Bond University

DOCTORAL THESIS

Case Based Reasoning in E-Commerce.

Sun, Zhaohao

Award date:
2003

Case Based Reasoning in E-Commerce

Ph.D. Thesis

by

Zhaohao Sun

School of Information Technology

Bond University

E-mail: zhaohao_sun@staff.bond.edu.au

A dissertation submitted

in fulfilment of the requirements for the Degree of

Doctor of Philosophy

at

School of Information Technology

Bond University

Gold Coast, Qld 4229

Australia

Supervisor: Associate Professor Gavin Finnie

December 2002

Certification of Thesis

This thesis is submitted to Bond University in fulfilment of the requirements for the Degree of Doctor of Philosophy.

This thesis represents my own work and contains no material which has been previously submitted for a degree or diploma at this University or any other institution, except where due acknowledgement is made.

Signature: ____________________________________ Date:______________________

Abstract

The revolution of the Internet and the WWW are changing traditional commercial activities such as shopping, brokerage, negotiation and retailing. Intelligent techniques for e-commerce have drawn increasing attention since the end of the last century. However, how to make e-commerce intelligent and the customers more satisfied remains a big issue. Applying intelligent agents or multiagent systems and CBR in e-commerce has been among the most rapidly growing areas of research and development in information technology in the last few years. CBR potentially has a large role to play in facilitating e-commerce, because it is experience-based reasoning, which plays an important role in business. However, applying CBR in multiagent e-commerce is still in its infancy, although there are some studies on CBR in multiagent negotiation and auction. There are also no systematic studies on integration of CBR, MAS and e-commerce from both a mathematical and a logical as well as an information technology viewpoint. There are few studies on applying CBR in multiagent brokerage.

This thesis will fill this gap by examining intelligent techniques such as case-based reasoning (CBR) and their applications in e-commerce, and providing a unified treatment of integrating CBR, MAS and e-commerce.

The philosophy of the thesis is that just as human agents play an important role in traditional commerce using their intelligence, intelligent agents will play the same role in e-commerce through their possessing intelligent techniques. In order to realize this philosophy, this thesis will make some contributions to CBR, intelligent agents and MAS, and e-commerce. Three of them will be briefly mentioned as follows, while the rest will be mentioned in the concluding remarks of each chapter in this thesis. The first contribution of this thesis is to provide a general theory of CBR based on similarity-based reasoning, in which the thesis introduces a new theory of similarity metrics, a novel process model for CBR (the R^5 model), examines abductive CBR and deductive CBR, and develops algorithms of rule-based and fuzzy rule-based case retrieval. It also shows that CBR is a process reasoning, in which a traditional reasoning paradigm plays a pivotal role in each stage of the process.

The second contribution is to develop efficient intelligent techniques and methodologies for multiagent e-commerce, in which the thesis provides deeper insight into multiagent e-commerce by classifying it into three categories: multiagent auction systems, multiagent mediation systems, and multiagent brokerage systems.

The final contribution of this thesis is to develop the unified methods, models and architectures for multiagent e-commerce, in particular for multiagent brokerage, and then integrate CBR, multiagent systems, and e-commerce in CMB, which is a CBR system for multiagent brokerage.

In order to make the above mentioned contributions, the thesis is undertaken at three different levels: a theoretical level, a technological level, and an implementation level.

Key words: e-commerce, intelligent agent, multiagent system (MAS), case-based reasoning (CBR), brokerage, bargaining, broker, Internet, World Wide Web(WWW), auction, knowledge based systems, e-business.

Dedication

This thesis is dedicated to

my wife, Yanxia Huo

for her constant support, tolerance and unselfish love,

and

my son, Lizhe Sun

you are a wonderful and smart young man, full of intelligence, promise and enthusiasm.

Mom and I are very proud of you,

and

my parents, brothers and sisters

for their love, patience and support,

and

My motherland, China,

my beloved countries, Germany and Australia

Acknowledgements

As with all dissertations, many people helped make this project possible, both through their endless support, help and courage. I apologize in advance to those who are overlooked here. For example, over the years, I have benefited from the information of the e-mail lists about e-commerce, intelligent agents, artificial intelligence, soft computing, and CBR on the Internet, which helped me to know the current issues and trends in the related fields. Therefore I would, first of all, like to thank all e-friends encountered in the Internet for their papers and ideas available in the Internet. I would then like to thank all the authors of the included articles cited in this thesis (including other publications of mine). This thesis could have not existed without their efforts.

I would also like to thank all of my teachers from year 1 to PhD research for their love, education, and heuristics.

I would still like to thank all of my friends, colleagues, and fellow students accompanying me to play, learn, study, and work in this world for their various sympathy, understanding, cooperation, and support.

I would like to thank to the School of Information Technology, Bond University for launching an innovative PhD-program, and the Australian Government for its OPRS (IPRS) scholarship. I could never have finished this thesis without the financial support of the School of Information Technology and the mentioned scholarship.

Toward the PhD research I like to be most indebted to the following important persons:

- Professor Li, Zhishan, former President of Hebei university in China, who gave me a firm logical and mathematical foundation and consistent support
- Professor Li, Xingwen, former Dean and Vice President of Hebei University, who guided my interest of transferring from Mathematics to Computer Science and Information Technology. He also provided constant encouragement and support, and guided me to teach subjects such as Artificial Intelligence, Database Systems, Software Engineering, Knowledge Engineering, Formal Language and Automata, Logic for Computer Science, and Discrete Mathematics
- Professor Li, Jianqiang, current Party Secretary of Hebei Normal University, one of my closest friends, who gave me endless support from philosophy, politics and individual life

- Professor Li, Tianzhu, former Dean of Faculty of Computer Science, Hebei University. I have benefited from discussions with him about Relational Database, Logical Database, Object-oriented Database, Object-oriented Programming, and Artificial Intelligence over many years
- Professor Zimmermann, H.J., who told me what a greatest scientist in the world is and made me get a lot of research experience in Soft Computing, Intelligent Systems, and Operations Research
- Professor Cromme, L., who gave me a lot of chances of involvement in industrial projects which included software development and process optimization in industry
- Herr Dipl.-Math. Klaus Weber for his insightful comments and suggestions at various stages of the thesis's development
- Professor Neville de Mestre and Mrs Margaret de Mestre for their understanding and support.

I would especially like to thank my supervisor, Associate Professor Gavin Finnie, a model of intellectual brilliance, humility, and cooperation as well as personal kindness. His endless encouragement, patience, cooperation, and support led me to effective work and productive publications.

Furthermore, I want to express my deep appreciation for the unfailing support and encouragement of my wife and son, Huo, Yanxia and Sun, Lizhe. They demonstrated remarkable and unforgettable patience and tolerance as I left them for Germany and then for Australia.

The Chinese government, German government, and Australian government are gratefully acknowledged for their sequential support during my realizing this goal.

Special thanks to the referees of this thesis.

Last, but definitely not least, my thanks, gratitude and apologies to all those who have helped me in their own unique way that I have failed to mention here.

Zhaohao Sun

Gold Coast, Qld

Australia

December 2002

Permission

This dissertation entitled:

Case-based Reasoning in E-Commerce

written by Mr. Zhaohao Sun

has been approved for the School of Information Technology

Chair of Committee

Committee Member

Date

This final copy of this dissertation has been examined by the signators, and we find that both the content and form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Chapter Contents

Contents

List of Figures

1 Introduction

With the dramatic development of the Internet and the World Wide Web (WWW), e-commerce has drawn increasing attention since the end of last century. Intelligent techniques for e-commerce have also become a significant research point in the fields such as information technology and information systems. This thesis will examine intelligent techniques such as Case-Based Reasoning and their applications in electronic-commerce or e-commerce.

1.1 Foundation of Thesis

The revolution of the Internet and the WWW has been changing traditional commercial activities such as shopping, brokerage, negotiation and retailing into e-commerce activities such as e-shopping, e-brokerage, e-negotiation and e-retailing since the end of last century [294].

E-commerce is a virtual market, where secure electronic transactions take place. Therefore, service/resource consumers and service/resource providers meet in this virtual space and do business activities. These service/resource consumers and providers can be end users, ordinary software programs, or intelligent agents. Customers can purchase a large selection of merchandise items from an ever-increasing number of Internet stores [179].

E-commerce has become more and more important in industrial applications and research. Basically speaking, there are two forms of e-commerce applications [53]: ones that simply put existing products and means of selling online, and others that create new ways of selling online using intelligent techniques. The first category is a natural mapping from traditional commerce, while the latter can be considered as an intelligent transformation from traditional commerce to intelligent e-commerce, which involves the birth of new business processes made possible by the Internet and new technology to make it successful [294]. Applying intelligent techniques, most of them from Artificial Intelligence (AI), in e-commerce belongs to the latter category [292]. The intense competition among Internet-based businesses to acquire new customers and retain the existing ones has made intelligent techniques an indispensable part of e-commerce [294].

There are many intelligent techniques that have been applied in e-commerce. These include expert systems (ESs) or knowledge-based systems (KBSs), case-based reasoning (CBR), intelligent agents and multiagent systems (MASs), data mining and knowledge discovery, as well as fuzzy logic, to name a few [294]. Much research and development has successfully been done to apply them to e-commerce such as customer service systems and virtual personalities [292]. This thesis will investigate the techniques (mainly) from AI that can be used in e-commerce projects, for example, CBR, KBSs, Intelligent Agents and MASs. Among them the thesis emphasizes the application of CBR and MASs to e-commerce. Why does this research bring CBR and e-commerce together? The idea is to make the present and future e-commerce more user-friendly, secure, and efficient.

CBR is an intelligent systems method that enables information managers to increase efficiency and reduce cost by substantially automating processes such as diagnosis, scheduling, and design. A case-based reasoner works by matching new problems to "cases" from a historical case base and then adapting successful solutions from the past to current situations. For example, CBR systems (CBRSs) have achieved significant practical success in customer support and help desk operations [292]. More recently, product search, recommendation, configuration, and negotiation have been the target of research and commercial activity for applying CBR in e-commerce. However, it seems that these applications and activities are still discrete or isolated. New challenges for the theoretically oriented research as well as for applications of CBR to e-commerce projects require new retrieval, reuse, and adaptation techniques as well as hybrid reasoning. Therefore, one of the goals of this thesis is to examine CBR and its applications in e-commerce.

Agent technologies have recently been applied to e-commerce to improve search effectiveness and reduce transaction costs. Many research studies or commercial projects on multiagent-based e-commerce[1] have been undertaken such as AuctionBot, BargainFinder, and Market Maker [32][104][188][209]. Intelligent agents are rapidly gaining popularity in e-commerce [332], in which agents have been playing the roles of buyers, sellers, intermediaries, and information providers [183]. Therefore, another of the goals of this thesis is to examine MASs and its applications in e-commerce.

Automation of negotiation, which corresponds to negotiation-based e-commerce, has received a great deal of attention from the MAS community, because such endeavours have the important potential for significantly reducing negotiation time and removing some of the reticence of

1. For brevity, "multiagent" stands for "multiagent-based"

humans to engage in negotiation and then facilitating the intelligent negotiation agents that are able to perform negotiation on behalf of users [183]. Auction, mediation, and brokerage can be considered as three concrete forms of negotiation [292]. These attempts correspond to the following three different kinds of e-commerce: auction-based e-commerce, mediation-based e-commerce, and brokerage-based e-commerce [292]. There are a number of studies on multiagent-based auction [352], brokerage [286], negotiation, mediation [105] and the bargaining process [85], although there are few studies that examine their interrelations. For example, Maes et al. have done considerable research on mediation-based e-commerce using MASs [188], which has also been drawn increasing interest in European countries [66][67]. Therefore, the final goal of this thesis is to examine multiagent-based negotiation, auction, mediation, brokerage and their applications in e-commerce.

1.2 Philosophy of PhD Thesis

The Internet and the WWW have been changing our teaching and research, our social life, and also our traditional commerce. E-commerce (doing business and commerce on-line) has been drawing increasing attention since the middle of the 1990s. This thesis will look into intelligent techniques, in particular CBR and MASs in e-commerce. The philosophy of the thesis is that:

Just as human agents play an important role in traditional commerce using their intelligence, intelligent agents will play the same role in e-commerce through their possessing intelligent techniques.

This philosophy is illustrated in Fig. 1.1 In order to realize this philosophy in the thesis, there are a few objectives, which also refine the goals in the previous section, that the thesis attempts to make.

The first objective is to improve the understanding of CBR by providing a general theory of CBR based on similarity-based reasoning, in which the thesis introduces a new theory of similarity metrics, a novel process model for CBR (the R^5 model), taking into account case base building, integration of abductive CBR and deductive CBR, and algorithms of rule-based case retrieval.

The second objective is to integrate CBR and e-commerce through intelligent agents or MASs, in which the thesis provides a deeper insight into multiagent e-commerce by classifying it into three categories: multiagent auction systems, multiagent mediation systems, and multiagent brokerage systems, and examining multiagent techniques for e-commerce. The thesis emphasises the important role of C^3N (cooperation, coordination, communication, and negotiation) in a MAS. This is one of the most significant differences from an expert system (ES), which will be examined in Chapter 4.

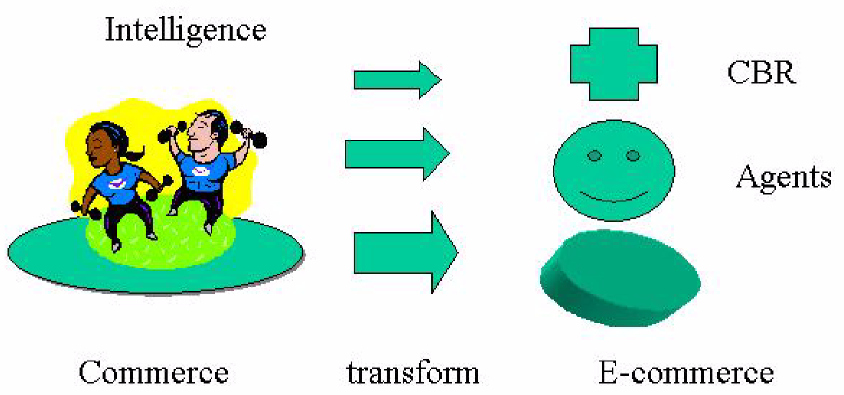


Fig. 1.1 Philosophy of the PhD thesis

The final objective of this thesis is to develop a general theory of multiagent-based electronic bargaining processes based on an agent chain, and treat CBR, MAS and e-commerce in a logical way. It then integrates them under one roof, CMB, which is a CBR system for multiagent brokerage.

Based on this consideration, the thesis will mainly contribute

- To improve the understanding of intelligent techniques for e-commerce and intelligent e-commerce such as multiagent e-commerce
- To develop formal methods of CBR which will move CBR towards a theoretical foundation such as case base building with similarity relations, new models for the CBR cycle, similarity-based case-based recommendation, and case-based adaptation
- To develop efficient intelligent techniques and methodology for multiagent e-commerce; and
- To propose the unified methods, models and architectures for multiagent e-commerce in particular for multiagent brokerage
- Based on these, an architecture for CMB, which is an e-brokerage system integrating CBR and MASs will then be developed to efficiently perform brokerage online.

1.3 Methodology of the Thesis

In order to realize the above mentioned philosophy and objectives, the thesis will be undertaken at three different levels: a theoretical level, a technological level, and an implementation level.

The first level means that some aspects such as CBR, compromise, and bargaining in this thesis will be treated from a mathematical and logical viewpoint.

The second level means that some other aspects such as MASs, architecture of bargaining, and brokerage will be treated using the logical tools of information technology such as flowcharts and knowledge-based techniques.

The third level means that some aspects such as CMB in the thesis will be implemented using J++ or Visual Basic (VB).

It should be noted that it is obvious that this research differs in flavour from the majority of scientific papers. It presents no new theorems in some chapters, has no experimental results in other chapters, and does not describe a novel application in still other chapters. Rather, it represents a rational or qualitative analysis and logical investigation of an important and fast growing area of information technology such as intelligent techniques in e-commerce, like Jennings did in [140]. This analysis and investigation are based on the above-mentioned three different levels, and will try to treat every point at each of these mentioned three levels as much as possible.

1.4 Outline of the Thesis

The general structure of this thesis is following the Boolean algebra. In such a way, the thesis will first examine CBR, e-commerce, and MASs as independent chapters (or as atoms in the Boolean algebra). Then it examines their interrelations, and finally integrates them all under one roof. Based on the above idea, this thesis is divided into three parts including 7 chapters, besides this chapter and other two chapters, as shown in Fig. 1.2:

- Part I. Fundamentals of the thesis, which includes three chapters that are at the second level in Fig. 1.2: case-based reasoning (Chapter 2), e-commerce (Chapter 3), intelligent agents and multiagent systems (Chapter 4)

- Part II. Interrelations of above mentioned three chapters, which includes three chapters that are at the third level in Fig. 1.2: the relationship between CBR and e-commerce (Chapter 6), the relationship between CBR and MASs (Chapter 7), and the relationship between MASs and e-commerce (Chapter 8)
- Part III. Integration of three mentioned aspects under one roof; that is, Chapter 9, which is at fourth level.

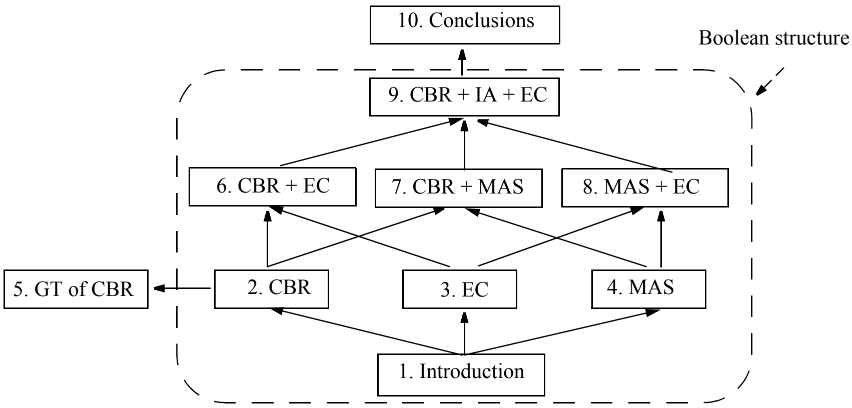


Fig. 1.2 Basic structure of PhD-thesis

The exceptions to the Boolean structure of the thesis are Chapter 5 and Chapter 10 as shown in Fig. 1.2. Chapter 5 develops a general theory of CBR and moves CBR towards a firm theoretical foundation. Chapter 10 is concluding remarks. In what follows, some more details will be provided for each of these three parts, which constitutes the basic contents in each of chapters.

1.4.1 Fundamentals: Part I

Chapter 2, 3, and 4 cover the fundamentals of the thesis. Chapter 2 begins with knowledge and experience for CBR. This chapter will review the fundamentals of CBR, such as case representation, case retrieval, and case adaptation. It examines the relationship of rule-based ESs (RBESs) and CBRSs. It also investigates the relationship of CBR, traditional reasoning, and fuzzy reasoning. It shows that CBR is a process reasoning, in which a traditional reasoning paradigm plays a pivotal role in each stage of the process. Finally it proposes a theoretical foundation for case adaptation and a cyclic case adaptation model, which is a kind of similarity-based reasoning.

Chapter 3 examines e-commerce. This chapter first explores the evolution from traditional commerce to e-commerce. Then it examines three important chains for both traditional commerce and e-commerce; that is, the value chain, supply chain, and agent chain as well as their relationships. This chapter also discusses transaction-based e-commerce; that is, business-to-business (B2B) e-commerce, business-to-consumer (B2C) e-commerce, and consumer-to-consumer (C2C) e-commerce with models and examples. Like blood flowing in our own body, rich information flows in any commerce and plays an even more important role in e-commerce. How to obtain the right knowledge in the right place at the right time is also a big issue of e-commerce with the more and more heavy information overload in the Internet. Therefore, this chapter finally explores information overload, search, and brokerage.

Chapter 4 investigates intelligent agents and multiagent systems (MASs). This chapter will briefly examine basic features and architectures of intelligent agents, intelligent brokers, and MASs. It also examines the relationship between intelligent agents and ESs as well as MASs, in which knowledge-based models of integrating ESs into MAS are proposed. Then it proposes a multiagent-based architecture for information brokering as an example of architecture of MAS, which can help the customers to access the information on the Internet.

Chapter 5 will provide a general theory of CBR, which is based on similarity or similarity-based reasoning. Similarity is at the heart in CBR, just as relations are at the heart of relational database [308]. The motivation of which is that CBR lacks theoretical foundation or formal methods. To this end, this chapter will first extend the concept of similarity given by Zadeh and examine similarity relations, fuzzy similarity relations, and similarity metrics. Then it proposes a theoretical formalization for building case bases with three novel algorithms. It also proposes a R^5 model for case based reasoning. Furthermore, it examines abductive CBR and deductive CBR and proposes a knowledge-based model for integrating abductive CBR and deductive CBR. Finally it proposes rule-based models for case retrieval based on similarity relations, fuzzy similarity relations, and similarity metrics, and fuzzy rule-based case retrieval based on composite rule of inference of Zadeh [355].

1.4.2 Interrelations: Part II

Chapter 6 through Chapter 8 in the thesis cover interrelations of CBR, e-commerce, and MASs. Chapter 6 will propose a unified architecture for a CBR-based e-commerce system, which covers almost all research and development activities in CBR applications in e-commerce such as intelligent support for e-commerce, product recommendation, product configuration, and product negotiation. It also gives new insight into the traditional CBR cycle through decomposing case adaptation into problem adaptation and solution adaptation and providing three different cycles for the extended CBR systems, which not only improves the understanding of case adaptation in traditional CBR, but also facilitates the refinement of activity of CBR in e-commerce and intelligent support for e-commerce. Then it investigates CBR in intelligent support for e-commerce, product recommendation, product configuration, and product negotiation respectively.

Chapter 7 examines the integration of CBR and multiagent systems (MASs). More specifically, it first examines the relationship between case-based reasoning (CBR) systems and multiagent systems (MASs), and proposes knowledge-based models of multiagent CBR systems from both logical and knowledge-based viewpoints, which is an important generalization of almost all attempts that apply CBR in MASs at a high level. Then this chapter investigates the case base and case retrieval in a distributed setting and examines the integration of case-based reasoning capabilities in a BDI architecture. This chapter also discusses CBR for agent team cooperation. Finall this chapter proposes an agent architecture using CBR to model an agent negotiation strategy.

Chapter 8 first reviews intelligent agents in e-commerce; that is, agent-based e-commerce. Then it examines multiagent negotiation, which is the core of multiagent negotiation-based e-commerce. Further it classifies multiagent-based e-commerce into multiagent-based auction, multiagent-based mediation, and multiagent-based brokerage, and gives a brief survey of related works in each. Then it investigates multiagent brokerage and examines the principles of bargaining and compromise in brokerage, and argues that bargaining and compromise play an important role in negotiation, in particular in brokerage.

Based on the characteristics of buyer agents, seller agents, and brokers, this chapter also proposes an architecture of a multiagent-based intelligent broker for the bargaining process, and argues that such an architecture is an abstraction of human agents and brokers working in bargaining processes of brokerage.

1.4.3 Integration: Part III

Applying CBR and MASs in e-commerce has drawn increasing attention in the CBR community. Chapter 9 focuses on the integration of CBR, MASs, and e-commerce. This chapter first examines parsimony principles of intelligence and artificial redundancy in MASs, which are practical strategies for implementing any MAS as well as intelligent system. Then it presents a framework of CMB, which is a multiagent system integrating case-based reasoning and electronic brokerage. The key ideas behind it are that some agents have CBR ability while other agents have not, based on parsimony of intelligence in MAS. Further it investigates the analysis and design for implementing the proposed CMB.

The last chapter of the thesis, Chapter 10, summarizes the thesis and makes some concluding remarks, in which it gives an overview of key research challenges and also attempts to extrapolate to future work.

Part - I: Fundamentals

Part I is Fundamentals of the thesis, which includes three chapters that are at the second level in the Boolean structure: Chapter 2: case-based reasoning, Chapter 3: e-commerce, Chapter 4: intelligent agents and multiagent systems, and the first exception to the Boolean structure of the thesis: Chapter 5: a general theory of case-based reasoning, as shown in the shaded area of Fig. I

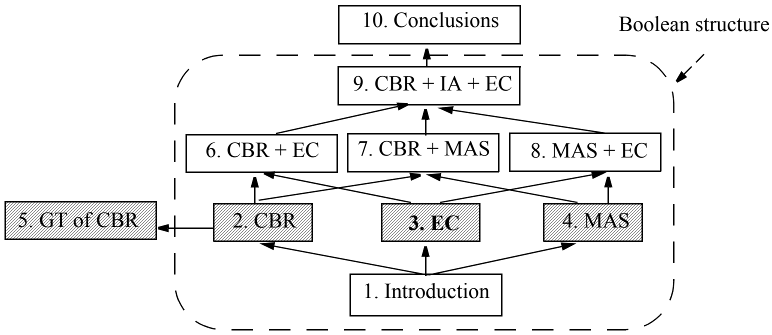


Fig. I. Part I in the Boolean structure of PhD-thesis

2 Case-based Reasoning

This chapter is the first chapter in the Part I of the thesis. It is also the basis for Chapter 5, 6, and 7, as shown in the shaded area of Fig. 2.1. This chapter will review the fundamentals of case-based reasoning (CBR) from a different viewpoint compared to the traditional studies of CBR, and examine the relationship of expert systems (ESs) and CBR systems. Then it reviews process models of CBR, and argues CBR as a process reasoning. It also investigates the relationship of CBR, traditional reasoning, analogical reasoning, and fuzzy reasoning. Finally, it proposes a theoretical foundation for case adaptation and a cyclic case adaptation model.

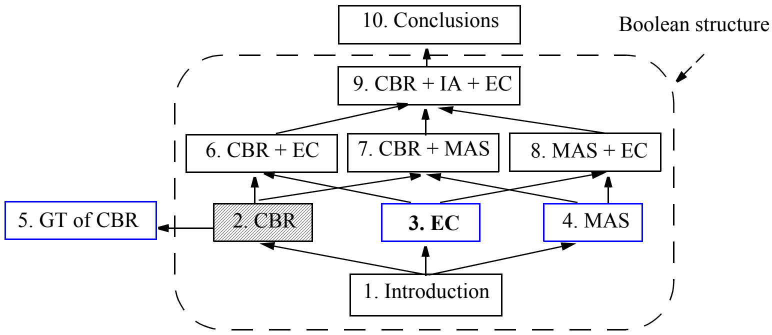


Fig. 2.1 Chapter 2 in the Boolean structure of PhD-thesis

2.1 Introduction

CBR is an approach to problem solving based on the retrieval and adaptation of cases, CBR systems are a particular type of analogical reasoning systems which nowadays have an increasing number of applications in different fields and specialized software products over the two decades [1][72][192][260][261][317]. CBR research has focused on issues such as process models [1], the case-based planning [109], the organisation of case bases [202], the efficient algorithms for case retrieval [13], the assessment of the similarity of cases, and the adaptation of post solutions to the current problem [124]. However, few attempts have been made at moving CBR towards a theoretical foundation, although attempts have been made to develop methodologies for CBR based on soft computing for the last few years [223] (p 241). Moreover, little work has been done yet at providing the existing process models of CBR with theoretical foundation, although they are essentially descriptive. This chapter and Chapter 5 (see later) will attempt to resolve these issues by providing a unified theory of CBR based on similarity-based reasoning. More specifically, this chapter will review and examine the fundamentals of CBR from a different viewpoint compared to the traditional studies of CBR. First of all, it discusses the relationship of knowledge and experience, and examines the relationship of expert systems (ESs) and CBR systems. Then it reviews process models of CBR, and argues CBR as a process reasoning, in which similarity-based reasoning plays an important role in each process phrase. It also investigates the relationship of CBR, traditional reasoning, analogical reasoning, and fuzzy reasoning. Finally, it proposes a theoretical foundation for case adaptation and a cyclic case adaptation model, which is a kind of similarity-based reasoning.

The rest of this chapter is organised as follows: Section 2.1 examines the relationship between knowledge and experience, which are the fundamentals of intelligent systems. Section 2.2 offers a comparative study on expert systems and CBR systems. Section 2.3 reviews process models of CBR. Section 2.4 shows that CBR is a process reasoning paradigm. Section 2.5 and 2.6 investigate case representation, case indexing, and case retrieval. Section 2.7 examines the logical basis of case adaptation and proposes a cyclic case adaptation model. The last section ends this chapter with some concluding remarks.

2.2 Knowledge and Experience

This section examines knowledge and experience, and considers case as a form of representation for experience.

2.2.1 Knowledge

There is no consensus on what knowledge is[1]. Over the millennia, the dominant philosophies of each age have added their own definition of knowledge to the list. Basically speaking, knowledge as a construct or an atom is defined as understanding the cognitive or intelligent system possesses that is used to take effective action to its system goal [329].

- The knowledge is specific to the intelligent or cognitive system that created it, while data is specific to the database system
- The often used definition of knowledge as information made actionable refers to the observable output of knowledge, not knowledge itself. Encyclopedias, handbooks, manuals, other reference material, speeches, lectures, conversations contain only information, not knowledge

1. Wenig, R.G. http://members.aol.com/rgwenig/defknow.htm.

- Understanding written and oral material requires a cognitive system (i.e. a human) to transform the information contained in that material into knowledge.

Knowledge became an important construct in AI in the 1970s. At that time, AI researchers believed that more powerful intelligent systems required much more built-in knowledge about the domain of application [214] (p 10). With expert systems (ESs) becoming an important application of AI in the 1980s, knowledge seems to be a necessary part for any expert system, in which the knowledge base and inference engine are main parts [115].

2.2.2 Experience

It is not easy to define what experience[1] is, just as it is hard to define what knowledge is. Generally speaking, however, experience can be taken as previous knowledge or skill one obtained in everyday life. For example, Peter avoided a traffic tragedy on Gold Coast highway yesterday, because he drove carefully and focused on the drive. This is a typical experience for driving. In other words, experience is previous knowledge which consists of problems one has met and the successful solution to the problems. Therefore, experience can be taken as a specialization of knowledge.

In his theory of recollection, Plato believed more than 2000 years ago that when you think that you are discovering or learning something, you are really just recalling what you already knew in a previous existence [212] (p 301). In other words, experience is an important part for intelligent activities.

In CBR terminology, a *case* usually denotes a *problem situatio*n [1]. A previous experienced situation, which has been captured and learned in a way that it can be reused in the solving of future problems, is referred to as a past case, previous case, stored case, or retained case. Correspondingly, a new case or unsolved case is the description of a new problem to be solved.

According to Kolodner [152], a case is a contextualized piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goal of the reasoner. Following this

1. In Chinese, experience is divided into categories: one is “Jingyan”, another is “Jiaoxun”. The first is previously successful experience, while the latter is previously unsuccessful experience. Sometimes, Jiaoxun is more important for development of one’s intelligence. Jiaoxun has been studied in the context of learning from failure.

direction, a case can be considered as an experience, which is not only a summary of a solution to a previously encountered problem, but also can be reused in the solving of similar problems in the future. In other words, the solved problem in the case is a representative of a class of similar problems, while its solution to this problem is also the representative of a class of similar solutions.

Based on the above discussion, it can be asserted that a case represents specific knowledge tied to a context. It records knowledge at an operational level. Thus, CBR systems (CBRSs) are a kind of knowledge-based system. Further, a case records experiences that are different from what is expected. Thus, a case is an operational definition of experience. Therefore, CBR is a special form of experience-based reasoning, and CBRSs are intelligent systems simulating experience-based reasoning.

It should be noted that cases are not the only type of knowledge that intelligent systems need in order to function. If one builds a full cognitive model, one should include in it both cases and abstractions of those cases. The organisation of abstractions and cases in a case base changes dynamically over time and with experience.

Generally speaking, case-based systems[1] have new experiences each time they are used. Those new experiences can be recorded in their case base so that the case-based systems can evolve into better reasoners over time [152] (p 11). But if one puts every experience in that is different in any small way from those that are already in the case base, it could easily become overwhelmed with all the cases. Thus, from a viewpoint of the cognitive model, it is difficult to predict exactly which cases should be recorded, when, and for what intentions [152] (pp 12-13). According to Kolodner, it is thus necessary to consider whether every experience in which a difference is encountered is worthy of recording in a case base. Our intuition tells us probably not. So, how can one distinguish which experiences with differences are worthy of being remembered as separate cases? The answer to this question is very difficult to formulate. None has tried to do it from a theoretical viewpoint, although Kolodner has given an empirical explanation of it.

1. This research prefers to use CBR system(s) rather than case-based systems, in order to avoid the confusion of CBR and case-based systems in the CBR community.

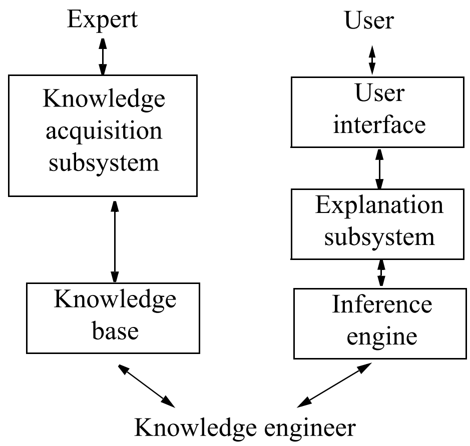


Fig. 2.2 Basic structure of an expert system followed [214]

However, this thesis will examine this problem in more detail in Section 5.3, in which it will propose models for similarity-based case base building.

2.3 From Expert Systems to Case-based Reasoning Systems

This section will examine the relationship between expert systems (ESs) and case based reasoning systems (CBRSs).

2.3.1 Expert Systems

As previously mentioned, expert systems (ESs) used to be one of the success stories of Artificial Intelligence (AI) research in the 1980s-1990s. In what follows, the section will focus on rule-based expert systems (RBESs), which are an important part of ESs. An ES[1] mainly consists of a knowledge base and an inference engine. The knowledge base contains the knowledge used by human experts, in contrast to knowledge gathered from textbooks or nonexperts. The inference engine consists of all the processes that manipulate the knowledge base to reduce information requested by the user- forward or backward chaining, for example [214] (p 282) shown as in Fig. 2.2. Thus, an ES can be formalized as:

$$\text{ES} = \text{Knowledge base} + \text{Inference engine} \tag{1}$$

In RBESs, knowledge is represented as facts about the world (i.e. relationships between entities) and rules for manipulating the facts [317] (p 7). Each rule has a simple form:

$$\text{IF P Then Q} \tag{2}$$

1. Any program that functions as an expert can be called an expert system [214] (p 281)

where P and Q are compound propositions. The mathematical foundation for RBES is mathematical logic: propositional logic and predicate logic, because the reasoning (forward chaining) in RBES is following *modus ponens*:

$$\frac{\begin{array}{l} P \\ P \to Q \end{array}}{\therefore Q} \tag{3}$$

where $P \to Q$ is a logical form of production rule: IF P Then Q. (3) means that if P and $P \to Q$ both are (logically) true, one can logically infer that Q is true applying *modus ponens* [282].

While computerizing experience is still the most important task for research and development of CBR, computerization of knowledge is the most significant contribution of ESs to computer science, which leads to Knowledge Engineering and Knowledge-based Systems [290]. The latter is still a very broad subfield in AI.

The understanding of software has evolved significantly in the past forty years. Before 1968, software was a program [290]. Then the researchers believed that documentation during software development was a necessary part for a successful software development so that software can be formalized:

$$\text{Software} = \text{program} + \text{documentation} \tag{4}$$

In the 1970s' researchers believed that [337]

$$\text{Programs} = \text{data structures} + \text{algorithms} \tag{5}$$

This idea had an important influence on software development as well as teaching in computer science. In the 1980s, researchers found that the interfaces of the programs played an important role in making the customers satisfied. Thus programs can be formalized as:

$$\text{Programs} = \text{interfaces} + \text{data structures} + \text{algorithms} \tag{6}$$

Since the 1990s, ESs have been embedded in much popular software, at the same time, knowledge and information have become an important topic for research and development of Information Technology; for example, knowledge engineering, and so on. Therefore, software can be formalized as:

$$\text{Software} = \text{program} + \text{ES} \tag{7}$$

where the ES is a help assistant or wizard, which is a new form of existing (rule-based) ESs.

However, despite the undoubted success of ESs such as RBESs in many sectors, developers of these systems have met several problems [315]:

- Knowledge acquisition is still a difficult process
- Implementing an expert system is a difficult process requiring special experience and skills and often taking many years
- Once implemented model-based KBS are often slow and are unable to access or manage large volumes of information.

Finally, there is the problem at the heart of RBESs, namely, the knowledge itself [317] (p 10). The rule-based approach assumes that there is a generally accepted body of explicit knowledge that most practitioners in the domain can agree upon. This is often difficult to achieve. Over the last decade, case-based reasoning (CBR) that seems to address the problems identified above has attracted increasing attention [272][273][311][317][331]. The next section will turn to CBR.

2.3.2 Case-based Reasoning: A Brief Introduction

The memory of experience plays an important role in problem solving [260]. For example, an expert encountering a new problem is usually reminded of similar cases experienced in the past, remembering the results of those cases and perhaps the reasoning behind those results [244]. New problems are solved by analogy with old ones and the explanations are often couched in terms of prior experiences. For example, medical expertise and legal education is also case-oriented.

CBR is a reasoning paradigm based on previous experiences or cases; that is, a case-based reasoner solves new problems by adapting solutions that were used to solve old problems [317] (p 15). Therefore, CBR can be considered as a form of reasoning combining deduction and experience-based reasoning, briefly [292]:

$$\text{CBR} = \text{Deduction} + \text{Experience-based reasoning} \tag{8}$$

Further, similarity-based reasoning is a special form of experience-based reasoning, because there is an experience principle in business activities, for example, "Two cars with similar quality features have similar prices". Therefore, Eq.8 can be specialized as reasoning combining deduction and similarity-based reasoning; that is:

$$\text{CBR} = \text{Deduction} + \text{Similarity-based reasoning.} \tag{9}$$

Based on Eq.9, Eq.8 is extended to the following reasoning model, which can be called *generalized modus ponens*:

$$\frac{P', P' \sim P, P \rightarrow Q}{\therefore \;\; Q'} \tag{10}$$

where P, P', Q, and Q' represent compound propositions, $P' \sim P$ means that P and P' are similar. Q and Q' are also similar. This is a logical model of any similarity-based reasoning. This is also a logical foundation for existing CBR systems, in particular for case retrieval.

The CBR approach is based on two tenets about the nature of the world [167] (pp 3-4). The first tenet is that the world is regular: similar problems have similar solutions. Consequently, solutions for similar prior problems are a useful starting point for new problem solving. The second tenet is that the types of problems an agent encounters tend to recur. Consequently, future problems are likely to be similar to current problems. When the two tenets hold, CBR is an effective reasoning strategy.

CBR tasks are often divided into two classes: interpretation and problem-solving. Interpretive CBR uses prior cases as reference points for classifying or characterizing new situations; problem-solving CBR uses prior cases to suggest solutions that might apply to new circumstances [167] (p 7). This thesis basically focuses on the problem- solving CBR.

