van, 5, 25, 33, 47
Leung, 30, 31
Li, 28
Lieberman, 42, 44
Liggett, 20, 28
Ligtenberg, 4-5, 25, 33, 47
Lin, 27
Lindsay, 28
Longley, 4, 26-27
Lynch, 1
Ma, 56
Maes, 43
Malczewski, 14, 29-31
Mangina, 99
Manson, 46
Mantei, 17
Maren, van, 18, 28
Moloney, 18, 28
Ndumu, 42, 44
Neves, 46
Norman, 50
Nwana, 38, 42, 44
O'Sullivan, 100
Ossowski, 38
Page, 100
Parker, 46
Perlman, 17
Peuguet, 22
Piper, 29
Pohl, J., 32
Pohl, K.J., 32
Qiao, 19-20, 47, 62
Ramanathan, 18, 29
Rand, 100
Ranjithan, 27, 70
Rao, 39
Raper, 46
Ravi, 24, 64, 107
Ricordel, 99
Riolo, 100
Rodrigues, 46
Rutledge, 3, 57
Sansone, 32
Schelhorn, 100
Sengupta, 46
Sester, 29
Shaw, 23
Shi, W., 28
Shi, X., 4, 6
Simon, 3, 57

Singh, 24
Smith, 28
Soufi, 63
Spaccapietra, 6, 22, 45
Stephens, 40
Stillwell, 6, 15, 17, 19, 21, 27
Strong, 17
Stylianou, 17, 31, 32
Sycara, 37-39, 40-41, 44, 63
Tabak, 29
Tang, C., 45, 46
Tang, M.X., 20
Thurstain-Goodwin, 100
Thurston, 22
Timmermans, 23, 56, 63
Tobias, 100
Tobón, 16-18
Toriwaki, 28
Torrens, 26
Tortorella, 32
Turban, 29-31
Tweed, 4
Uljee, 129
Underkoffler, 29
Van Dyke Parunak, 37, 39
Vanhout, 129
Vento, 32
Verplank, 17
Vries, de, 29
Wachowicz, 4, 33, 47
Wang, 23
Webster, 21
Weibel, 22
Wooldridge, 38-40, 42, 45, 50
Wu, 26, 117
Xiang, 20
Xie, 4, 21
Xu, 45-46
Yang, 28
Yasuda, 28
Yeh, 4, 6, 14-15, 19-21, 25, 30, 47, 62
Yeung, 29
Yokoi, 28
Zambonelli, 37, 39
Zamenopoulos, 47
Zellner, 100
Zlatanova, 28



SUBJECT INDEX

- accessibility, 19, 46, 66, 70, 74, 82, 112, 121, 125-126, 128, 133-134, 138-139
- adjacency, 19, 46, 66, 70, 74-75, 82, 112, 121-122, 126, 128, 135-136, 138-139
- agent
 - anatomy, 38
 - autonomy, 39, 43, 55, 66
 - categories, 42-44
 - definition, 38-39
 - physical environment, 40, 49-50
 - social environment, 40
- alternative plan evaluation, 16, 50, 59, 70, 162
- alternative plan generation, 7, 16, 27, 47, 49, 56, 59, 68-70, 72, 90, 96, 99, 102, 105-106, 115, 163-164
- anthropomorphy, 8, 38, 49, 51, 55, 60, 66, 96, 160
- assigning behaviour
 - compare precision, 88, 109, 120, 148
 - iteration limit, 88, 109, 120, 148
 - priority policy, 110, 120, 148, 151, 153
- assigning strategy
 - cell finalization, 88, 105, 109-110, 120, 144, 148
 - weight set adjustment, 87, 94, 109-110, 120, 148, 150
- assignments
 - assessment, 89, 91-93, 109, 120
 - completion method, 105, 109, 148
 - logit model, 82-83, 104
- beliefs
 - future state belief, 81, 85, 87, 90-91, 94, 103, 142
 - goal state belief, 77, 81, 85, 90-92, 103, 142
- claiming behaviour
 - clustering, 70, 80-81, 85, 120, 130, 139-141, 153
 - clustering intensity, 79-80, 104, 108, 120, 139-141, 153
 - neighbour distinction, 79, 108, 120, 141
 - scaling policy, 81, 85, 109, 120, 139, 146-147, 149, 153
- claiming strategy
 - accentuation, 86, 109, 120, 136, 142-145, 151, 153
 - opportunism, 85-86, 108, 120, 141-143, 153
- claims
 - agent decision model, 73, 77, 85, 89, 102, 110-112, 116, 121-123, 128, 133-138, 153, 162-163, 166
 - best set, 86-87, 109, 122, 142, 147
- collaboration, 40, 64
- complexity, 1, 4, 16-17, 22, 24, 27, 34, 37, 43, 50, 60, 68, 160
- component-based software development, 21, 48
- conditional probability table, 80, 83
- consultation, 2, 4-5, 7, 64-65, 67, 94
- cooperation, 4, 7, 40, 42, 68, 89, 91
- decision support, 1, 6, 15, 24-25, 30, 41, 47, 55, 159, 164
- Decision Support Systems, 14, 29-32, 35
- design support, 29, 159
- direct neighbourhood, 75, 110, 121-122