A CBR system draws its power from a large case base. In order to be successful, CBR systems must answer the following questions:

1. How are cases organized in a case base[1] ?
2. How are relevant cases retrieved from a case base?
3. How can previous cases be adapted to new problems?
4. How are cases originally acquired?

In what follows, each case of these questions will be discussed in some detail.

To 1. To use a case base effectively, it is necessary to have a rich indexing mechanism. Some features are only important in certain contexts. Because important features vary from domain to domain, a CBR system must learn a proper set of indices from experience. Both the inductive and

1. Throughout of this thesis, case base is used instead of memory or library, although the latter also appears in some literature.

explanation-based learning techniques have been used for this task. This problem will be examined further in Section 2.7.

To 2 and 3. The result of the retrieval process is usually one case or a set of cases. The next step is to take the best case and adapt it to the current situation (matching). One method for choosing the best case is the use of preference heuristics [152]. Here are some examples:

- Goal-directed preference- prefer cases that involve the same goal as the current situation
- Salient-feature preference- prefer cases that match the most important features, or those that match the largest number of important features
- Specificity preference- prefer cases that match features exactly over those that match features generally
- Frequency preference- prefer frequently matched cases
- Ease of adaptation preference- prefer cases with features that are easily adapted the new situations.

Since even the best case may not match the current situation exactly, it will usually have to be adapted. At the simplest level, this involves mapping new cases onto old ones. When old cases represent entire problem-solving episodes, adaptation can be quite complex. This question will be discussed further in Section 2.8.

To 4. In fact, most CBR systems draw on a small case base that is entered by hand. The large bodies of on-line texts, such as legal cases, can eventually be transformed into large case bases. Another approach is to bootstrap gradually from rule-based search into CBR. The idea is to start solving problems with a heuristic search engine. Each time a problem is solved, it is automatically stored in a case base. As the case base grows, it becomes possible to solve some new problems by performing CBR. This idea is very similar to some of the learning techniques. This also brings up the issue of whether it is better to store whole cases in the case base or to store smaller bits of control knowledge instead. There are a number of trade-offs involved. Central to CBR is the idea that stored cases can be adapted and modified.

2.3.3 A Comparison of RBESs and CBR Systems

As is known, a RBES breaks a problem down into a set of individual rules that each solves part of the problem [317]. Rules are combined together to solve a whole problem. However, to create these rules by hand, one has to know how to solve the problem, and this task can be extremely complex and time consuming. CBR systems differ basically in that to use them, one does not need to know how to solve a problem, only to recognise if a similar problem was solved in the past. If so, the CBR system can be used instead of a RBES to easily solve this problem. However, if the similar problem was not solved in the past, one has to use a RBES instead of CBR system to try to solve this problem.

Further, although some consider CBR as rule-based reasoning (RBR) with very big rules [167][195][317], there are real differences between CBR systems and RBESs in that:

- Partial matching: in the CBR system many cases can not be matched exactly in all details. Patterns may be used to recognise and store generalizations about cases, but they are not themselves considered to be cases
- Adaptation: If the customer believes that the solution has not completely met his requirements, they may require some features of the solution adjusted. In this case, one has to decide which details to throw away, which to replace, and which to keep in case adaptation. Further, partial matching implies adaptation. If the current cases can’t be matched exactly, case adaptation will be required to resolve discrepancies
- CBR does not require an explicit domain model and so elicitation becomes a task of gathering case histories
- Implementation is reduced to identifying significant features that describe a case, an easier task than creating an explicit model
- By applying database techniques large volumes of information can be managed, and CBR systems can learn by acquiring new knowledge as cases making maintenance easier.

From a viewpoint of system development, the goal of ESs and that of CBR systems are also different. While the goal of ESs is to create a software counterpart of a human expert, although this goal has not been realized, the goal of CBR systems is basically to simulate experiences of human experts. In this sense, a CBR system can be considered as a subsystem of a KBS (including ESs). In fact, a CBR system is also a KBS, because experience is special knowledge or, experience can be represented by knowledge. Therefore, CBR systems and RBESs can be combined in many ways [167] (pp 22-23). Cases may guide interpretation of rules; cases may be used to focus rule-based reasoning (RBR); or the CBR system may be one component in a RBES.

Furthermore, the CBR system can provide an alternative to RBESs, and is especially appropriate when the number of rules needed to capture an expert's knowledge is unmanageable or when the domain theory is too weak or incomplete [28].

Furthermore, traditional RBESs and CBR systems share the common theoretical foundation: mathematical logic; that is, propositional logic and predicate logic, because RBESs and CBR systems basically perform deductive reasoning. More specifically, RBESs perform reasoning based on *modus ponens,* while CBR systems perform reasoning based on *generalized modus ponens* (see Eq.10). Because *generalized modus ponens* is an extended form of *modus ponens,* CBR systems can be considered as a general form of RBES.

It should be noted that CBR also has a close relation with memory-based reasoning (MBR) [277] and analogical reasoning (AR), because MBR is often considered a subtype of CBR which can be viewed as fundamentally analogical [167]. MBR systems solve problems by retrieving stored cases (precedents) as a starting point for new problem-solving [167] (p 13). However, its primary focus is on the retrieval process, and in particular on the use of parallel retrieval schemes to enable retrieval without conventional index selection. Parallel models can lead to very fast retrieval, but also raise new questions to address about the criteria for knowledge access.

CBR might be viewed as a particular form of analogical reasoning (AR) [73]. The latter has been investigated for a long time in AI and the interest in this research has been considerably renewed by the development of CBR [74]. Further, while CBR solves new problems and interprets new situations by applying analogous prior episodes [167] (p 13), research on analogy was originally more concerned with abstract knowledge and structural similarity, while research on CBR is more concerned with forming correspondences between specific episodes based on pragmatic considerations about the usefulness of the result.

However, from the theoretical viewpoint, MBR and AR share the same reasoning paradigm: that is, *generalized modus ponens, or similarity-based reasoning,* although they stem from different real world scenarios. Therefore, the relationship of CBR, MBR, AR, and RBES can be summarized and shown in Fig. 2.3, in which deduction provides the foundation for RBES and similarity-based reasoning, while similarity-based reasoning is the basis for MBR, CBR, and AR as well as fuzzy reasoning (see later). This is also the answer to why similarity and similarity assessment are pivotal in the mentioned fields, in particular in CBR. The thesis will turn back to this issue later.

2.3.4 Summary

This section examined the relationship between RBESs and CBR systems from both a logical and knowledge-based viewpoint. It argued that CBR can be considered as an experience-based reasoning, which is thus a similarity-based reasoning. Because experience is a special expression form of knowledge, CBR is still a form of knowledge-based reasoning. In this sense, CBR systems are also a kind of knowledge-based systems. Further, as knowledge plays an important role in RBES, experience is pivotal in CBR. Computerizing experience is the most important task for research and development of CBR.

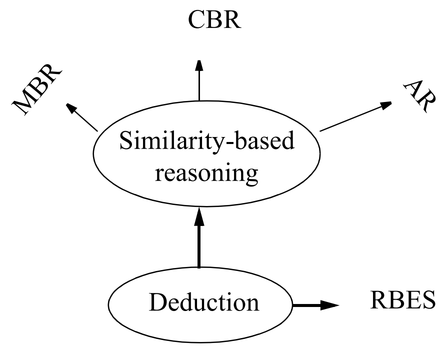


Fig. 2.3 Basic relations of CBR, MBR, AR, and RBES

2.4 Process Models of Case Based Reasoning

A process model is defined in terms of processes, methods, products, goals, and resources [172]. In the last twenty years, there have been a few process models of CBR proposed that attempt to provide better understanding of CBR [88]. This section will review some representatives of these.

2.4.1 Allen's Process Model

Allen [5] has paid attention to the fact that CBR may be considered as a process, because he considers CBR as five-step problem solving process:

- *Presentation*: A description of the current problem is input to the system
- *Retrieval*: The system retrieves the closest-matching cases stored in a case base
- *Adaptation*: The system uses the current problem and closest-matching cases to generate a solution to the current problem[1]
- *Validation*: The solution is validated through feedback from the user or the environment

- *Update*: If appropriate, the validated solution is added to the case base for use in future problem solving.

However, Allen [5] has not paid much attention to the important role that similarity based reasoning plays in the main stages of this process.

2.4.2 Aamodt-Plaza Model- R^4-Model of CBR

At the highest level of generality, Aamodt and Plaza [1] introduced a process model of the CBR cycle. This model is commonly called the R^4 model [89][315], because the process involved in this model can be represented by a schematic cycle comprising the four *R*s, shown in Fig. 2.4.

1. Retrieve the most similar cases
2. Reuse the cases to attempt to solve the problem
3. Revise the proposed solution if necessary; and
4. Retain the new solution as a part of a new case.

A new problem is matched against cases in the case base and one or more similar cases are retrieved. A solution suggested by the matching cases is then reused and tested for success. Unless the retrieved case is a close match the solution will probably have to be revised producing a new case that can be retained. Therefore, a CBR problem-solving cycle consists of case

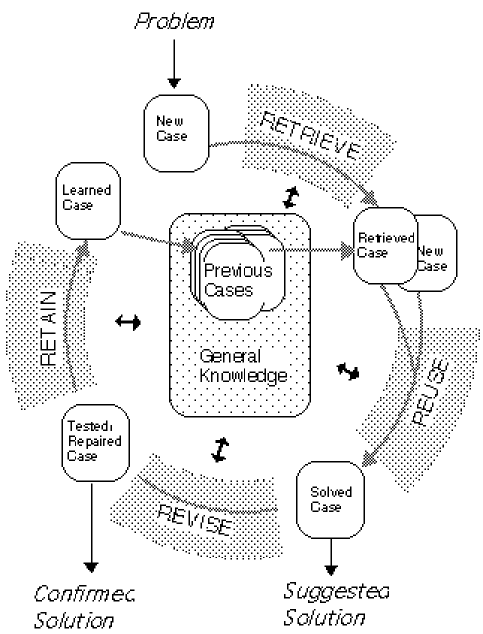


Fig. 2.4 Aamodt's process model after [1]

1. Note that the differences in adaptation power depend on how well the domain is understood [314].

retrieval, adaptation, and case updating [220]. Case retrieval is a process of finding and retrieving a case or a set of cases in the case base that is considered to be similar to the current problem. Case adaptation is a process where the solutions of previous similar cases with successful outcomes are modified to suit the current case, keeping in mind the lessons from previous similar cases with unsuccessful solution. Case updating is a process of revising cases in the case base or insertion of new cases in the case base.

2.4.3 Leake's Model

Leake illustrates the basic solution generation process of case-based problem solving in Fig. 2.5. This is called Leake's model. In this model, when the CBR system gets a new problem from the user interface, it normalizes it into a problem description, p_0, and then retrieves the case base and searches for a prior problem description which is most similar to the current problem description, p_1, that is, $p_1 \approx p_0$. The solution of the retrieved problem description, s_1, is used as the starting point for generating a solution to the new problem s_0.

Based on the above mentioned models such as the Aamodt-Plaza model, the whole CBR process includes the following path: p_0-p_1-s_1-s_0, in which, the transformation from a new problem to p_0 is case representation, the transformation process from p_0 to p_1 is case retrieval, transformation from p_1 to s_1 is case base building[1], and the transformation process from s_1 to s_0

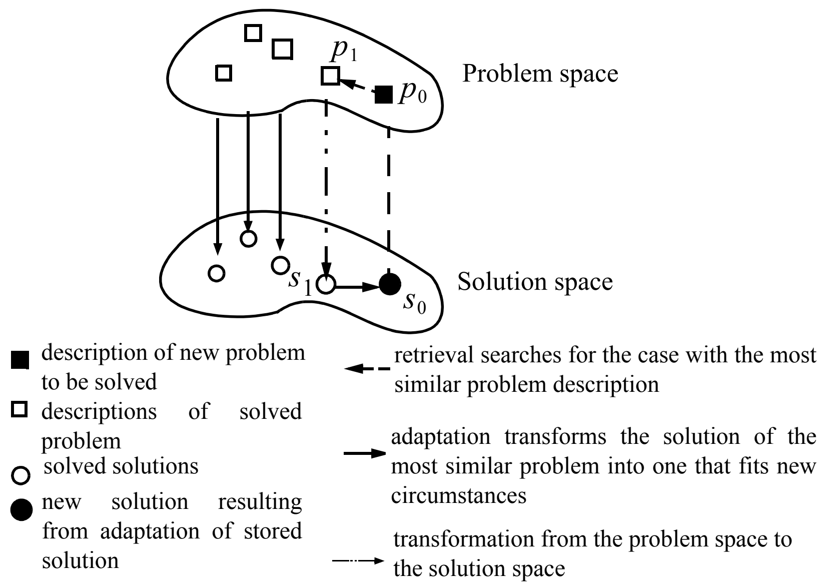


Fig. 2.5 Leake's Model of CBR based on [167]

is case adaptation. In other words, the whole CBR process mainly consists of case representation, case base building, case retrieval, and case adaptation, which will be examined respectively in more detail later.

2.4.4 Further Remarks on the Existing Process Models of CBR

So far, three process models of CBR have been introduced. All mentioned models basically describe the major process stages for performing CBR; that is, case retrieval, case update, and case adaptation. Further, the Aamodt-Plaza model [1] stresses the cyclic feature of CBR in the process model, while Leake's model emphasizes the solution obtained based on two different types of similarity in the problem space and solution space. In comparison to other mentioned models, the R^4 model provides a better understanding of CBR, because it not only covers the essential process description of the CBR cycle, but also provides a nested task decomposition (subprocess description) and related problem solving method descriptions.

However, all mentioned process models of CBR can only be regarded as descriptive process models; that is, every model has covered CBR at a very high or general level. The advantage of descriptive process models (i.e. Allen's model) is that nonprofessionals can understand CBR easily. The disadvantage of it is that the professionals can't easily formulate CBR from a theoretical viewpoint based on the mentioned model. The next section will try to improve the mentioned process model in order to build a firm bridge from a descriptive process model to a theoretical model of CBR.

2.5 Case-based Reasoning: A Process Reasoning

This section shows that CBR is a process reasoning, which differs from not only traditional mathematical reasoning but also fuzzy reasoning or similarity-based reasoning. Furthermore, it proposes a formal approach for CBR as process reasoning based on fuzzy similarity metrics, and treats case adaptation as similarity-based reasoning. Finally, it argues that the proposed model is a more reasonable formalization of CBR than the existing modelling for CBR not only from an implementation-oriented viewpoint but also from a theoretical viewpoint.

1. In an existing CBR system, p_1 and s_1 are stored in the case base as a case. However, from a viewpoint of system development, finding the solution s_1 to the problem p_1 belongs to case base building.

2.5.1 Introduction

As is known, theoretical and empirical studies on CBR have focused on viewing CBR as a traditional logical reasoning or fuzzy reasoning as well as a descriptive process model [1][315], which ignore, in essence, the difference of CBR and traditional logical reasoning [72][231]. In other words, they have not gone beyond either fuzzy reasoning or simple logical reasoning. CBR in their studies is only a scenario for development of their general idea, since their work is still based on one-step reasoning. Furthermore, as previously mentioned in Section 2.4, the proposed process models are basically descriptive. It is difficult for these mentioned models to evolve into a unified theoretical treatment. This section will fill this gap, and it is organized as follows: first of all it discusses the relationship between CBR and other reasoning paradigms, and then examines CBR as a process reasoning, where case adaptation is also considered as a similarity-based reasoning, although it is based on a different similarity assessment; then it proposes a formal approach to CBR as a process reasoning and finally ends this section with a few remarks.

2.5.2 CBR, Traditional Reasoning, and Fuzzy Reasoning

Generally speaking, reasoning is a fundamental task in mathematics and philosophy. Reasoning is also an important method in Artificial Intelligence (AI), in which it is mainly based on the reasoning in mathematical logic such as propositional logic and predicate logic [279][1]. The popular application of reasoning in AI is in expert systems (ESs), in particular rule-based expert systems (RBESs), because RBESs mainly consist of reasoning and knowledge [283][287] as mentioned in Section 2.3. In what follows, the section will examine reasoning in propositional logic, fuzzy logic, CBR, and their characteristics.

In propositional logic, reasoning is performed by a number of inference rules [252][282], in which the most commonly used is *modus ponens* (*m.p.*):

$$\begin{array}{l} P \to Q \\ \underline{P \qquad\quad} \\ \therefore\ Q \end{array} \tag{11}$$

where P and Q represent compound propositions. One of the most important features of this reasoning is that it satisfies the transitive law, and then this reasoning can be performed as many

1. There is no intention of discussing nonmonotonic reasoning in AI in this research.

times or steps as required with the preservation of validity of the result of the inference. This means that the reasoning in traditional logic is a multistep reasoning.

As mentioned in Section 2.3, the reasoning of RBESs is based on (11) [287]. From a logical viewpoint, a RBES is a logical system, which consists of a knowledge base and an inference engine corresponding to the language (knowledge) and inference rule(s) in a traditional logical system [214]. However, the essential difference between a RBES and a traditional logical system lies in that the former possesses much more knowledge than the latter, while the latter is richer in inference rules and can perform a multistep process of reasoning.

Fuzzy reasoning in fuzzy logic is basically generalized from traditional logic with the exception of its computational process [154][182][295][342]. Its reasoning is based on the following *generalized modus ponens* [280][355]:

$$\begin{array}{l} P \rightarrow Q \\ \underline{P'\quad\quad} \\ \therefore\ Q' \end{array} \tag{12}$$

where P and Q represent fuzzy propositions, P' is approximate to P; that is, $P' \sim P$. (12) is also commonly represented in the following form in fuzzy logic [355]:

$$\begin{array}{c} \text{If } x \text{ is } P \text{ Then } y \text{ is } Q \\ \underline{x \text{ is } P' \qquad\qquad\qquad} \\ \therefore\ y \text{ is } Q' \end{array} \tag{13}$$

For instance,

$$\begin{array}{ll} \text{IF a tomato is red} & \text{THEN the tomato is ripe} \\ \quad \text{This tomato is very red} & \\ \hline \quad \text{Conclusion:} & \text{This tomato is very ripe} \end{array} \tag{14}$$

In fact, many other reasoning methods also follow, in some sense, the model (12), for example, CBR [231], analogical reasoning (AR) [167], and similarity based reasoning [72][355], as mentioned in Section 2.3, although they have different interpretations and operational algorithms for performing their own reasoning based on different real world scenarios.

However, fuzzy reasoning does not satisfy the traditional transitivity law (see Section 5.2.3), although it is quasi-transitive (or $\otimes$ transitive [72]). Thus the multiple/sequential use of fuzzy reasoning will lead to *fuzzy degeneration* [87]. Therefore, fuzzy reasoning is basically one-step-reasoning in order to avoid such fuzzy degeneration [87]. Successful applications of fuzzy reasoning do not come from its multistep process of reasoning based on its quasi-transitivity, but from its linguistic computation and more precise understanding of the fuzzy characteristics of certain domain knowledge. Reasoning in CBR does not belong to either form of the above mentioned reasoning paradigms. It can be considered as a new kind of reasoning; that is, a process reasoning. A *process reasoning* is a reasoning paradigm that infers information about a domain using process or multistage methods, and there exists a traditional reasoning paradigm which plays a vital role in every main stage of the process. In what follows, the section will argue that CBR is a process reasoning in more detail.

2.5.3 CBR as Process Reasoning

As mentioned in Section 2.4, a typical reasoning in CBR mainly consists of (case) Retrieve, Reuse, Revise, and Retain. Each of these four components is a complex process. For example, case retrieval is a complex operation in the case base [1]. Furthermore, case retrieval and case adaptation are two main stages in the CBR, in which similarity-based reasoning plays an important role. For instance, case retrieval is based on similarity-based reasoning [72] (also see Section 5.6), case adaptation is also based on it, but maybe on a different similarity-based reasoning (see later). In fact, case base building is also based on similarity-based reasoning (see Section 5.3 and [293]). Thus, CBR is a process reasoning, in which similarity-based reasoning dominates each of main stages; that is, case base building, case retrieval, and case adaptation, as shown in Fig. 2.6.

The rest of this section will examine similarity-based case adaptation in some detail. To this end, an e-sales process is used as a scenario.

During a possible adaptation of the retrieved products (synonymous to product configuration), the retrieved products are tailored based on similarity-based reasoning to best fit the customers' demands, if necessary [335] (p 103). The reason for performing similarity-based reasoning is that the sales agent usually looks at the difference between the retrieved products and the demands of the customer, and then examines what prior case adaptation experience is useful to this difference. Then s/he will deal with this case adaptation based on this found similar experience case in the past; that is, based on similarity-based reasoning. However, the similarity-based reasoning is different from that used in case retrieval, because the similarity here is defined on the *case adaptation base*, in which the meta-knowledge and strategies are from treating case adaptation in the past sale business, while the similarity metric used in case retrieval is relevant only to the case base. Therefore, case adaptation is another process with similarity-based reasoning.

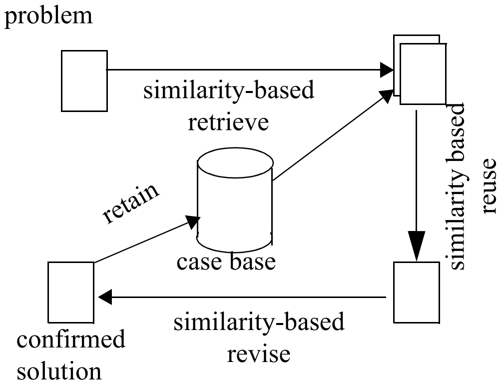


Fig. 2.6 Overall inference of CBR as a process reasoning

It should be noted that similarity-based reasoning shares the common essence with fuzzy reasoning, as mentioned above. Thus it can only be a one-step-reasoning if it is considered in a fuzzy setting. This means that CBR is also not a multistep reasoning in the terms of process reasoning.

2.5.4 Summary

This section examined fuzzy reasoning, case-based reasoning, analogical reasoning and their relationships. It argued that CBR is a process reasoning, different from those other mentioned reasoning paradigms and the process models mentioned in Section 2.4. The feature of CBR as a process reasoning is that case base building, case retrieval, and case adaptation are all based on similarity-based reasoning.

2.6 Case Representation

The most important element in a CBR system is the case base itself [126][205][354]. This is a repository of past problem solutions and is the basis for the whole reasoning process. As such, the way in which a case is represented is critically important for CBR, just as knowledge representation is in KBSs [41]. For example, a good case representation will allow the important features of the problem to be identified and reasoned about. It will also promote the effective and efficient search of the case base.

The case representation process is one of the most fundamental phases in designing a CBR system [112][258]. The case representation should contain all information that describes a situation that has a direct impact on the outcome or the solution of that situation [194]. Depending on the complexity of the situation, cases can be represented in a flat form or a complex or hierarchical form or in other structured manners. What is at issue are methodologies of representing cases in a manner that allows for efficient retrieval, easy maintenance, and that provides for transmission over a network [112]. There are a variety of ways of representing the information in the computer using a wide range of representational formalism including frames, semantic nets, rules, and relational database techniques or a combination of different knowledge representations [226]. This section will examine some models of these case representations.

2.6.1 Tuple-based Case Representation

The case in the case base contains the past experience that is the content of the case and the context in which the experience can be used [72][317]. Typically a case comprises a problem description and solution description. The problem description describes the state of the world when the case occurred while the solution description states the derived solution to that problem, and/or the state of the world after the case occurred.

Cases are often represented using flat or structured or nested attribute-value vectors [172] (p 3) or can be given as n-tuples of completely, incompletely or fuzzily described attribute values, this set of attributes being divided in two non-empty disjoint subsets: the subset of problem description attributes and the subset of solution (or outcome) description attributes, denoted by P and Q [1] respectively [72]. Therefore, a *case* will be denoted as:

$$c = (u_1, \ldots, u_k, v_1, \ldots, v_{n-k}) \qquad (15)$$

where $p = (u_1, \ldots, u_k)$, $p \in P$, is a k-tuple standing for a concrete problem description and $s = (v_1, \ldots, v_{n-k})$, $s \in Q$ is a n-k-tuple which represents the corresponding solution description. Assume that a finite set C of known cases is given, called the *case base.* Thus, C can be denoted as $C = (P, Q)$. A current problem description, denoted by p_0, for which the precise

1. Q denotes solution instead of *S* in [72], because *S* will stand for similarity.

values of all attributes belonging to P are given. Then CBR aims at estimating the values s_0 of the attributes in S, for the current problem.

The tuple-based case representation is also called attribute-value representation, according to [172] (p 2). From a theoretical viewpoint, the tuple-based case representation is essentially the same as the relational data model [251][308] (p 85). Therefore the case base can be implemented using relational database technology.

2.6.2 Rule-based Case Representation

According to AI [244], a rule is said to be a structure consisting of some conditions and a conclusion, a numerical weight is, in some cases, associated with a conclusion or rule itself. A rule is usually denoted as IF A THEN B or $A \to B$.

Rules can be considered as one of the best choices as an appropriate or even necessary form for expressing various kinds of knowledge. At least the following reasons support the use of rules [278]:

- Rules are the most common form of knowledge representation
- Rules are precise but its generalized form, fuzzy rules, allow incorporation of certainty assessment, uncertain knowledge, and plausible inference schemes using fuzzy logic [355]
- Rules ensure modularity in representation, making the representation easy to construct and manipulate, and making it easy to incorporate new knowledge and change existing ones
- Representation with rules facilitates explanation and improves human comprehensibility in many other ways
- Rules share the same form of inference rule in logic, which can make one easily believe that the aim of rule representation is knowledge reasoning. In other words, it has become common sense that rules and reasoning are twins in KBSs or in AI
- Other knowledge representation schemes can be transformed into rule-based schemes.

For example, one can transfer relation schemes as well as tuple-based case representation schemes into rules. Let $c = (p, s)$ (see Eq.15) be the tuple-based case representation of a case c in the previous section, then the case c is represented as a rule $p \to s$; that is:

$$\text{IF } u_1, \ldots, u_k \text{ THEN } v_1, \ldots, v_{n-k}. \tag{16}$$

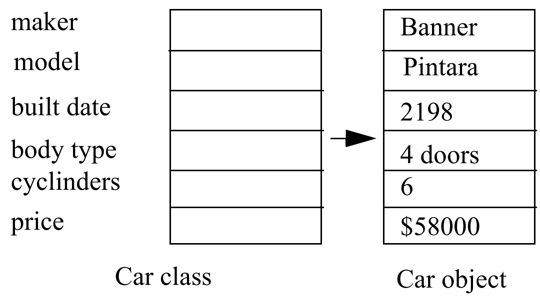


Fig. 2.7 A class representing car

Because rule-based case representation can also be regarded as an alternative of tuple-based case representation, it can thus be implemented using relational database technology, although there are different explanations of cases.

2.6.3 Object-oriented Case Representation

There is an increasing interest in object-oriented case representations [22][24][64][131], because some current state-of-the-art CBR systems are object-oriented, and allow structural domain knowledge to be represented [172] (p 341). For example, Bergmann et al. integrated rule-based general knowledge in CBR with an object-oriented case representation. Such representations are motivated by object-oriented technology [308] (p 27), in which the world to be modelled is thought of as composed of objects. Object-oriented case representations have proven to be flexible and efficient and can be applied to domains in which cases have complex structures [22].

In object-oriented case representations, cases are represented as a collection of objects, each of which is described by a set of attribute-value pairs. The structure of an object is described by an object class[1] that defines the set of attributes (also called slots) together with a type for each attribute. Thus, a case class can have a number of case objects with similar properties. For example, a car class and one of its objects are shown in Fig. 2.7.

It should be noted that “similar properties” of the objects in a class are in two different ways [308] (p 27):

- The real-world concepts represented by the objects of a class should be similar. For example, all customers of a bank can be grouped into one class, and all accounts at the bank into another class

1. Strictly speaking, a class consists of a type and possibly one or more methods that can be executed on objects of that class [308] (p 15).

- The properties of objects in a class must be same.

Classes are arranged in a class hierarchy diagram, which is a tree in which sub-classes inherit attributes as well as their definition from their parent class. For example, a travel agency basically arranges for customers' *vacation*; that is to provide a satisfactory supply to meet the requirement of *transportation* and *accommodation* of customers [22]. One can illustrate *vacation*, *transportation,* and *accommodation* using a class hierarchy diagram, shown in Fig. 2.8. In this diagram, class *transportation* and class *accommodation* are subclasses of the class *vacation*.

2.6.4 Summary

This section mainly discussed three kinds of case representations: tuple-based case representation, rule-based case representation, and object-oriented case representation. The first is motivated from relational database technology [308], the second stemmed from rule-based (expert) systems, and the last is motivated from object-oriented technology and can be used to represent complex knowledge or experience as case. All these are the fundamentals for building a case base in CBR.

2.7 Case Indexing and Case Retrieval

This section will examine case indexing and case retrieval, which are important aspects in CBR, and then propose a rule-based model for case retrieval.

2.7.1 Case Indexing

Using indexes to speed up the retrieval of data is one key technique in most database systems [317] (p 20). CBR also uses indexes to speed up retrieval in case base.

Information within a case in the case base is of two types:

- Indexed information that is used for retrieval

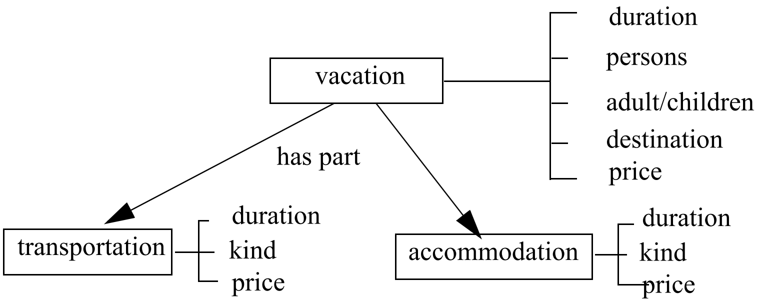


Fig. 2.8 A class hierarchy diagram

- Unindexed information that may provide contextual information of value to the user but not to be used directly in retrieval.

The process of case indexing is one of assigning labels to the case when entered in the case base to ensure its retrieval at the appropriate moment [194].

There are many different techniques available to index cases; such as, choosing indexes using similarity- and explanation-based methods and indexing vocabularies as well as using features and dimensions of cases [194][317]. It also includes techniques like index elaboration, abstraction, mutation, and index transformation or generation so as to provide a different view of the case base, leading the reasoner to previously inaccessible cases.

2.7.2 Case Retrieval

Case retrieval is of primary importance to the overall effectiveness of any CBR system, because [28][221]:

- Retrieving the case ensures the best solution within the system's capability
- Retrieving cases must include some computation of the similarities and difference between the input problem and the retrieved cases. All subsequent case modification uses this computation as a basis.

Case retrieval starts with a (partial) problem description, and ends when a best matching previous case, in which the problem description is most similar to the current problem description, has been found [1]. Its subtasks are referred to as *identify features*, *initially match*, *search*, and *select*, executed in that order. The identification task basically comes up with a set of relevant problem descriptors, the goal of the matching task is to return a set of cases, in which the problem descriptions are sufficiently or most similar to the current problem description- given a similarity threshold of some kind, and the selection task works on this set of cases and chooses the best match.

Given a description of a problem, the retrieval algorithm, using the indexes, should retrieve the cases with the most similar problem(s) to the current problem description [194][317] (p 239). The retrieval algorithm relies heavily on the indexes and the structure and organisation of the case base to direct search to appropriate cases. Heuristic search and matching techniques may be used to retrieve an ordered set of useful cases from the case base. Several retrieval algorithms are now available for case retrieval such as concept refinement and parallel search techniques. The issue of choosing and ranking a best matching case has been addressed using several approaches such as analogy, similarity metrics, combinations of analytical, and qualitative or multi-attribute similarity [172], Case Retrieval Nets [172] (p 79), validated retrieval [223] (p 11), inductive retrieval as well as nearest-neighbour retrieval (NNR), in which the last two are used in commercial CBR tools [317].

Nearest-neighbour retrieval (NNR) is a simple technique that provides an assessment of how similar the problem description attached a case in the case base is to the current problem description, based on the following evaluation function [98][152] (p 355):

$$\sum_{i=1}^{n} w_i \times sim(f_i^I, f_i^R) \qquad (17)$$

where w_i is the importance of dimension (slot) i, and $\sum_{i=1}^{n} w_i = 1$. sim is the similarity function for primitives. f_i^I and f_i^R are the value for feature f_i in the input and retrieved cases, respectively.

Several CBR systems implement versions of the NNR algorithm such as MEDIATOR [152]. In fact, the NNR algorithm is perhaps the most widely used technology in CBR since it is provided by the majority of CBR tools [318].

However, at least the similarity function in the NNR algorithm is still problematic because nobody has thoroughly studied it, although Richter has paid a lot of attention to this issue (see [223]). Further, the NNR has another major weakness, namely, retrieval speed. To find the best matching case, the current problem must be compared to the problem description of each case in the case base [317] (p 32). Moreover, a similarity comparison must be calculated for every indexed attribute. Thus, if there were 100 source cases with a single indexed feature, 100 similarity calculations would be required. If the cases had 10 index features, then 1000 similarity calculations would be required. This means that this algorithm can become inefficient as either the size of a case-base increases or the number of indexed attributes increases.

Validated retrieval, proposed by Simoudis [270], consists of two phases. Firstly, the retrieval of all cases that appear to be relevant to a problem is based on the main features of the query case. The second phase involves deriving more discriminating features from the group of retrieved cases to determine whether they (the cases) are valid in the current situation. The advantage of this method is that inexpensive methods can be used to make the initial retrieval from the case base, while more expensive methods can be used to make the second phase as they are applied to only a subset of this case base [223] (p 11).

2.7.3 Summary

This section examined case indexing and case retrieval, which have been drawn a lot interest not only from CBR but also from database and information retrieval, because they basically share the retrieval techniques. However, almost all retrieval algorithms have a common disadvantage; that is, they confuse similarity and distance. The concept of distance in Euclidean space is precise, while the concept of similarity in the mentioned fields is still confusing. For more detail see Section 5.2.

2.8 Case Adaptation and Case Evaluation

This section reviews case adaptation, and examines case adaptation using similarity- based reasoning. It then proposes a theoretical foundation for case adaptation and a cyclic case adaptation model. It also argues that case adaptation is a process of search and retrieval in the *case adaptation base*.

2.8.1 Overview of Case Adaptation

Generally speaking, "adaptation" denotes all changes of a system so that it becomes suitable for a given situation [106]. Adaptation is also applied to those self-modifications that enable systems to survive in a changed environment. This meaning, however, is too broad to be of value in some cases. It is better to discuss case adaptation in a domain dependent setting.

According to Voss [314], without case adaptation, CBR systems are restricted both in scope and application (also see [147]). To reuse cases effectively in new situations they must be adapted to account for differences between retrieved cases and some new problems. Research in this field has produced a diversity of adaptation techniques to cope with a wide range of tasks, domains, and knowledge sources.

From Section 2.4.3, the case adaptation begins once a matching case is retrieved, as shown in Fig. 2.9. If the retrieved solution is ideal for solving the current problem, then the process of case adaptation needs not start. Otherwise, case adaptation will really start. At this stage, a CBR

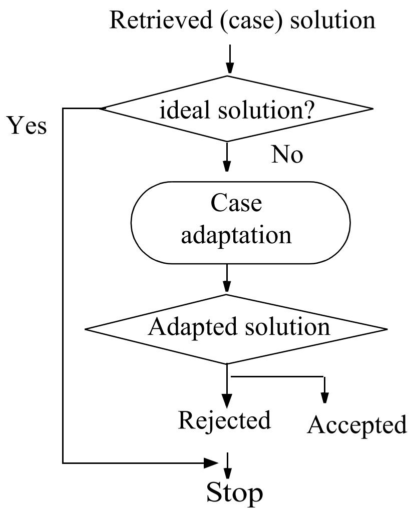


Fig. 2.9 Case adaptation

system should adapt the solution stored in the retrieved case to the current problem [317] (p 34). Adaptation looks for prominent differences between the retrieved problems and the current problem, and then applies formulae or rules that take those differences into account when suggesting a solution.

In general, there are two kinds of adaptation in CBR [317]: Structural adaptation and derivational adaptation. The former applies adaptation rules or formulas directly to the solution stored in the case base, while the latter reuses adaptation rules or formulas that generated the original solution to produce a new solution to the current problem.

Central questions for case adaptation are which aspects of a situation to adapt, which changes are reasonable for adapting them, and how to control the adaptation process [167] (p 23). Answering these questions may require considerable domain knowledge or empirical experience. Many CBR systems depend on that knowledge being encoded *a prior* into rule-based production system. Consequently, correct case adaptation requires that those rules capture both a theory of case adaptation, and the needed aspects of the domain theory to carry out changes. However, in many cases a reliable domain theory is lacking for using CBR. As a result, developers defining adaptation rules must re-confront the knowledge acquisition problem for rule-based systems, which is also a bottleneck and CBR was aimed at avoiding.

2.8.2 A Logical Basis of Case Adaptation

From a theoretical viewpoint, case adaptation stemmed from the similarity-based reasoning in CBR, because the latter can be formalized as the following general model:

$$\frac{p_0, p_0 \sim p_1, p_1 \to s_1, s_1 \sim s_0}{s_0} \tag{18}$$

where, p_0 is the problem description of the customer, $p_0 \sim p_1$ means that p_0 and p_1 are most similar, $p_1 \to s_1$ is the case retrieved from the case base C based on similarity-based retrieval mechanism. $s_1 \sim s_0$ means that s_0 and s_1 are most similar, and s_0 is the most satisfactory solution to the requirement of the customer. It should be noted that in practice s_1 is only the retrieved solution, which is the previously successful solution to problem p_1. It is uncertain that s_1 can be the solution to the current problem of the customer p_0. If it can, case adaptation is not required. Otherwise, the CBR system has to perform case adaptation to find the most similar solution s_0 to meet the requirement of the customer p_0. Therefore, the process of finding s_0 such that $s_1 \sim s_0$ and s_0 is the most satisfactory solution to the requirement of the customer, p_0, is a theoretical foundation of case adaptation.

2.8.3 A Cyclic Case Adaptation Model

The adapted products[1] are offered to the customer and, at the same time, the difference of the offers and demands is explained to the customers. The customer evaluates these offers during the case adaptation phase, which results in a set of evaluated products. The customer can state that he accepts certain products or parts of the products or he may state that something is not appropriate. If the adapted products have been accepted by the customer, the case adaptation ends.

This also means that the adaptation process consists of, at least, search and evaluation, where the case adaptation case is retrieved and the meta-knowledge or strategy will be given to perform the new case adaptation. For example, in the sale process of the traditional shop, if a customer visits an auto dealer, and asks the sales person, if he could buy a car with property X, the sales person will first retrieve all available cars in the dealership, and try to meet the customer's requirements. If not, the sales person uses his available resources (e.g. chain partners) to search for the required car. If a chain partner has a suitable car, he will have a satisfied customer. This step does not occur in the traditional CBR cycle, nor in the model of [335].