DSS-ES integration, 31
 DURP initiative, 2, 24
 expected utility, 73-75, 77-78, 88, 90,
 110-112, 122, 133-138, 162-163
 Expert Systems, 30-32, 35
 Geographical Information Systems, 13-
 15, 17-18, 20-23, 25-26, 28, 33-34,
 45-46, 56, 60, 159-161
 GIS-VR integration, 28, 57, 161
 grid representation, 72, 75, 101-103,
 106, 117, 124, 162
 human organization, 8, 37, 41
 human-computer interaction, 16-18
 IMRO information model, 24, 64, 107
 interaction protocol, 70, 90, 153
 sequential contributions, 94
 simultaneous contributions, 91, 94,
 105, 116
 inventory, 3, 15, 57-58, 65, 70-72, 90,
 159, 162
 iterative proportional fitting, 84, 87-88
 land use configuration, 1, 8, 59, 102,
 122, 124
 land use planning, 8-9, 25, 51, 55-56, 63,
 69, 161
 land use specialists, 4, 7, 27, 30-32, 47,
 49, 55, 63-66, 69, 96, 105, 116, 118,
 153, 160, 162-163, 166
MASQUE
 agent roles, 70, 76, 90, 103, 108-109,
 119, 135, 139, 142, 148
 agent tasks, 73, 81-82, 84, 86, 107,
 109, 113-114, 120, 123, 126, 128,
 136, 143, 146-147
 Domain Agents, 61, 63-66
 domain consultants, 64, 94
 domain managers, 61, 64-70, 73, 90,
 162
 Interface Agents, 60-61, 71
 knowledge component, 36, 55-56, 59,
 64, 66-69, 161-162
 large-task agents, 125-126, 128, 130,
 133, 135-136, 139-140, 142, 144-
 145, 147, 153
 phase guide, 61
 process guide, 61
 small-task agents, 125-126, 130, 133-
 136, 138-140, 143-145, 147, 150,
 153
 system architecture, 56
 system guide, 60
 Tool Agents, 61-63, 71
 tool managers, 61-63
 tool operators, 63
 micro-simulation, 40, 43-44, 46-47, 50,
 56, 59, 63, 117, 162-164
 modularity, 42, 50, 55, 66, 96, 160
 multi-actors, 4, 47, 161
 multi-agent systems, 40-44
 multi-agents
 applications, 38, 41-42
 concepts, 9, 46-47, 55-56, 66, 117,
 159, 162, 164-165
 organization, 41, 48-49
 negotiation, 4, 7, 30, 40, 47
 plan animation, 112-115
 plan evolution, 118
 plan outcome, 118
 plan performance, 118
 plan program, 58, 61, 65, 67-68, 70-73,
 90, 106, 117, 124, 128, 152, 162-164,
 166
 plan program scenarios, 58, 72
 plan proposals, 64, 67, 70, 73, 90, 162
 plan verification, 70, 89, 93