1. This research uses the e-sales process as a scenario.

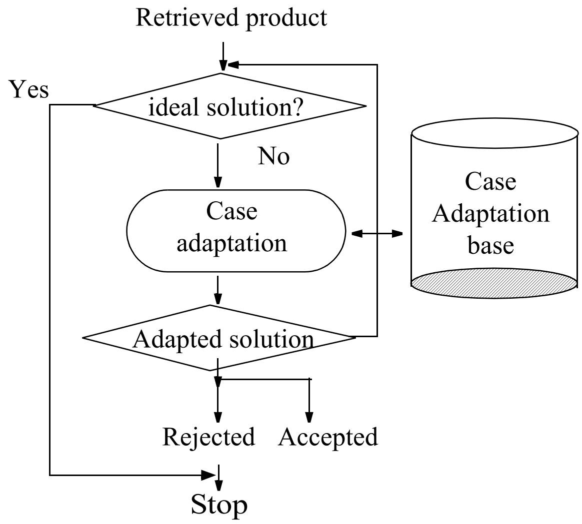


Fig. 2.10 A cyclic case adaptation

In another case, a new step called *refinement* is introduced. This step does not occur in the traditional CBR cycle. By refinement the customer will tune his requirements. Therefore, refinement is, in essence, a kind of adaptation and based on the evaluation (see later) given by the customer. Therefore, the case adaptation can be considered a cyclic process. The successive refinement of customer's requirements may lead to a cyclic sales process, as shown in Fig. 2.10. In general, this cycle is applicable in all situations where an iterative retrieval in a decision support situation takes place.

2.8.4 Summary

This section reviewed case adaptation, and proposed a logical foundation of case adaptation, and a cyclic case adaptation model.

2.9 Concluding Remarks

This chapter reviewed the fundamentals of CBR, such as case representation, case retrieval, and case adaptation. It examined the relationship of RBESs and CBR systems. It also investigated the relationship of CBR, traditional reasoning, and fuzzy reasoning. It showed that CBR is a process reasoning, in which a traditional reasoning paradigm plays a pivotal role in each stage of the process. Finally it proposed a logical foundation for case adaptation and a cyclic case adaptation model, which are based on similarity-based reasoning.

It should be noted that what is discussed in this chapter can only be considered as descriptive CBR or empirical CBR [248]. In the last few years, theoretical CBR has attracted increasing interest. This is an important direction, because further research and development of CBR will confront potential difficulties without a firm theoretical foundation. In this direction, Plaza et al. [231][232] apply model logic and Dubois et al. [72] apply fuzzy logic to CBR. Richter and Bergmann discuss similarity in CBR [223]. Furthermore, soft computing might support CBR towards a firm theoretical foundation [223][342]. However, there are still some important issues about CBR. For example, what is the theoretical foundation of CBR? Can theoretical CBR be treated in a unified way? These issues will be resolved in Chapter 5.

3 E-Commerce

This chapter is the second chapter in the Part I of the thesis. It is also the basis for Chapter 6 and 7, as shown in the shaded area of Fig. 3.1. This chapter first explores the evolution from traditional commerce to e-commerce. Then it examines three important chains for both traditional commerce and e-commerce; that is, value chains, supply chains, and agent chains as well as their relationships. This chapter also discusses transaction-based e-commerce; that is, business-to-business (B2B) e-commerce, business-to-consumer (B2C) e-commerce, and consumer-to-consumer (C2C) e-commerce with models and examples. Like blood flows in our own body, rich information flows in any commerce and plays an even more important role in e-commerce. Finally, this chapter explores information overload, search, and brokerage with models.

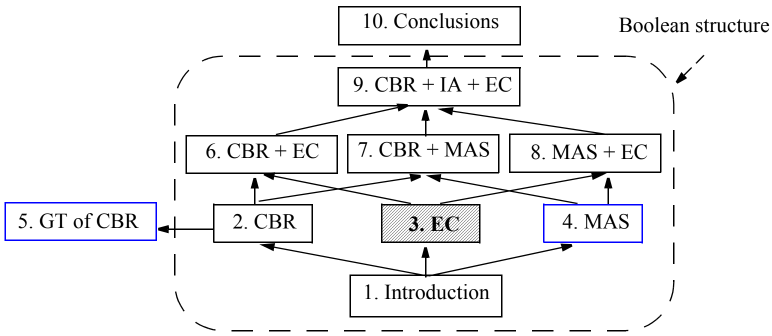


Fig. 3.1 Chapter 3 in the Boolean structure of PhD-thesis

3.1 Introduction

The last decade has seen an explosion in the growth and use of the Internet. Rapidly evolving network and computer technology, coupled with the exponential growth of services and information available on the Internet, is heralding a new era of commerce; that is, e-commerce [155][210][224]. Hundreds of millions of people will soon have pervasive access to a huge amount of information available on the Internet, which leads to important opportunities for e-commerce.

In fact, e-commerce has been around in various forms since the late 1960s [303]. In sectors such as retail and automotive, electronic data interchange (EDI) for application-to-application interaction is being used regularly [300]. For defence and heavy manufacturing, e-commerce lifecycle management concepts have been developed that aim to integrate information across larger parts of the value chain, from design to maintenance, such as computer-assisted lifecycle support. It should be noted that these kinds of e-commerce have been very limited until now. Recently, however, with the increasing popularity of the WWW and explosive growth of the Internet, e-commerce has become an area of growing importance. The Internet not only supports application-to-application e-commerce similar to that already known from EDI, but also person-to-person and person-to-application forms of e-commerce [300] (pp 3-4). E-commerce based on the Internet is set to become a very important way of doing business. More and more organisations are facing the challenge of this new technology [294][344]. However, e-commerce can be considered as a further evolution from traditional commerce, although such an evolution is a little revolutionary. Therefore, this chapter first explores the evolution from traditional commerce to e-commerce. Then it examines value chains, supply chains, and agent chains as well as their relationships. Transactions play an important role in any commerce. Thus, this chapter discusses transaction-based e-commerce; that is, B2B e-commerce, B2C e-commerce and C2C e-commerce with models and examples. How to obtain the right knowledge in the right place at the right time is also a big issue of e-commerce with the increasing heavy information overload in the Internet. Therefore, this chapter finally explores information overload, search, and brokerage with models.

The rest of this chapter is organised as follows: Section 3.2 discusses the relationship between traditional commerce and e-commerce. Section 3.3 examines e-market and its characteristics. Section 3.4 investigates three important chains in e-commerce; that is, value chain, supply chain, and agent chain. Section 3.5 discusses transaction-based e-commerce with models and examples. Section 3.6 investigates information overload and search. Section 3.7 examines information brokerage and proposes a model for information flow in information brokerage. Finally this chapter is ended with a few concluding remarks.

3.2 Traditional Commerce vs E-Commerce

This section mainly investigates the main players of traditional commerce and trade types and then looks at how these elements are transformed into e-commerce.

3.2.1 Traditional Commerce

The origins of traditional commerce occurred before recorded history when our ancestors first decided to specialize their everyday activities such as hunting and farming [263] (p 4). Instead of each family unit having to hunt for meat, grow crops, and make tools, families developed skills in one of these areas and traded some of their products for other needs. For example, the tool-making family would exchange tools for grain from the crop-growing family, which leads to a kind of trade type; that is bartering, which still exists in some underdeveloped places in the world. Services were brought and sold in these primitive economies, too. Later, bartering basically gave way to the use of currency, making transactions easier to settle. The inception of currency makes the trade type more complex and diverse such as brokering, which facilitated the development of modern commerce. However, the basic mechanics of trade is essentially the same; that is, one member of society creates something of value that another member of society desires. At a higher level, the main players in commerce are still buyers, sellers, and brokers [86].

The first player is the buyer. Buyers are customers who purchase certain products or services[1] [178]. A buyer begins by identifying a need [263]. Once buyers have identified their specific needs, they must find products that will meet those needs. Buyers may consult sellers to gather information about specific features and capabilities of products they are considering. After selecting a product, the buyer must select a seller[2] that can supply that product [263]. When the buyer is satisfied that the purchased product or service has met the terms and conditions agreed to by both the buyer and the seller, the buyer will pay for the purchase [263]. After the sale is complete, the buyer may have further contact with the seller regarding warranty claims, upgrades, and regular maintenance.

The second player is the seller. A seller is a product provider [178]. Sellers often undertake market research to identify potential customers needs [263]. Even businesses that have been selling the same product for many years always look for ways to improve and expand their offerings. It is important for sellers to make potential buyers aware that a new product exists and acquire the new buyers [323]. Once customers needs have been identified, sellers will provide the products to the customer, if available. Otherwise, the seller will search for the products that he feels will meet those needs. This search activity includes visiting the production companies. It

1. In fact, a service is also a product, for example, the service of a bank is its main product. Thus, in this research product stands for both product and service, if it does not lead to any misunderstanding.
2. For brevity, seller stands also for vendor.

should be noted that in the history of commerce, the seller used to also be a producer (see Section 3.4.3). The separation between sellers and producers is an important stage of commerce development.

The last and most important player for traditional commerce is the broker. Brokers are intermediaries who help the buyers and sellers to complete a transaction [178][289]. Further, the importance of brokers lies in that they provide a buffer against the interest conflict between the buyer and the seller; that is, the broker can help the buyer and the seller to resolve their conflict. In current terms, the broker can be considered an agent of both a buyer and a seller [85]. However, the broker only exists in certain conditions, for instance, a real estate agent is a broker [84]. One does not find a broker in the supermarket.

These players' activities in traditional commerce form different trade types. Each type has a set of activities to be performed. There are six different trade types [178]: barter, bargaining, bidding, auction, clearing, and contract[1].

1. Barter

Barter is a trade type in which both sides offer their products for an exchange rather than for money [178]. A deal is reached if both sides have a higher preference on what the other is offering than those of their own goods. It should be noted that barter is an early trade style in the history of commerce, although it still exists in current commerce.

2. Direct trade

Direct trade is also a simple trade type in which the buyer negotiates terms with the seller directly until an acceptable deal is reached [178]. Usually, the buyer finds a seller, examines product price or other terms, and negotiates to obtain a better deal. If the deal fails, the buyer finds another seller to negotiate again. Direct trade is also a simple trade style in the history of commerce. This trade style usually occurs in retail shops for end customers.

3. Bidding

Bidding is a trade type that involves a buyer and many potential sellers [178]. The buyer compares the received bids and chooses the best. Bidding is one of most important type for investing in international projects. Further, bidding is also an e-trade type in e-commerce.

---

1. [178] has a different classification, although the rest in this subsection is basically the update of [178].

4. Contract

Contract is a trade type in which both the buyer and the seller are governed by a set of mutually agreed rules [178]. If there is no contract, then both sides need to negotiate for an agreement. If a contract already exists, then ensuring accurate implementation of individual orders under the regulation of the contract becomes the key.

5. Auction

Auction is a trade type that involves a seller (agent), many potential buyers (or buyer agents) [143], and the seller agent who governs the auction. The seller basically doesn't participate in trading or auctioning publicly, and tells the seller agent what the reserved price is, which is a reference price for the bottom (start) price of the product during auctioning. The buyers bid sequentially to compete for the product to be sold.

Auction is still a common trade type in modern commerce. It is usually performed for some special kind of goods such as ancient art works. In e-commerce there is also considerable attention to auction since a lot of e-auction Websites (e.g. eBay) are available in the Internet. Auction in e-commerce will be investigated in Chapter 8.

6. Brokering

Brokering is a trade type involving multiple buyers, multiple sellers, and a broker. A typical example is the real estate agent. Both buyers and sellers submit their requests. The broker tries to match the requests. The main issue in brokering is bargaining [292]. Brokering is attracting more and more attention in information technology and e-commerce world [84]. Brokering in e-commerce will be investigated in Chapter 8.

Several of these may be combined in order to reach a deal. For example, the seller agent and the buyer agent are also involved in direct trade. The above classification in terms of the different players in the trade or transaction can be summarized as follows, where nb and ms stand for n buyers and m sellers, c for condition.

Type 1: barter, 1b to 1s, that is, one buyer exchanges (rather than money) his goods with a seller.

Type 2: direct trade, 1b to 1s

Type 3: bidding, 1b to ms

Type 4: contract, 1b to 1s (c)

Type 5: auction, nb to 1s (c)

Type 6: brokering, nb to ms (c)

For this research, barter, direct trade, and contract will not be discussed any more. The rest of the mentioned trade types share some activities such as bargaining, negotiation, and brokering. All these will be discussed in more detail in Chapter 8.

3.2.2 Definition of E -Commerce

Because e-commerce is attracting more and more people with different backgrounds and needs to share the chance and challenge of e-commerce, there have been many different ways to define e-commerce. For example, Han [110] defines e-commerce as the process of sharing business information, maintaining business relationships, and conducting business transactions by means of telecommunication networks. In this research, e-commerce is defined as the exchange of information, goods, or services within business through the use of Internet technology [263] (p 5), while the new way of doing business via the Internet is defined as e-business (for "electronic" business). The main activities of e-commerce can be formed in a five-layer-structure activity [211], as shown in Fig. 3.2.

EFT and other payment systems are the transactional foundation upon which a great deal of e-commerce follows. There are already a great number of transactions in economy that occur at this level. Key activities at this level include use of Automatic Teller Machines (ATMs), as well as credit card payments, electronic payrolls and may more.

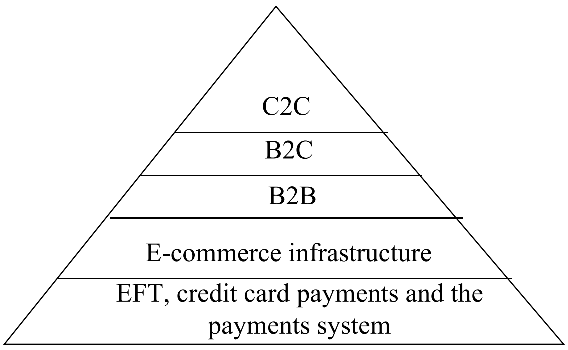


Fig. 3.2 Layers of e-commerce activity after [211]

E-commerce infrastructure includes network service providers, hardware, software and enabling services.

B2B e-commerce represents the major proportion of e-commerce activity.

B2C e-commerce can involve electronic transactions in marketing, ordering and paying, after sales service and, in the case of intangible or virtual goods and services, even delivery.

C2C consists primarily of Websites where consumers deal directly with one another, such as on-line communities and free personal classified pages.

3.2.3 Advantages and Disadvantages of E-commerce

E-commerce is not simply the Web-based automation of existing activities in traditional commerce, but involves the birth of new business processes made possible by the Internet and new technology to make it successful [303]; that is, e-commerce is transforming the way that products, services, and even information are bought, sold, and exchanged. E-commerce also changes the way that organisations interact with customers and business partners. All these lead to that e-commerce has substantial advantages over traditional commerce. However, different players (buyers, sellers, and brokers) in traditional commerce have different viewpoints about advantages over traditional commerce. Sellers believe that e-commerce can increase sales, decrease costs and make customers more satisfied and then can help increase profits. Buyers (customers) think that e-commerce provides them with an increasing bargaining power on the e-market because of rich information available on the Internet. Further, both buyers and sellers share that e-commerce increases the speed and accuracy with which businesses can reduce costs on both sides of transactions. Brokers believe that e-commerce can improve negotiating price and delivery terms, because the Web can provide competitive bid information very effectively, but e-commerce may lead to dis-intermediation.

However, no matter how their relative power changes, players in traditional commerce still exist in e-commerce, but may perform functions differently and in a different role. For example, the players in traditional commerce will be replaced by the intelligent agents in e-commerce; that is, intelligent buyer agents, intelligent seller agents, and intelligent broker agents, which will be discussed in next chapters.

Although e-commerce is attractive and important for future business; its transaction process is often complicated. Involved parties may need to collect and analyse information, negotiate contracts, execute transactions safely, and provide follow-up services over the Internet. Further, some business processes may not lend themselves to e-commerce because of lacking the available technology [263] (p 11). Finally, some consumers are still somewhat fearful of sending their credit card numbers over the Internet [303]. The legal environment in which e-commerce is conducted is full of unclear and conflicting laws. Most of the disadvantages of e-commerce today, however, stem from the newness and rapidly developing pace of the underlying technology. These disadvantages will disappear as e-commerce matures and becomes more available to and accepted by the general population.

3.2.4 Evolution of E-Commerce

Although e-commerce has only developed for about 10 years with the development of the Internet, it has evolved in the following stages[1]:

1. Creating Websites and announcing the information of products
2. Interactive questioning and answering on the Internet
3. Information technology based industrialized e-commerce.

At the first stage, many companies believed that e-commerce is information retrieval on the Internet. At the second stage they believe that interaction between the company and its customers is more important. At the third stage, they have to believe that e-commerce consists of information flow, capital flow, material flow and agent flow[2]. The optimized configuration of these four flows is the necessary condition for successful e-commerce, which will be discussed later in this chapter.

3.3 E-Market and its Characteristics

An e-market is the place for doing e-commerce. The widespread deployment of the Internet has opened up a worldwide market for both sellers and buyers [157]. Buyers can now compare many more suppliers worldwide. Similarly any sellers can now offer goods and services to customers worldwide.

The concept of an e-marketplace extends beyond the Internet, Intranets, and extranets. It denotes an all-encompassing marketplace or network which will be the vehicle for sending,

1. http://www.people.com.cn/GB/it/51/20010507/458683.html
2. Flow can be replaced with chain.

receiving, and using all kinds of digital data, information, and services [119]. Basically speaking, an e-marketplace consists of the host computer and the definition of allowable interactions and communication techniques [332]. The host must provide a set of interaction scenarios, constraints within which the agents must operate, communication language, and a hosting procedure. The e-marketplace may be provided by a retailer or financial institution, an Internet mall, an auction site, or a site provided for buyer and seller agents to meet and negotiate deals, For example, Market Maker is an e-marketplace that provides customers for selling and buying in the Internet [104].

According to Sarker, Butler, and Steinfield [259], there are at least four characteristics in the e-market:

1. The number of organizations involved in a complete seller-buyer exchange will be greater than in a comparable exchange in a traditional market
2. Channel functions related to attracting a community of potential customers will more likely be performed by cybermediaries[1]
3. Customers will interact with a greater number of cybermediaries than similar customers in traditional markets. Furthermore, the number of information channels will be greater in an e-market than in comparable traditional markets
4. The number of sellers using cybermediaries to perform a particular channel service will decline at a slower rate than the number of sellers in a comparable traditional market.

These characteristics of e-markets cause new challenges and criteria for an e-marketplace. The following criteria are essential for an e-marketplace, based on the lessons from many early unsuccessful attempts at e-markets [145] (p 272):

- *Critical mass of buyers and sellers*. The e-marketplace should be the first place customers go to find the products and services they need
- *Opportunity for independent evaluations and for customer dialogue and discussion*. In an e-marketplace, not only do users buy and sell products or services, they also compare notes on who has the best products and whose prices are outrageous. The ability to openly evaluate the wares offered is a fundamental principle of an e-marketplace

1. Cybermediaries are organizations that operate in e-markets to facilitate exchanges between producers and consumers [259]. For example, intelligent agents are cybermediaries.

- *Negotiation and bargaining*. No marketplace is complete if it does not support negotiation. Buyers and sellers need to be able to bargain over conditions of mutual satisfaction, including money, terms and conditions, delivery dates, and evaluation criteria
- *New products and services.* In an e-marketplace, consumers can make requests for products and services not currently offered and have reasonable expectation that someone will turn up with a proposed offering to meet that request
- *Seamless interface*. The biggest barrier to e-trade is having all the pieces work together so that information can flow seamlessly from one source to another
- *Recourse for disgruntled buyers.* An e-marketplace must have a recognized mechanism for resolving disputes among buyers and sellers. Markets typically include a provision for resolving disagreements by returning the product or through arbitrage in other cases.

Finally, it should be noted that e-commerce brings new challenges. For example, in a physical marketplace such as a retail store, a salesperson can use clues such as the customer's questions, dress style, and body language to better assess interests [99] (p 224). However, in an e-marketplace no one sees the customer, and the goal is to let the customer do as much shopping as possible for himself or herself. E-store owners have to do an interesting piece of detective work; that is, based on customer browsing behaviour and purchase history, one should construct a model of who the shopper is, which requires sophisticated data analysis capability.

3.4 Three Chains for E-Commerce

Any business or commerce activity at least includes the dynamic change or management of three chains: a supply chain, a value chain, and an agent chain, as shown in Fig. 3.3. The agent chain can be called a customer chain if everyone can be considered as a customer in business. These three chains are made up of a complex structure in both traditional commerce and e-commerce, and influence the consequence of any business activity in an interdependent way. Further, information flow in these three chains plays a pivotal role for a successful commerce system, just as blood does for a healthy person. This section will discuss these three chains in some depth.

3.4.1 Value Chain

In a market society, any business activities basically aim at the most profit. These business activities can be taken as a sequence of activities that create value for the company. This sequence

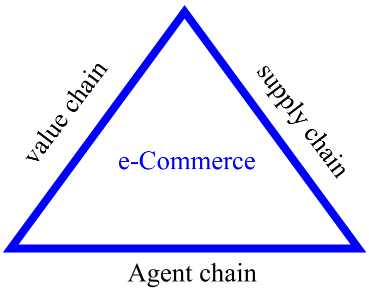


Fig. 3.3 Three important chains in commerce

of activities is essentially viewed as a value chain. Strictly speaking, a value chain is a way of organising the activities that each strategic business unit undertakes to design, produce, promote, market, deliver, and support the products or services it sell, as well as human resources management and purchasing [263] (p 24). The value chain breaks these activities down to strategically relevant categories in order to understand the behaviour of cost and the existing and potential sources of differentiation and then ascertain a company's competitive advantage [56] (p 62). Two strategic categories of the value chain are primary and support activities. Primary activities constitute the physical production of the product, the sale and transfer to the buyer, and post-sales service, etc. [56] (p 62). Supporting activities constitute human resource management, purchasing, etc. in order to help the primary activities.

According to this definition, a model for the value chain is shown in Fig. 3.4. The essence of a value chain is that one stresses that the value dynamically flows up and down in the value chain, and neglects the other kind of flows such as material or information. Further, the linear structure

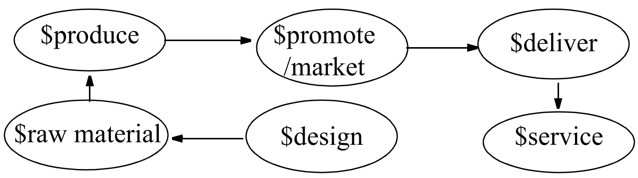


Fig. 3.4 A model of a value chain

of the value chain from $design to $service is only illustrated as an example. In practice, the structure of a value chain for a company is much more complex. However, according to graph theory, given the number of nodes in the value chain, for example, n, then the most complex variant of the value chain is a n-complete graph-based structure [282]. A concrete value chain for a company is a subgraph of a n-complete graph-formed structure[1], as shown in Fig. 3.5.

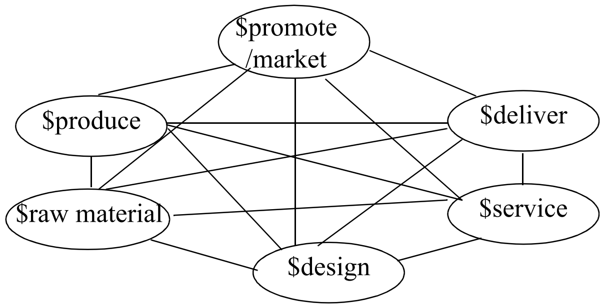


Fig. 3.5 A model of a complete graph-based value chain

3.4.2 Supply Chain (SC)

A supply chain is essentially a business process that links manufacturers, retailers, customers, and suppliers in the form of a "chain" to develop and deliver products as one "virtual" organisation of pooled skills and resources [146] (p 285). The goal is to obtain benefits by streamlining the movement of manufactured goods from the production line into the customer's hands, by providing early notice of demand fluctuations and coordination of business processes across a number of cooperating organisations.

The supply chain can be broken into three parts: an upstream part, internal part and downstream part [305]. The upstream part encompasses all the activities involved in material and service inputs from suppliers, the internal part involving in the manufacturing and packaging of products, and the downstream part involved in the distribution and sale of goods to distributors and customers. A general model of a supply chain is shown in Fig. 3.6.

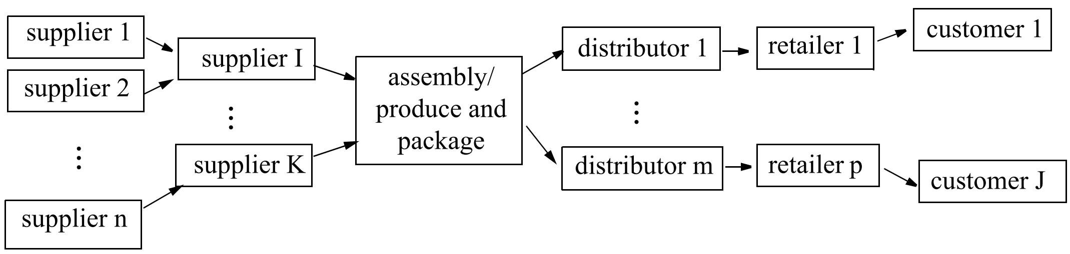


Fig. 3.6 A model of a supply chain

Before the inception of the Internet, many of the processes in the supply chain, especially in upstream and downstream activities, had been managed with paper transactions [305]. Now these transactions are being done digitally using the Internet, because digital information flow makes it possible for a company to create a boundaryless organisation [99] (p 217). The old phrase supply

1. Strictly speaking, it is a weighted complete graph, where weights stand for the value in value chain.

chain implies links in a linear form, looking back from the retailer to distribution to transportation to manufacturing. Today's supply chain is replaced by a "supply network", a web of partnerships enabled by digital information flow. Everyone who touches the product must add value, and communications go both forward as well as back. Companies in the supply network aren't restricted to their places in line by heavy chains of processes but can interact and do business with multiple vendors as they need to. In this way, the supply network integrates the value chain and the supply chain, and extends them into a complete graph-formed value-supply network.

It is noted that e-commerce can be used to link all the members in the supply chain for a seamless flow of goods, services, and information about purchases, payments, delivery schedules, and so on [303] (p 34). SCM (supply chain management) has been a hot buzzword for the past years, but linking many business partners manually has been extremely difficult. Everyone from the raw material supplier to the ultimate seller in the chain must be able to get accurate information quickly and easily about orders, shipping, and customer responses to products and services. Business partners must decide together which areas they will try to link first. Each partner then focuses on key internal groups that have the most to gain by adopting e-commerce.

3.4.3 Agent Chain

Value chains and supply chains have been discussed so far. However, human agents have been an active and decisive factor in traditional commerce. Therefore, only taking into account value chains and supply chains is not sufficient for e-commerce. Agent chains are also important for e-commerce [44]. In some cases, agent chains might decide the dynamic development of value chains and supply chains.

Agents[1] are everywhere in human society. It could be said that one occupation corresponds to one kind of agent [84]. People daily encounter travel agents, seller agents, buyer agents, tax agents, and so on. In business activities there is a special kind of agent, i.e. brokers or merchants. It is these agents who specialize the commerce activity and facilitate the development of modern commerce. With the development of this kind of specialization and information technology, part of or the whole function of some human agents is replaced by intelligent agents (which will be discussed in Chapter 4). However, the history of agents goes back thousands of years [85][125].

1. Agents and brokers mentioned in this section are human agents and human brokers respectively.

At that time, the seller separated from the producer, and the buyer and merchant appeared as a special job i.e.

$$\text{producer} \quad \rightarrow \quad \text{seller} \quad \rightarrow \quad \text{buyer} \tag{1}$$

The seller (merchant) could be considered as the agent of the producer from a producer viewpoint. If a seller wished to sell the goods of the producer to a buyer, he[1] must be also regarded as the agent of the buyer from the viewpoint of the buyer, because the buyer thought that the seller assisted him to buy the goods. So the seller in (1) worked as an agent, an intermediary or advisor, of both producer and buyer in the bargaining process. In other words, the relationship of the agent and broker is

$$\text{producer agent} \quad \equiv \quad \text{broker} \quad \equiv \quad \text{buyer agent} \tag{2}$$

Further separation or refinement or specialization in business activities continued with social development in human history. The seller sells his goods either directly or via a seller agent. The seller agent asks a broker for help to find an appropriate buyer to finish selling the goods. The buyer might also buy the goods not directly but via a buyer agent. In that case, the broker usually stands between seller agent and buyer agent. Therefore, the relationship of agent and broker evolves into

$$\text{seller} \rightarrow \quad \text{seller agent} \rightarrow \quad \text{sell-buy broker} \rightarrow \quad \text{buyer agent} \rightarrow \quad \text{buyer} \tag{3}$$

In fact, in modern business activities, there are many different agents on behalf of sellers. There are also many agents on behalf of buyers. Some of them can function as a broker on some occasions. Therefore, there is an agent chain or broker chain that links the (initial) seller and the (end) buyer:

$$\text{seller} \rightarrow \text{agent}_1 \rightarrow \text{agent}_2 \rightarrow \ldots \rightarrow \text{broker} \rightarrow \ldots \rightarrow \text{agent}_{n-1} \rightarrow \text{agent}_n \rightarrow \text{buyer} \tag{4}$$

(4) can be considered in another way; that is, one can view that the agent_i is always the buyer of the agent_{i-1} and the seller of the agent_{i+1}. Based on this idea, for each $i = 2, \ldots, n-1$, agent_{i-1}, agent_i and agent_{i+1} constitute a subchain of seller-broker-buyer, which is the fundamental part of any commerce. For brevity, let agent_1 be the initial seller, and agent_n the

1. For brevity, he, his or him stands for he/she, his/her, or him/her respectively in this research.

end-buyer, then the agent chain becomes a sequence of subchains of seller-broker-buyer. The transfer of the goods from agent_1 to agent_n changes into a sequential and cyclic movement of such a subchain of seller-broker-buyer, as shown in Fig. 3.7.

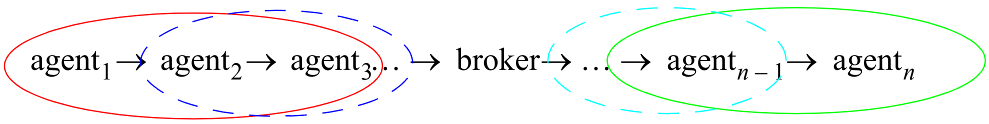


Fig. 3.7 Subchains of seller-broker-buyer in a agent chain

In this way, some agents in the agent-chain sometimes play a tri-role; that is, agent_i will work as either a seller-agent, or a broker, or a buyer agent. Such a tri-role of an agent in the agent chain might facilitate the healthy and effective development of commerce, although business activities have been specialized in the current time.

As mentioned previously, a chain is essentially a linear structure. It is only useful for research or abstraction of the practical case. It is not a real simulation of the real world agent structure actively working in the traditional commerce. The real abstraction is an integration of a linear structure; that is, an agent chain, and a local cyclic linear structure; that is, a seller agent, a broker agent, and a buyer agent. If one also takes into account the complete graph in an agent chain, then the agent chain becomes a more complex structure, called an agent network, as does the generalized structure of value chains and supply chains. However, with the fast development of e-commerce, one has to take into account the generalized structure from a research viewpoint.

In current business activities, how to manage the agent chain and how to optimize the agent chain are always a big issue for business management. This is not a new problem in commerce, because customer relations and human factors are always a big issue for real commerce. However, the agent chain or agent network is a new viewpoint, which will lead to a new challenge for commerce, in particular for automating commerce; that is, agent chain management (ACM) might be an important topic, just as SCM is. Further, based on above discussion, it can be argued that multiagent systems (MASs) are better than a single agent in simulating real business activities with the agent chain or agent network, which will be discussed in Chapter 4. But there are still no available multiagent systems to implement such a model.

It is worth noting that in commerce, a buyer or buyer agent is always taken as a customer. Thus, an agent chain can sometimes be viewed as a customer chain. How to manage the customer chain and how to improve the relationship of customers in the customer chain then become a big issue of customer relationship management (CRM).

So far, this section examined three kinds of chains. The dynamic change of these three chains might lead to the evolution of traditional commerce to e-commerce. The first two have been drawn increasing interest in e-commerce which are fallen in the categories of value chain management (VCM) and supply chain management (SCM). The agent chain can be considered in the ACM, which is an important part in intelligent agents or MASs with further development of MASs in e-commerce. MASs will be discussed in next chapter.

3.5 Transaction-based E-Commerce

Enterprises are increasingly deploying the e-commerce framework in three classes of applications based on the transaction mode: business to business (B2B), business to Consumer (B2C), and Consumer to Consumer (C2C)[1], as mentioned in Section 3.2. The first two of them have been attracted a lot attention in these days and become the primary forms of e-commerce, comparing to the last one. In what follows, these three kinds of transaction-based e-commerce will be discussed with models and examples.

3.5.1 Business to Business (B2B) E-commerce

As its name implies, B2B e-commerce is all about two businesses conducting transactions on the Web [6][299]. For example, Company A purchases product P of Company B using the Web. Company A will negotiate with Company B about this product using the Web. Both companies might use their own agents/intelligent agents during the negotiation. After careful bargaining, they agree on the sale. Then lawyers will draw up electronic contracts which are digitally signed. Finally Company A transfers funds from his account to B’s account and the deal is reached, as shown in Fig. 3.8.

B2B e-commerce essentially involves industrial markets, which are defined as individuals or groups that purchase a specific type of product for resale, for use in making other products, or for use in daily operations [300] (p 139).

1. http://www.indiawebdevelopers.com/services/ecommerce.asp

Many experts believe the next wave of Internet use will be driven by B2B e-commerce [56] (p 270). Thus, B2B e-commerce is one of the principal forms of e-commerce [303] (p 9). B2B e-commerce is used by businesses perhaps most commonly to improve communication within the organisation and to cut the cost and increase the efficiency of business processes. Some B2B e-commerce systems such as eBay (http://www.ebay.com) are very impressive. However, these sites are not seller-centred but buyer-centred. The key idea behind these is that if the buyers are there, the sellers will come[1].

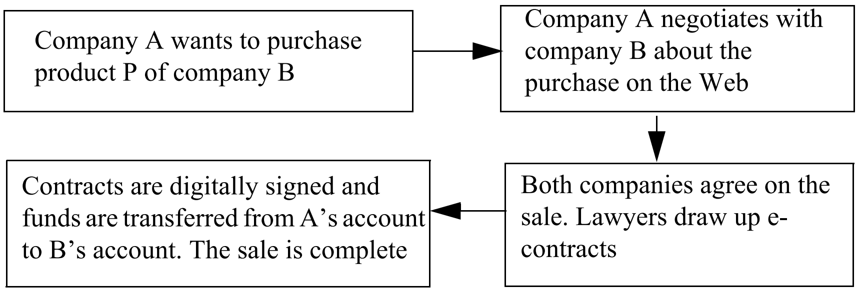


Fig. 3.8 B2B e-commerce model after [299]

3.5.2 Business to Consumer (B2C) E-Commerce

Another principal form of e-commerce is B2C e-commerce [303] (p 9)[56] (p 270), which is one that has been most completely developed in e-commerce. B2C e-commerce[2] is applications that provide an interface from businesses directly to their consumers. The most common example of a B2C application is a retail Website featuring the business's products or services that can be directly purchased by the consumer such as www.amazon.com. The basic model of B2C e-commerce is shown in Fig. 3.9.

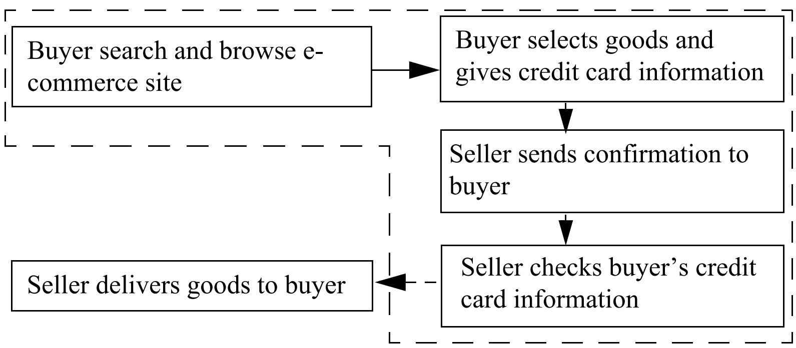


Fig. 3.9 A model of B2C e-commerce after [299] (p 265)

1. See http://www.cnn.com/2001/TecH/internet/09/24/web.auctions.idg/index.html
2. See http://www.peterindia.com/E-businessOverview.html.

B2C e-commerce can involve electronic transactions in marketing, ordering, and paying, after sales service and, in the case of intangible or virtual goods and services, even delivery (as mentioned in Section 3.2.2). B2C e-commerce is used by customers for the convenience of purchasing products or services over the Web. Businesses use B2C e-commerce to attract new customers, to reach new markets, and promote products and services.

B2C e-commerce is also called consumer-oriented e-commerce [110]. In consumer-oriented e-commerce, companies deploy virtual storefronts to sell their goods and services directly to the customer.

B2B e-commerce and B2C e-commerce share at least two major goals. One of these major goals is to complete the transaction on time [299]. For example, in the case of B2C e-commerce, the seller has to minimize the time between the time of purchase and the time the buyer gets his goods. Another of the major goals is to complete the transaction without taking into account distance and time to provide even more satisfaction for customers. But there are also differences between these two kinds of e-commerce. The major difference is how a business is carried out. This is similar to the real world. In B2B transaction, people can give credit cards, cash or checks to make a purchase. In the Web world, credit cards are used most often. In B2C transactions, corporations have company accounts that are maintained and the accounts are billed at certain times. This is still valid in e-commerce.

3.5.3 Consumer to Consumer (C2C) E-commerce

C2C e-commerce consists primarily of Websites where consumers deal directly with one another, such as online communities, free personal classified pages, auction houses. This obviously means that the company facilitating the transaction must find some non-traditional revenue stream. This could be a small cut of the transaction, a service fee, advertising, or some combination of these. The basic model of C2C is shown in Fig. 3.10.

E-bay (www.ebay.com) is an example of a C2C e-commerce application that is popular with consumers. Every one can open his own store and display and sell all of his items in the e-bay world. Every one can also search and buy books, toys etc. in the e-bay stores.

The current e-commerce is mainly supported by the various kinds of companies, small and giant ones. C2C e-commerce is thus relatively negligible. However, C2C e-commerce is an interesting and relatively new piece of the e-commerce world[1]. The development of the Internet and the WWW makes the traditional distance zero and the communication free. Then the development of C2C e-commerce will become an important complementary part for facilitating B2B e-commerce and B2C e-commerce.

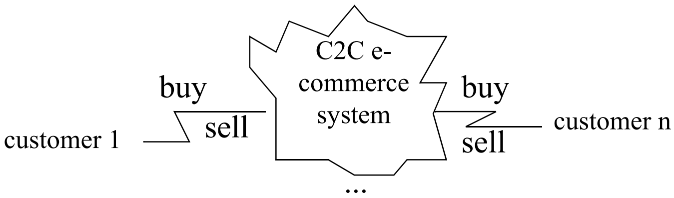


Fig. 3.10 A model of C2C e-commerce

3.5.4 Summary

In this section three kinds of transaction-based e-commerce were explored with models and examples. Using Web technology, these three kinds of transaction-based e-commerce will facilitate the buying and selling of goods and services and the transfer of funds. They also embrace such intercompany and intracompany processes as procurement, order handling, production, marketing, sales, and distribution.

3.6 Information Overload and Search

This section will examine how to lessen information overload and obtain the right information in the Internet at a right time. Although the scenario is information for investors, it is still valid for a more general setting.