planning
 Choice phase, 3, 47, 57, 66, 70, 96
 communication, 17-18
 decentralization, 4, 27
 decision-making quality, 6, 29
 Design phase, 3, 47, 57, 65, 96
 goals and objectives, 3-4, 16, 19, 27, 57-58, 65-67, 72-73
 Intelligence phase, 3, 57, 65
 multidiscipline, 4, 9, 19-20, 31-32, 35, 45, 56, 63, 66, 69, 96, 159-160
 plan development process, 2, 4-5, 7, 23, 27, 49, 57, 60, 63-66, 68-70, 81, 131, 140, 151, 161
 problem definition, 3, 7, 19, 27, 30, 35, 162
 public participation, 4, 14, 17, 23, 27, 122
 Planning Support Systems
 definition, 15
 framework, 7-9, 33-35, 55, 159, 161-162
 general architecture, 56
 information component, 6, 21, 26, 34, 159
 models component, 6-7, 24, 26, 34-35, 56, 62-63, 159-161
 principal requirements, 19, 46, 96
 user groups, 14-16, 27, 33, 45, 60
 user modes, 17-18, 34
 visualization component, 6, 21, 26-27, 34-35, 56, 159-161
 population, 75, 111-112, 121-122, 136, 138
 problem-solving, 6-7, 16, 40-42, 48, 51
 project input file, 72, 106, 111, 163
 relative suitability, 73, 162
 resolution, 22, 75, 101, 107, 112, 117-118, 121, 124, 128-132, 136, 152, 164-165
 scenarios, 25, 27-28, 47, 59, 61
 software design, 42, 44, 46, 161
 software engineering, 37, 55, 165
 soil condition, 71, 77
 Spatial Decision Support Systems, 14, 29, 31
 specialized knowledge, 7, 9, 17, 30, 35, 42, 56, 63-64, 70, 73, 96
 strategic actions, 76, 84
 system
 adaptability, 34, 68, 160, 164
 deployment, 6, 36, 45
 extensibility, 34, 68, 160, 164
 flexibility, 7-8, 15, 20, 25, 33-34, 44, 68, 160, 164
 functionality, 8, 14, 16, 19, 20-22, 25, 28, 30, 33-34, 45, 48, 56, 58-59, 61, 68, 99, 106, 164
 intelligence, 7-8, 17, 30-32, 34-35, 45, 50, 55-56, 68, 159-160, 164
 understandability, 7-9, 20, 38, 55, 59, 116, 145, 160, 165
 uncertainty, 4-5, 7, 27, 73, 75, 78, 91, 122, 162
 urban environment, 1-2, 5-6, 46, 48-50, 65
 urban planning team
 team, 49-50, 64, 66, 68-70, 96, 116, 118
 members, 64, 67, 102
 urban systems, 4, 26-27, 37, 48, 96, 161
 usability, 7, 16, 18, 21, 26, 34, 55, 59, 68, 116, 160-161, 164-166

user

acceptance, 6-7

assistance, 34, 36, 42, 44, 56, 59, 160

guidance, 34, 36, 42, 44, 56, 59-61,
160

user interface, 17-18, 33-35, 42, 46, 55-
56, 60, 105

viewpoint approach, 70, 122, 152, 162

virtual reality, 28

SAMENVATTING

(DUTCH SUMMARY)

In een reactie op de aanhoudende behoefte aan meer geavanceerde en meer intelligente beslissingsondersteuning in de ruimtelijke planning, heeft dit onderzoek zich ten doel gesteld de mogelijkheden te verkennen van een uitgebreidere toepassing van multi-agent technologie in het kader van Planning Support Systemen (PSS). Hiertoe is een studie in drie stappen uitgevoerd. Ten eerste is een inventarisatie verricht van PSS ontwikkelingen tegen de achtergrond van het veronderstelde ideaalbeeld, aangevuld met een verkenning van de mogelijkheden die multi-agent technologie biedt in dit perspectief. Vervolgens zijn de conceptuele principes ontwikkeld voor het structureren van een PSS raamwerk op basis van een concept dat de planningsondersteunende potenties van multi-agent technologie volledig benut. Dit is verder uitgewerkt met de specificatie van een multi-agent model dat gericht is op een van de kerntaken in ruimtelijke planning: het genereren van alternatieve plannen. Tenslotte is een prototype applicatie ontwikkeld op grond van dit model en is een reeks van gevoeligheidsanalyses uitgevoerd ter illustratie van de veelzijdigheid van multi-agent concepten met betrekking tot de ondersteuning van ruimtelijke planning.

DEEL I - INVENTARISATIE

Het ideaalbeeld van PSS wordt normaliter voorgesteld als een systeem bestaande uit drie componenten: een *informatie* component die GIS functionaliteit biedt, een *modellen* component die doelgerichte modellen en procedures in het systeem onderbrengt, en een *visualisatie* component die inzicht kan bieden in de gevolgen van beslissingen. Het bewerkstelligen van een allesomvattend, geïntegreerd systeem is echter een zware opgave wanneer men zich het sterk lokatiespecifieke karakter van ruimtelijke planning realiseert, alsmede de diverse en continue veranderende vraag naar gegevens en de evolutie van beslissingsondersteunende instrumenten in zowel technologische als methodologische zin. Benodigd is een raamwerk dat behalve allesomvattendheid ook flexibiliteit van systemen toelaat in termen van uitbreidbaarheid en aanpasbaarheid.

In het laatste decennium is vast komen te staan dat het enige passende antwoord op

deze vraag ligt in de ontwikkeling van intelligente oplossingen, in het bijzonder voor het regelen van informatie- en controlestromen, het gebruik van onzekere of vage gegevens, en kennisvergaring en -beheer. Tezamen met de in de praktijk zichtbare matigende rol van GIS, lijkt de koers gezet in de richting van een PSS raamwerk waarin de verzameling van instrumenten (of *modellen*) de centrale component vormt die de benodigde kennis in zich huisvest, terwijl de informatie en visualisatie componenten vanuit de achtergrond een ondersteunende rol spelen. Uit deze voorstelling van zaken volgen evenwel een aantal punten van zorg. Door het onderbrengen van kennis in de mechanismen van de *modellen* component wordt aan flexibiliteit ingeboet ten aanzien van het onttrekken, bijwerken en aanvullen van kennis. Ontrafeling van kennis en modellen in twee aparte maar onderling verbonden componenten lijkt een duidelijke verbetering. Daarnaast wordt ook de begrijpelijkheid voor gebruikers, en dus de bruikbaarheid van een systeem, op het spel gezet wanneer de begeleiding en assistentie van gebruikers te wensen overlaat.

Multi-agent technologie biedt veelbelovende concepten en technieken die het de potentie geven om PSS te verbeteren wat betreft zowel gebruik als functionaliteit. Het antropomorfische karakter van de technologie biedt de mogelijkheid om PSS vorm te geven als personele organisaties, waarbij werkelijke personen gerepresenteerd worden door softwarematige equivalenten ('agents'). Voor planmakers – gewend om te werken met teams van specialisten in een veelheid van disciplines die allen unieke bijdragen leveren aan het proces – is dit een hoogst herkenbare abstractie in tegenstelling tot de meer mechanische abstracties die ten grondslag liggen aan traditionele systemen. Complexiteit en verscheidenheid worden bewerkstelligd door modulariteit, dat wil zeggen dat aan iedere agent specifieke probleemoplossende kennis en vaardigheden worden toegekend, waarbij de interactie tussen agents synergie teweegbrengt met betrekking tot het probleemoplossend vermogen van het totale systeem. De kansen om multi-agent technologie toe te passen in PSS ontwikkelingen liggen op drie vlakken. Ten eerste biedt het de middelen om de gebruikersondersteuning binnen een systeem te verbeteren door agents aan te stellen voor het uitvoeren van gebruikerstaken en het bieden van begeleiding bij het doorlopen van procedures en processen. Ten tweede biedt het de mogelijkheid om individuen, hun gedrag en hun interacties in directe zin te representeren, wat niet alleen de vraag naar disaggregate modellen ter beschrijving van stedelijke systemen kan beantwoorden, maar ook een oplossing aandraagt voor het expliciet onderbrengen van de kennis van verschillende specialisten in het systeem. Ten derde kan het het doel van software-ontwerp dienen met betrekking tot het koppelen van

componenten (bestaand of nieuw), het organisatorische structureren van systemen en het creëren van lerende en zich aan gebruikers aanpassende systemen.