3.6.1 Information Overload

With the explosive growth in the Internet, there is an unprecedented abundance of digital information available on the Web and the Internet. The magnitude of information available on the Web is so great that it is more and more difficult for a person to collect, filter, evaluate, and use information available in the Internet for problem solving [227]. For example, investment information overload from the explosive growth of the World Wide Web is making it difficult to search on-line in order to find an investment which really meets one's needs [286]. Despite enormous efforts in categorising information, linking information, and search and retrieval as well as many well-known search engines available on-line, this overload has not yet been tamed.

1. See http://www.peterindia.com/E-businessOverview.html

Another example is that while writing this section the author used "investment information" as search key words. Google, a popular and comprehensive search engine, found about 2,760,000 Web items, and found 3,010,000 Web items for "e-commerce". However, the most comprehensive search engine covers no more than 16 percent of the available Internet [204], although Google can search 2,469,940,685 Web pages[1]. Therefore, information overload is a critical issue facing the information society. Under these conditions, information providers (e.g. Commonwealth Securities) face the problem of creating and building the most effective new-media channels to get their information to the appropriate information customers. The information customers face the problem of how to get the appropriate required information using as little searching time as possible. Individuals recognize that timely and relevant information is critical to business success, and that information truly is power. Few information customers such as investors have enough time to perform costly and ineffective on-line searches. They often lose patience facing such an information overload.

3.6.2 Search Engines

As information in the Internet grows exponentially, new ways of finding the right information and resolving information overload are needed. At the moment, search engines are still a very popular means of finding information and resolving information overload [119], because the first attempt to deal with the information overload on the Web was the search engine [21]. Modern, powerful search engines such as Openfind[2], Yahoo, and Google provide an example of how digital technology can be used to retrieve information. The most important feature of any search engine is ease of use and accuracy, the search speed, for example, of Google, is so satisfactory that it becomes of secondary importance [303] (p 135).

Search engines such as Yahoo generally only provide key word input and then search according to the input of the user [21]. If the user requires further limited search requirements, Yahoo uses another page to let the user select retrieval methods and search domain and simple time limitation. It doesn't deal with the semantic refinement of the key words, but searches its

1. This data was recorded on 11.09.02
2. http://www.openfind.com

information database and finds the related information and sends it to the user, which includes much information of no value.

Search engines work most effectively when information is indexed, graded and categorized as they are posted [21]. Since most searchable documents are now on the Web, and the Web didn't grow out of this philosophy, there is a definite practical limit to the performance one may expect of future search engines no matter how finely tuned. Search engines work with these constraints. While some additional effectiveness may be expected in such areas as indexing behaviour (e.g. more sophisticated parsing and integration of <meta> and <title> tags with the index of the document body), it is unlikely that even the best groomed search engine can satisfy all of our long-term needs. It appears as if the Internet will continue to be over-indexed for the foreseeable future. A partial solution is to develop personal software agents to help information search which will be discussed in next chapter.

3.6.3 Summary

This section discussed information overload and information search. In fact, how to resolve information overload and get the right information in the Internet at right time becomes a big issue not only for e-commerce but also for all kind of e-activities. However, it should be noted that on the one side, the information overload brings a big burden; on the other side, one has to face the fact that information overload is a natural phenomenon during the fast growing Internet and the WWW. Nobody can change this situation thoroughly with the still extraordinary growth of the WWW. Therefore one should change the traditional way of thinking about information overload, and admit that it is such information overload in the Internet that make us have a good understanding about our intelligence and artificial intelligence, because information overload in the Internet will lead to new intelligent techniques.

3.7 Information Brokerage

As a result of information overload, the problem of locating information sources, accessing, filtering and integrating information and coordinating information retrieval and problem solving efforts of information sources have become very critical. This requires new methods of searching, filtering, and organising information that are appropriate for the decision support task in e-commerce [8]. If the first step of reducing information overload in the Internet is to use search engines, then the second step for reducing information overload is to delegate some activities to intelligent agents [178]. Apparently, the delegation of tasks is an important response to the above problem, because through delegation the desktop, server, and the middle-ware can assume more of the end user's work. Information brokering is one of the most important forms of delegation and also an important form of e-commerce.

Information brokering is the business of buying and selling information as a commodity. It has been around us for a long time and is very popular on the Internet [173]. Information brokering is mainly involved in three components: information customers, information providers, and information brokers. A general architecture of information brokering is shown in Fig. 3.11. For

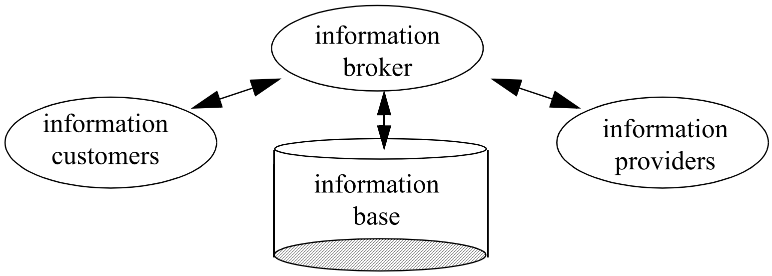


Fig. 3.11 A general architecture of information brokering

example, investment information customers are mainly investors such as personal investors for e-investment. They can consult the investment information base (repository) through investment information brokers [127]. They can also get the requested investment information from the investment information brokers, which might be fee-oriented [286].

Investment information providers are mainly stockbrokers, listed investment companies, research organisations for investment, and any others, e.g. Reuters. They supply investment information for information brokers. They also provide other financial services on-line such as chart analysis and portfolio tools in order to make themselves more popular in the information-rich environment of the Internet. The comprehensive search engines available on the Internet such as Yahoo can also be considered information providers, because they provide information about the stocks, funds, or other investment services.

An investment information broker is an individual or organisation who on demand seeks to answer questions using all sources available and is in business for a profit. In open cyberspace, investment information brokers (a more general term is agents) are paid and have to pay for any investment services they provide to their customers.

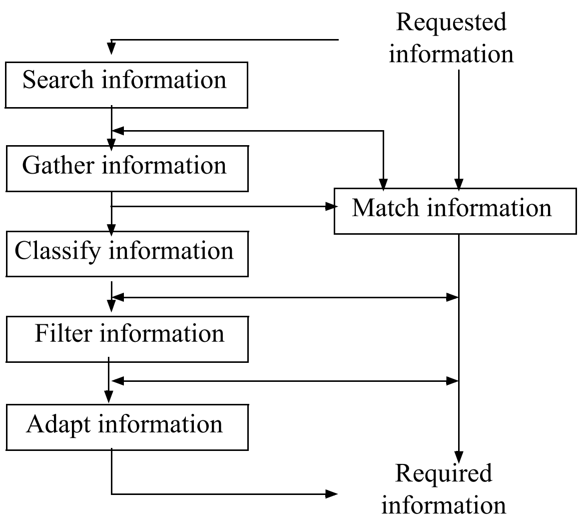


Fig. 3.12 Information flow in information brokering

A good information broker will not just take a customer's question and answer it as early as possible [286]. The information broker should be concerned with several aspects of the investment and probe the customer for as much information about the investment as possible. Further, the information broker should accept information from all available information providers. He should also retrieve, select, gather, organise, store, analyse and evaluate, filter, adapt, and disseminate information or return recommendations to requesting information agents in various ways [286], in order to meet the information customer's needs with high quality information, as shown in Fig. 3.12. In fact, intelligent information agents and intelligent information brokers for information brokering co-operate with one another, and can take away much of the information burden an information customer is confronted with in e-commerce [286]. In the next chapter a model for an intelligent information broker based multiagent system will be proposed and discussed in detail.

3.8 Concluding Remarks

This chapter discussed the evolution from traditional commerce to e-commerce, and examined three kinds of chains in e-commerce: value chains, supply chains, and agent chain. It showed that the linear structure of the traditional value chain, supply chain, and agent chain can be replaced by the most complex structure; that is, a complete graph-based structure, because the dramatic development of the Internet and WWW makes communication free of the time constraints and distance essentially zero. This chapter also discussed transaction-based e-commerce: B2B e-commerce, B2C e-commerce, and C2C e-commerce with models and examples and argued that C2C e-commerce is also an important supplement to the major forms of e-commerce: B2B e-commerce and B2C e-commerce. How to obtain the right knowledge in the right place at the right time is also a big issue for e-commerce with the increasing information overload in the Internet. Therefore this chapter finally examined information overload and information brokerage with models.

It should be noted that e-commerce applications are just about to leave their infancy [224]. New technologies and techniques are still required to enable users to do business on-line with confidence and without taking into account time, distance, and make e-commerce more intelligent, more effective, and more personalized. One solution to these problems is to use CBR through intelligent agent technology, which will be discussed in the following chapters in detail. First of all, the next chapter will provide fundamental treatment of intelligent agents and multiagent systems (MASs).

4 Intelligent Agents and Multiagent Systems

This chapter is the third chapter in the Part I of the thesis. It is also the basis for Chapter 7 and 8, as shown in the shaded area of Fig. 4.1. This chapter will briefly investigate basic features and architectures of intelligent agents, intelligent brokers, and multiagent systems (MASs). It also examines the relationship between intelligent agents and expert systems (ESs) as well as MASs, in which a knowledge-based model of integrating ESs in MAS is proposed. Then it proposes a multiagent-based architecture for information brokering as an example of the architecture of MAS, which can help customers to access information on the Internet.

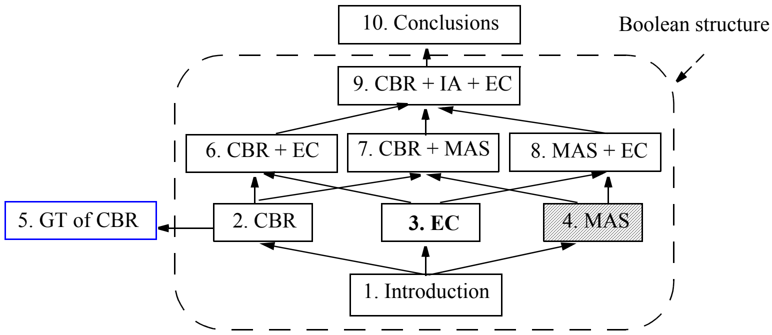


Fig. 4.1 Chapter 4 in the Boolean structure of PhD-thesis

4.1 Introduction

With the rapid development of the WWW, intelligent agents and MASs are among the most rapidly growing areas of research and development in computer science [32][125][180]. They help private and business users in their search for information and performance of tasks in a networked, digital world and improve the decision making in their ordinary business activities [52][84][201]. Many experts believe that they provide a new category of software that will gain greatly in importance in the coming years [31] [32][139].

Agent research crosses many disciplines and has been influenced by areas such as artificial intelligence (AI), distributed computing, software engineering, sociology, economics, object-oriented systems, artificial life, and game theory [86].

This chapter will investigate basic features and architectures of intelligent agents, intelligent brokers, and MASs. It also examines the relationship between intelligent agents and intelligent brokers, the relation between expert systems (ESs) or knowledge-based systems (KBSs) and MASs. The main idea stressed here is that the intelligence level of the MAS can be improved through coordination, cooperation, communication, and negotiation among the agents within the MAS, although each of them may be less intelligent than an ES. Then it proposes a multiagent-based architecture for information brokering, which can help the customers to access information on the Internet. The key idea behind the architecture is that the task of a human information broker should be done by a few cooperative intelligent information agents within a MAS.

The remainder of the chapter is structured as follows. Section 4.2 discusses the essence of intelligent agents. Section 4.3 and Section 4.4 examine two special intelligent agents: mobile agents and intelligent brokers. Section 4.5 discusses the fundamentals and features of MASs, and examines the relationship of coordination, cooperation, and communication. Section 4.6 argues the relationship between ESs and MASs. Section 4.7 highlights the architectures of MASs and proposes a multiagent architecture for an information broker. Finally this chapter is ended with a few concluding remarks.

4.2 Intelligent Agents

In these days, one only uses a click of a web browser to retrieve information with a search engine. Let’s go one step further from search engines and meet intelligent agents, which make you a click on a browser and you can get all the information and services you could possibly want. Intelligent agent[1] technology has become important in both AI and mainstream computer science. It is now making the transition from universities and research labs to industrial and commercial applications, such as information retrieval and e-commerce [86]. This section will examine fundamental features of agents.

4.2.1 What is an Intelligent Agent?

Researchers in the agent world have offered a variety of definitions, each hoping to explicate his or her use of the word “agent”. These definitions range from the simple to the lengthy and demanding. They are used in different concepts and contexts from cognitive modelling of human behaviours to information agents of the Internet. Generally speaking, the concept agent is used in the subject matter of two contexts [227]. The first is a software engineering context where an agent is a “softagent” interacting only with software entities in a computer software world. Some

1. Hereafter, “agent” means an abbreviation for “intelligent agent,” for brevity.

of the examples are information agents, Internet agents, and database management agents. The second context of the term agent is a cognitive and engineering attempt to the explanation, modelling, and simulation of human mental functions. An intelligent agent is considered as some abstraction from human persons to the specification of various professional, social, and psychological roles. For example, seller agents and buyer agents in the e-market belong to this category [227].

However, there is not a universally accepted definition of agenthood in the literature [139][227] owing to the aforementioned two different contexts of the term of agents. There are, however, several widely accepted concepts which characterise agent systems. Further, almost every definition of the agent is a specification of the definition from the Webster's dictionary; that is, an agent is: "an entity which acts for or in the place of another by authority from him, either as a representative or as an instrument by which a guiding intelligence achieves a result." The following definitions are among them:

- Individual problem solving entities are called agents in distributed artificial intelligence (DAI) systems [137]
- Agents can be as simple as subroutines, or each of the system components might be considered as agents. As such they represent some cognitive speciality. Typically they are larger entities with some sort of persistent control (e.g. distinct control threads within a single address space, distinct processes on a single machine, or separate processes on different machines)[100]
- By the term *agent*, Wellman refers to a module that acts within the mechanism according to its own knowledge and interests [330]
- An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors [96]
- An agent is a self-contained program capable of controlling its own decision making and acting, based on its perception of its environment, in pursuit of one or more objectives [207]
- An agent is a software program designed for performing a specific task based on its own knowledge and the message it received [178]
- A software agent is a computer program that functions as a cooperating personal assistant to the user by performing tasks autonomously or semi-autonomously as delegated by the user with a common understanding agent more than a task perception and execution program [227]

- An entity is a software agent if and only if it communicates correctly in an agent communication language like ACL [100]. This means that the entity must be able to read and write ACL messages, and it means that the entity must abide by the behavioural constraints implicit in the meanings of those messages.

In order to differentiate an agent from an ordinary program, Franklin [96] analysed 11 different definitions of agents available at that time and examined the requirements constitute the essence of being an agent, and then he formalized them into the following definition:

"An *autonomous agent* is a system situated within and a part of the environment that senses that environment and acts on it, over time, in pursuit of its own agenda and to effect what it senses in the future. The environment here can be computer operating systems, a computer screen or its memory, databases, networks or the Internet, etc. The agents live in all these artificial environments."

This definition yields a large and varied class of agents as was to be expected of one requiring only the essence. No doubt it's too general to be useful as is. Adding additional requirements for different purposes will produce useful subclasses of agents, which will be discussed in next section. But first, there are a couple of basic points to clarify.

Autonomous agents are situated in some environment. If the environment is changed then one may no longer have an agent. A robot with only visual sensors in an environment without light is not an agent. Systems are agents or not with respect to some environment. Some agents discussed above require that an agent "can be viewed" as sensing and acting in an environment; that is, there must exist an environment in which an agent lives.

What are about ordinary programs? A payroll program in a real world environment could be said to sense the world via its input and act on it via its output, but is not an agent because its output would not normally effect what it senses later [209]. A payroll program also fails the "over time" test of temporal continuity. It runs once and then goes into a coma, waiting to be called again. Most ordinary programs are ruled out by one or both of these conditions, regardless of how one stretches to define a suitable environment. Therefore, all agents are programs, but not all programs are agents based on the above definitions. Furthermore, agents differ from 'traditional' software/program in that they are personalized, semi-autonomous, proactive, adaptive, and so on.

These qualities make agents particularly useful for the information-rich and process-rich environment of e-commerce.

In this research, intelligent agents are considered as autonomous and adaptive computer programs operating within software environments such as databases or the Internet [86]. They are the software counterpart of human agents existing in the society and business. Typical tasks performed by intelligent agents could include collecting, filtering, and processing information, scheduling appointments, locating information, alerting to commerce opportunities and making travel arrangements, etc. For example, information agents filter and coherently organize unrelated and scattered data; and autonomous agents are able to accomplish unsupervised actions. Individuals are capable of handling these routine tasks and have been doing so for years. But intelligent agent technology holds the promise of easing the burdens on users by automating such tasks [32][125][250].

4.2.2 Features of Intelligent Agents

Intelligent agent technology combines AI (reasoning, planning, natural language processing, etc.) and system development techniques (object-oriented programming, scripting languages, human-machine interface, distributed processing, etc.) to produce a new generation of software that can, based on user preferences, perform tasks for users [250][125]. In fact, because the varieties of agent definitions, there are many features or attributes of agents that have been discussed in agent technology. The main fundamental features of agents (also see [143]), which are shared in the current research and differ from traditional software, are listed in Table 4.1.

In practice, some features are often associated with the notion of an intelligent agent in literature [34][32][94][95][250] such as adaptability, autonomy, cooperativity, mobility, proactivity, and reasoning capability. It is necessary to examine them in some more detail.

- Adaptive behaviour is the ability to learn and improve with experience. This means that learning behaviour is the necessary condition of adaptive behaviour. For example, learning agents could learn the user's habits and preferences over time and either respond to requests or act on the user's behalf based on what they learned
- Autonomy is the ability of agents to handle human user-defined tasks independently of the user and often without the user's guidance or presence. The user does not become directly involved in executing the task. Once he has specified how and when a task should be performed, the

intelligent agent is delegated to perform it when the right conditions are met. Autonomy seems to be more useful in distinguishing agents from other kinds of software, as opposed to the general notion of intelligence [230]. Further, autonomy seems to be central to agenthood

Table 4.1 Features of agents

Features	Other names	Meaning
adaptive	learning	change its behaviour based on its previous experience [96][304]
autonomous		exercise exclusive control over their internal state and behaviour or they can act on their own [140][143][227]
goal-oriented	proactive, initiative	does not simply act in response to the environment [21][96][143][353]
character		believable “personality” and emotional state
cooperative		ability of agents to work with other agents to achieve a common goal.[143][353]
communicate	socially able	able to exchange information with other entities (agents, humans, objects, their environment) [96][227]
commitment		if an agent advertises a willingness to perform a service, then it is obliged to perform that services when asked to do so [100]
mobile		ability to move from one location to another while preserving their internal state [353]
flexible		actions are not scripted [96][140]
goal-oriented	proactive, initiative	does not simply act in response to the environment [21][96][143][353],
reactive	sensing & acting	responds in a timely fashion to changes in the environment [96][227]
temporally continuous		is a continuously running process [96]
reasoning	rational	ability to infer and extrapolate based on current knowledge an experiences - in a rational, reproducible way [227]
inductive		agents are allowed to adapt and learn from their environment by recording the success and the failure of their past actions [106]
Interoperative		[353]
proactive		able to exhibit opportunistic, goal-directed behaviour and take the initiative where appropriate [140]
personalised		MIT [209]
persistent		maintain a consistent internal state over time which is not changed capriciously [227]
planning		synthesise and choose between different courses of action intended to achieve its goals [227]
social ability		able to interact, with other agents in order to complete their own problem solving and to help others with their activities [140]

- Cooperativity is the ability of agents to work with other agents to achieve a common goal. Agents are usually developed to provide expertise in a specific area and can, through cooperative work, jointly accomplish larger and more complex tasks

- Mobility is the ability of agents to traverse one Web server to another in a self-directed way, carrying actions for remote execution. Further, mobility can be classified into: data mobility, control mobility, and mobile problem solving methods (PSMs) [197]. Using mobile PSMs an agent can share and reuse other cooperative agents' experiential knowledge
- Proactivity is the ability of agents to take action initiatively, rather than waiting for something to happen and then acting as a result of it
- Reasoning capability is the ability of agents to operate in a decision-making capacity in complex, changing conditions. This property is usually associated with making inferences, having the competence to choose among different strategies or the capability to plan a task
- Rationality: an ideal rational agent is defined as follows: for each possible percept sequence, it acts to maximize its expected utility, on the basis of its knowledge and the evidence from the percept sequence [125] (p 4).

An intelligent agent does not necessarily have all these abilities. It should be tailored to a special problem specification. Further, it is useful to combine some listed features and define a special agent to model a real word problem. For example, a flexible agent is responsive, proactive, and social [140]. In fact, each of the listed features is certainly a facet of human real intelligence. What features a researcher stresses are basically pragmatic and easy to be treated from the perspective of the researcher.

In agent technology, almost every agent satisfies the first two properties [21][76]. Adding other properties produces potentially useful classes of agents, for example, mobile agents. As to a particular agent, it may possess only a few features of the above listed. Because agents are everywhere in the real world, people daily encounter travel agents, real estate agents, and so on. Agents may be usefully classified according to the subset of these properties that they enjoy which can correspond to a certain agent in the real world. This is the reason why there are so many fundamental features introduced in agent technology. This also indicates that agent technology has drawn much attention among the researchers with different background in the real world. From the Boolean algebra viewpoint, if there are N different features, then 2^N different kinds of agents can be obtained. If there are M different kinds of agents in the real world, then there should be at least number N so that

$$2^N \geq M \tag{1}$$

If one finds this N, then he can use it to classify agents based on a Boolean lattice[1]. However, there are no investigation into finding out the number N, although there are many studies about the classification of agents from a pragmatic viewpoint.

4.2.3 Intelligence Level of Intelligent Agents

Turban [304] classified intelligence of intelligent agents into following four levels:

Level 0- Retrieve documents as specified by users, help in navigation. The example is a browser.

Level 1- user-initiated searches using key words, etc. The example is search engines.

Level 2 -software agents such those that match user's profiles with items in catalogues. The example is information monitoring and alert agents.

Level 3 -learning agents with a deductive component.

However, it is probably reasonable to say that the intelligence level of an agent can be correlated to the degree to which it implements the features listed in the previous subsection. It is thus better to think of agents as providing a range, or different levels, of intelligence, just as people have different intelligence in human society [34].

Further, the level of intelligence of an agent depends heavily on the advance of information technology and its living environment. For example, if one can compute the price on the market with the fastest speed, then he might be called the cleverest man in some places. With the popularity of computers in the world, the computation ability has changed its role as the highest level intelligence into a basic ability of implementing the "basic intelligence of human beings". What is the highest level intelligence of human beings at present becomes open. However, it is basically valid that which human ability has not yet been implemented, or which aspects of human intelligence have not been best studied is the highest level intelligence. This is a basic criterion, different from the work of Turing in 1950 [306]. Based on this idea, the intelligence of performing tricks of an agent might be the highest level intelligence of an agent. Performing

1. One of the differences between a Boolean lattice and a set is that the latter's elements has no order or hierarchy even using the inclusion.

tricks is the necessary condition not only to win the battle or war but also to be successful in commerce [285].

4.3 Mobile Agents

This section will examine mobile agents that are among the fundamental technologies needed to build distributed applications [44][297].

Mobile agents are agents capable of roaming networking environments such as the WWW, interacting with foreign hosts, performing tasks on behalf of their owners and returning “home” having performed the duties set them [216][156].

The mobile agent concept grows out of three earlier technologies: process migration, remote evaluation, and mobile objects- all developed to improve on remote procedure calling (RPC) [48] for distributed programming [338]. Early systems supporting process migration allowed an entire address space to be moved from one computer to another. One goal of this mechanism was to reduce network bandwidth (compared to RPC) when multiple RPC calls are needed to execute an application. While process migration allowed an entire process to be transferred to a remote host, this mechanism did not allow an easy way to return data back to the source node without the entire process returning as well.

Next came remote evaluation programming, allowing one computer to send another computer a request in the form of a program (rather than an entire process address space) [338]. The remote computer receiving such a request executes the program referenced in the request within its own local address space and returns the results to the sending computer. Remote evaluation systems improved on process migration by allowing remote programming to occur without having to transmit the process control data from the source to the destination host.

Mobile objects (based on object-oriented programming (OOP)) extended remote evaluation by capturing more program behaviour within the mobile object. Such objects can migrate from node to node while carrying executable code and data in the form of object-specific properties, and potentially other embedded executable objects [338]. A number of mobile systems were developed in the 1980s. For example, the Emerald system developed at the University of Washington, which led most directly to mobile agents.

Mobile agents are also autonomous to some extent, because they themselves can decide dynamically where and when to travel to a particular destination node based on some embedded mobility meta-data to perform some required work [338]. Mobile agents improve on all these earlier technologies for distributed programming by providing a way for executable code, program state information, and other data to be transferred to whichever host the agent deems necessary to carry out the actions specified in an application.

There are at least six main benefits from using mobile agents [162]:

1. They reduce the network load
2. They overcome network latency
3. They encapsulate protocols
4. They adapt dynamically
5. They are naturally heterogeneous
6. They are robust and fault-tolerant.

Several applications clearly benefit from the mobile agent paradigm. These include personal assistance, secure brokering, distributed information retrieval, telecommunication networks services, workflow applications and groupware, monitoring and notification, information dissemination, and parallel processing. The most important application of mobile agents for this research is e-commerce.

Mobile agents are well suited for e-commerce [162][193]. A commercial transaction may require real-time access to remote resources, such as stock quotes and perhaps even agent-to-agent negotiation. Different agents have different goals and implement and exercise different strategies to accomplish them. Mobile agents, embodying the intentions of their creators can act and negotiate on their behalf, which will be discussed in more detail in Chapter 8.

4.4 Intelligent Brokers

Intelligent agents will play an important role in e-commerce for information and services, but they will not be the only actor. Intelligent brokers are another, for example, sometimes a more important one in such a place, just as the broker in the traditional bargaining process [185]. In fact, as special case of intelligent agents, intelligent brokers have also drawn an increasing interest in Web intelligence system, because brokerage services are an emerging development satisfying information needs [119].

4.4.1 What is an Intelligent Broker ?

Just with the definition of an agent, there is not a unified idea about a broker. Generally speaking, a broker is a kind of intermediary. Brokers are mediators standing between the parties of a contract (or transaction), usually buyers (clients) and sellers (servers)[1], and perform functions necessary to the fulfilment of a contract [8][61]. In the financial organisations, brokers have existed for many years, and, in fact, they have become an indispensable part of many financial transactions. The proliferation of financial brokers is based upon the efficiency with which they can serve the needs of both would-be lenders and would-be borrowers. The efficiency comes from the many resources the brokers can pool together, which helps them develop specialised knowledge and achieve the economy of scale. From a viewpoint of commerce, brokers are an important part of intermediaries that consist of two kinds of people: broker and agent. The latter is either a selling agent or a buying agent, as illustrated in Fig. 4.2.

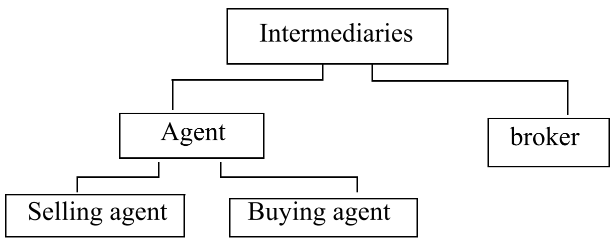


Fig. 4.2 Intermediaries

A buying agent can automatically collect information on vendors and products that may fit the needs of the company, evaluate the different offerings, make a decision with these merchants or brokers and products to pursue, negotiate the terms of transactions with these merchants and finally place orders and make automated payments [209].

Selling agents can dynamically tailor merchant or broker offerings to each customer [209]. In Market Maker[2], a selling agent is analogous to a classified advertisement. The selling agents are

---

1. This research basically uses buyer and seller instead of client and server respectively, although many studies use the latter two notions in network computing.
2. http://web.mit.edu/is/isnews/v14/n04/140401.html

proactive, they try to sell their products, by going into the marketplace, contacting interested buying agents and negotiating with them to find the best deal.

In business practice, the broker usually makes the "best possible deal" not only on the seller's behalf but also on the buyer's behalf. Therefore, the broker is the agent of both the seller and the buyer. Further, many real-life brokers perform searching, delivery or payment services, and many other functions, which do not explicitly involve the negotiation/bargaining function [85].

An intelligent broker is a software counterpart of a human broker, for example, in the business activities. In some cases, an intelligent broker is able to select, configure, and adapt problem solving methods (PSMs) for a test case [19]. The intelligent broker handles requests for reasoners from various customers. Based on these requests, it accesses different libraries available on the Web and searches them for candidate PSMs, which are adapted and configured into a knowledge system for the customer.

Brokers have also been discussed in other studies of e-commerce. For example, Ba et al. [8] investigate the information integration using brokers, WWW, and structured documents based on their client-broker-server architecture. The broker here mainly organises distributed, structured information. The broker idea is developed to encapsulate the notion of intermediation between distributed information sources and information users, the task of which is carried out by a centralized intelligent broker who is equipped with software agents that perform various broker functions.

It should be noted that the reason for differentiating brokers from agents is that some agents have no intelligence from a perspective of AI. However, an intelligent broker must have some intelligent ability. In fact, it must have strong reasoning ability to cope with all possible cases facing it.

4.4.2 Features of Intelligent Brokers

There are at least a few features of intelligent brokers that differ from those of intelligent agents to some extent; that is, intermediation, matchmaking, delegation, and bargaining.

Intermediation: In current business activities, the buyer and seller, even the broker, don't necessarily know where the relevant information resides that is critical for decision making [8].

Thus intermediation is one of the main tasks for intelligent brokers to facilitate activities of buyer agents and seller agents on the e-market.

Matchmaking: Another main task of an intelligent broker is matchmaking, which is a process whereby consumers seeking goods and services with given specifications are put in contact with providers whose goods and services match the specifications. Providers may also reach consumers in a similar fashion (http://www.igec.umbc.edu/) (also see [61][65]).

Delegation: Delegating some of the tasks of finding, classifying, filtering, organising, and adapting information from humans to intelligent brokers is at least a possible solution to resolve information overload in the Internet [19]. According to Jennings [140], the future of computing will be completely driven by delegating to, rather than manipulating computers. In order to realize delegation, the intelligent broker must be at least autonomous, proactive, responsive, and adaptive.

Compromise and bargaining: Some brokers perform the compromise/bargaining function [264] during negotiation with buyer agents and seller agents, which will be examined in more detail in Chapter 8.

4.4.3 Information Broker

According to Trepper [303] (p 20), easy access to valuable information is a strong requirement, together with the need for new technologies to build innovative intermediation platforms. An intermediation platform is a hardware and software based solution that brings parties with common interests (either commercial or social) together. New information brokerage services will be at the centre of the e-marketplace.

An information broker is a system that helps users locate the databases that are most likely to contain answers to their queries. To perform this service, brokers use summary information about the available databases. Brokers must be able both to query and to update this summary information. A central problem in broker design is to find a representation for summary information that is both effective in its ability to select appropriate information resources and efficient to query and maintain [301]. One of information brokers is the Electronic Channel Broker (ECB), which utilizes exact and approximate similarity query processing methods to perform matchmaking along with EDI (Electronic Data Interchange) for exchange of business data (http://www.igec.umbc.edu/).

4.5 Multiagent Systems

Multiagent systems (MASs) have been studied for many years, and various types of such systems have been developed [79][86][217][218][266][327][328]. This section examines the fundamentals and features of MASs.

MAS technology is now one of the most important, exciting, and fast moving areas of information technology, which is now making the transition from universities and research labs into industrial and commercial applications [32][125]. For example, TabiCan is a real commercial service site on the Internet. Using MAS, it provides airline tickets and package tours consisting of plane flight and hotel stays [346].

From the viewpoint of DAI, a MAS[1] is a loosely coupled network of problem-solver entities that work together to find answers to problems that are beyond the individual capabilities or knowledge of each entity [94]. More recently, the term MAS has been given a more general meaning, and it is now used for all types of systems composed of multiple agents showing the following characteristics:

- Each agent has incomplete capabilities to solve a problem
- There is no global system control over agents
- Data are decentralized
- Computation is asynchronous.

One of the more important factors fostering MAS development is the increasing popularity of the Internet, which provides the basis for an open environment where agents interact with each other to reach their individual or shared goals [94].

The following four concepts are vitally important to MASs: coordination, cooperation, communication and negotiation [17][32][85][125][133][134][246][286]:

- Coordination is a property of a system of agents performing some activities in a shared environment. The degree of coordination is the extent to which they avoid extraneous activity by reducing resource contention, avoiding deadlock, and maintaining applicable safety conditions

1. In DAI, a MAS is also called a *community* [137].

- Cooperation is coordination among nonantagonistic agents and arises as they plan and execute their actions in a coordinated way to achieve their goals
- Communication forms the basis of the cooperation and is formed from the communication protocols and the resulting communication methods
- Negotiation means a compromise for both parties and causes a degradation of their results. The overall aim of all negotiation activities is to permit a constructive cooperation from within the group of independently operating agents that have their own goals.

For a single intelligent agent, these concepts need not be of importance as it could do all the work on its own. However, their importance becomes evident in the MASs; standards-based mechanisms and means to coordinate, cooperate, and communicate with all kinds of agents are at the root of the MASs [86]. The rest of this section will discuss the first three and their relationship in some more detail, and leaves the last to be discussed in Chapter 8.

4.5.1 Cooperation

Cooperation is often presented as one of the key concepts which differentiates MASs from other related disciplines such as distributed computing, object-oriented systems, and KBSs [68].

Generally speaking, to cooperate is to act with another or others for a common purpose and for common benefit [68]. For independent commercial organisations to cooperate, they must generate an explicit agreement to act for a common purpose and for common profit. The motivation to cooperate is derived from their individual motivations to maximise profit while minimising their costs. Therefore, cooperation is beneficial to both parties in some cases. In MASs, successful cooperation can be generated between agents that are not a priori cooperative through negotiating a mutually acceptable agreement to which they are both committed that describes how they are to act (i.e. a binding agreement of cooperative intent.)

From a higher level, using the cooperation metaphor it is possible to decompose the system into a number of simpler and logically separate agents that could work on dedicated areas of the problem [137].

As to cooperation, three primary objectives related to the agent's role in a social problem solving context should be supported [137]. First, it has to establish new social interactions (e.g. find an agent capable of supplying a desired piece of information). Second, it has to maintain

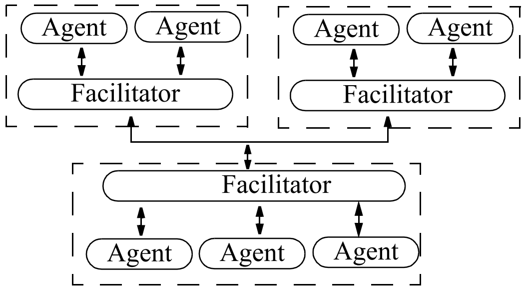


Fig. 4.3 A federated system after [100])

cooperative activity once it has been established, tracking its progress until successful completion (e.g. send out relevant intermediate and final results to interested agents). Finally it has to respond to cooperative initiations from other agents.

4.5.2 Coordination

As mentioned in the last section, coordination is an important feature of a system of agents performing some activities in a shared environment. In fact, in regard to agent models (representations of oneself and other agents in a MAS), the identification, design, and implementation of strategies for cooperation based on agent models have been key research issues since the early years of the field of DAI [197]. This subsection will look at some aspects about coordination in MASs.

Once having a language and the ability to build agents, there remains the question of how these agents should be organised to enhance cooperation. Two very different approaches have been explored in [100]: direct communication (in which agents handle their own coordination) and assisted coordination (in which agents rely on special system programs to achieve coordination), because coordination enables the agents to operate in a shared environment [125].

The advantage of direct communication is that it does not rely on the existence, capabilities, or biases of any other programs [100]. Two popular architectures for direct communication are the contract-net approach of Davis and Smitch in 1983 and specification sharing. One disadvantage of direct communication is cost. If the number of agents is small, this is not a problem. However, in a setting like the Internet, with millions of programs, the cost of broadcasting bids or specifications and the consequential capabilities of those messages of prohibitive. Another disadvantage is implementation complexity. Each agent is responsible for negotiating with other agents and must contain all of the code necessary to support this negotiation.

A popular alternative to direct communication that eliminates both of these disadvantages is to organise agents into what is often called a *federated system*. Fig. 4.3 illustrates the structure of such a system in the simple case in which there are just three machines, one with three agents and two with two agents apiece. As suggested by the diagram, agents do not communicate directly with each other. Instead, they communicate only with system programs called facilitators, and the latter communicate each other.

4.5.3 Communication

The most obvious autonomy-preserving interactions are communications. Enabling heterogeneous programs written by different people, at different times, in different languages, and with different interfaces to communicate and interoperate has attracted attention in MASs [100]. This section will examine some attempts to facilitate the communication in MAS, in particular agent communication languages (ACLs).

Researchers in the ARPA Knowledge Sharing Effort have proposed agent communication languages (ACLs) as the means to allow the exchange of knowledge among software agents in order to facilitate their cooperation and coordination [197]. Generally speaking, an ACL(Fig. 4.4)

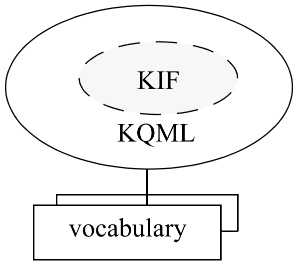


Fig. 4.4 An ACL

consists of three main components [125][197]: an open-ended vocabulary of words appropriate to a common application area, an inner language, KIF (Knowledge Interchange Format), to encode the information content communicated among agents, and an outer language, Knowledge Query and Manipulation Language (KQML), to express the intentions of agents.

KIF is a prefix version of the language of first order predicate calculus with various extensions to enhance its expressiveness [100]. It provides for the encoding of simple data, constraints, negations, disjunctions, rules, quantified expressions, metalevel information, and so forth. Despite these extensions and restrictions, the core language retains the fundamental characteristics of first order logic, including compactness and the semi-decidability of logical entailment.

KQML provides the agent designer with a standard syntax for messages, and a number of performatives that define the force of a message [340]. Example performatives include tell, perform and reply. KQML *Messages* are similar to KIF expressions [100]. Each message is a list of components enclosed in matching parentheses. The first word in the list indicates the type of communication. The subsequent entries are KIF expressions appropriate to that communication, in effect the “arguments”.

Nowadays, modern ACLs, such as KQML, FIPA ACL (the Foundation for Intelligent and Physical Agents), and KAoS, reflect the consensus that agent communication is best analysed by viewing the messages which agents exchange as designed to achieve certain ends. ACL designers analyse agent communication as composed of *intentional* actions [123]. However, when the agents interact by exchanging messages a higher level of interaction concerned with the conventions that they share during the exchange should be addressed. Such a level of interaction is not supported by KQML, whereas coordination languages- like COOL- allow such conventions to be explicitly expressed.

It should be noted that two prominent ACLs are KQML and the FIPA ACL. The FIPA ACL is rapidly spreading to replace KQML as the ACL of choice [123]. Further, cooperation and coordination is goal of agents using communication. This is the essence between communication, cooperation, and coordination.