DEEL II - PRINCIPES

Om een voorstelling te geven van hoe deze geïdentificeerde kansen gecombineerd dienen te worden binnen de context van PSS, is een conceptueel raamwerk geconstrueerd voor een multi-agent planning support systeem, genaamd *MASQUE*, dat zich richt op de ondersteuning van lokale planvorming. Ten opzichte van de traditionele PSS architectuur zijn daarbij twee cruciale veranderingen naar voren gebracht die de bruikbaarheid verbeteren en de functionaliteit versterken. Ten eerste is een reorganisatie van componenten doorgevoerd, inhoudende (i) een samensmelting van de *informatie* en *visualisatie* componenten tot een geïntegreerde GIS/VR component met een dienende rol, (ii) een centrale positionering van de *modellen* component tezamen met een nieuw toe te voegen *kennis* component, en (iii) de inpassing van verschillende typen van agents om het samenspel van componenten alsmede de interactie tussen gebruiker en systeem te dienen. Ten tweede is hierbij het belang benadrukt van het vormgeven aan PSS op basis van het planvormingsproces, waarbij de organisatie van agents zorgdraagt voor de koppeling van systeem en proces.

Het voorgestelde PSS raamwerk vereist een organisatie van agents bestaande uit drie typen. Ten eerste is een groep van ‘interface agents’ benodigd voor het monitoren van gebruikersgedrag ten aanzien van systeem structuur, functionaliteit en het onderliggende planvormingsproces, teneinde te assisteren bij optredende problemen door middel van het geven van gerichte suggesties, waarschuwingen, ideeën, enzovoort. Ten tweede dient een groep van ‘tool agents’ te worden ondergebracht in de *modellen* component, die – georganiseerd volgens de fasen in het planvormingsproces – gebruikers kunnen assisteren met het organiseren van instrumenten (d.w.z., het toevoegen, construeren, aanpassen en evalueren van instrumenten) als wel het selecteren en gebruiken van specifieke instrumenten. Ten derde is een groep van ‘domain agents’ vereist die de verschillende grondgebruik-specialisten betrokken in de planvorming representeren en, vanuit de *kennis* component opererend, assistentie kunnen verlenen gedurende het gehele proces, van probleemdefinitie tot het genereren en evalueren van alternatieven.

Gegeven de bepleite toevoeging van een *kennis* component aan het conceptuele PSS raamwerk, alsmede de gelijktijdige noodzaak om de mogelijkheden aan te tonen van

andere multi-agent toepassingen dan micro-simulatie, is de focus gericht op de werking van agents in de *kennis* component. Een model is gespecificeerd dat beschrijft hoe deze agents – genaamd *domain managers* – gezamenlijk sets van alternatieve plannen genereren in een entropy-minimaliserend zoekproces gericht op een stapgewijze opmaak van planvoorstellen vanuit verschillende oogpunten. Laatstgenoemde duidt erop dat – op basis van een gebiedsbeschrijving ('inventarisatie') en een lijst van grondgebruikseisen ('plan programma') – elke agent (i) de ontwikkeling van een planvoorstel initieert en leidt, en op verzoek (ii) deelneemt in de planvoorstellen van anderen. Onder de aanname van een rasterweergave, drukken agents op verzoek van een initiator claims uit voor cellen (d.w.z., stukken grond), en wel door op grond van beschikbare informatie de *relatieve geschiktheid* van beschikbare cellen vast te stellen in termen van *verwacht nut*. Initiators verzamelen alle claims, verwerken deze in toewijzingen ('assignments') en stellen deze toewijzingen weer voor aan alle deelnemende agents.