4.5.4 Relationship of Coordination, Cooperation and Communication

Franklin [97] explores a collection of examples of coordination without communication and then argues that coordination with or without communication is a property of MASs. He also observes that the main mechanism of coordination without communication is repeated and frequent sampling of the environment, and responding thereto. The advantage of using coordination without communication as a control architecture in MASs is that it can possibly save computing resources.

It is worth noting here that coordination, cooperation, and communication are expensive, requiring additional architecture, and more intelligence. If a system can accomplish its tasks without communication, then it is better. In fact, the Boolean structure can be used to coordination, cooperation, and communication, which are stood by C_1, C_2, and C_3 respectively, in order obtain a Boolean model for MASs, as illustrated in Fig. 4.5. This structure extends the idea

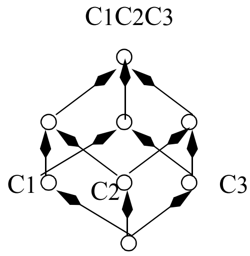


Fig. 4.5 CCC based on Boolean lattice

of coordination without communication introduced by Franklin [97] to a complete systematic (logical) treatment of MASs with 3 "C" because it contains all possible combinations of coordination, cooperation, and communication from a logical viewpoint. Moreover, this model classifies MASs into eight different categories. The simplest category is MASs without any of these three "C"s, which can be referred to the classical knowledge-based systems (KBSs), while the most complicate category is the MASs with all these three "C"s. Further, this model also paves the way from KBSs to MASs.

4.5.5 Summary

This section examined cooperation, coordination, and communication. In fact, they all belong to the category of social interaction [138]. Communication is the basis for social interaction. Automating the social interaction of human beings is one of most important goals of MASs, different from that of ESs that aims to automate the individual intelligence of human experts. Further, the proposed model for the relationship of cooperation, coordination, and communication paves the smooth way from KBSs to MASs, which will be discussed in the next section.

4.6 ES = MAS ?

As is known, in the 1980's Expert systems (ESs) used to be one of the most exciting research fields in computer science [287]. ESs were one of most successful applications in AI even in the early 1990s [341]. MASs are among the most rapidly growing areas of research and development in AI communities with the rapid development of the Internet and WWW [85][125][214]. This section overviews ESs and MASs, and examine their relationships, and then proposes a model for integrating ESs and MASs, which is a generalization of the ideas of Jennings in [137]. It also argues that an ES as a kind of a MAS and simulation of human intelligence depends not only on the knowledge and reasoning of human experts but also on cooperation, coordination, and communication among the agents within an intelligent system.

4.6.1 Expert Systems

ESs employ human knowledge to simulate expert performance, and they present a human-like facade to the user [114]. Knowledge in ESs means those kinds of data that can improve the efficiency or effectiveness of a problem solver. Three major types of knowledge fitting this description are: facts, beliefs, and heuristics. Facts express valid propositions, beliefs express plausible propositions, and heuristics express rules of good judgment in situations where valid algorithms generally do not exist.

The origin of the ES might be found in the research on the General Problem Solver (GPS) in the late 1950's [115][284], from which one major insight gained was the importance of domain-specific knowledge or expert knowledge [184]. Expert knowledge is a combination of a theoretical understanding of the problem and a collection of heuristic problem-solving rules that experience has shown to be effective in the domain. ESs are constructed by obtaining this knowledge from human experts and coding it into a form that a computer may apply to similar problems. This reliance on the knowledge of human domain expert for the system's problem-solving strategies is a major feature of ESs.

Early research of ESs arose in universities in the mid-1960s and emphasized matching the performance of human experts [114][287]. DENDRAL was the first to achieve expert performance and identified the chemical molecular structure of a material from its mass spectrographic and nuclear magnetic resonance. Whereas DENDRAL was one of the first programs to effectively use domain-specific knowledge to achieve expert level problem-solving performance, MYCIN established the methodology of contemporary ESs [184]. MYCIN uses expert medical knowledge to diagnose and prescribe treatment for spinal meningitis and bacterial infections of the blood.

From an engineering viewpoint [287], an ES can be regarded as a process of the following sequential phases: knowledge acquisition, knowledge representation, knowledge matching, knowledge reasoning, knowledge explanation, and knowledge utilization, each of which has become a separate research field.

ES techniques are principally used for following three reasons [116]:

- to improve the reasoning of the application system
- to increase the flexibility of the application system
- to increase the human-like qualities of the system.

Because of the heuristic, knowledge-intensive nature of expert-level problem-solving, ESs generally [184]

- support inspection of their reasoning process, both in presenting intermediate steps and in answering questions about the solution process
- allow easy modification, both in adding and in deleting skills from the knowledge base
- reason heuristically, using (often imperfect) knowledge to obtain useful problem solutions.

The research of ESs developed rapidly in the 1980s, due to the 5th Generation Computing in Japan [284], the development of microcomputers, and the success of a few other ESs such as HEARSAY-II and R1, which had been developed in the 1970's. Since then, thousands of ESs (or KBSs) have been developed and deployed in industrial and commercial settings and have permeated nearly every area of industry and government such as finance, airlines, and management [116].

Generally speaking, ESs have made significant progress in the following aspects: the key role of knowledge in intelligence simulation, effective knowledge representations, and more powerful reasoning techniques. In spite of such significant progress of ESs, it would be a mistake to overestimate the ability of this technology. Current deficiencies include [184]:

- Difficulty in capturing "deep" knowledge of the problem domain
- Lack of robustness and flexibility
- Inability of providing deep explanations
- Little learning from experience
- Knowledge acquisition is still a bottleneck for developing ESs.

Furthermore, research and development of ESs has been fading since last few years, possibly because there have been a number of widespread negative perceptions about the state of ES technology and corresponding results. These perceptions and beliefs include the following [116]:

- The technology failed when transferred from the laboratory
- The companies that employed the technology lost money and withdrew from the area. In the mid-1980s, there were hundreds of technology-based companies offering ESs development tools for sale. Most technology-based ES tool providers have disappeared since then [116]
- The technology did not live up to the claims made for it by its proponents, and users were disappointed
- The expert is simply not following any rules! That in turn explains why ESs are never as good as human experts. If one asks the experts for rules one will, in effect, force the expert to regress to the level of a beginner and state the rules he still remembers but no longer uses [70].

Facing this situation, the founder of ES, Feigenbaum himself has also to admit that ESs are very different from experts [287], "Part of learning to be an expert is to understand not merely the letter of the rule but its spirit. The expert knows when to break the rules, and understands what is relevant to his task and what isn't. ESs do not yet understand these things". In fact, if one is more realistic, then he can assert that ESs should not be expected to perform as well as human experts, nor should they be seen as simulation of human expert thinking, if one only limits his endeavour in the architecture, goal, and available technology of ESs. In other words, it is more realistic to weaken the goal of achieving an individual expert status. This is the reason why intelligent agents can be considered as a weaker form of an ES.

Further, ESs must be methods-poor even if they are knowledge-rich. This is an important result and one that has only recently become well understood in AI [184]. It is better for ESs to be both knowledge-rich and methods-rich. However, in many cases, ESs possess only one reasoning model such as *modus ponens*. This is just the reason why the ESs are not smart as expected, because in the classical mathematical logic there are at least a few dozen reasoning models which are the abstraction of human reasoning.

4.6.2 Comparison of ES and MAS

This section examines the goals, functionalities, and architectures of ESs and MASs. It also discusses the relationship between them and shows that the ES can be viewed as an important component of a MAS from a knowledge-based viewpoint.

4.6.2.1 Goals

Human experts differ from others in the quality and quantity of knowledge they possess. Experts know more, and what they know makes them more efficient and effective [114]. ESs solve problems that are normally solved by human experts [244]. To solve expert-level problems, ESs need access to a substantial domain knowledge base, which must be built as efficiently as possible. They also need to exploit one or more reasoning mechanisms to apply their knowledge to the problems they are given [287]. Then they need a mechanism for explaining what they have done to the users who rely on them. Therefore, an ES is a software counterpart of a human expert, and its goal is to simulate the intelligence of an individual human expert.

However, the goal of MASs is to solve problems that are normally solved by human agents. More generally, they mimic the role of an intelligent, dedicated and competent personal assistant [34]. Because human agents are ubiquitous, it could be asserted that one occupation corresponds to one kind of agent [85]. This also means that there are many more agents than experts in human society. Therefore, from a statistical viewpoint, the average intelligence of a human agent is lower than that of a human expert. Based on this idea, it can be asserted that intelligent agents are a weakened form of ESs, and they have thus more application possibilities than those of ESs. Furthermore, the work of a human expert in a special domain can be at first decomposed and then satisfactorily done by a certain number of intelligent agents within a MAS in a cooperative way [85][86]. The intelligence of the MAS can be improved through their cooperation, coordination, communication, and negotiation, although every agent within the MAS is less intelligent than an ES. Therefore, MASs can be used to implement what ESs can do in simulating the human expert in a special domain. In other words, an ES can be considered as one kind of a MAS.

As is mentioned, ESs and MASs aim at simulating intelligence either of human experts or of human agents. However, under some conditions, a human expert can also be regarded as a special agent. For example, an agent involved in the bargaining process can be viewed both as a business expert from the viewpoint of business and as a business agent from a more general perspective [287].

4.6.2.2 From ES to MAS

One of the best researched ESs are rule-based ESs (RBESs). RBESs address the need to capture, represent, store, distribute, reason about, and apply human knowledge electronically [115]. With the current state of the art they provide a practical means of building automated experts in applications where job excellence requires consistent reasoning and rewards practical experience. Although AI researchers have developed several alternatives, only the RBESs approach consistently produce ES solvers [287].

The key ideas of RBESs have affected many other areas of computing [115]. Two of these concern the development of MASs; that is, RBESs for communications architecture and macrorules in the form of pattern-directed modules for distributed architectures and systems of cooperating systems, which can be characterized by cooperating agents [34]. One well-known example is the blackboard system [32][184], which was first presented in the HEARSAY-II research, and is very important for communication among the agents within MASs. The blackboard represents an extension to the agenda of traditional AI systems and RBESs and makes the first attempt to support the process of distributed problem solving through the use of suitable structures [32]. A blackboard is a central global data base for the communication of independent asynchronous knowledge sources focusing on related aspects of a particular problem, and thus provides all agents within a MAS with a common work area in which they can exchange information, data, and knowledge.

Therefore, intelligent agents and MAS technology is a further development of ES technology taking cooperation, coordination, communication, and negotiation into account [115][287].

4.6.2.3 From MAS to ES

As is mentioned, cooperation is essential to MASs. In fact, the notion of using cooperating agents within the MAS has at least two attractive features at the abstract modelling level [240]:

- Using a collection of problem-solvers makes it easier to employ divide-and-conquer strategies, in order to solve complex, distributed problems. Each agent only needs to possess the capabilities and resources to solve an individual, local problem
- The idea of several agents cooperating to solve a problem that none could solve individually is a powerful metaphor for thinking about various ways that individual elements can be combined to solve complex problems.

Using these features, one can overcome the difficulties facing research and development of ESs such as cooperation and coordination in ESs. Some ES literature has already paid attention to the cooperation of the ES with other computer systems [114]. For example, [116] shows that ESs increasingly can add value in industrial and commercial settings by interacting in a cooperative way with other computing systems and human operators.

Communication capability has also become one of important techniques for ESs [114]. ESs communicate with knowledge engineers, experts, databases, and other computing systems. Just as humans access and interact with these various sources, an ES needs to speak to each in its own appropriate language. ESs communicate with knowledge engineers through structure editors that allow them to access and modify components of the knowledge base easily. ESs communicate with experts through sample dialogues with explanations that elucidate their lines of reasoning and highlight for the expert where to make knowledge base changes. Beyond their interactions with people, ESs also interact with other computing systems. For example, ESs incorporate means to access and retrieve information from online databases. In this way ESs can apply their knowledge automatically and directly to the vast stores of data that now commonly reside on-line.

Based on this discussion, MAS technology also fosters research and development of ESs. It can thus be asserted that the simulation of human intelligence depends not only on the computerized knowledge and reasoning of human experts, to which ESs have paid much attention, but also on the cooperation, interaction, and communication between an intelligent system and other computing systems, which MASs have emphasized. This is because the human intelligence depends not only on the possession of the knowledge and reasoning methods, but also heavily on the community where the human being lives or works.

4.6.2.4 Architectures

Although the architecture of the ESs undergoes substantial modification as the ESs advance in complexity, for example, blackboard architecture [115][184], the simple model of a RBES mainly consists of a knowledge base (KB), an inference engine (IE), and a working memory [114]. In a more concise way, ES = KB + IE. A KB consists of rules and facts. Rules are the most widely used way of representing domain knowledge in ESs [115]. Rules always express a conditional, with an antecedent and a consequent component. Facts constitute the other kind of data in a knowledge base and express assertions about properties, relations, propositions, etc. The IE applies the knowledge to the solution of actual problems [184].

In contrast to the simple architecture of an ES, the architecture of a MAS is complex. A MAS basically consists of a number of agents, which perform their own special task and might share a general knowledge base[1] in some cases. The other components of MAS are modules for coordination, cooperation, and communication among the agents. Thus some of the agents in a MAS can be viewed as a quasi-ES, while the others can be viewed as abstract (computational) objects, which have the problem solving capacity of an ES [92]. A quasi-ES means that an agent consists of a user (or agent)-agent interface, a procedure repository, and a processing engine [34]. The repository, similar to a KB, contains facts and rules supporting reasoning. The processing engine, similar to an IE, also contains the agent's current understanding of the user and the instructions received from the user. The agent uses its embedded agent processing engine and the related data in the agent repository to perform tasks and exchange information via a view. A set of views defines the standards of interaction between the user and the agent. The above consideration can be summarized as the following important relationship between ESs and MASs:

$$MAS = \sum_{1}^{n} A_i + C \approx \sum_{1}^{n} ES_i + C = \sum_{1}^{n} (KB_i + IE_i) + C \qquad (2)$$

where A_i is agent i within the MAS, ES_i is the quasi-ES corresponding to agent i, $i = \{1, 2, \ldots, n\}$. C is the above-mentioned modules for coordination, cooperation, communication, and negotiation among the agents. $\approx$ stands for "is similar to". Therefore, a MAS can be viewed as a kind of ES. Furthermore, it is practical to simulate each agent within the MAS using ES technology as much as possible, while making good use of MAS technology to deal with coordination, cooperation, communication, and negotiation among the agents in order to improve the intelligence of the MAS.

A concrete example of model (2) is the cooperation and control subsystem of GRATE[2] mentioned in [137][340]. This subsystem has three main problem solving modules: cooperation

1. From an ES perspective, this is the architecture of a blackboard system [32][184].
2. GRATE (Generic Rules and Agent model Testbed Environment) is a general framework for constructing communities of cooperating agents for industrial applications [137].

module, situation assessment module, and control module. Each of these modules is implemented as a separate ES. Communication between the modules is via message passing. Further, Jennings et al. have conducted an experiment of transforming two standalone and preexisting ESs for diagnosing faults in a particular accelerator into a MAS of cooperating agents [137]. Therefore the above investigation can be considered a generalization of the ideas of Jennings in [137].

4.6.3 Summary

This section investigated ESs and MASs and their goals, functionalities, architectures, and interrelationships, and showed that high-level intelligence of a system requires a more complex system structure than low-level intelligence does in most cases. The intelligence level of the MAS can be improved through coordination, cooperation, communication, and negotiation among the agents within the MAS, although each of them may be less intelligent than an ES. It thus emphasized that simulation of human intelligence depends not only on the computerized knowledge and reasoning of human experts, to which ESs have paid much attention, but also on cooperation, coordination, and communication among the components (agents) within an intelligent system, which MASs have emphasized, as shown in Fig. 4.6. Therefore, ES technology and MAS technology complement each other and their integration will facilitate the research and development of intelligent systems. Chapter 8 will consider an intelligent broker involved in the bargaining process as both a business expert and a business agent and propose two separate architectures using ES technology and MAS technology respectively.

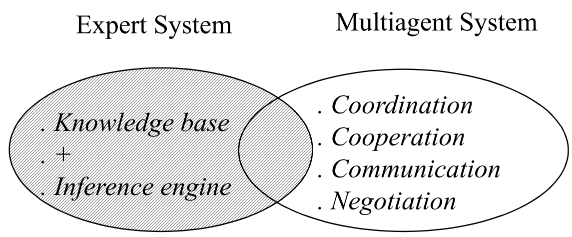


Fig. 4.6 Characteristics of ESs and MAS

4.7 Architecture of Multiagent Systems

This section will discuss architectures of MAS with examples. At first it reviews the agent architecture from the viewpoint of MASs, and then proposes a multiagent architecture for an information broker. The key idea behind it is that the task of a human broker should be decomposed and done by a few intelligent agents within a MAS in a cooperative way.

Wooldridge and Jennings [340] reviewed and classified agent architectures into three categories: deliberative architectures, reactive architectures, and hybrid architectures.

A deliberative agent architecture is one that contains an explicitly represented, symbolic model of the world, and in which decisions (for example about what actions to perform) are made via logical (or at least pseudo-logical) reasoning, based on pattern matching and symbolic manipulation. From the definition, it can assert that deliberative agent architectures are heavily influenced by traditional symbolic AI. It can be called a symbolic paradigm. One of these architectures is GRATE, which is a layered architecture in which the behaviour of an agent is guided by the mental attitudes of beliefs, desires and intentions and joint intentions [340].

Opposite to a deliberative agent architecture, a reactive agent architecture is one that does not include any kind of central symbolic word model, and does not use complex symbolic reasoning. The famous example of this kind of architecture is *subsumption* architecture introduced by R. Brooks in [281][340]. The architecture is based on his basic ideas that:

- Intelligent behaviour can be generated without explicit representations of the kind that symbolic AI proposes
- Intelligent behaviour can be generated without explicit reasoning of the kind that symbolic AI proposes
- Intelligence is an emergent property of certain complex systems.

Many researchers have suggested that neither a completely deliberative nor completely reactive architecture is suitable for building agents. They have argued the case for hybrid systems, which attempt to integrate deliberative architectures and reactive architectures; that is, a MAS consists of two subsystems, one is deliberative; another is reactive. Often, the reactive subsystem is given some kind of precedence over the deliberative one, so that it can provide a rapid response to important environmental events.

The rest of this section is devoted to a multiagent-based architecture for information brokering (also see information brokerage in Chapter 3 and [286]), which is a kind of hybrid architecture, shown in Fig. 4.7. The architecture consists of an information searching agent (also see [45]), an information gathering agent, an information managing agent, an information matching agent, an information filtering agent, an information adapting agent, and a transaction agent. In other words, the task of a human information broker could be decomposed and done by a few agents within a MAS respectively and cooperatively. This architecture thinks of each information agent as essentially another information source [19], however, it draws on already existing information repositories and applications and combines them with organisational and business model components.

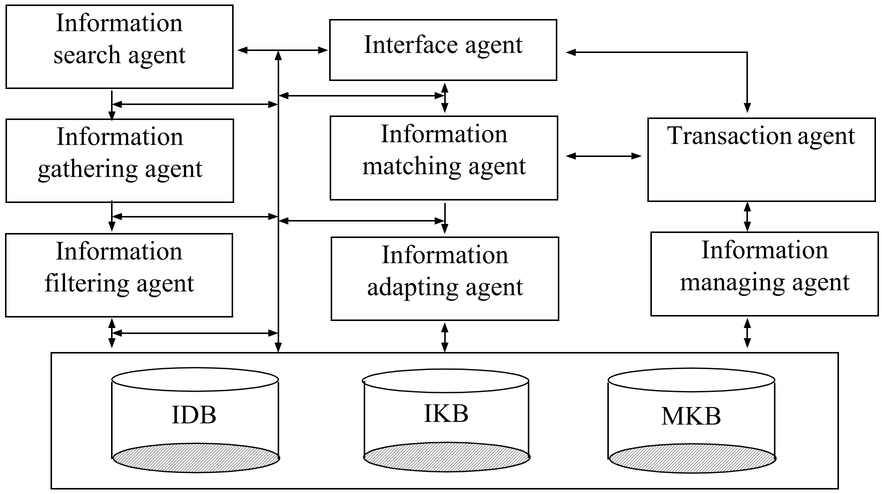


Fig. 4.7 Multiagent architecture for investment information brokering

Based on the investment information scenario, this architecture also includes a few databases; that is, IDB (investment database), IKB (investment knowledge base), and MKB (market knowledge and database), which constitute so-called information spaces [150]. Traditionally, those information spaces resided in databases, but they are now best exemplified by the Web, and are becoming increasingly virtual, dynamic and heterogeneous all the time.

Briefly, the interface agent accepts the request for information from the customers and then forwards it to the information managing agent for sorting or to the information matching agent for matchmaking. The information matching agent searches, at first, the available information in the relevant databases (i.e. IDB or IKB) and performs matchmaking between the requested information and the searched information. If the matchmaking is not successful, the information searching or gathering agent is asked to retrieve or gather relevant information on the Internet. The information matching agent will still perform matchmaking between the requested information and the retrieved or gathered information. If the matchmaking is still not successful, the information filtering agent is asked to classify and filter the gathered information. If the matchmaking between the requested information and filtered information is still not successful, the information adapting agent may have to tailor the available information in order to meet the investor (as the customer)'s need. In what follows, some agents in the architecture are examined in some more detail.

- The information search agent is a mobile agent that proactively roams around a variety of Internet search engines such as Yahoo, Openfind, and FinanceWise [286]. It simultaneously interacts and collaborates with them in order to access information resources, retrieve information related to or "similar to" the requested investment information indirectly from individual Web sites [121][150][286]
- The information-gathering agent [85] collects the retrieved information from the information search agent and puts it in the corresponding data or knowledge bases IDB or IKB. The agent has to refine and extract the retrieved information before performing collecting. All gathered information can be used by the information matching agent to perform matchmaking with the information requested and then the agent will decide if the latter has matched and met the needs of the information of the investor. The gathering agent also retrieves a wide variety of research data, such as company profiles and reports, real-time stock quotes, market updates, and information about other investment vehicles – stocks, bonds, mutual funds, treasury bills, options, precious metals certificates and then puts them in MKB. All the information is useful for investor agents to select which investment is appropriate
- The information filtering agent [150] filters the gathered information from the gathering agent according to a clustering algorithm. Information filtering is a stepwise process, in which the searched information will be filtered according to different needs or standards. The agent also offers the filtered investment information (or sends the related message about the filtered investment information) to the information matching agent, if necessary. In some cases, the agent can easily perform the relevant matchmaking based on the filtered information
- The information managing agent [85][150], similar to the data source agent in [240], manages the information in IDB, IKB, and MKB using different management methodologies
- The information adapting agent [150] adapts the investment information using both the filtered investment information and information in IDB and IKB in order to meet the request for investment information from the customers, if no filtered information satisfies the request

- The transaction agent analyses and computes the transaction cost of investment information brokering, which the information managing agent and the information matching agent suggest. This agent is similar to the computational module agent [240], which provides a wrapper for a module that performs some specialist computation, such as statistical analysis.
- IDB mainly stores structured investment data such as stock price, while IKB stores the unstructured investment information such as listed company profiles and reports. MKB includes general market information such as the political situation in a country or a region, which may affect the stock market or listed investment companies.

4.8 Concluding Remarks

This chapter investigated basic features and architectures of intelligent agents, intelligent brokers, and MASs. It further discussed the relationship of coordination, cooperation, and communication with Boolean algebra. It also examined the relationship between intelligent agents and ESs as well as MASs, in which a knowledge-based agent architecture and a model of integrating ESs into MAS were proposed, which generalizes the ideas of Jennings et al. in [137]. The main idea stressed here is that the intelligence level of the MAS can be improved through coordination, cooperation, communication, and negotiation among the agents within the MAS, although each of them may be less intelligent than an ES. Then it proposed a multiagent-based architecture for information brokering, which can help the customers to access the information on the Internet. The key idea behind the architecture is that the task of a human information broker should be done by a few cooperative intelligent information agents within a MAS. Chapter 8 will examine applications of intelligent agents and MASs in e-commerce.

5 A General Theory of Case Based Reasoning

This chapter is the first exception to the Boolean structure of the thesis, although it is included in Part I of the thesis, as shown in the shaded area of Fig. 5.1. It is also the basis for Chapter 6 and 7, although this is not shown in Fig. 5.1. This chapter will attempt to provide a general theory of CBR, and move CBR towards a firm theoretical foundation based on similarity-based reasoning. To this end, this chapter will first extend the concept of similarity and examine similarity relations, fuzzy similarity relations, and similarity metrics. Then it provides a theoretical formalization for building case bases with three novel algorithms. It also proposes the R^5 model for CBR. Furthermore, it examines abductive CBR and deductive CBR and proposes a knowledge-based model for integrating abductive CBR and deductive CBR. Finally, it proposes rule-based models and fuzzy rule-based models for case retrieval.

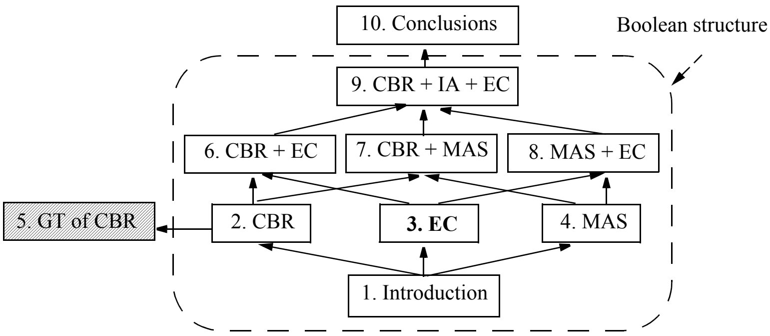


Fig. 5.1 Chapter 5 is outside the Boolean structure of PhD-thesis

5.1 Introduction

As mentioned in Chapter 2, case-based reasoning (CBR) is a reasoning paradigm that exploits analogies and similarities with previously solved problems [214]. CBR systems are a particular type of analogical reasoning system which has an increasing number of applications in different fields such as in intelligent Web-based sales service and Web-based planning as well as multiagent systems (MASs) [72][167][232]. As is well known, the goal of CBR is to infer a solution for a current problem description or enquiry in a special domain from solutions of a family of previously solved problems, the case base [72]. Theoretical and empirical works have focused among others on the definition and elucidation of similarity measures [213], on retrieving the relevant cases, on extrapolating pieces of knowledge from cases in the case base, on logical modelling of the inference mechanism [231], on empirical comparison of different similarity measures on a number of domains, and on the management of incomplete, imprecise or uncertain descriptions of cases [73][107], as well as on fuzzy set-based modelling of the inference mechanism [74][72]. For example, Dubois et al. [74][72] propose a fuzzy set-based model for basic CBR inference, and use it to treat imprecise or fuzzy descriptions in CBR. Plaza et al. [231] also introduce a PPR model (Precedent-based Plausible Reasoning) using fuzzy similarity relations. This model is based on approximation entailment and proximity entailment as well as being equipped with modal propositional logic. Unfortunately, these studies seem to view CBR as a traditional logical reasoning or fuzzy reasoning and treat CBR as intelligent retrieval, i.e. it seems that the CBR systems have degenerated into intelligent retrieval systems. Most of the CBR systems (e.g. [167][319]) do not include case base building, at least from a theoretical viewpoint. There is a lack of a theoretical treatment of CBR in a unified way, although the latter could form the basis for further robust research and development of CBR. This chapter will attempt to fill this gap by providing a unified theoretical formalization of case base building, CBR models, and case retrieval with similarity based reasoning. To this end, the rest of this chapter is organised as follows: Section 5.2 will first extend the concept of similarity given by Zadeh [348], and examine similarity relations, fuzzy similarity relations, and similarity metrics. Section 5.3 provides a theoretical formalization for building case bases with three novel algorithms. Section 5.4 proposes the R^5 model for CBR. Section 5.5 proposes integration of abductive CBR and deductive CBR. Section 5.6 and Section 5.7 propose rule-based models and fuzzy rule-based models for case retrieval. Section 5.8 concludes this chapter with some concluding remarks.

5.2 Similarity and Metrics in Case-based Reasoning

Similarity is at the heart of CBR, because case base building, case retrieval, and even case adaptation all use similarity or similarity-based reasoning[1]. However, there has been no essential development in this aspect from a theoretical viewpoint, because purely casuistic CBR systems assume that the only represented knowledge is a specific collection of cases with their solutions - plus a similarity relation [231]. In order to resolve this disadvantage, Plaza, et al. [231] indicate

1. Although connectionist reasoning and learning are often said to be similarity based, it usually involves utilizing previous similar training cases, either individually or collectively in a statistical way, this section has no intention to discuss them in this research any more. See [278] for detail.

that CBR systems also have general knowledge K about the domain of application and characterize K as the ability of the CBR system to infer new propositions about a current problem p_0 given the initial true propositions about p_0. Therefore they, in effect, only extend the case in the case base without new insight into similarity relations, although they define the similarity relation S_I on the *input* space and the similarity relation S_O on the *output* space[1]. But there are still unresolved issues; namely, where are the similarity relations defined, and where does a case come from? Dubois et al. [72] define one similarity measure on the set of problem description attribute values U and another on the set of solution attribute values V. But the relationship between U, V and the case base, C, in the associated CBR system are still unclear. In order to overcome these disadvantages, this section will first examine similarity, similarity relations, fuzzy similarity relations, similarity metrics, and possible world of problems and solutions, etc. from a theoretical viewpoint, which are all fundamental for investigating CBR, in particular case base building [293] and case retrieval [291].

Furthermore, there is some confusion using similarity, similarity measures [23], and similarity metrics in CBR, in particular in domain-dependent CBR systems. This section attempts to resolve this confusion by providing a unified framework for similarity, similarity relation, similarity measure, and similarity metric and their relationship. It also thoroughly extends the concept of similarity relations introduced by Zadeh [348] and some of the well-known results in the theory of relations to similarity metrics. Further, it introduces six different types of similarity relations and corresponding similarity metrics, one of which is Zadeh's similarity relation introduced in 1971. Such extension might be of significance in case base building and case retrieval in CBR as well as in various applied areas such as soft computing, pattern cognition, information retrieval, Web intelligent systems in which similarity plays an important role in system behaviours.

5.2.1 Introduction

The concepts of similarity and similarity relations play a fundamental role in many fields of pure and applied science [87]. The notion of a metric or distance, $d(x, y)$, between objects x and y has long been used in many contexts as a measure of similarity or dissimilarity between elements of a

1. They assume that similarity relation S_I is given, while S_O is unknown.

set. Thus, there exist a wide variety of techniques for dealing with problems involving similarity, similarity relations, similarity measures, and similarity metrics. For example, fuzzy logic, CBR, and information retrieval provide a number of concepts and techniques for dealing with similarity relations and similarity measures as well as similarity metrics, many of which are quite effective in dealing with the particular classes of problems that motivated their development.

This section does not intend to add still another technique to the vast armamentarium which is already available. The purpose of it is rather to introduce a unifying point of view based on the available theory and application of fuzzy logic [348] and CBR [172]. This is accomplished by examining the notions of similarity, similarity relations, similarity measures, and similarity metrics as well as distance functions in [73][72][172][231][316][348][355], thereby discussing the relationships between these concepts and influences on CBR. The main contribution of the proposed approach consists of providing a unified conceptual framework for the study of fuzzy similarity relations and similarity metrics (or measures), thereby facilitating research and development of CBR and fuzzy logic with their applications. The most important contribution is to extend the concept of similarity relation introduced by Zadeh in 1971 with new insight into similarity.

In what follows, this section will focus on reviewing and defining some of the basic notions with this conceptual framework and exploring their elementary implications and the relationships between them. Although the proposed approach might be of significance in areas such as pattern recognition, decision processes, intelligent information retrieval, data mining, natural language processing, Web search engine, system modelling, approximation, and multiagent systems, this section shall make no attempt to discuss its applications in these or related problem areas. This section will use CBR and e-sales as scenarios, if required.

5.2.2 Similarity Relations

Similarity is the core concept in CBR, because it is used not only in case retrieval but also in case adaptation as well as in case base building [293]. The concept of a similarity is a natural generalization of similarity between two triangles in the plane and between two matrices in mathematics [87]. More precisely:

Definition 1. A binary relation S on a non-empty set X is called a *similarity relation* provided it satisfies

(R) $\forall x, xSx$

(S) if xSy then ySx

(T) if xSy, ySz then xSz

The conditions (R), (S), and (T) are the reflexive, symmetric and transitive laws. If xSy, x and y is called similar, denoted as $p \sim q$ for convenience [252].

Example 1. Matrices B and C in $M_{n,n}$ are *similar*, denoted as $B \sim C$ if $C = PBP^{-1}$ for an invertible P. It is easy to prove that $\sim$ is a similarity relation on $M_{n,n}$ [252] (p 283).

This example implies that the concept of a similarity relation here is a generalization of the similarity between matrices in $M_{n,n}$.

Example 2. Let f be a function with domain A and codomain B; namely, $f: A \to B$, and define xSy if $f(x) = f(y)$. Then S is a similarity relation on A.

It is obvious that the similarity relation S in this example has the following property: if x_1 and x_2 are similar in the sense of S, then x_1 and x_2 have the same solution, that is, $f(x_1) = f(x_2)$. This reflects that "similar problems have the same solution" in the e-sale settings, at least in some cases. For example, in a shoe shop, the seller may put many different pairs of shoes together and sell these for the same price, i.e. \$188.00. In this case, the seller views those mentioned shoes as "similar".

Based upon the aforementioned idea, a seller agent does know "a similar query (problem) of customers has a similar answer". This is common sense in business. CBR also shares this common sense, based on the so-called Analogous Assumption: Whenever a problem description p_1 is similar to a problem description p_2 one can assume that what one can infer from p_2 is similar to being true for p_1 [231]. Now the problem arises: how to use this common sense in the practical transaction process. In order to solve this problem, it is, first of all, necessary for an agent to introduce a certain similarity relation and then use it to form a partition of the possible world of problems W_p and make the similar problems into a similarity class. In the latter part of this chapter, Section 5.6 will discuss it in more detail from a rule-based viewpoint.

It should be noted that the similarity relation proposed here is identical to the equivalence relation in discrete mathematics [252][87]. However, the former is more important than the latter in the context of CBR, because similarity relations rather than equivalence relations play an important role in CBR. Thus, this treatment is different from the idea of Zadeh [348] in that Zadeh considered a similarity relation, which is frequently cited in fuzzy literature without further consideration, as a fuzzy one and as a generalization of the concept of an equivalence relation, while this research views Zadeh's similarity relations as fuzzy similarity relations (see the next subsection). Fuzzy similarity relations are a fuzzification of a similarity relation rather than an equivalence relation [87].

5.2.3 Fuzzy Similarity Relations

As an extension of similarity relations, fuzzy similarity relations were introduced by Zadeh [348] and have attracted research attention since then [25][73][222][347][349][355]. Fuzzy similarity relations have been also used in CBR in particular in case retrieval [72][73][74][231] and case base building [293]. This subsection will examine fuzzy similarity relations from a new viewpoint. For the sake of brevity and simplicity, it uses standard fuzzy set theory notation for operations min $\wedge$, max $\vee$, although there are many alternative choices for these operations available in fuzzy set theory [355]. S is still used to denote a fuzzy similarity relation if there is not any confusion arising.

Definition 2. A fuzzy binary relation, S, on a nonempty set X is a fuzzy similarity relation[1], if it is reflexive, symmetric and transitive [222][348], i.e.,

$$S(x, x) = 1 \tag{1}$$

$$S(x, y) = S(y, x) \tag{2}$$

$$S \geq S \circ S \tag{3}$$

where $\circ$ is the composition operation of fuzzy binary relations based on $\wedge$ and $\vee$ operation.

1. The notation $S(p, q)$ is used for the membership $\mu_S(p, q)$, although the latter is commonly used in the fuzzy set literature.

A more explicit form of Eq.3 is [348]:

$$S(x, z) \geq \underset{q}{\vee} (S(x, y) \wedge S(y, z)) \tag{4}$$

The revised form of this definition was given by Ovchinnikov in 1991 [222]. Dubois and Prade [73] used the revised form for fuzzy similarity relations directly in 1994. The main difference between the definitions of Zadeh and of Ovchinnikov lies in that instead of Eq.4, Ovchinnikov viewed the following model as max-min transitivity.

$$S(x, z) \geq S(x, y) \wedge S(y, z) \tag{5}$$

This is simpler than that used by Zadeh, because if the cardinality of the set is less than or equal to 3, then Eq.4 coincides with Eq.5. This extension has some advantages, if one examines in depth the relation between similarity and metric in the Euclidean space. For detail see Section 5.2.8.

It should be noted that Eq.4 of Zadeh is a direct extension of the traditional composite relation [252]:

$$S(x, z) = \underset{q}{\vee} (S(x, y) \wedge S(y, z)) \tag{6}$$

where $\vee$, $\wedge$ are Boolean operations.

Dubois et al. [72][74] believe that in CBR, the transitivity is not always compulsory. However, in discrete mathematics [252][282], a binary relation only having reflexivity and symmetry is called a compatible relation, which is rather different from a similarity relation, in particular in partition of a nonempty set. Thus, transitivity is here necessary, because building a case base based on (fuzzy) partition requires the transitivity of a similarity relation (see Section 5.3), while Dubois et al. investigate mainly case retrieval using fuzzy similarity relations. Finally S satisfies the separating property: $\forall x, y \in X, S(x, y) = 1$ if and only if $x = y$ [72][74].

Example 3. Let $X = \{x_1, x_2, x_3\}$. Suppose a binary relation S on X is defined by

$$S = \begin{bmatrix} 1 & a & b \\ a & 1 & a \\ b & a & 1 \end{bmatrix}$$

Then S is a fuzzy similarity relation on X if and only if $a \leq b$ [222].

Zadeh considered Eq.4 as max-min transitivity based on the composition operation of fuzzy relations [348], while Ovchinnikov, Dubois and Prade used Eq.5 for max-min transitivity without any explanation. Unfortunately, they all have ignored the influence of the concept of metrics or distance on their definition of similarity relation, which will be discussed at somewhat greater length in Section 5.2.8.

It should be noted that a fuzzy similarity relation does not satisfy the traditional transitivity law, although it is quasi-transitive (or $\otimes$ transitive [72]), which, unfortunately, is too weak so that sequential use will lead to *fuzzy degeneration* [87]. In other words, if fuzzy reasoning is performed for many steps sequentially, using the traditional transitive law, the consequence will easily lose validity. For example, there are no exercises of fuzzy reasoning with ten inference (even more than one) steps in many textbooks of fuzzy logic such as [355]. Further, in the normal life one can easily say “10001 is similar to 10000” without taking membership degree into account. Here “is similar to” is a fuzzy similarity relation according to our *intuitive expectation* [348], because it reflects what we think about “similarity”. Using this similarity and the transitive law one can at once conclude “10001 is similar to 9999”, since “10000 is similar to 9999”. After having performed this similarity-based reasoning for 10000 times, one comes to the conclusion that “10001 is similar to 1". This is a *fuzzy paradox*, which leads to a *fuzzy degeneration*. If one takes membership degree into account and replaces min with product (a T-norm), then from Eq.4 the membership value of compound similarities decreases. In this example, assume that $\mu(10001, 10000) = 0.99$, then

$$\mu(10001, 9999) = t[\mu(10001, 10000), \mu(10000, 9999)] = 0.99^2$$

and finally we have $\mu(10001, 1) = 0.99^{10000} \approx 0$. Therefore, the degree of similarity between 10001 and 1 is, in essence, zero, which is the same as our intuitive expectation. However, if t-Norm is used as a min-max-function, then $\mu(10001, 1) = 0.99$ [293].

This hints that fuzzy similarity in fuzzy logic is not a real simulation of the similarity relation in human social life. The above paradox can be called “Beauty-Ugliness” paradox, because one can use this “similarity” relation to get the conclusion that “a beautiful lady is similar to a ugly woman”. This paradox and the *fuzzy degeneration* will be discussed in more detail in Section 5.2.6

5.2.4 Metric and Metric Space

A *metric space* is a nonempty set X in which a *metric* (or distance function) d is defined, with the following properties [255]:

(a) $0 \le d(x, y) \le \infty$ for all x and $y \in X$

(b) $d(x, y) = 0$ if and only if $x = y$

(c) $d(x, y) = d(y, x)$ for all x and $y \in X$

(d) $d(x, y) \le d(x, y) + d(y, z)$ for all x, y and $z \in X$.

As is well known, property (d) is called the *triangle inequality*, which is based on the property of Euclidean geometry.

The popular metric space is Euclidean space R^n, and R^1 is the real line, while R^2 is the plane in Euclidean geometry. Almost all CBR systems are based on Euclidean space R^n.

5.2.5 Similarity and the Nearest Neighbour Algorithm

This section will review similarity and the nearest neighbour algorithm, which plays a major role in CBR.

Similarity can be formalized in a relational and in a functional way [172]. The relational approach uses a four-place relation $R(x, y, u, v)$ meaning "x and y are at least as similar as u and v are." This allows the definition of the nearest neighbour notion

$$NN(x, z) \Leftrightarrow \forall y R(x, z, x, y) \tag{7}$$

meaning z is a nearest neighbour to x. If the nearest neighbour is unique, then NN is also used as a function symbol. A refinement is when k nearest neighbours are considered for some $k(k \ge 1)$.

However, similarity formalized in a relational way is a binary relation, as mentioned in Section 5.2.2. It is irrelevant to how near any two neighbours are. Nearness involves a distance concept and should be assessed by a metric or distance function[1], which will be discussed again in more detail in the following sections.

The typical Nearest Neighbour Algorithm, which was, for example, implemented in REMIND (Cognitive Systems 1992) [89][152][315], is shown in Eq.8.

1. Metric is more mathematical flavour than distance function although they are same in [255]

$$\frac{\sum_{i=1}^{n} w_i \times sim(f_i^I, f_i^R)}{\sum_{i=1}^{n} w_i} \tag{8}$$

where w_i is the importance weighting of a feature i represented as numerical values between 0 and 1. Nearer neighbours have values closer to 1 while more distant neighbours have values closer to 0. f_i^I and f_i^R are the values for feature i in the input and retrieved cases respectively, sim is the similarity function for primitives. It is this similarity function that makes the nearest neighbour algorithm different from a mathematical expectation formula, which is the generalization of arithmetic average value. However, it seems that nobody has studied how to formalize this similarity function mathematically, based on problem domains, although common CBR systems use this algorithm to perform case-based reasoning.

In the relational approach [172], similarity is treated as a partial ordering. Such partial orderings can be realized by numerical functions which are called similarity (or dual distance) measures $sim(x, y)$ or $d(x, y)$, respectively. Both similarity measures sim and distance measures d induce four-place relations R_{sim} and R_d in an obvious way. If $R_{sim}(x, y, u, v) \Leftrightarrow R_d(x, y, u, v)$ holds, R_{sim} and R_d are called compatible.

It should be noted that the relationship between similarity and distance from the above discussion is still unclear, although a new concept of similarity measure has been introduced.

In order to reduce arbitrariness some assumptions are common:

1. $0 \le sim(x, y) \le 1$
2. $sim(x, x) = 1$

The intention of (1) is normalization and (2) implies that each object is itself its nearest neighbour. This is often the case for the following conditions:

3. $sim(x, y) = sim(y, x)$ (symmetry)
4. $d(x, z) \le d(x, y) + d(y, z)$ (*triangle inequality,* in terms of distance measures)

The notion of a distance function $d(x, y)$ is dual; here, simply all orderings are reversed. For an attribute-value representation, a simple distance function is the Hamming distance. If problems are coded as n-dimensional real vectors, classical mathematical metrics like the Euclidean or the Manhattan distance are often used.

From the above discussion, it seems that there is no discussion on the relationship between similarity and metric in a mathematical way, although Richter's idea [172] that

> *from a mathematical viewpoint, the notion of similarity is equivalent to the dual distance concept. However, both notions emphasize different aspects and have given rise to different computational approaches.*

is correct. However, his thinking is still at an empirical level. Furthermore, the notion of the nearest neighbour should be directly based on the notion of either "distance" or metrics rather than on the notion of similarity, which will be discussed once again in Section 5.2.8.

5.2.6 Similarity and Metrics

In many applications like CBR, metrics are used to measure the similarity between two objects such as cases in CBR. Then it would be reasonable to assume that

$$S_1(x, y) = 1 - |x - y| \tag{9}$$

and say that x and y are similar with respect to S_1 if $|x - y| < \varepsilon$, where $|\ |$ is the Euclidean distance function and ε is a small number (in relation to $|x - y|$). But then, S_1 is not transitive from a mathematical viewpoint, which is inconsistent with our intuitive expectation that the similarity relation is transitive. However, the fuzzy similarity relation S_2 in the following example is transitive in some sense [348].

Example 4. A fuzzy similarity relation possessing transitivity. Suppose that

$$S_2(x, y) = e^{-\beta|x - y|}, \qquad x, y \in X \tag{10}$$

where β is any positive number. In the definition the max-product transitivity is employed. Under this condition, S_2 satisfies Eq.5 and therefore it is a fuzzy similarity relation.

In practice, S_1 rather than S_2 is often used to measure the similarity between two objects. Thus, two questions arises as follows:

1. What is the difference between S_1 and S_2?
2. Can we resolve the inconsistency between transitivity and our intuitive expectation [348] ?

Let us try to answer them and assume that $\beta = 1$ for convenience.

As is well known, the Taylor series expansion of the function $e^{-|x-y|}$ is [255]:

$$e^{-|x-y|} = \sum_{n=0}^{\infty} \frac{|x-y|^n}{n!} = 1 - |x-y| + |x-y|^2/2! + \dots \tag{11}$$

Then

$$S_2 - S_1 = |x-y|^2/2! + \dots = O(|x-y|^2) \tag{12}$$

Thus, if ε is chosen as small as possible, i.e. p and q are quite similar, almost equal, then the difference between S_1 and S_2 can be insignificant. In other words, the inconsistency between transitivity and our intuitive expectation results from the insignificant difference in Eq.12; that is, $O(|x-y|^2)$. Therefore, S_1 can be referred to a fuzzy similarity relation and used to perform similarity-based reasoning with transitivity only if the number of transitive reasoning steps is limited. In this case, the similarity-based reasoning with S_1 can not lead to obviously *fuzzy degeneration* mentioned in the previous section (Section 5.2.3) because of Eq.12. This result can be also easily extended as follows: Let

$$S_3(x, y) = 1 - cd(x, y), \tag{13}$$

and say that x and y are similar with respect to S_3 if $d(x, y) < \varepsilon$, where d is a metric, $0 < c < 1$ is a constant, and ε is a small number (in relation to $d(x, y)$).

5.2.7 Similarity Metrics

This subsection will discuss similarity metrics in CBR based on integration of similarity relations and metrics mentioned. It argues that it is similarity metrics rather than similarity measures that should be used to assess the similarity between problems or between solutions in CBR. It begins with the following example borrowed from [278].

According to [278], similarity $\sim$ is a measure of the amount of overlap between the corresponding feature sets, F_A and F_B, of the source and target concepts or propositions, A and

B, as shown in Fig. 5.2. More specifically, the similarity measure between A and B is: $A \sim B = \frac{|F_A \cap F_B|}{|F_B|}$. It is easy to show that $\sim$ here is only reflexive. However, it is neither symmetric nor transitive. The reason is that a similarity measure should not be made to be either symmetric or transitive, according to Sun [278]. However, according to discrete mathematics [252], $\sim$ can only be considered as not a similarity relation but a reflexive relation. Further, Burkhard [36] and Dubois [72] believe that similarity is often considered as a symmetric measure, although in practice, what "similarity" is maybe asymmetric in daily usage.

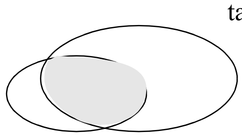


Fig. 5.2 Similarity in [278]

Generally speaking, similarity in mathematics is considered as a relation, while similarity in CBR is considered as both a relation [72] and a measure [28][36][278] as well as a metric [20]. This confusion between similarity relation, similarity measure [28][272][316], and similarity metric [20][316] is so popular that these three concepts are *de facto* the same in CBR. However, there is still no general definition for the concept of a similarity metric in CBR, although many CBR publications are involved in it [168][172][272][316]. No one seems to have any idea about how differentiate these concepts or what the relationship is between them, although Burkhard and Richter [36][172] have been aware of the difference between similarity measure and distance or metric (see Section 5.2.5), and given some useful formalization. However, they have not examined them thoroughly. In what follows, this section will fill this gap.

Briefly speaking, measures assess the size of any subset in a mathematical system, e.g. a Borel field [63], while metrics evaluate the distance between any two elements in a mathematical system, e.g. a Banach space [255]. In Euclidean space R^2, a measure can be considered as the generalization of the notion of area, while a metric can be viewed as the distance between two points. In Euclidean space R^1, a measure can be considered as the generalization of the notion of *"length" of any interval* (if the interval I has endpoints x and y, $x \le y$, then the length of I is

$l(I) = y - x$ [255]) and as a function from a certain subset of the power set of R^1 to $[0, \infty)$, while a metric d is a function $R^1 \times R^1 \to R^1$ with some conditions. Thus, measures and metrics are two different concepts. Because one uses it to assess the similarity degree between the features of two cases, he should use a similarity metric rather than a similarity measure to investigate the similarity involved in CBR. To perform a theoretical analysis, the following is needed:

Definition 3. A relation, denoted by S_m, on non-empty X, is a similarity metric if it satisfies

1. S_m is a similarity relation in X
2. $1 - S_m$ is a metric on X; that is, it is a function from $X \times X$ to $[0, 1]$, provided that

- for any $x, y \in X$, $S_m(x, y) = 1$ if and only if $x = y$
- for all $x, y \in X$, $S_m(x, y) = S_m(y, x)$
- for all $x, y, z \in X$,

$$S_m(x, z) \geq S_m(x, y) \wedge S_m(y, z) \tag{14}$$

where $\wedge$ is min operator. Eq.14 in this definition is called the *similarity inequality*. It should be noted that the similarity metric here, S_m, can not directly satisfy the *triangle inequality* (see Section 5.2.8). Eq.14 is motivated by the concept of fuzzy similarity relations given in [222], which is as Eq.5 in this section.

In comparison with the definition of fuzzy similarity relations given in [72][222] and in Section 5.2.3, the similarity metric here is at first a traditional similarity relation, and then just a metric (maybe to some extent), because the similarity between two elements is the necessary condition to further discuss how similar they are, which coincides with our intuitive expectation [87]. In practice, our first concern is whether x and y are similar, then we ask how similar they are. In fact, in some cases, such as case base building [293], (fuzzy) similarity relations rather than similarity metrics are essential. However, metrics and in particular similarity metrics play an important role in case retrieval in CBR [291]. Therefore, the integration of similarity relations and metrics into similarity metrics is of practical significance.

5.2.8 Relationships between Similarity Metrics and Traditional Metrics

Usually similarity metrics are used to evaluate the similarity between two cases in CBR [291]. The question arises: what is the relationship between the proposed similarity metric and the traditional metric. In what follows, this section discusses this at somewhat greater length and at the same time extends the concept of similarity introduced by Zadeh in 1971 [348].

Suppose that S_m on a non-empty X is a similarity metric. Let

$$d(x, y) = 1 - S_m(x, y), \forall x, y \in X \tag{15}$$

Then it holds, based on Definition 3:

- For any $x \in X$, $d(x, x) = 1 - S_m(x, x) = 0$
- For any $x, y \in X$, $d(x, y) = 1 - S_m(x, y) = 1 - S_m(y, x) = d(y, x)$, and
- For any $x, y, z \in X$, using Eq.14,

$$\begin{aligned} & d(x, z) \\ & = 1 - S_m(x, z) \le 1 - (S_m(x, y) \wedge S_m(y, z)) \\ & = (1 - S_m(x, y)) \vee (1 - S_m(y, z)) \\ & = d(x, y) \vee d(y, z) \le d((x, y) + d(y, z)) \end{aligned}$$

That is,

$$d(x, z) \le d(x, y) + d(y, z) \tag{16}$$

triangle inequality is valid. Thus [87]:

Proposition. $1 - S_m$ is a metric or distance function.

This proposition demonstrates that the similarity inequality implies the triangle inequality indirectly. On the other hand, the triangle inequality is the generalization of the R^2 property that the "sum of the lengths of any two sides of a triangle is greater than the length of the remaining side" [63], demonstrated in Fig. 5.3. However, define

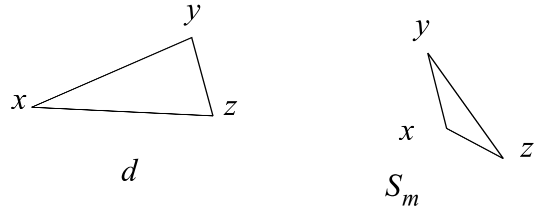


Fig. 5.3 Triangle inequality and similarity inequality

$$S_m(x, y) = 1 - d(x, y), \forall x, y \in X, \tag{17}$$

then one can easily find that the longest edge $d(x, z)$ in the sense of distance function d becomes the shortest edge $S_m(x, z)$ in the sense of S_m and R^2 (see Fig. 5.3). This characteristic leads to consider

$$S_m(x, z) \le S_m(x, y) \wedge S_m(y, z) \tag{18}$$

as an important feature in the similarity metric, when $d(x, z) \ge d(x, y) \vee d(y, z)$ demonstrated in Fig. 5.3 In other cases; that is, $d(x, z) \le d(x, y) \vee d(y, z)$, Eq.18 is not valid, for example, when the distance of $d(x, z)$ is the shortest among them. However, in such cases, similarity inequality Eq.5 or Eq.14 is satisfied. This result differs from that of fuzzy similarity relations [73][74][222][348]. Thus, it is necessary to examine the relationship between $d(x, z), d(x, y), d(y, z)$ and $S_m(x, z), S_m(x, y), S_m(y, z)$ in a unified way. Taking the commutativity of $\vee$, $\wedge$ into account, it is sufficient to consider the following three cases:

1. $d(x, z) \ge d(x, y) \vee d(y, z)$
2. $d(x, y) \le d(x, z) \le d(y, z)$
3. $d(x, z) \le d(x, y) \wedge d(y, z)$

In what follows, each of these cases will be considered in more detail. The consideration of their relationship with identity relation “=” is left to the readers.

From the first case,

$$S_m(x, z) \le S_m(x, y) \wedge S_m(y, z) \tag{19}$$

which leads to

$$S_m(x, z) \le S_m(x, y) \vee S_m(y, z), \tag{20}$$

because $S_m(x, y) \wedge S_m(y, z) \le S_m(x, y) \vee S_m(y, z)$. It should be noted that Jacas and Valverde [129] extended Eq.20 to define S-triangular inequality and then a S-pseudometric.

From the second case, the following holds:

$$S_m(x, z) \le S_m(x, y) \vee S_m(y, z) \tag{21}$$

but $S_m(x, z) \le S_m(x, y) \wedge S_m(y, z)$ is not valid, although the following is valid:

$$S_m(x, z) \geq S_m(x, y) \wedge S_m(y, z) \tag{22}$$

As mentioned early, Eq.22 is, in essence, used to define the transitivity in the fuzzy similarity relations by Ovchinnikov [222]. This means that Ovchinnikov [222] only considered one of three mentioned cases, when he defined his concept of similarity.

In the third case, neither

$$S_m(x, z) \leq S_m(x, y) \wedge S_m(y, z) \text{ nor } S_m(x, z) \leq S_m(x, y) \vee S_m(y, z) \tag{23}$$

hold. However, from this case, the following holds:

$$S_m(x, z) \geq S_m(x, y) \vee S_m(y, z) \tag{24}$$

Eq.24 is, in essence, used to define the transitivity in the fuzzy similarity relations by Burkhard [36], and leads to

$$S_m(x, z) \geq S_m(x, y) \wedge S_m(y, z) \tag{25}$$

These results imply that the definition of fuzzy similarity relation in [222] is irrelevant to the triangle inequality. Because the definition of fuzzy similarity relation in [222] is a simpler form of the definition of a similarity relation given by Zadeh [348], the latter is also irrelevant to the triangle inequality. In fact, the definition of fuzzy similarity relations in [222][348] only reflects one of the above results; that is, Eq.25. Therefore four different types of fuzzy similarity relation concepts based on Eq.19, 20, 24, and 25 respectively will be introduced, following the discussion of Zadeh [348], Dubois [73], and Ovchinnikov [222], as follows.

Definition 4. A fuzzy binary relation S on a nonempty set X is a *type-1 fuzzy similarity relation*[1], if it is reflexive, symmetric, and type-1 transitive; i.e.,

$$S(x, x) = 1 \tag{26}$$

$$S(x, y) = S(y, x) \tag{27}$$

$$S(x, z) \leq \bigvee_q (S(x, y) \wedge S(y, z)) \tag{28}$$

Definition 5. A fuzzy binary relation S on a nonempty set X is a *type-2 fuzzy similarity relation*, if it is reflexive, symmetric, and type-2 transitive; i.e.,

1. This research uses the notation $S(p, q)$ for the membership $\mu_S(p, q)$, although the latter is commonly used in the fuzzy set literature.

$$S(x, x) = 1 \tag{29}$$

$$S(x, y) = S(y, x) \tag{30}$$

$$S(x, z) \leq \underset{q}{\vee} S((x, y) \vee S(y, z)) \tag{31}$$

Definition 6. A fuzzy binary relation S on a nonempty set X is a *type-3 fuzzy similarity relation*, if it is reflexive, symmetric, and type-3 transitive; i.e.,

$$S(x, x) = 1 \tag{32}$$

$$S(x, y) = S(y, x) \tag{33}$$

$$S(x, z) \geq \underset{q}{\vee} S((x, y) \vee S(y, z)) \tag{34}$$

Definition 7. A fuzzy binary relation S on a nonempty set X is a *type-4 fuzzy similarity relation*, if it is reflexive, symmetric, and type-4 transitive; i.e.,

$$S(x, x) = 1 \tag{35}$$

$$S(x, y) = S(y, x) \tag{36}$$

$$S(x, z) \geq \underset{q}{\vee} S((x, y) \wedge S(y, z)) \tag{37}$$

Each of these four types of similarity concepts corresponds to the generalization or induction of some cases in nature and society. The similarity relation introduced by Zadeh [348], Dubois [73], and Ovchinnikov [222] is type-4, while the similarity introduced by Valverde [129] is type-2. Their results based on either type-2 or type-4 similarity relations can be extended to the cases of other two types of similarity relations, in order to obtain a complete study on all kind of similarity relations. Further, four different types of similarity metric can also be introduced in a similar way, each of which corresponds to one of the above-mentioned similarity relations. They are also called *type-1 similarity metric*, *type-2 similarity metric*, *type-3 similarity metric*, and *type-4 similarity metric*, denoted as S_{m1}, S_{m2}, S_{m3}, and S_{m4} respectively.

Based on Eq.28, 31, 34, and 37, it is easy to know that S_{m1} is stronger than S_{m2}, while S_{m4} is weaker than S_{m3} from a viewpoint of similarity.

We have already come to two points: The first is that we hope to define the similarity metric as not only a similarity relation from a traditional viewpoint but also as a metric or distance function, to some extent. The second is that we have found that Eq.19, 20, 24, and 25 are valid respectively under different conditions if taking Euclidean space into account. These two results suggest to further introduce:

Definition 8. A relation, denoted by S_m, on a non-empty X, is a similarity metric if it satisfies

1. S_m is a similarity relation in X
2. $1 - S_m$ is a metric on X; that is, it is a function from $X \times X$ to $[0, 1]$, provided that

- for any $x, y \in X$, $S_m(x, y) = 1$ if and only if $x = y$
- for all $x, y \in X$, $S_m(x, y) = S_m(y, x)$
- for all $x, y, z \in X$, either

$$S_m(x, z) \le S_m(x, y) \wedge S_m(y, z) \text{ or } S_m(x, z) \ge S_m(x, y) \vee S_m(y, z) \tag{38}$$

where $\wedge$ is min operator. Eq.38 in this definition is called the *similarity inequality.*

However, it is not easy to verify if the similarity inequality (Eq.38) is valid from a pragmatic viewpoint. Therefore, we introduce the following:

Definition 9. A relation S_m on a non-empty X, is a type-5 similarity metric if it satisfies

1. S_m is a similarity relation in X
2. 1- S_m is a metric on X, that is, it is a function from $X \times X$ to $[0, 1]$, provided that

- for any $x, y \in X$, $S_m(x, y) = 1$ if and only if $x = y$
- for all $x, y \in X$, $S_m(x, y) = S_m(y, x)$
- for all $x, y, z \in X$,

$$S_m(x, z) \ge S_m(x, y) + S_m(y, z) - 1 \tag{39}$$

where $\wedge$ is min operator. Eq.39 in this definition is also called the *similarity inequality.* Eq.39 is also a variant of the following equation

$$1 - S_m(x, z) \le (1 - S_m(x, y)) + (1 - S_m(y, z)) \tag{40}$$

Therefore it satisfies the *triangle inequality.*

Similar to the previous discussion, Eq.39 can also be used to correspondingly define a *type-5 similarity relation*.

The previously discussion is, in essence, based upon the *triangle inequality* in Euclidean space R^2. In fact, there is another property in Euclidean space R^2, which is parallel to the *triangle inequality*; that is, "difference of the lengths of any two sides of a triangle is less than the length of the remaining side". More formally,

$$d(x, z) \geq d(x, y) - d(y, z) \tag{41}$$

Assume

$$S_m(x, y) = 1 - d(x, y), \forall x, y \in X \tag{42}$$

Then, based on Eq.41, it holds:

$$S_m(x, z) \leq S_m(x, y) - S_m(y, z) + 1$$

Therefore, another definition of a similarity relation and a similarity metric can be introduced, called the *type-6 similarity relation* and *type-6 similarity metric* respectively, for example,

Definition 10. A relation S_m on non-empty X, is a *type-6 similarity metric* if it satisfies

1. S_m is a similarity relation in X
2. $1 - S_m$ is a metric on X; that is, it is a function from $X \times X$ to $[0, 1]$, provided that

- for any $x, y \in X$, $S_m(x, y) = 1$ if and only if $x = y$
- for all $x, y \in X$, $S_m(x, y) = S_m(y, x)$
- for all $x, y, z \in X$,

$$S_m(x, z) \leq S_m(x, y) - S_m(y, z) + 1 \tag{43}$$

where $\wedge$ is min operator. Eq.43 in this definition is also called the *similarity inequality*.

As is known, metrics have played a vital role in mathematics and engineering, in particular in functional analysis and engineering computation, while similarity plays a similar role in many fields in computer science, such as CBR, IR, and pattern recognition. However, there is no theoretical insight into similarity metrics. Almost all similarity metrics and measures for "neighbourhood" are domain dependent. There is also a misunderstanding about similarity measures and similarity metrics in CBR. The proposed results basically fill this gap. The six different types of similarity metrics introduced here can be used easily in CBR. In the following context, a fuzzy similarity relation or similarity metric is always referred to one of the proposed six different types of fuzzy similarity relations and corresponding similarity metrics, if it is not mentioned specifically.

5.2.9 Local Similarity vs Global Similarity

Let p_1 and p_2 be two problems in the possible world of problems W_p, in which every problem has n (feature) attributes. The attribute-value representation of the problem in W_p can be taken as a n-tuple vector; that is:

$$p_1 = \{u_1, \ldots, u_n\}$$

$$p_2 = \{v_1, \ldots, v_n\}$$

For every $i \in \{1, \ldots, n\}$, there is a similarity metric S_{m_i} on the domain of attribute A_i; that is, $S_{m_i}: A_i \to R^1$, S_{m_i} is called a local similarity metric, and $S_{m_i}(u_i, v_i)$ is the similarity degree between u_i and v_i.

Besides those mentioned in Section 5.2.8, typical general-purpose local similarity metrics are the following *Canberra metrics* (Eq.44, 45, and 46)[71]

$$S_{m_i}(u_i, v_i) = 1 - \frac{|u_i - v_i|}{|u_i| + |u_i|} \tag{44}$$

$$S_{m_i}(u_i, v_i) = 1 - \frac{|u_i - v_i|}{|max(u_i, v_i)|} \tag{45}$$

$$S_{m_i}(u_i, v_i) = 1 - \frac{|u_i - v_i|}{max(u_i, v_i) - min(u_i, v_i)} \tag{46}$$

Strictly speaking, the above discussion basically belongs to local similarity, which deals with the values of an individual attribute or feature of a problem. However, a problem/solution description has a number of attributes in a CBR system. Therefore how to get an overall similarity assessment for a problem/solution description based on the local similarity assessment is an important part in CBR.

The evaluation of global similarity between two multiple-feature descriptions is obtained by aggregating the evaluation of local similarities for each feature [72]. The aggregation has to be done in such a way that the resulting similarity relation should preserve properties, like reflexivity, symmetry and possibly a certain kind of transitivity of the local or individual similarities. In what follows, the section will examine the theoretical global similarity based on local similarity discussed above from the viewpoint of CBR.

Let f be a composite function from $R \times \ldots \times R$ to R. Then the global similarity degree of p_1 and p_2, $S_g(p_1, p_2)$, can generally be considered as (see [36][71])

$$S_g(p_1, p_2) = f(S_{m_1}(u_1, v_1), \ldots, S_{m_n}(u_n, v_n)) \tag{47}$$

where S_g is a global similarity metric. If f is a linear function such that

$$S_g(p_1, p_2) = \sum_1^n w_i \cdot S_{m_i}(u_i, v_i) \tag{48}$$

where w_i is the weighted value of attribute A_i, which reflects the relative importance of corresponding A_i within the problem in W_p and satisfies $w_i \in [0, 1]$ and $\sum_1^n \boldsymbol{w_i} = 1$ (normalized weights), Eq.48 is called the weighted Hamming similarity metrics, because its form is essentially the same as the weighted Hamming distance.

Another popular (weighted) global similarity metric is the Euclidean similarity as follows:

$$S_g(p_1, p_2) = \sqrt{\sum_1^n w_i^2 \cdot S_{m_i}^2(u_i, v_i)}, \tag{49}$$

owing to that its form is essentially the same as the traditional Euclidean distance.

From the viewpoint of fuzzy logic, the following global similarity metrics are useful when dealing with the global similarity assessment of problems with incomplete knowledge and uncertainty.

$$S_g(p_1, p_2) = \bigvee_i S_{m_i}(u_i, v_i), \tag{50}$$

$$S_g(p_1, p_2) = \bigwedge_i S_{m_i}(u_i, v_i), \tag{51}$$

where $\vee$ and $\wedge$ are max and min operators in fuzzy logic [355].

5.2.10 Summary

This section examined similarity relations, similarity measures, similarity metrics, and distance functions in a unified way and built an important relationship between similarity metrics and Euclidean metrics. It examined similarity measures, similarity metrics and distance functions in a novel way and extended the concept of similarity relation introduced by Zadeh et al. and proposed six different types of fuzzy similarity relations and corresponding similarity metrics, each of which corresponds to some cases in nature and society. The core idea behind this is the integration of similarity relations and metrics into similarity metrics, based on investigation of similarity relations and traditional metrics used in CBR. Another original idea is to understand the relation between transitivity in fuzzy similarity relations and the triangle inequality in the plane from a new viewpoint, which has been ignored by other researchers. Because similarity measures and metrics are frequently used to assess the similarity between two objects in a confused way, it is better to use similarity metrics rather than similarity measures in CBR. It also argued that similarity metrics proposed in the last subsection can be easily used in CBR. The preceding analysis extended some of the well-known results in the theory of relations to similarity metrics. It appears that such extension may be of use in case base building and case retrieval in CBR as well as in various applied areas in which similarity plays an important role in system behaviour such as database, data mining, and Web search engine.

5.3 Case Base Building with Similarity Relations

While the theory and practice of CBR has benefited greatly from recent advances in case representation, similarity assessment, case retrieval, and case adaptation, there have been only modest advances in case base building, in particular from a formal viewpoint. This section will show that case base building can be based on both similarity relations and fuzzy similarity relations, and then present three algorithms for case base building. Thus case base building is a form of similarity-based reasoning.

5.3.1 Introduction

Most CBR systems (e.g. [167][231]) do not include case base building, at least from a theoretical viewpoint, although this is the foundation for performing case retrieval and then case adaptation. This section will attempt to fill this gap by providing a theoretical formalization of building the case base in CBR and presenting three algorithms for case base building. In contrast to the popular use of similarity relations in case retrieval, e.g. [72][231], the proposed approach introduces a similarity relation and fuzzy similarity relation for partitioning the possible world of problem descriptions and the possible world of solutions. It then creates the case base of a CBR system and argues that the case base can be built using both similarity relations and fuzzy similarity relations with three proposed algorithms. Therefore case base building is a form of similarity-based reasoning.

This section is organized as follows: Section 5.3.2 introduces the possible world of problems and solutions. Section 5.3.3 investigates similarity classes on the possible world of problems and solutions. Section 5.3.4 discusses cases and case base building. Section 5.3.5 examines the refinement of case bases. Section 5.3.6 investigates case base building based on fuzzy similarity relations. Section 5.3.7 ends this section with a few concluding remarks.

5.3.2 Possible World of Problems and Solutions

This subsection discusses the case base of a CBR system in a broader domain; that is, the possible world of problems and the possible world of solutions.

After the failure of GPS (general problem solver) in early AI to capture general purpose reasoning or intelligence, intelligent systems have only served to solve certain types of problem in a special field or in a narrow *domain* [293]. Any CBR system can thus only give the answers to problems in a *possible world*[1], which corresponds to a real world scenario. Based on this idea, the *possible world of problems*, W_p, and the *possible world of solutions*, W_s, are the whole world of an agent (see [214]) to use CBR to do everything that he can. If an agent considers a CBR system as a function or transformation h from W_p to W_s, it is meaningless to discuss the image of $h(x)$ if $x \notin W_p$. Therefore, the agent can only know and play in the world $W_p \times W_s$, shown in Fig. 5.4 For example, in a CBR e-sale system, the possible world of problems W_p can consist of

- Properties of goods
- Normalized queries of customers
- Knowledge of customer behaviour

1. This term is affected by the terminology in modal logic and AI [214].

- General knowledge of business (similar to K in [231])
- etc.

and the possible world of solutions W_s consists of

- Price of goods
- Customized answers to the queries of customers
- General strategies for attracting customers to buy the goods
- etc.

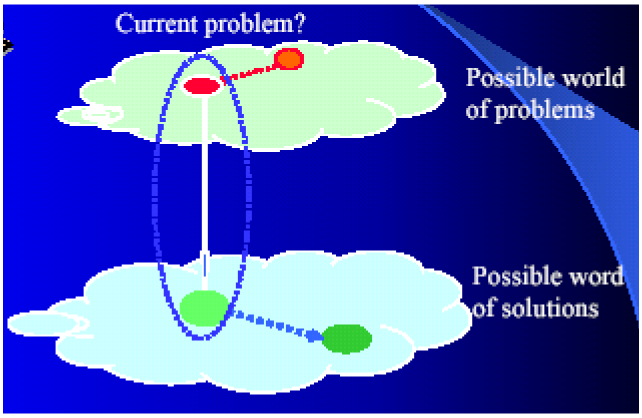


Fig. 5.4 CBR World

It should be noted that if the case base in the CBR system is denoted as $C = (P, Q)$, where P is the subset of problem descriptions and Q is the subset of solution descriptions, then it is obvious that P and Q are subsets of W_p and W_s respectively [293]. For example, in the above mentioned CBR e-sale system, W_p can represent all possible goods on the market, while P is only all available products (goods) for customers in this system. Further, the requirement of a customer or buyer agent is a desirable good p_0, which may not be in P but in W_p. However, this point is not clear in Leake's model (including problem space and solution space) of CBR (see Section 2.3.3). Therefore, the introduction of possible world of problems and possible world of solutions is of significance, which will be seen more in the following.

5.3.3 Similarity Classes on the Possible World

Definition 11. let S be a similarity relation on W_p. For each $p \in W_p$, define

$$[p] = \{q | pSq, q \in W_p\} \tag{52}$$

$[p]$ is called a *similarity class* containing p and p a representative element[1] of $[p]$. The set of all similarity classes of W_p is denoted by $[W_p]$.

As is known, there is a one-to-one correspondence between a similarity relation on W_p and a partition of W_p [252][282]; namely, if S is a similarity relation on W_p, then $[W_p] = \{[p] | p \in W_p\}$ is a partition of W_p, denoted by W_p / S. Conversely, if $\{A_i\}$ is a partition of W_p, then the sets A_i are the similarity classes corresponding to some similarity relation on W_p. In other words, any similarity relation on W_p determines a corresponding partition of W_p. Thus, in terms of reasoning, partitioning of sets can be viewed as similarity-based reasoning.

Example 5. Let f be a function with domain W_p and codomain W_s, namely, $f: W_p \to W_s$, and define pSq if $f(p) = f(q)$, where = means identity between two elements in W_s. Then S is a similarity relation on W_p and the similarity classes are the nonempty sets $f^{-1}(s)$, where $s \in W_s$.

It is obvious for this example that for any similarity class $[p]$ with respect to S, if $p_1, p_2 \in [p]$ then p_1 and p_2 have the same solution; that is, $f(p_1) = f(p_2)$.

Finally, in contrast to the nearest neighbour algorithm mentioned previously in Section 5.2.5, a most similar problem model (MSPM) for case retrieval in CBR will be proposed as follows:

Let S_m[2] be a similarity metric on the possible world of problems W_p, p_0 be a current problem (a normalized enquiry) and similar to $p \in W_p$ in the sense of S_m. Then the most similar problem p_{most} is the problem that satisfies:

$$max\{S_m(p_0, q), \forall q \in [p]\} \tag{53}$$

where $[p]$ is the similarity class with the representative p.

1. In practice, one element from a similarity class is chosen as the representative element.
2. It is one of six different types of similarity metrics.

5.3.4 Case and Case Base Building

This subsection will investigate cases and case bases based on similarity relations on the possible world of problems W_p and similarity relations on the possible world of solutions W_s. This differs from other studies, in which similarity relations are mainly used to treat case retrieval [72][231].

In many studies such as [72][172], cases are denoted as $n + m$-tuples of completely, incompletely or fuzzily described attribute values, this set of attributes being divided in two non-empty disjoint subsets, i.e. the subset of problem description attributes (n-tuples) and the subset of solution or outcome attributes (m-tuples), denoted by P and Q respectively. A case, c, can be denoted as an ordered pair (p, s), where $p \in P$ and $s \in Q$. The case base C is the set of (stored) known cases [74]. Unfortunately, such studies neglect the relationship between C and W_p. One can imagine that the seller agent in the selling process always classifies the products and customers using his special "similarity relation" before he performs the mentioned "a similar query of customers has a similar answer". This suggests that the relationship between C and W_p is important. The classification performed by the seller agent can be considered as a partition of the possible world of problems W_p, which can be realized based on the similarity relation. That is, let a relation S on W_p be a *similarity* relation. Then $[W_p] = \{[p] | p \in W_p\}$ is a partition of W_p with respect to S. Furthermore, for any two problems $p_1, p_2 \in [p]$, p_1 is similar to p_2 with respect to similarity relation S on W_p, and they can have similar, or in particular, the same solutions in the possible world of solutions W_s. In such a way, it is sufficient to choose the representative element $p \in [p]$ and its corresponding solution $s \in W_s$ to constitute a case $c = (p, s)$ and store it in the case base. Therefore, we conclude that

- A case c in the case base of a CBR system consists of a representative element p of a similarity class $[p]$ in terms of similarity relation S on W_p and its corresponding solution s in the possible world of solutions W_s, denoted as $c = (p, s)$

- The case base is made up of the representative elements p_i of all disjoint similarity classes in the partition of W_p in terms of S and their own corresponding solution[1] s_i in the possible world of solutions W_s; that is:

$$C = \left\{ (p_i, s_i) \middle| \bigcup_1^n [p_i] = W_p, [p_i] \cap [p_j] = \varnothing, \text{if } i \neq j, i, j \in \{1, \ldots, n\} \right\} \tag{54}$$

where $s_i \in W_s$ is a solution of p_i. Define $P = \{p_i | (p_i, s_i) \in C\}$, $Q = \{s_i | (p_i, s_i) \in C\}$. Then P is the set of precedent problem descriptions, and Q is the set of solution descriptions. C is a case base with respect to the partition $[W_p] = \{[p] | p \in W_p\}$.

This result, shown in Fig. 5.5, is also based on the following idea: one classifies similar

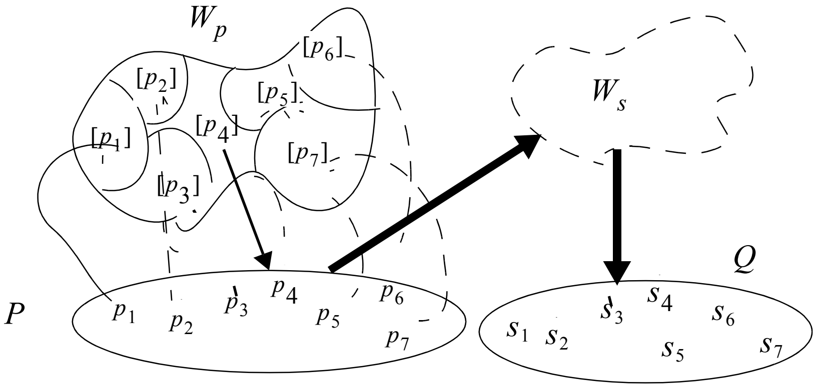


Fig. 5.5 From W_p and W_s to case base C-I

problems into a class, then finds a representative problem from this class and solves it thoroughly in order to "get twice the result with half the effort". If one finds out the solution to the representative problem, then he can use this solution to solve all other problems in that similarity class including the mentioned representative problem. It is reasonable to define the similarity relation on W_p rather than on P, which is a part of the case base. Now, the above discussion can be summarized as an algorithm-I, which creates a case base for a CBR system using a similarity relation in W_p:

Step 1. Define the possible world of problems W_p and the possible world of solutions W_s

1. If there are more than one solution, one of them can be selected as s_i.

Step 2. Define a similarity relation S on W_p

Step 3. Find the partition of W_p with respect to S

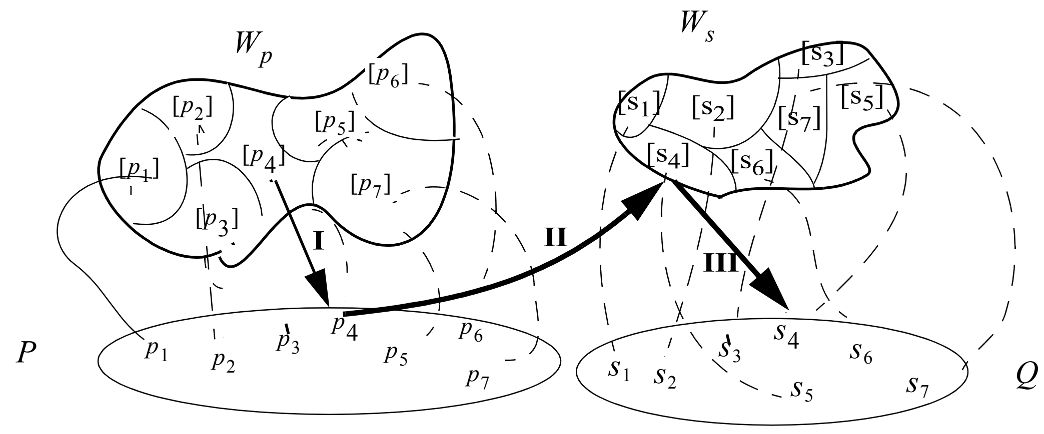
Step 4. Find the representative elements p_i of all disjoint similarity classes in the partition of W_p in terms of S and then constitute them into a set of precedent problem descriptions P

Step 5. For every representative element $p_i \in P$ find its corresponding solution s_i in the possible world of solutions W_s. The set of all the corresponding solutions from this step is called the set of solution descriptions Q

Step 6. Create the case base $C = (P, Q)$

Step 7. End.