Omdat grondgebruiken veelal onderling samenhangen, zijn agents afhankelijk van elkaars claims. Doordat claims gelijktijdig gemaakt worden, vindt de besluitvorming van agents plaats onder onzekere omstandigheden. Op een probabilistische wijze, handelen zij zodoende op grond van hun verwachtingen met betrekking tot welke claims het meest aantrekkelijk voor hen zouden zijn ('goal state beliefs'), gegeven hun verwachtingen met betrekking tot de meest waarschijnlijke uitkomst van het planvoorstel op grond van alle op een bepaald moment bekende gegevens ('future state beliefs'). Daar komt bij dat de noodzaak tot het balanceren van persoonlijke en collectieve belangen (d.w.z., een gezamenlijke oplossing) het formuleren van claims (d.w.z., een balancering van verwachtingen) en toewijzingen (d.w.z., een balancering van claims) maakt tot strategische handelingen. Het gedrag van agents met betrekking tot het maken van claims en toewijzingen is gevat in kleine sets van parameters, terwijl hun besluitvorming verondersteld is plaats te vinden op basis van individuele beslismodellen – bestaande uit regels in de vorm van nutsfuncties – waarmee de relatieve geschiktheid van cellen middels een gewogen meting beoordeeld wordt op grond van zowel fysieke als ruimtelijke kenmerken.

DEEL III - ILLUSTRATIES

Op basis van het gespecificeerde model is een prototype applicatie ontwikkeld voor het genereren van alternatieve plannen: de *MASQUE* Alternative Plan Generator. Daartoe is

eerst gekeken naar de mogelijkheid om gebruik te maken van bestaande instrumenten voor het ontwikkelen van multi-agent applicaties. Gezien de duidelijk verschillen in speerpunten is hun bruikbaarheid en toepasbaarheid echter zeer beperkt te noemen. Aldus is besloten de applicatie losstaand hiervan te ontwikkelen, en wel in de object-geörienteerde programmeertaal Java binnen de omgeving van Borland® JBuilder™.

Het voor de applicatie gekozen structureel ontwerp bestaat uit drie samenhangende onderdelen: de omgeving waarmee de agents te maken hebben, de team organisatie die zij vormen, en de mentale concepten die zij gebruiken. De implementatie heeft zich ten doel gesteld om een team van vier agents – specialisten op het terrein van wonen, werken, groen en voorzieningen – alternatieve plannen te laten genereren voor raster-gebaseerde omgevingen waarin een studiegebied en een omliggende gebied (zijnde een configuratie van één of meer van dezelfde grondgebruiken) onderscheiden worden. De resulterende applicatie vereist slechts beperkte input in de vorm van een ‘project input file’ waarin de benodigde gebiedsbeschrijving (d.w.z., afmetingen en bestaand grondgebruik) en het plan programma (d.w.z., de vraag per grondgebruik) worden gegeven; standaardinstellingen voor zowel gedragsparameters als beslisregels worden automatisch geladen. De instellingen en uitvoer van het model zijn georganiseerd in het input/output kader van de applicatie, dat bestaat uit drie tabbladen. Het eerste tabblad (‘Behaviour’) bevat de velden waarin het individuele gedrag van agents ingesteld kan worden. Het tweede tabblad (‘Decision models’) stelt gebruikers in staat om voor elke agent het beslismodel te formuleren op basis waarvan een agent de relatieve geschiktheid van cellen vaststelt. Na activering van het model, worden de gegenereerde plannen afgebeeld in het derde tabblad (‘Results’) samen met de door agents gemeten prestaties per plan. Animaties bieden de mogelijkheid om de stapsgewijze opbouw van plannen te bestuderen en te vergelijken.

De demonstratie van het ontwikkelde multi-agent model is de vorm gegeven van een serie van gevoeligheidsanalyses, inhoudende een bestudering van de effecten van het systematisch veranderen van gedragsparameters and beslisregels. De bevindingen hebben zich veelbelovend getoond in de zin dat het mechanisme van het model in staat is gebleken om op effectieve wijze het proces te laten resulteren in sets van alternatieven die zowel onderscheidend als bruikbaar te noemen zijn. Bovendien hebben de uitgevoerde tests laten zien dat het model goed reageert op veranderingen in omstandigheden (d.w.z., gebiedsomschrijving en plan programma) gezien het feit dat de effecten duidelijk en verklaarbaar bleken. Een minpunt was wel de robuustheid ten aanzien van veranderingen in resolutie, die minder bleek dan wenselijk geacht. Daar staat weer tegenover dat alle

geteste typen van beslisregels zich effectief hebben getoond, logischerwijs in wisselende mate. Ook de veranderingen in het gedrag van agents – ten aanzien van zowel het claimen en het toewijzen van cellen – hadden zichtbare en veelal verklaarbare effecten. Op basis van de gevonden verschillen in (de mate van) effecten kan geconcludeerd worden dat de gedragsparameters van het model in complementaire zin werken zoals was voorzien.

DISCUSSIE

De studie zoals gerapporteerd in dit proefschrift heeft de toepassingen van multi-agent technologie geïdentificeerd waarvan verwacht mag worden dat deze de ontwikkeling van meer intelligente en flexibele PSS – in termen van werking, functionaliteit, bruikbaarheid, uitbreidbaarheid en aanpasbaarheid – mogelijk maken. De principes voor een volledige benutting van de geboden kansen zijn samengebracht in een conceptueel PSS raamwerk, geformuleerd in termen van componenten en agent typologie. Gezien de complexiteit waarmee planmakers in werkelijkheid te maken hebben en de daaruit op te maken noodzaak van intelligente beslissingondersteuning, kan dit onderzoek beschouwd worden als een doelgerichte en betekenisvolle onderneming.

Huidige PSS ontwikkelingen laten zien dat de toepassing van multi-agent concepten beperkt blijft tot de ontwikkeling van agent-gebaseerde micro-simulatie instrumenten voor het evalueren van alternatieve plannen. In deze studie is gesteld dat, op geheel complementaire wijze, multi-agent concepten ook gebruikt kunnen worden voor het genereren van alternatieve plannen, wat kracht is bijgezet met de ontwikkeling van een prototype applicatie. De uitgevoerde serie van tests heeft inzicht geboden in de details van het gebruik van multi-agents voor deze toepassing en, daardoor, in de verschillen met de conventionele toepassing. Aldus kan deze studie mede beschouwd worden als zijnde zowel origineel als waardevol.

De ontwikkelde applicatie geeft gebruikers de mogelijkheid om op snelle wijze compacte sets van alternatieve plannen te ontwikkelen om de oplossingsruimte van een project te verkennen, de kansen en bedreigingen in kaart te brengen, de afhankelijkheden tussen verschillende grondgebruiken (dan wel agents) te visualiseren, enzovoort. Als zodanig stelt het hen in staat informatie te verkrijgen die van waarde is bij het zoeken naar goede alternatieven. Echter, een aantal punten moet hierbij in gedachten worden gehouden die voortkomen uit het feit, dat de huidige implementatie bovenal gedreven is door de noodzaak om de veelzijdigheid van multi-agent concepten te zichtbaar te maken.

Dientengevolge geeft de applicatie gebruikers toegang tot alle gedragsparameters en beslisregels. Hoewel dit een voorwaarde is voor het testen van het onderliggende model, is het niet optimaal vanuit het oogpunt van eindgebruikers. Daarnaast heeft het tot gevolg gehad dat de ontwikkeling en het testen van de applicatie gericht is geweest op het bestuderen van de mechanismen van het model teneinde de conceptuele verschillen met de gebruikelijke toepassing van multi-agents in ruimtelijke planning aan het licht te brengen, en niet op validatie en het afleveren van een kant-en-klaar instrument. Hiertoe zullen aanvullende inspanningen vereist zijn.

In algehele conclusie resteert de vraag of de PSS wetenschappers en ontwikkelaars zich wensen te conformeren naar een algemeen raamwerk zoals naar voren gebracht in deze studie, ongeacht de vraag of de voordelen voor zich spreken. Redelijkerwijs kan het lokatiespecifieke karakter van ruimtelijke planvorming gezien worden als een factor die conflicteert met het idee van een algemeen toepasbaar raamwerk en, zodoende, slechts de ontwikkeling van opzichzelfstaande applicaties toestaat. Dit als een voldongen feit beschouwen is evenwel niet in lijn met de vooruitgang die geboekt wordt op het gebied van component-gebaseerde software ontwikkeling en de rol die multi-agent technologie hierin kan spelen. Deze studie heeft een poging gedaan om dit in gedachten te brengen en, zodoende, de deur te openen naar de ontwikkeling van PSS die ideaal zijn vanuit het oogpunt van zowel ontwikkelaars als gebruikers.