It is worth noting that there is, in practice, a similarity relation, T, on W_s, too, which is motivated by [74][72]. Thus a representative of a similarity class in W_p, e.g. p_i is mapped to an adequate representative of an similarity class in W_s, e.g. s_i. For case retrieval, given a problem or an enquiry $p_0 \in W_p$, one firstly decides which similarity class $[p_i]$ that p_0 belongs to, then goes to the possible world of solutions W_s and looks up an appropriate solution s_i in all possible similar solutions $[s_i]$. For case base building one can generalize from the concrete class of problems (i.e. find the representative of a similarity class), then look for all possible similar solutions in the possible world of solutions and then generalize from the similarity class of solutions, i.e. find a representative. However, the similarity relation S on the possible world of problems W_p has to be defined in advance. The similarity relation T on the possible world of solutions W_s depends on the similarity classes in the possible world of problems W_p: From each similarity class $[p_i]$ a representative p_i is chosen. For each representative p_i, we find a set of possible solutions (similar solutions) in the possible world of solutions, $\{s_{ij} | j \in J\}$. If these sets are disjoint they give a partition of W_s, i.e. $\{s_{ij} | j \in J\} = [s_i]$, which corresponds to a similarity relation, called T, on W_s. Finally we choose a representative s_i from $[s_i]$. All the pairs (p_i, s_i) constitute the case base. Therefore, Fig. 5.5 should be changed into Fig. 5.6.



I = generalization, II = search solutions, III = generalization

Fig. 5.6 From W_p and W_s to case base C-II

The algorithm-I can be also slightly extended as algorithm-I* which creates a case base for a CBR system using similarity relations on W_p and on W_s.

Step 1. Define the possible world of problems W_p and the possible world of solutions W_s

Step 2. Define a similarity relation S on W_p

Step 3. Find the partition of W_p with respect to S

Step 4. Find the representative elements p_i of all disjoint similarity classes in the partition of W_p in terms of S and then constitute them into a set of precedent problem descriptions P

Step 5. For every representative element $p_i \in P$ find all possible solutions (similar solutions) in the possible world of solutions, $\{s_{ij} | j \in J\}$

Step 6. Define a similarity class $\{s_{ij} | j \in J\} = [s_i]$ of a certain similarity relation T on W_s and decide a partition of W_s, i.e.

$$W_s = \bigcup_{i=1}^{n} [s_i] \cup \left(W_s - \bigcup_{i=1}^{n} [s_i] \right) \tag{55}$$

Step 7. Choose a representative s_i in similarity class $[s_i]$ of a partition of the possible world of solutions W_s. The pair (p_i, s_i) becomes a case in the case base. The set of all the corresponding solutions from this step is called the set of solution descriptions Q

Step 8. Combine P and Q into the case base $C = (P, Q)$

Step 9. End.

So far, the relationship between similarity relations in the possible world of problems and similarity relations in the possible world of solutions has been investigated. It also argued that case base building in a CBR system can be a process of similarity-based reasoning based on algorithm-I and algorithm-I*. The next subsection will extend the proposed algorithm to a "recursive" algorithm, owing to the refinement and adjustment of the partition of W_p, and demonstrate that case base building in a CBR system is a cyclic process of similarity-based reasoning.

5.3.5 Refining Case Bases

It appears that case bases are very domain dependent with the result that there are no studies and in particular no theoretical studies on the refinement or improvement of case bases. This section attempts to give some new insight into this question.

It is obvious that many similarity relations can be defined on W_p. Further, if necessary, a *similarity relation base* can be built, similar to a case base, in order to implement a CBR system. Different similarity relations on W_p lead to different partitions of W_p and then form different case bases. A new problem arises owing to different similarity relations: Which of these different similarity relations is better in practice, for example, for the e-sale business. This question has been neglected in CBR and fuzzy reasoning, with no studies on the comparison of similarity relations in both fields. This section discusses it in some detail from an algebraic viewpoint.

Definition 12. Let $\{A_i\}$ and $\{B_j\}$ be two partitions of W_p. Partition $\{A_i\}$ is called *finer* than $\{B_j\}$ if for every $A_k \in \{A_i\}$ there exists a set B_j such that $A_k \subseteq B_j$. $\{B_j\}$ is called *coarser* than partition $\{A_i\}$, if $\{A_i\}$ is *finer* than $\{B_j\}$.

According to this definition, it is obvious that if the similarity relation S on W_p determines the coarsest partition of W_p; that is, $[W_p] = \{W_p\}$, then $P = \{p_0\}$, where p_0 is any given element in W_p. In this extreme case, the case base will have only one case. Thus it is not a real case base in any existing CBR system. In another extreme case, the similarity relation S on W_p determines the finest partition of W_p; that is, $[W_p] = \{[p] = \{p\} | p \in W_p\}$. This means that every single element in W_p forms a similarity class with respect to S. In this case, $P = W_p$. Therefore, the case base is the largest and provides a corresponding solution to every problem in the possible world W_p. This is, in general, not feasible in any existing CBR system, because it would require full understanding of all problems. Usually, any partition corresponding to a similarity relation involved in CBR research and development lies between these two extremes. One can examine if this partition of W_p is finer than another one based on Definition 12. In practice, it is worth refining a partition corresponding to the similarity relation on W_p if the existing case base is not satisfactory[1] based on the experience of case retrieval or if the current case base is to be updated. If so, two loop processes, an *inner loop* and an *outer loop*, are proposed to perform the refinement of the partition. In the inner loop, the partition will be changed such that the result is neither finer nor coarser than the original one, because it is easily shown that "finer" as a binary relation is a partial order $\leq$. This will be repeated for a given number of iterations (if P and Q are not satisfactory). When the maximum number of loops has been reached, and P and Q are still not satisfied then the outer loop will be entered, where the partition is refined once. For brevity, the inner loop is called *microadjustment* and the outer loop *refinement*. For example, let $W_p = \{a_1, a_2, a_3, a_4, a_5, a_6\}$, and S_0, S_1, S_2, S_3, S_4, S_5, S_6 be similarity relations on W_p, and their corresponding partitions of W_p are:

- $W_p/S_0 = \{\{a_1, a_2, a_3, a_4, a_5, a_6\}\}$
- $W_p/S_1 = \{\{a_1, a_2, a_3\}, \{a_4, a_5, a_6\}\}$

1. Which is based on the statistics of case retrieval.

- $W_p/S_2 = \{\{a_1, a_2\}, \{a_3, a_4\}, \{a_5, a_6\}\}$
- $W_p/S_3 = \{\{a_1\}, \{a_2, a_3\}, \{a_4, a_5\}, \{a_6\}\}$
- $W_p/S_4 = \{\{a_1\}, \{a_2\}, \{a_3, a_4\}, \{a_5\}, \{a_6\}\}$
- $W_p/S_5 = \{\{a_1, a_2\}, \{a_3\}, \{a_4\}, \{a_5, a_6\}\}$
- $W_p/S_6 = \{\{a_1\}, \{a_2\}, \{a_3\}, \{a_4\}, \{a_5\}, \{a_6\}\}$

The partial order "finer" between the partitions is illustrated in Fig. 5.7 It is easy to see that W_p/S_0 is the coarsest partition of W_p, W_p/S_6 is the finest partition of W_p, However, there are no "finer" or "coarser" relationships between W_p/S_1 and W_p/S_2, nor among W_p/S_3, W_p/S_4 or W_p/S_5. In *microadjustment,* if W_p/S_1 (i.e. its corresponding P) is not satisfied, then one can choose W_p/S_2 as an alternative, carrying out the inner loop. If W_p/S_2 is still not satisfied, then either W_p/S_1 or W_p/S_2 will be refined and either W_p/S_3 or W_p/S_4 or W_p/S_5 are obtained, carrying out the outer loop. etc. The concrete order of microadjustment and refinement is application dependent and has to be chosen in advance or in accordance with the degrees of satisfaction or dissatisfaction of the partition to be microadjusted or refined.

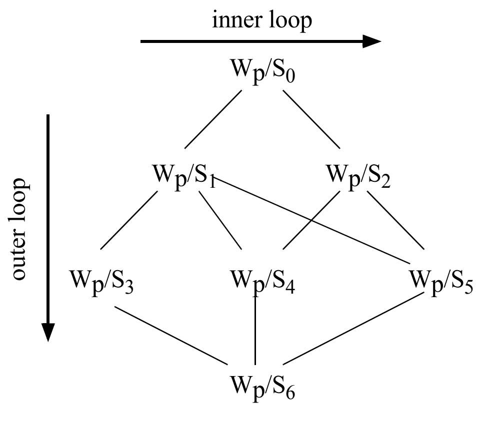


Fig. 5.7 Hasse diagram for the "finer" relation in Definition 12, the lower the finer

Based on this consideration, the transition from the possible world of problems W_p and the possible world of solutions W_s to the case base is also a refinement process of partition and repartition. Therefore, algorithm-I* will be extended to algorithm-II, which has two loops, in order to create a satisfactory case base for a CBR system based on similarity-based reasoning as follows:

Step 1. Define the possible world of problems W_p and the possible world of solutions W_s

Step 2. Define a similarity relation S on W_p

Step 3. Find the partition of W_p with respect to S

Step 4. Find the representative elements p_i of all disjoint similarity classes in the partition of W_p in terms of S and then constitute them into a set of precedent problem descriptions P

Step 5. For every representative element $p_i \in P$ find all possible solutions (similar solutions) in the possible world of solutions, $\{s_{ij} | j \in J\}$

Step 6. Define a similarity class $\{s_{ij} | j \in J\} = [s_i]$ of a certain similarity relation T on W_s and decide a partition of W_s, i.e.

$$W_s = \bigcup_{i=1}^{n} [s_i] \cup \left(W_s - \bigcup_{i=1}^{n} [s_i] \right) \tag{56}$$

Step 7. Find a representative, s_i, in similarity class $[s_i]$ of a partition of the possible world of solutions W_s. The pair (p_i, s_i) becomes a case in the case base. The set of all the corresponding solutions from this step is called the set of solution descriptions Q

Step 8. IF P and Q are satisfied, THEN create the case base $C = (P, Q)$ and GOTO Step 11, otherwise GOTO Step 9

Step 9. IF the maximal number of iterations is not exceeded THEN *microadjust* the similarity relation, $S \to S^*$, and GOTO Step 3 (outer loop), otherwise GOTO Step 10

Step 10. IF the maximal number of iterations is not exceeded THEN refine the partition, $S \to S^x$, GOTO Step 3 (outer loop), otherwise GOTO Step 11

Step 11. End.

It should be noted that which partition of W_p is better also depends on the cardinality of the case base and the concrete application settings. For example, there are about 10,000 cases in the case base involved in a distributed CBR application for engineering sales support [319], although there are a few hundred cases in most CBR systems [172].

There remains a question; that is, how does one deal with adding a new case $\tilde{c} = (\tilde{p}, \tilde{s})$ to the case base of a CBR system? This question is of practical significance, because it is a frequent action for any running CBR system to add a new case to its case base. This question involves case retrieval and case reuse, because case retrieval should be performed to know if the problem description $\tilde{p}$ belongs to a certain similarity class $[p_i]$. If $\tilde{p} \in [p_i]$ then there are two possibilities:

1. $\tilde{s} \in [s_i]$. In this case, $(\tilde{p}, \tilde{s})$ will not be required to put to the case base
2. $\tilde{s} \notin [s_i]$ for any i. In this case, W_s has to be re-partitioned.

If $\tilde{p} \notin [p_i]$ for any i, then the possible world W_p should be repartitioned or a new similarity relation on W_p should be chosen so that the $\tilde{p}$ belongs to a certain similarity class in terms of the new partition of W_p. Then $\tilde{p}$ can be added as the representative element of the mentioned similarity class and its corresponding solution $\tilde{s}$, as a new case, into the case base.

Because fuzzy similarity relations are an extension of similarity relations, the above proposed approach can be extended to the fuzzy similarity setting. The next section will turn to a fuzzy similarity based model for case base building.

5.3.6 Case-base Building based on Fuzzy Similarity Relations

Fuzzy similarity relations have been used in CBR, in particular in case retrieval [72][73][74][231]. However, there are still no studies on applying fuzzy similarity relations to case base building, although the latter is an important basis for case retrieval and case adaptation. This section will extend discussions in the previous section using fuzzy similarity relations and fill the mentioned gap. For brevity, S is still used to denote a fuzzy similarity relation in this section if there is not any confusion arising.

Definition 13. Let S be a fuzzy similarity relation in W_p. For $\alpha \in [0, 1]$, a α-level-set of fuzzy similarity relation S is, denoted by S_α, a non-fuzzy set in $W_p \times W_p$ defined by

$$S_\alpha = \{(p, q) | S(p, q) \geq \alpha\} \tag{57}$$

Then the following consequences are valid [348]:

$$\alpha_1 \geq \alpha_2 \Rightarrow S_{\alpha_1} \subseteq S_{\alpha_2} \text{ (nested sequence)} \tag{58}$$

$$S = \sum_\alpha \alpha S_\alpha,\ 0 < \alpha \leq 1 \text{ (resolution identity)} \tag{59}$$

where $\sum$ stands for the union and support$(\alpha S_\alpha) = S_\alpha$ with

$$\alpha S_\alpha(p, q) = \begin{cases} \alpha, \text{ for } (p, q) \in S_\alpha \\ 0, \text{ elsewhere} \end{cases} \tag{60}$$

Conversely [348], if the S_α, $0 < \alpha \leq 1$, are a nested sequence of distinct similarity relations in W_p with $\alpha_1 > \alpha_2 \Leftrightarrow S_{\alpha_1} \subset S_{\alpha_2}$, S_1 non-empty, then, for any choice of α in $(0, 1]$ which includes $\alpha = 1$, S is a similarity relation in W_p.

This implies that every α-level set of a fuzzy similarity relation is a traditional similarity relation. Furthermore, from a fuzzy similarity relation S one can get crisp similarity relations S_α, $\alpha \in [0, 1]$. From a nested sequence of crisp similarity relations S_α (with the above mentioned properties) one can get the fuzzy similarity relation S.

Since traditional similarity relations play an important role in partitioning a set, fuzzy partitions will be examined in order to build a case base using a fuzzy partition [25]. The core idea behind this is that every α-level set can be used, which is a traditional similarity relation, to form its corresponding partition of W_p.

Let S be a fuzzy similarity relation in W_p with a membership function $S(p, q)$. With each $p \in W_p$, p is associated a fuzzy similarity class denoted by $S[p]$. This class is a fuzzy set in W_p which is characterized by the membership function

$$S[p](q) = S(p, q) \quad \forall q \in W_p \tag{61}$$

Thus, $S[p]$ is a fuzzy similarity class centred at p. Furthermore, $S_\alpha[p]$ is a non-fuzzy set in W_p and given by $S_\alpha[p] = \{q|S(p, q) \geq \alpha\}$. Based on Eq.59 the following result is obtained at once:

$$S[p] = \sum_{\alpha} \alpha S_\alpha[p] \tag{62}$$

where

$$\alpha S_\alpha[p](q) = \begin{cases} \alpha, \text{ for } q \in S_\alpha\ [p] \\ 0, \text{ elsewhere} \end{cases} \tag{63}$$

Therefore, $S_\alpha[p]$ is a similarity class with the representative $p \in W_p$ and the elements in $S_\alpha[p]$ have the same similarity degree α. Since every S_α is a traditional similarity relation in W_p, its corresponding partition is

$$[S_\alpha] = \{S_\alpha[p]|p \in W_p\} \tag{64}$$

Because a fuzzy similarity relation is reflexive, then we have $domS_\alpha = W_p$, which is useful for partition of W_p.

In the e-sale setting, every fuzzy similarity relation S on W_p might be based on the seller agent's experience, and is seller-centred. It is only some α-level set of a fuzzy similarity relation S that is meaningful for decision making in selling process, because two problems with very low similarity (in this case, α is very small) do not certainly have the same or similar solutions. Therefore it is necessary to introduce

Definition 14. Let S be a fuzzy similarity relation in W_p, a constant $b, 0 < b < 1$ is called a *domain-similarity threshold* iff for any $p, q \in W_p$ with $S(p, q) \geq b$ then the similarity between p and q is application-feasible.

Therefore, in practice, one can only care about the α-level set of a fuzzy similarity relation in W_p with $\alpha \geq b$. Now, assume that $\alpha \geq b$, the partition of the α-level set of a fuzzy similarity relation S in W_p, S_α, is

$$[S_\alpha] = \{S_\alpha[p_i]|p_i \in W_p, i = 1, \ldots, n\} \tag{65}$$

Then, similar to the discussion in Section 5.2.2, the representative element of every $S_\alpha[p_i]$, p_i is selected, and its possible solution is found, and then its corresponding solution $s_i \in W_s$ is chosen to constitute a case $c_i = (p_i, s_i)$ and store it in the case base. Therefore, the case base is built based on the fuzzy similarity relation too; that is,

$$C = (P, Q) \tag{66}$$

where, $P = \{p_1, \ldots, p_n\}$ is the set of precedent problem descriptions, $Q = \{s_1, \ldots, s_n\}$ is the set of corresponding solution descriptions. Finally the discussion in this section is summarized as an algorithm- III as follows:

Algorithm-III creates a case base for a CBR system using a fuzzy similarity relation in the possible world of problems W_p and the possible world of solutions W_s.

Step 1. Define the possible world of problems W_p and the possible world of solutions W_s

Step 2. Define a fuzzy similarity relation S on W_p

Step 3. Decide the domain-similarity threshold b

Step 4. Select $\alpha = b$, then find the α*-level set* of the fuzzy similarity relation S, S_α, which is a traditional similarity relation on W_p

Step 5. Find the partition of W_p with respect to S_α; that is,

$$[S_\alpha] = \{S_\alpha[p_i] | (p_i \in W_p), i = 1, \ldots, n\} \tag{67}$$

Step 6. Select the representative element p_i of each $S_\alpha[p_i]$, p_i, and then constitute them into a set of precedent problem descriptions P

Step 7. For every representative element $p_i \in P$ find all possible solutions (similar solutions) in the possible world of solutions W_s, $\{s_{ij} | j \in J\}$. Define a similarity class $\{s_{ij} | j \in J\} = [s_i]$ of a certain similarity relation T on W_s and decide a partition of W_s, i.e.

$$W_s = \bigcup_{i=1}^{n} [s_i] \cup \left(W_s - \bigcup_{i=1}^{n} [s_i] \right) \tag{68}$$

Step 8. Choose a representative s_i in similarity class $[s_i]$ of a partition of the possible world of solutions W_s. The pair (p_i, s_i) becomes a case in the case base. The set of all the corresponding solutions from this step is called the set of solution descriptions Q

Step 9. If P and Q are satisfied, then create the case base $C = (P, Q)$ and End; else: Select another α such that $\alpha_{\text{new}} > \alpha_{\text{old}}$ and return Step 4

Step 10. If P and Q are still unsatisfactory, then return to Step 2 (that is, define a new fuzzy similarity relation S' on W_p)

Step 11. End.

It is worth noting why $\alpha_{\text{new}} > \alpha_{\text{old}}$ in Step 9. This is because if α becomes greater, the crisp partition gets finer! Thus, this rule is, in fact, a refinement of the partition of W_p, as mentioned previously. Thus, Step 9 corresponds to what *outer loop* (*refinement*) and Step 10 corresponds to *inner loop* (*microadjustment*). Therefore, algorithm III is an extension of algorithm II. The difference between them lies in the following: In algorithm II the way of refinement is not restricted and the partition can be refined arbitrarily, while in algorithm III the refinement is given by the fuzzy similarity relation which defines all α-level sets.

5.3.7 Summary

This section argued that a case base can be built based on both similarity relations and fuzzy similarity relations and proposed three algorithms for building case-bases. Thus case base building is a form of similarity-based reasoning. The main difference of this research from others is that similarity relations are not only used for case retrieval but also used to create case bases as well as for case adaptation [88]. This approach is thus the foundation and an extension for the logical and fuzzy approach to case based reasoning, because case base building is an important basis for performing case retrieval and case adaptation.

5.4 R^5 Model for Case-based Reasoning

As mentioned in Section 2.3, there have been many models for CBR that attempt to provide better understanding of CBR. However, they all assume that the case base is also ready for the first process, case retrieval, although they discuss the representation of cases and believe that a case-based reasoner is heavily dependent on the structure and content of its collection of cases. In fact, it seems that everyone believes that representation of cases is important for CBR, but there are no unified ways to integrate it into the models of CBR. Furthermore, it seems that almost all existing models are empirical and descriptive, and it is difficult to extend these models to a theoretical CBR. It is obvious that CBR can't develop robustly further without a firm theoretical foundation.

This section proposes a R^5 model, in which Repartition, Retrieve, Reuse, Revise and Retain are the main tasks for the CBR process. It argues that the proposed R^5 model is a new approach to using similarity-based reasoning to unify case base building, case retrieval, and case adaptation.

From an engineering viewpoint, a knowledge-based system such as a rule-based expert system (see Chapter 4) can be regarded as a process of the following sequential phases: Knowledge acquisition, knowledge representation, knowledge reasoning, knowledge interpretation, and knowledge utilization [287]. In comparison with this, it seems that there is not a stage in the CBR models mentioned above that corresponds to knowledge acquisition. This means that there has not been much discussion about case acquisition, although case representation has been much discussed in CBR. As mentioned in Section 2.3, in the R^4 model, *Retrieval* is the first step in the process of CBR, which means that the case representation and case base are already ready for performing CBR. However, this is not the case in many applications.

Therefore, it is of significance to extend the R^4 model to the R^5 model, shown in Fig. 5.8. In this proposed R^5- model, Repartition, Retrieve, Reuse, Revise, and Retain are the main process steps in the CBR [88]. While the other process steps are the same as those in the R^4 model mentioned in Section 2.4 or in [1], Repartition here is considered as the fifth R and used to form a satisfactory case base $C = (P, Q)$ based on partitioning on W_p, as discussed in the previous section (Section 5.3). Furthermore, Repartition provides the theoretical foundation for case retrieval, because of the one-to-one correspondence between the partition of W_p and the similarity relations on W_p. Thus, case base building and case retrieval can be treated as a similarity-based reasoning in a unified way. It should be noted that case adaptation in this model is also similarity-based reasoning. Therefore, the proposed model is a further instantiation of CBR as a process reasoning in Section 2.5, and can facilitate the use of similarity-based reasoning to unify case base building, case retrieval, and case adaptation.

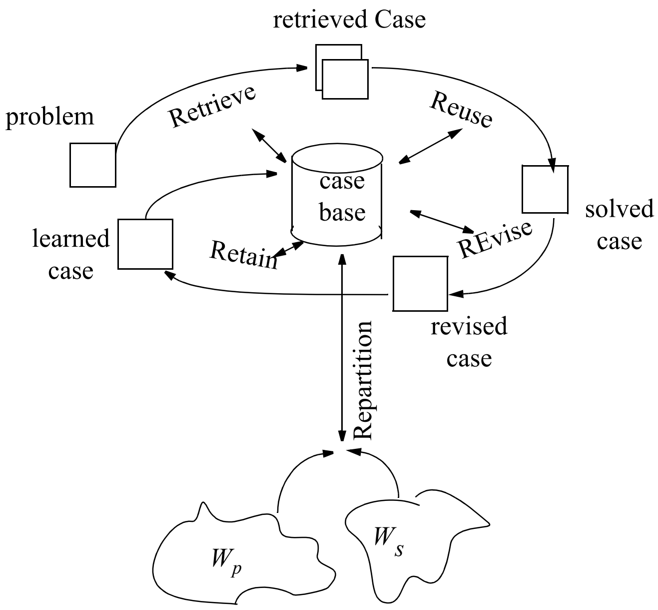


Fig. 5.8 The R^5 model of CBR

It should be noted that the result of this section is a direct consequence of Section 2.5 and Section 5.3. The central idea behind the R^5 model is that case base building is an important task of CBR and the case base can be built based on partitioning of the possible world of problems and solutions, which is considered as the fifth R [88]. The core idea different from other studies is that similarity relations and fuzzy similarity relations are not only used for case retrieval but also used for creating the case base, although it might be used in different stages in a different way. Therefore, the proposed approach provides a new attempt towards using similarity-based reasoning to unify case base building, case retrieval, and case adaptation and therefore facilitate the development of theoretical CBR with applications.

5.5 Abductive Case-based Reasoning

This section will introduce abductive case-based reasoning (CBR) and show that abductive CBR and deductive CBR can be integrated in clinical process and problem solving. Then it provides a unified formalization for integration of abduction, abductive CBR, deduction, and deductive CBR. The proposed approach demonstrates that the integration of deductive CBR and abductive

CBR is of practical significance in problem solving such as system diagnosis and analysis and will facilitate research of abductive CBR and deductive CBR.

5.5.1 Introduction

As is well known, abduction and deduction play a fundamental role in problem solving [11][49]. In particular abduction seems to be a basic reasoning component in activities such as explanation [165] and diagnosis [49][302] as well as analysis. Abduction has drawn much attention in AI fields [244][49][165][302]. In [49] Console et al. introduced an interesting relation between abduction and deduction and showed that abduction can be reduced to deduction on a transformed (completed) domain theory that explicitly contains the assumption that all the direct explanations of an event have been represented. Recently, CBR has been shown to play an important role in explanatory or abductive reasoning tasks like diagnosis and explanation [235][289], one of which is case-based explanation [165]. In most AI views, explanations are treated as deductive proofs. Abductive reasoning systems build their proofs by nondeductive methods, and additional assumptions may be required for those proofs to apply [165]. However, their view is fundamentally the same in that if the abductive assumptions were shown to be true the resulting explanation would be considered a deductive proof. The case-based approach explicitly treats explanations as plausible reasoning chains that may be implicit. However, there is a lack of a theoretical treatment towards integration of deductive CBR and abductive CBR. There is also no unified treatment of the relationship between abduction, deduction, and CBR. This section attempts to show that abductive CBR and deductive CBR can be integrated in clinical process and problem solving. Then it provides a unified formalization for integration of abduction, abductive CBR, deduction, and deductive CBR. This section also proposes the transformation from abduction to abductive CBR and from deduction to deductive CBR.

5.5.2 Abduction and Deduction

This subsection will examine abductive reasoning and deductive reasoning with two examples and show that clinical reasoning and problem solving in general can be considered as an integration of abductive reasoning and deductive reasoning [289]. At first, it examines abduction and deduction in clinical processes. The goal here is not clinical data or knowledge modelling, but only a computational or logical understanding.

As is well known, the clinical process basically consists of the diagnosis and treatment of patients. Diagnosis is a judgement (or explanation) about what a particular illness is, made after making an examination of the symptoms of a patient. Its goal is to explain symptoms observed from the patient in the clinic [302]. The explanations for the observed symptoms are the basis for treatment. Treatment is a concrete solution to the illness of the patient based on the explanation descriptions of the diagnosis.

Example 6. Consider a concrete case of diagnosis and treatment happening in a normal day in the clinic. The doctor examines the patient and gets

Symptom: dizziness.

He has the following medical knowledge (domain theory):

$$\{\text{ flu} \to \text{fever, infection} \to \text{fever, fever} \to \text{dizzyness, fever} \to \text{no interest in eating }\}$$

and diagnoses that the patient has flu and tells the explanation to the patient. Then he completes the prescription which includes 10 tablets of “Aspirin” as he has the medical knowledge “$\text{flu} \to \text{Aspirin}$”.

During the above process, the doctor has used two different reasoning paradigms: abductive reasoning and deductive reasoning. From a logical viewpoint, his diagnosis result is following the process of abductive reasoning:

He derives the explanation, “fever”, from the symptom, “dizziness” and his knowledge “$\text{fever} \to \text{dizzyness}$.” Then he derives the explanation, “flu,” from the just derived explanation, “fever,” and his knowledge “$\text{flu} \to \text{fever}$.” Therefore, his reasoning towards the satisfactory diagnosis is following the model of abduction or abductive reasoning [244][302]:

$$\begin{array}{r} P \to Q \\ \underline{\quad\quad Q} \\ \therefore P \end{array} \tag{69}$$

where P and Q represent compound propositions in a general setting. In medical diagnosis, $P \to Q$ is a form of general relation: disease $\to$ symptom.

Abduction is the term currently used in the AI community for generation of explanations for a set of events from a given domain theory [46][289]. From a logical point of view, abduction is an unsound reasoning [244][302]. However, it has similar properties to those of other nonmonotonic logics proposed and studied in the AI literature. For example, it shares declarative and computational properties with other forms of nonmonotonic reasoning [289]. Thus, abduction is a very useful kind of nonmonotonic reasoning, in particular for reasoning towards explanation in (system) diagnosis [302] and analysis in problem solving, which will be examined in more detail later.

It should be noted that most current rule-based diagnosis systems use knowledge of the form [234]

$$\text{observation and knowledge of situation} \quad \rightarrow \quad \text{problem}$$

to express knowledge about the potential cause of an observation. For example, MYCIN expresses its knowledge in terms of rules of the form

$$\text{symptom} \quad \rightarrow \quad \text{disease [CF]}$$

where CF is a certainty factor that represents a subjective evaluation of the rule's quality. The diagnosis task consists of matching rule symptoms and observed symptoms, accumulating the conclusions suggested by relevant rules, and ranking the conclusions by a simple arithmetic function on certainty factors. However, the rules above are the wrong way around: diseases result in symptoms, rather than symptoms in diseases. In other words, a diagnosis is not a logical consequence of our observations about a patient. In fact, exactly the opposite is the case, it is the observations that should be shown to be logical consequence of our knowledge and the diagnosis. Based on this idea, Poole and Goebel [234] uses an alternative formulation of the rules in their system, Theorist, which is a logic programming system that uses a uniform deductive reasoning mechanism to construct explanations of observations in terms of facts and hypotheses:

$$\text{problem} \quad \rightarrow \quad \text{observation}$$

where knowledge is expressed in terms of problems and the observations that consequently arise. For example, the medical diagnosis task would use rules of the form:

$$\text{disease} \quad \rightarrow \quad \text{symptom}$$

to encode the observable symptoms of diseases. This form of representation is more appropriate for expressing textbook knowledge of diseases, as it records what is known without any requirement for heuristic measures like certainty factors and experience record. Therefore, Poole and Goebel [234] have similar ideas to that in this research about diagnosis. However, they have not generalized their idea to a more general reasoning paradigm: abduction or abductive reasoning.

After having obtained the precise explanation for the symptoms of the patient, the doctor derives the treatment from the explanation “flu” and his knowledge of treatment “flu $\rightarrow$ Aspirin;” that is, “10 tablets of Aspirin.” This is deductive reasoning, its general model is well-known *modus ponens (m.p.)* mentioned in Section 2.4. Deduction is a fundamental reasoning paradigm in traditional logic and mathematics. It also has widespread application in almost every academic field.

So far, this section has shown that the clinical process is an integration of abductive reasoning and deductive reasoning, as shown in Fig. 5.9. The cycle in Fig. 5.9 starts with the patient

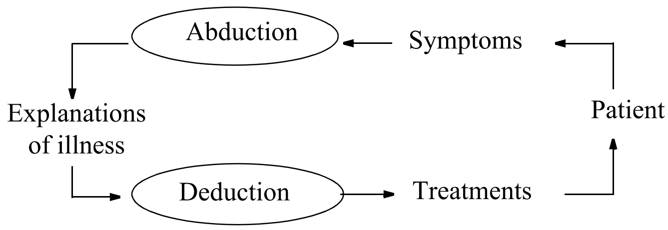


Fig. 5.9 Integration of abduction and deduction in clinical process

showing symptoms and completes with treatments. Further, in a more general setting, diagnosis is a process of obtaining satisfactory explanation for a particular problem, made after making an examination of a system with the presence of some faults. Therefore, abductive reasoning can be applied to more general situations such as system diagnosis. The rest of this subsection will look into abduction and deduction in problem solving.

The notation of explanation and analysis is basic in many human behaviors. In particular, in any intelligent system such as an expert system there is a subsystem to explain the reasoning process to the user. Reasoning towards explanation and analysis is a fundamental task in many of problem solving activities investigated by the AI community [302]. In what follows, problem solving is decomposed into analysis and reasoning. The analysis process in problem solving will be shown as mainly abductive reasoning, while the reasoning process is mainly deductive reasoning, with an example in propositional logic, borrowed from [282].

Example 7. Prove $P \rightarrow \neg\neg Q, \neg R \rightarrow P, \neg Q \Rightarrow R$

As is known, this is a non-trivial problem for an undergraduate student studying propositional logic. It is better to decompose this problem solving into two phases: analysis and reasoning: Abductive reasoning is performed in the analysis phase, while deductive reasoning is usually performed in the reasoning phase.

1. Analysis.

(1) Because of R, $\neg R \to P$ is transformed into $\neg P \to R$. Then $\neg P$ is derived from $\neg P \to R$ and R based on abductive reasoning.

(2) Because of $\neg P$, $P \to \neg\neg Q$ is transformed into $\neg Q \to \neg P$. Then performing abductive reasoning, $\neg Q$ is derived from $\neg Q \to \neg P$ and $\neg P$.

(3) Because in the hypotheses there is also $\neg Q$. Thus it can conclude that this formula is provable and the analysis phase is finished.

It should be noted that the above transformations are logically equivalent. Further, it is interesting to note that during the analysis phase, the abductive reasoning chain (in reverse) is obtained without taking logical equivalence formulas into account:

$$\neg Q,\ \neg Q \to \neg P,\ \neg P,\ \neg P \to R,\ R.$$

which is just the main deductive reasoning chain in the below reasoning phase. Therefore, after having obtained the abductive reasoning chain from R to $\neg Q$, it is easy to prove the formula under consideration.

2. Reasoning.

Based on the results of above abductions in the analysis phase, this phase performs deductive reasoning as that in propositional logic.

Proof	Explanations
$\neg Q$	(hypothesis)
$P \to \neg\neg Q$	(hypothesis)
$\neg Q \to \neg P$	(contrapositive (2))
$\neg P$	(m.p. (1), (3))
$\neg R \to P$	(hypothesis)
$\neg P \to R$	(contrapositive)

R (m.p. (4), (6))

Generally speaking, many textbooks do not discuss the analysis phase in problem solving. They usually only provide standard solutions to problems. However, this is insufficient for class teaching. During teaching, lecturers sometimes have to use such methods to instruct the students to improve their ability of analysing and solving problems. Therefore, problem solving can also be considered as an integration of abductive reasoning and deductive reasoning: Abductive reasoning is performed to get a satisfactory analysis in order to perform deductive reasoning to solve the problem. In other words, abductive reasoning is a necessary condition for performing deductive reasoning towards problem solving in some cases.

Forward chaining and backward chaining are well-known concepts in AI [256] (p 272). More specifically, one can start with the sentences in the knowledge base and generate new conclusions that in turn can allow more inferences to be made. This is called *forward chaining*. Forward chaining is usually used when a new fact is added to the knowledge base and its consequences should be generated. The theoretical foundation of forward chaining is previously mentioned *modus ponens* (see Section 2.4). Alternatively, one can start with something that is to be proved, find implication sentences that would allow one to conclude it, and then attempt to establish their premises in turn. This is called *backward chaining*, because it uses *modus ponens* backwards. Backward chaining is normally used when there is a goal to be proved, according to Russell and Norvig [256]. Backward chaining is commonly used in RBESs to enable a hypothesis to be tested and explained [317] (pp 8-9). This mimics human problem-solving strategies, for example, in medical diagnosis some of human problem-solving strategies. When one is sick, his doctor often hypothesizes using knowledge of what the possible cause of his illness may be. Doctors then try to confirm their hypothesis by looking for characteristic symptoms or by performing certain tests. If these do not confirm their hypothesis, they will think of another illness and test that hypothesis. This problem solving strategy is often referred to as *generate and test* and has been used successfully by many expert systems, particularly in diagnosis. However, based on the above discussion, the theoretical foundation of backward chaining is the basic reasoning model of abduction (see Eq.69). Furthermore, the theoretical foundation of Prolog and most other logic programming languages is also abduction, because they are based on backward chaining [256] (p 313). Therefore, this subsection proposes a new insight into backward chaining and its relation to

Prolog and most other logic programming languages: abduction and deduction exist in AI in a "symmetric" way.

5.5.3 Abductive CBR and Deductive CBR

This subsection will demonstrate that abductive CBR and deductive CBR are an extension of abductive reasoning and deductive reasoning respectively. it then shows that abductive CBR and deductive CBR can be integrated in diagnosis, explanation, and problem solving. It begins with the evolution from abduction to abductive CBR.

As has been previously shown, abductive reasoning is a kind of explanation-oriented reasoning [302]. Diagnosis is a process of deriving an explanation of the symptoms based on the observations by the doctor of the patient and it can be considered as an abductive reasoning [49]. In fact, in clinical practice, a doctor usually first observes a particular patient's mouth, eyes, and body temperature, etc. and gets all possible symptoms of the patient. These symptoms can trigger a reminder of previous cases he has met. Prior experiences then play an important role in getting the exact explanation for the symptoms of the patient. Therefore, diagnosis or the process of explaining the symptoms is not only abduction, but also an experience-based reasoning. The experiences also play a pivotal role in the analysis phase of the problem solving. For example, why was $P \rightarrow \neg\neg Q$ not at first selected in the last section for analysis? Because CBR means reasoning based on previous cases or experiences, abductive CBR can be considered as the reasoning combining abduction and experience-based reasoning, briefly:

$$\text{Abductive CBR} = \text{Abduction} + \text{Experience-based reasoning} \qquad (70)$$

An important experience principle in the diagnosis is "most similar symptoms result from most similar illness". Based on this principle, the doctor comes to the conclusion that the illness of the patient is most similar to the illness that he experienced last week. This is not only an experience-based reasoning but also a similarity-based reasoning. Thus, similarity-based reasoning is an operational form of experience-based reasoning (see 2.2). In fact, similarity-based reasoning has played an important role in experience-based reasoning as shown in the CBR literature [152][167]. Therefore Eq.70 can be specialized as a form of reasoning combining abduction and similarity-based reasoning:

$$\text{Abductive CBR} = \text{Abduction} + \text{Similarity-based reasoning} \qquad (71)$$

Similarity-based reasoning is also very important in performing experience-based reasoning in the analysis phase of problem solving, because common sense is used in problem solving such as in mathematics "Two problems are similar, if they have similar explanations". For example, in case-based explanation, the first criterion for selecting likely explanations is experience in similar situations: Explanations of new situations are considered most plausible if they have applied in similar prior situations [165]. Therefore, the analysis of problem solving can be considered as a kind of abductive CBR.