MOGELIJKHEDEN VOOR VERDER ONDERZOEK

Er zijn verschillende aspecten die in aanmerking komen voor verder onderzoek. Zo ligt verdere verbetering van het ontwikkelde prototype voor de hand, met name wat betreft de aangetroffen gevoeligheid voor resolutie. Hoewel de resultaten wellicht al verbeteren wanneer het model in een serie van opeenvolgende runs wordt toegepast waarbij de resolutie stapsgewijs toeneemt, is nadere studie vereist om na te gaan in hoeverre de gevoeligheid een gevolg is van het toegepaste interactie protocol en of verbetering mogelijk is door aanpassing of vervanging van het protocol. In laatstgenoemde geval zal een onderzoek naar werkwijzen van planmakers een goed startpunt vormen.

Hoewel zo veel mogelijk afgeleid van planologische kengetallen zijn de door agents gebruikte beslisregels vrij eenvoudig van aard. Om de agents qua kennis een betere afspiegeling te laten zijn van de personen die zij representeren is het wenselijk dergelijke personen te raadplegen om hun kennis in regels te kunnen vatten. Daarnaast is er ook een

behoefte om meer agents toe te voegen, waarbij een agent ten aanzien van verkeer nog de meeste inspanning zal eisen. Als het team van agents eenmaal is uitgebreid en meer verfijnde kennis is ingebracht, zou validatie van het model uitgevoerd kunnen worden door een groep van werkelijke planmakers in een ‘role-playing game’ te laten deelnemen dat gespiegeld is aan het multi-agent model.

Er is aanvullend onderzoek nodig om inzicht te krijgen in de interactieve effecten van gedragsparameters. Dit is mede gerelateerd aan de behoefte om de juiste instellingen per agent te bepalen in een gegeven project, wat idealiter een taak van agents zelf zou moeten zijn in plaats van een gebruikerstaak. Om dit te realiseren zullen agents uitgerust moeten worden met leervaardigheden die, gegrondvest op ‘case-based reasoning’ technieken, hen in staat stellen het succes van hun strategisch gedrag te interpreteren in termen van de kenmerken van een project (d.w.z., omgevingskenmerken en plan programma). Omdat het leren van agents door middel van ‘case-based reasoning’ ook van belang is voor het voeden, herzien en uitbreiden van hun beslismodellen, vormt het een fundamenteel onderwerp voor verdere studie.

Tenslotte zullen bij het ontwikkelen van operationele systemen op basis van het voorgestelde multi-agent PSS raamwerk gedetailleerde ontwerp- en implementatievragen naar voren komen die momenteel moeilijk te voorzien zijn. Echter, de nieuwigheid van de aangedragen typen agents ligt niet in hun werking, kennis, vaardigheden of interactie, maar slechts in hun toepassing in de context van PSS. Zodoende zullen de in andere onderzoeksvelden gerapporteerde ervaringen met multi-agent applicaties een voorname bron vormen om op te bouwen. Desondanks kan en mag het optreden van problemen die voortkomen uit de toepassing in ruimtelijke planning niet bij voorbaat worden uitgesloten. Als dergelijke toepassingsgerichte problemen daadwerkelijk gaan optreden, vormen zij als vanzelfsprekend onderwerpen voor verder onderzoek.

CURRICULUM VITAE

Dick Saarloos was born in 1972 in Rotterdam, The Netherlands. After secondary school, he studied Building Engineering at the Polytechnic Faculty of Rotterdam (HR&O), specializing in the field of urban design. He graduated in 1996 after a final project in which he made a design proposal for a large-scale city development project in the region of Rotterdam.

After this, he entered the Faculty of Architecture, Building and Planning at Eindhoven University of Technology and followed a Master's program focussed on urban planning and decision support. In 1999 he graduated with distinction after a final project in which he developed a GIS application for supporting retail site selection decisions within downtown shopping areas based on pedestrian distributions. His stay at this university was prolonged when he was appointed as a Ph.D. candidate at the Urban Planning Group.

In 2006 he has been awarded a Postdoctoral Fellowship for Foreign Researchers by the Japan Society for the Promotion of Science (JSPS) to conduct research at the Transportation Engineering Laboratory of Hiroshima University, Japan. The focus of this research is on pedestrian-oriented agent-based modelling as a decision support instrument in CBD regeneration.