Based on Eq.71, Eq.69 can be extended to the following reasoning model:

$$\begin{array}{r} P \to Q \\ \underline{\quad Q'} \\ \therefore P' \end{array} \tag{72}$$

where P, P', Q, and Q' represent compound propositions, Q and Q' are similar in the sense of a certain similarity (see Section 5.2). This is a theoretical foundation for abductive CBR, in particular for similarity-based abductive case retrieval, which will be examined in more detail in Section 5.6.3.

Now the subsection will turn to look at the evolution from deduction to deductive CBR. As was shown in the previous subsection, treatment in the clinical process can be considered as a deductive reasoning. Further, after having obtained a satisfactory diagnosis, the doctor not only performs deduction but also uses his experience in the past for writing the prescription or performing treatment for the illness of the patient. Thus, experience-based reasoning plays an important role in deductive reasoning such as treatment. In fact, it is obvious that experience also plays a pivotal role in the reasoning phase of problem solving. For example, if $\neg R \to P$ is used as the first step of deductive reasoning in the example of the previous subsection, one might not know which will be the next step. Therefore, deductive CBR can be considered as the form of reasoning combining deduction and experience-based reasoning, briefly:

$$\text{Deductive CBR} = \text{Deduction} + \text{Experience-based reasoning} \tag{73}$$

This research prefers deductive CBR rather than CBR, because CBR is an extension of deductive reasoning from a logical viewpoint. If CBR is only used, one can't see the influence of deductive reasoning on CBR. Further, it seems that there is certain "symmetry" between abductive CBR and deductive CBR.

It is common sense in the clinical process that "similar illnesses usually result in similar treatments". This is not only an experience-based reasoning but also a similarity-based reasoning. Thus, similarity-based reasoning is a special form of experience-based reasoning in the treatment phase of clinical processes. Further, similarity-based reasoning is very important in performing experience-based reasoning in the reasoning phase of problem solving, because there is an experience principle in this phase such as in mathematics "Similar problems have similar solutions". Therefore, Eq.73 is specialized as a reasoning combining deduction and similarity-based reasoning; that is:

$$\text{Deductive CBR} = \text{Deduction} + \text{Similarity-based reasoning} \tag{74}$$

As is known, CBR solves new problems reapplying the lessons from specific prior reasoning episodes. A functional motivation for CBR is the principle that in a regularity in the world, similar problems have similar solutions [166]. When this principle holds, starting from similar previous solutions can be more effective than reasoning from scratch. Similarly, the functional motivation for abductive CBR is the principle that there is also a regularity in tasks such as diagnosis or analysis, similar symptoms result from similar illnesses. This principle leads to the conclusion that it is *effective* for generating new explanations by retrieving prior explanations (analysis) for similar symptoms (problems) and adapting those retrieved explanations (analysis) to fit the new symptoms (problems) if possible. However, different understanding of similarity usually leads to different case retrieval and then to different abductive CBR and deductive CBR, which will be discussed in more detail in Section 5.6.3.

Eq.74 can be expressed (see Section 2.4) the following reasoning model:

$$\begin{array}{l} P \to Q \\ \underline{P'} \\ \therefore\ Q' \end{array} \tag{75}$$

where P, P', Q, and Q' represent compound propositions, Q and Q' are similar in the sense of similarity (see Section 5.2). This is the basic reasoning model of similarity-based reasoning. Although it has the same form as that of generalized *monus ponens* mentioned in Section 2.4., both reasoning models have different explanations and come from different real world scenarios. Finally, Eq.75 is a theoretical foundation for deductive CBR, in particular for similarity-based deductive case retrieval (see Section 5.6.3)

So far, the relationship between abduction, abductive CBR and deduction, deductive CBR has been discussed respectively from a logical viewpoint. It has also shown that clinical process and problem solving are a form of reasoning for combining abductive CBR and deductive CBR, as shown in Fig. 5.10[1]. Similar to Fig. 5.9, in Fig. 5.10 the cycle starts with the patient showing symptoms and completes with treatments. Because deductive CBR is a kind of deductive (monotonic) reasoning, while abductive CBR is a kind of nonmonotonic reasoning, clinical process and problem solving is then an integration of traditional reasoning and nonmonotonic reasoning.

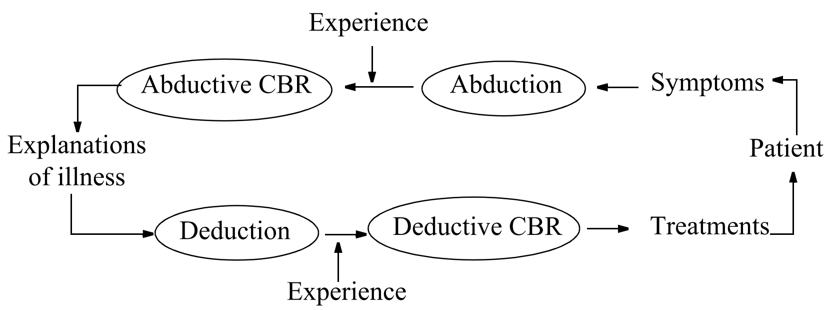


Fig. 5.10 Integration of abductive CBR and deductive CBR

5.5.4 Integration of Abductive CBR and Deductive CBR

This subsection will examine abductive CBR and deductive CBR from a viewpoint of knowledge-based systems (KBSs) and integrate the abductive CBR system and deductive CBR system with a knowledge-based model.

As has been shown, clinical process and problem solving are a form of reasoning combining abductive CBR and deductive CBR from a logical viewpoint. There has been an important influence of KBSs on CBR systems in most (deductive) CBR literature [1][152][315]. For example, the case base in the CBR system can be considered as a variant of the knowledge base in KBSs. Therefore, a deductive CBR system can be considered as an integration of deductive reasoning and a KBS [289].

From the viewpoint of AI, abductive reasoning has also been affected by the research of KBSs [166][244]. Thus, abductive CBR can also be considered as an integration of abductive reasoning

1. It is easy to give a similar diagram for integration of abductive CBR and deductive CBR in problem solving.

and KBSs. In fact, CBR-based abduction has been studied for many years [49][165][166][302], one part of which is case-based explanation [165][166].

Case-based explanation generates new explanations by retrieving explanations of relevant prior episodes and adapting them to fit the new situation in light of the explainer's need for information [166]. The prior experiences of the explainer are fundamental to focusing search for candidate explanations, and the motivation for explaining is reflected in both the explanation generation and selection processes. Therefore, abductive CBR can be considered as an extension of case-based explanation.

In case-based explanation, the most important criterion for judging plausibility is similarity-based [166]: Explanations of new anomalies are favoured if they are similar to explanations that applied to similar prior anomalies. This similarity judgment is done implicitly through the case retrieval process; retrieval of stored explanations is aimed at retrieving explanations from similar prior situations [165].

Based on the above consideration, Leake proposed a process model for case-based explanation in [166]; that is:

- Problem characterization: Generate a description of what must be explained, i.e., the information that a good explanation must provide
- Explanation retrieval: Use the results of the problem characterization step as an index for retrieving relevant explanations of prior episodes from memory
- Explanation evaluation: Evaluate the retrieved explanations' plausibility and usefulness. Generate problem characterizations for any problems that are found
- Explanation adaptation: If problems were found, use the evaluator's problem characterization to select adaptation strategies for modifying the explanation to repair the problems. Apply the strategies and return to the explanation evaluation phase to evaluate the new explanation.

In what follows, an integrated knowledge-based model for both the abductive CBR system and deductive CBR system is proposed, shown in Fig. 5.11. This model can also be used in clinic processes, because diagnosis and treatment in the clinic process is a special case of problem solving [289].

In this model, problem solving is decomposed into analysis and reasoning. Abductive CBR is performed in the analysis process, while deductive CBR is performed in the reasoning process

[289]. For clarity, an *abductive case base* is used in the abductive CBR system and a deductive case base in the deductive CBR system instead of case base respectively. Similar to the inference engine in expert systems [214], an abductive engine is used in the abductive CBR system and a deductive engine in the deductive CBR system for the reasoning mechanism in each case.

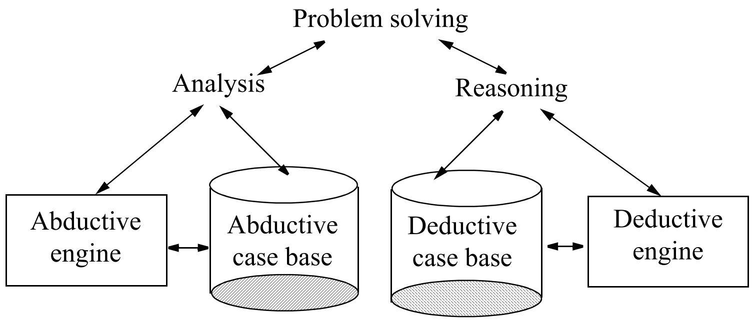


Fig. 5.11 A knowledge-based model of integrating abductive CBR and deductive CBR

However, working memories are ignored in each case in the figure. In fact, it is important that the user interface can differ in the analysis and reasoning phases in any problem solving. In particular, in the user interface, the user should know what the problems are, what the premise set and conclusions are, etc. The user interface might consist of some kind of natural language processing system that allows the user to interact with the system in a limited form of natural language. Therefore the problem solving system consists of two subsystems. One is an abductive CBR system; another is a deductive CBR system. The major part of both systems is the (either abductive or deductive) case base and the abductive engine or deductive engine. In terms of the CBR systems, the abductive case base consists of explanation-oriented facts and rules about the subject or problems at hand; the deductive case base consists of reasoning-oriented predicate-like facts and rules about the subject or problem solving available. The abductive engine and the deductive engine consist of all the processes that manipulate the (either abductive or deductive) case base to derive information requested by the user based on either abductive CBR or deductive CBR.

Based on the above discussion, the following process cycle is of significance for abductive CBR systems:

- Repartition for building an abductive case base
- Retrieve the most similar abductive cases

- Reuse the abductive cases to attempt to give the explanation to the problem(s)
- Revise the retrieved abductive cases
- Retain the new abductive case as a part of a new abductive case base.

This is called the R^5 model of an abductive CBR system [289], which corresponds to the R^5 model for deductive CBR systems introduced in Section 5.4.

5.5.5 Summary

This section showed that abductive reasoning and deductive reasoning can be integrated in the clinical process and problem solving. It argued that the theoretical foundation of backward chaining and Prolog as well as most other logic programming languages is abduction. After discussing the evolution from abduction to abductive CBR and that from deduction to deductive CBR the section integrated abductive CBR and deductive CBR, and proposed a unified formalization for integration of abduction, abductive CBR, deduction and deductive CBR. It also demonstrated that the integration of the abductive CBR system and deductive CBR system is of practical significance in problem solving such as system diagnosis and analysis. The proposed approach will facilitate research and development of abductive CBR and deductive CBR.

5.6 Rule-based Models for Case Retrieval

This section will provide a rule-based formalization of case retrieval in case-based reasoning (CBR) in a unified way, which includes the rule-based case retrieval based both on similarity relations and on similarity metrics as well as on fuzzy similarity relations. The proposed approach shows that the rule-based case retrieval and the separation between similarity relations and similarity metrics are significant for CBR.

5.6.1 Case Retrieval with Similarity Relations

Case retrieval based on similarity relations is not a new topic for research and development in CBR, because almost every CBR system uses this idea. However, they are basically domain-dependent and can only be viewed as a model for a special CBR principle. Further, there is no complete treatment of case retrieval from a logical viewpoint. In what follows, this section attempts to investigate case retrieval in a domain-independent way and proposes a general architecture for case retrieval from a logical viewpoint.

Definition 15. Let a relation S_p on the possible world of problems W_p and a relation S_s on the possible world of solutions W_s be similarity relations. Then (W_p, S_p) and (W_s, S_s) are called *similarity systems*.

Based on this definition, the goal of case retrieval is to create the relationship between the similarity with respect to S_p and that in the sense of S_s. From a rule-based viewpoint, rule-based case retrieval falls into the following different categories:

1. IF p_1 is similar to p_2 in the sense of S_p, THEN s_1 is similar to s_2 in the sense of S_s
2. IF p_1 is similar to p_2 in the sense of S_p, THEN s_1 is not similar to s_2 in the sense of S_s
3. IF p_1 is not similar to p_2 in the sense of S_p, THEN s_1 is similar to s_2 in the sense of S_s
4. IF p_1 is not similar to p_2 in the sense of S_p, THEN s_1 is not similar to s_2 in the sense of S_s.

where p_1 and p_2 are two problem descriptions in W_p; s_1 and s_2 are their corresponding solution descriptions in W_s respectively, S_p and S_s are similarity relations. In domain-dependent CBR systems, p_1 can be referred to as a current problem description or a normalized enquiry p_0, which is usually used in CBR publications such as [72][74], and s_1 to a potential solution of p_0, which will be obtained using CBR. Now each of these four categories will be discussed in some detail.

Category 1 reflects the traditional basic hypothesis of CBR: *Similar problems have similar solutions* [172][274] and it also reflects the CBR principle given by Dubois et al. in [72][74]: "Similar situations give (or may give) similar outcomes." It still reflects the previously mentioned Analogous Assumption in a broader domain (see previous section). It can be also considered as the primary motivation of CBR.

Category 2 reflects some practical cases from real life situations [289]. In business, one can also encounter such cases, for example, two similar used cars might be sold with very different prices. Dubois et al. classify these kinds of problems into non-deterministic problems [74], while the first category into deterministic problems. However, almost all CBR systems ignore this category [291][295].

Category 3 reflects another social phenomenon, which seems to have not been touched in CBR, because case retrieval is essentially the starting point for the CBR cycle [315], where a case is useful only if p_1 is similar to p_2 in the sense of S_p which has limited further insight into this category in the terms of CBR. However, perhaps it should be discussed from a fuzzy viewpoint, because "not similar to" can be, in essence, weakened and become an intermediate state between "similar" and "not similar to." [289]

Category 4 has been introduced in [289]. From a logical viewpoint, category 4 is logically equivalent to the following category, because category 4 is the contrapositive form of category 5:

5. IF s_1 is similar to s_2 in the sense of S_s, THEN p_1 is similar to p_2 in the sense of S_p.

This category reflects "Two problems are similar if the solutions are similar" [172]. However, the solutions are sometimes unknown, thus it is a paradox. This paradox is resolved by the observation that the similarity of solutions can be stated a *priori*; e.g., two solutions can be regarded as similar if they are equal or if the solution transformation is simple. Further, we call reasoning or problem following category 5 *abductive CBR,* a reasoning combining abduction and experience-based reasoning. The reason is that it can be considered as an integration of CBR and abduction in AI [244](see Section 5.5). Because abduction has already had an extensive research and applications history in AI [166], category 4, and in particular category 5 will certainly be of further interest in CBR research in the near future [289]. In fact, Leake has done some work in this aspect [165][166].

So far, five categories have been introduced for modelling rule-based case retrieval in CBR based on similar relations. From a viewpoint of logic, the first four categories are logically independent of each other, while category 4 and category 5 are logically equivalent. Further, it is usually insufficient to model case retrieval only using similar relations, because they can not be used to model the situations such as "x is possibly similar to y", "x is more similar to y" and "x is most similar to y" in a satisfactory way. This means that fuzzy similarity relations and similarity metrics should be used to model the mentioned situations.

It is worth noting that in some cases the implication from IF to THEN is fuzzy, Thus above five categories can be weakened into the five following categories from a viewpoint of fuzzy logic [289]:

1. IF p_1 is similar to p_2 in the sense of S_p, THEN it is possible that s_1 is similar to s_2 in the sense of S_s
2. IF p_1 is similar to p_2 in the sense of S_p, THEN it is possible that s_1 is not similar to s_2 in the sense of S_s
3. IF p_1 is not similar to p_2 in the sense of S_p, THEN it is possible that s_1 is similar to s_2 in the sense of S_s
4. IF p_1 is not similar to p_2 in the sense of S_p, THEN it is possible that s_1 is not similar to s_2 in the sense of S_s
5. IF s_1 is similar to s_2 in the sense of S_s, THEN it is possible that p_1 is similar to p_2 in the sense of S_p.

Dubois has discussed the first case in [72], while the other four cases have not been touched.

5.6.2 Case Retrieval based on Similarity Metrics

So far, this section has investigated the rule-based case retrieval based both on similarity relations and on fuzzy similarity relations. Furthermore, the strength of the implication between the premises and consequences of rules in the proposed categories, or the so-called strength of a rule, also plays an important role in CBR. For example, Dubois et al. [72] pay attention to the strength of the implication between the premises and consequences of the mentioned rules using the following CBR principle:

"The more similar are the problems in the sense of S_p, the more similar are the corresponding solutions in the sense of S_s."

This principle seems to be unconsciously affected by the research and development of fuzzy rule-based expert system (RBES), because the strength of rules attracted much attention in the 1990's. In order to examine the strength of rules and also model the just mentioned CBR principle, the concept of a similarity relation is insufficient to model "more similar to", because it requires a similarity metric, which must provide a quantitative measurement or comparison (see Section 5.2.7). In what follows, the subsection turns to case retrieval based on similarity metrics:

1. IF p_1 is more similar to p_2 in the sense of S_p, THEN s_1 is more similar to s_2 in the sense of S_s
2. IF p_1 is more similar to p_2 in the sense of S_p, THEN s_1 is not more similar to s_2 in the sense of S_s
3. IF p_1 is not more similar to p_2 in the sense of S_p, THEN s_1 is more similar to s_2 in the sense of S_s
4. IF p_1 is not more similar to p_2 in the sense of S_p, THEN s_1 is not more similar to s_2 in the sense of S_s.

where S_p on the possible world of problems W_p and S_s on the possible world of solutions W_s are similarity metrics. Now each of them will be examined in some detail.

Category 1 reflects the CBR principle mentioned in this subsection. This category can be formalized as:

$$\text{IF } S_p(p_1, p_2) \geq S_p(p_1, p_3), p_3 \in W_p, \text{ THEN } S_s(s_1, s_2) \geq S_s(s_1, s_3), s_3 \in W_s \tag{76}$$

In fact, the following model is more useful for a domain-dependent CBR system, because any case retrieval algorithm is aimed at the most similar problem and its corresponding solutions.

1'. IF p_1 is most similar to p_2 in the sense of S_p, THEN s_1 is most similar to s_2 in the sense of S_s

This category can be modelled as:

$$\text{IF } S_p(p_1, p_2) \geq S_p(p_1, p), \forall p \in W_p, \text{ THEN } S_s(s_1, s_2) \geq S_s(s_1, s), \forall s \in W_s. \tag{77}$$

In usual CBR systems, if p_1 is the current problem or normalized inquiry, then it is problems in the set of problems of case base, P, rather than all problems in W_p which are evaluated if they are similar to p_1 and how similar they are to p_1, therefore (77) can be simplified as:

$$\text{IF } S_p(p_1, p_2) \geq S_p(p_1, p), \forall p \in P, \text{ THEN } S_s(s_1, s_2) \geq S_s(s_1, s), \forall s \in Q \tag{78}$$

where Q is the set of solutions in case base, which corresponds to P.

In fact, (77) and (78) can be improved with the following two models respectively:

$$\text{IF } S_p(p_1, p_2) = \max_{\forall p \in W_p} S_p(p_1, p), \text{ THEN } S_s(s_1, s_2) = \max_{\forall s \in W_s} S_s(s_1, s) \tag{79}$$

$$\text{IF } S_p(p_1, p_2) = \max_{\forall p \in P} S_p(p_1, p) \text{ THEN } S_s(s_1, s_2) = \max_{\forall s \in Q} S_s(s_1, s) \tag{80}$$

The last two models can be considered as a theoretical foundation for the current mainstream in case retrieval studies. It is also useful for any areas in which information search or retrieval play an important role.

Category 2. reflects some practical cases in real life [289]. It is worth noting from a logical viewpoint that this category might be invalid, although "IF p_1 is similar to p_2 in the sense of S_p, THEN s_1 is similar to s_2 in the sense of S_s," which is the first category in the previous section. In this case, "not more similar" lies between "more similar" and "similar" from a viewpoint of fuzzy logic. Therefore, this category can be considered as a weaker form of category 1 mentioned in the previous section.

Category 3 reflects a dual problem to the category 2, which has not attracted much attention in CBR, because case retrieval is the initial point in the study on CBR.

As to category 4, it is also logically equivalent to the following category:

5. IF s_1 is more similar to s_2 in the sense of S_s, THEN p_1 is more similar to p_2 in the sense of S_p.

This category reflects "Two problems are more similar if the solutions are more similar" and therefore belongs to a special case of *abductive CBR*, which was mentioned in the previous section.

More specifically, this category can be formalized as:

$$\text{IF } S_s(s_1, s_2) \geq S_s(s_1, s_3), s_3 \in W_s \text{ THEN } S_p(p_1, p_2) \geq S_p(p_1, p_3), p_3 \in W_p \tag{81}$$

In fact, the following model might be more useful for an *abductive CBR*,

5'. IF s_1 is most similar to s_2 in the sense of S_s, THEN p_1 is most similar to p_2 in the sense of S_p

This category can be modelled as:

$$\text{IF } S_s(s_1, s_2) \geq S_s(s_1, s), \forall s \in W_s, \text{ THEN } S_p(p_1, p_2) \geq S_p(p_1, p), \forall p \in W_p \tag{82}$$

In the usual case retrieval, the model (82) can also be simplified as:

$$\text{IF} S_s(s_1, s_2) \geq S_s(s_1, s), \forall s \in Q, \text{THEN } S_p(p_1, p_2) \geq S_p(p_1, p), \forall p \in P \tag{83}$$

and considered as a theoretical foundation for studies of abductive CBR. In fact, (82) and (83) can be replaced with the following two models respectively:

$$\text{IF } S_s(s_1, s_2) = \max_{\forall s \in W_s} S_s(s_1, s), \text{THEN } S_p(p_1, p_2) = \max_{\forall p \in W_p} S_p(p_1, p) \tag{84}$$

$$\text{IF } S_s(s_1, s_2) = \max_{\forall s \in Q} S_s(s_1, s) \text{ THEN } S_p(p_1, p_2) = \max_{\forall p \in P} S_p(p_1, p) \tag{85}$$

These last two models can be considered as a theoretical foundation for *abductive CBR*.

From a logical viewpoint, "If P Then Q" means that if P is true then Q is at least true; that is, from a computational viewpoint, $t(P) \leq t(Q)$, where $t(\ )$ is truth value of a proposition. If t is a similarity metric, then it is essentially the same as that in [74], in the latter, t is intentionally replaced by two different similarity metrics S and T. Based on this idea, (80) and (85) can be specified as:

$$S_p(p_1, p_2) = \max_{\forall p \in P} S_p(p_1, p) \leq S_s(s_1, s_2) = \max_{\forall s \in Q} S_s(s_1, s) \tag{86}$$

$$S_s(s_1, s_2) = \max_{\forall s \in Q} S_s(s_1, s) \leq S_p(p_1, p_2) = \max_{\forall p \in P} S_p(p_1, p) \tag{87}$$

The last two models are useful for implementing a concrete CBR system.

5.6.3 Abductive Case Retrieval vs Deductive Case Retrieval

The previous subsections have focused on rule-based case retrieval, which is based on similarity relations, fuzzy similarity relations, and similarity metrics. Further, Section 5.5.3 introduced abductive CBR. Therefore, it is necessary to divide case retrieval into *abductive case retrieval* and *deductive case retrieval*. Case retrieval in abductive CBR is called *abductive case retrieval*, while case retrieval in deductive CBR is called *deductive case retrieval*. In what follows, the subsection describes some more about abductive case retrieval and deductive case retrieval in a parallel way.

For brevity, assume that the abductive case base is denoted as $B = (E, P)$, where E is a subset of explanation descriptions in W_e, the possible world of explanations that is associated with similarity S_e. P is a subset of problem descriptions W_p associated with similarity S_p. An abductive case is denoted as an ordered pair (e, p), where $e \in E$ and $p \in P$.

In the context of abductive CBR, if the current problem p' is similar to p of the case (e, p) in abductive case base B, then one can conclude that the explanation of p', e', is also similar to e, according to (72). However, it is obvious that different understanding of similarity leads to different abductive case retrieval. Based on this idea, abductive case retrieval can be examined taking into account similarity relations, fuzzy similarity relations, and similarity metrics respectively. It should be noted that the models under consideration can be considered as the specialization of (72), but denoted by a production rule. So it is easy to use these production rules and (72) to perform abductive CBR in different settings.

For deductive case retrieval, assume that the deductive case base is denoted as $C = (P, Q)$, where P is a subset of the possible world of problem descriptions W_p associated with a similarity S_p and Q is a subset of the possible world of solution descriptions W_s associated with a similarity S_s. A deductive case, c, is denoted as an ordered pair (p, s), where $p \in P$ and $s \in Q$. In the context of CBR, if the current problem p' is similar to p in the deductive case (p, s) in the deductive case base C, then one can conclude that the solution of p', s', is also similar to s, according to (75).

Based on the above terminology, one can discuss abductive case retrieval and deductive case retrieval either separately or in a unified way, just as in the previous subsections. In practice, abductive case retrieval and deductive case retrieval can be examined in a parallel way, based on the basic model of integration of abductive CBR and deductive CBR in Section 5.5.4.

5.6.4 Summary

This section provided a rule-based formalization of case retrieval in CBR, in which case retrieval was classified into five categories with respect to different understanding of similarity. Then it investigated them based on both similarity relations and similarity metrics as well as on fuzzy similarity relations. This section also proposed that abductive case retrieval and deductive case retrieval can be discussed and used in a parallel way, in practice. The proposed approach showed that the rule-based case retrieval and the separation between similarity relations and similarity metrics are significant for CBR.

As is well known, fuzzy inference is based on the fuzzy rule: IF fuzzy set A, THEN fuzzy set B can be represented by a fuzzy relation R such that $B = A \times R$ (compositional rule of inference) [355]. These relations are called “fuzzy implication operators”, e.g. Zadeh implication operator, Mamdani implication operator, etc. Therefore, as a natural generalization, this fuzzy relation or compositional rule of inference will be used to improve the proposed rule-based case retrieval models in next section.

5.7 Fuzzy Rule-based Case Retrieval

This section will provide fuzzy rule-based models of case retrieval and its implementation from a viewpoint of fuzzy logic. This is a further development of rule-based models for case retrieval discussed in the previous section, because the uncertain knowledge and inexact matching are involved in case retrieval. It can therefore argue that CBR is a unifying mechanism for integrating rule-based reasoning and similarity-based reasoning. Therefore, this is also a further insight into integration of abductive CBR and deductive CBR (see Section 5.5) from a logical viewpoint[1].

5.7.1 Uncertainty and Incompleteness in CBR

Uncertainty and incompleteness pervade the CBR reasoning process [28]. Uncertainty exists in the semantics of abstract features used to index the cases, in the evaluation of the similarity measures computed across these features, in the determination of relevancy and saliency of the similar cases, and in the solution adaptation phase. Incompleteness is present in the partial domain theory used in the indexing and retrieval, in the (usually) sparse coverage of the problem space by the existing cases, and in the description of the probe.

Soft computing, in particular fuzzy logic has been shown to treat uncertainty and incompleteness of knowledge successfully in the last few decades [29]. Therefore, this section will apply fuzzy logic to rule based case retrieval.

1. Sun [278] outlined a unifying mechanism for carrying out the basic processes of rule-based reasoning and similarity-based reasoning from a connectionist viewpoint.

5.7.2 Similarity Neighbourhood and Similarity Uniform Mapping

First of all, two concepts are introduced with respect of similarity metrics. Assume that (W_p, S_p) and (W_s, S_s) are two similarity systems (see Section 5.6.1) in which S_p on the possible world of problems W_p and a relation S_s on the possible world of solutions W_s are similarity metrics (see Section 5.2.7).

Throughout this section, Let r be a real number. The *open similarity disc* of radius $r > 0$ centred at p_0 means the set of problems p in W_p such that $S_p(p, p_0) > r$. Generally speaking, r is approximate to 1 in the case retrieval. The open similarity disc of radius r at p_0 is denoted as $S_p(p_0, r)$. Suppose $w > 0$ and (p_0, s_0) is a case in the case base C, then $S_s(s_0, w)$ is an open similarity disc of radius w at s_0.

Definition 16. A function f: $W_p \to W_s$ is called *uniformly similar* on the domain W_p if for any $r > 0$ there exists a $w > 0$ such that

$$\text{if } p_1, p_2 \in W_p \text{ and } S_p(p_1, p_2) > r \text{ then } S_s(f(p_1), f(p_2)) > w \tag{88}$$

Further for any $p \in P$, $(p, f(p)) \in C$, where C is the case base. f is called a *uniform solution function*.

The motivation for introducing this concept is from the concept of conformal mappings in complex analysis [161] and uniformly continuous functions in real analysis [255]. The goal of introducing this concept is to build a formal connection between W_p and W_s. This is an important condition for examining fuzzy rule-based case retrieval.

5.7.3 Fuzzy Rule-based Models for Case Retrieval

This subsection examines fuzzy rule-based models for case retrieval based on Zadeh's composite rule [355] by beginning with the following real world scenario in e-commerce.

A customer uses the interface of the e-commerce system, for example, CMB [292], to submit a requirement with the problem description p_0, which may be fuzzy and uncertain owing to its description in natural language. The search agent of CMB will search the case base C in CMB to try to find if there is a case $c_1 = (p_1, s_1)$ in C, such that p_0 is completely matched over p_1; that is, $p_0 \equiv p_1$. If so, the goods with the solution description s_1 are the most satisfactory solution to the requirements of the customer according to the experience of CMB. Otherwise, the search agent has to activate the similarity-based mechanism, which is based on similarity metric S_p, to search the case base C in CMB to obtain the most similar p_1, which is in a case $c_1 = (p_1, s_1)$ in C, such that $S_p(p_0, p_1) = \max_{\forall p \in P} S_P(p_0, p)$ with similarity degree r. Then the most satisfactory solution to the requirements of the customer is s_0 such that $S_s(s_0, s_1) = \max_{\forall s \in Q} S_s(s_1, s)$ with similarity degree w (see Section 5.7.2), according to the experience of CMB.

It should be noted that s_0 is only the most satisfactory good to meet the requirements of the customer. However, it may not be matched completely to the requirements of the customer. In this case, case adaptation is necessary, if the customer asks to tune the requirements or the product with an adjustment of the problem descriptions.

As is known, a case (p, s) can be represented as a rule (see Section 2.5.2); that is, $p \to s$. Therefore, above discussion can be expressed in the following brief form:

$$\frac{p_0, p_0 \sim p_1, p_1 \to s_1, s_1 \sim s_0}{s_0} \tag{89}$$

where, p_0 is the problem description of the customer, $p_0 \sim p_1$ means that p_0 and p_1 are most similar, with the similarity degree r, in the sense of S_p, $p_1 \to s_1$ is the case retrieved from the case base C based on the similarity-based retrieval algorithm. $s_1 \sim s_0$ means that s_0 and s_1 are most similar, with the similarity degree w, in the sense of S_s, and s_0 is the most satisfactory solution to the requirements of the customer with the similarity degree $r \times k \times w$, where k is the certainty factor of $p_1 \to s_1$. Usually, $k = 1$ because the case in the case base is the result of experience, i.e. a successful solution to a previous problem.

Model (89) is an implementation-oriented realization of the CBR world (see Fig. 5.4). It should be noted that in practice s_0 might be any one of $S_s(s_1, w)$, which is an open similarity disc of radius w at s_1 (see 5.7.2)[1]. This is the fundamental reason why case adaptation is necessary. The process of finding s_0 such that $s_1 \sim s_0$ is at least an important part of case adaptation. $S_s(s_1, w)$ can be also considered as the *selection world* of the customer using the CMB. A special case is $s_0 \equiv s_1$; that is, s_0 is identical to s_1. This degenerates from model (89) to the cases that many studies have implicitly or explicitly done such as [278]. In the later case (89) is simplified as:

$$\frac{p_0, p_0 \sim p_1, p_1 \to s_1}{s_1} \tag{90}$$

The rest of this subsection turns to fuzzy rule-based case retrieval. Assume that p_0 corresponds to $\tilde{P}_0$, $p_0 \sim p_1$ corresponds to $\tilde{F}_{01}$, $p_1 \to s_1$ corresponds to $\tilde{F}_{11}$, $s_1 \sim s_0$ corresponds to $\tilde{F}_{10}$, and s_0 corresponds to $\tilde{S}_0$. Then, according to the compositional rule of inference of Zadeh [355] and the above model (89), the following is obtained:

$$\tilde{S}_0 = \tilde{P}_0 \circ \tilde{F}_{01} \circ \tilde{F}_{11} \circ \tilde{F}_{10} \tag{91}$$

where $\tilde{P}_0$ is a fuzzy set in W_p. $\tilde{F}_{01}$, $\tilde{F}_{11}$, and $\tilde{F}_{10}$ are a (fuzzy) similarity metric, a fuzzy rule and a fuzzy similarity metric in $W_p \times W_s$ respectively, and $\tilde{S}_0$ is a fuzzy set on W_s. This is a theoretical foundation for fuzzy rule-based case retrieval. In the case $s_0 \equiv s_1$, $\tilde{F}_{10}$ is an unit matrix, (91) is then simplified into:

$$\tilde{S}_0 = \tilde{P}_0 \circ \tilde{F}_{01} \circ \tilde{F}_{11} \tag{92}$$

When, $\tilde{P}_0$, $\tilde{F}_{01}$, $\tilde{F}_{11}$ and $\tilde{S}_0$ are only a numerical similarity metric respectively, (91) is, essentially, degenerated into the form discussed in [278].

5.7.4 Summary

This section proposed fuzzy rule-based models for case retrieval based on compositional rule of inference of Zadeh [355], and its implementation from a viewpoint of fuzzy logic. Therefore it

1. The idea here is different from that in [72][74].

showed that CBR is a unifying mechanism for integrating rule-based reasoning and similarity-based reasoning.

5.8 Concluding Remarks

This chapter proposed a general theory of case-based reasoning. More specifically, it extended the concept of similarity given by Zadeh, examined similarity relations, fuzzy similarity relations, similarity metrics, and their relationships. It thus proposed six different types of similarity relations and corresponding similarity metrics. Then it provided a theoretical formalization for building case bases with three novel algorithms based on similarity relations and fuzzy similarity relations. It also proposed a R^5 model for case based reasoning. Furthermore it examined abductive CBR and deductive CBR and proposed a knowledge-based model for integrating abductive CBR and deductive CBR. Finally it proposed rule-based models for case retrieval based on similarity relations, fuzzy similarity relations and similarity metrics, and fuzzy rule-based models for case retrieval based on composite rule of inference of Zadeh [355]. The proposed approaches in this chapter can help towards a firm theoretical foundation of CBR, of which similarity or similarity-based reasoning is at the heart, just as relations are at the heart of relational database [108].

Part - II: Interrelations

Part II is Interrelations of the thesis, which includes three chapters that are at the third level in the Boolean structure: Chapter 6: Case-based reasoning in e-commerce, Chapter 7: CBR in Multiagent systems, Chapter 8: Multiagent systems in e-commerce, as shown in the shaded area of Fig. II

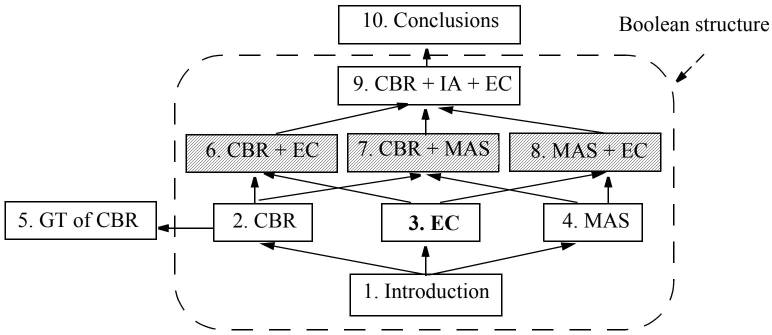


Fig. II. Part II in the Boolean structure of PhD-thesis

6 CBR in E-Commerce

This chapter is the first chapter in the second part of the thesis. It is also the basis for Chapter 9, as shown in the shaded area of Fig. 6.1. This chapter will propose a unified architecture for a CBR-based e-commerce system, and give new insight into the traditional CBR cycle. Then it investigates CBR in intelligent support for e-commerce, product recommendation, product configuration, and product negotiation respectively.

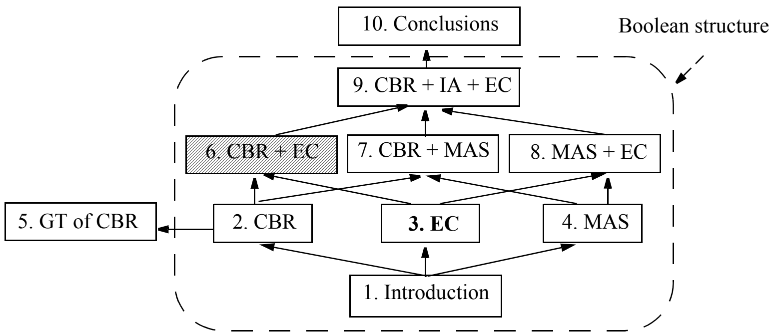


Fig. 6.1 Chapter 6 in the Boolean structure of PhD-thesis

6.1 Introduction

CBR in e-commerce is drawing increasing attention in both fields of AI and e-commerce [334]. For example, it is the first time that CBR in e-commerce was one of workshops in the International Conference on CBR 2001 (ICCBR'01)[1]. In fact, CBR systems have achieved significant practical success in customer support and help desk operations, which are critical e-commerce functions.

Adding intelligence to e-commerce systems as well as other Internet applications is an obvious role for CBR in e-commerce. E-commerce sets out to sell products without the intervention of a sales assistant and in the absence of human sales assistants [112]. Instead, there is a need for e-commerce to have intelligent software assistants to lubricate its sales process. CBR has proved an available technology to create these sales assistants, since catalogue data and data on user behaviour and preferences are basically available from traditional commerce. For example, one can use cases to describe commodities on sale, and use CBR to identify the case configuration that meets the customers' requirements.

1. http://www.ics.uci.edu/~burke/research/cbrec/cfp.html

CBR has found increasing applications in e-commerce as an assistant in e-commerce stores and as a reasoning agent for online technical support [112], as well as an intelligent assistant for sales support or for e-commerce travel agents [172]. The strength of CBR in this area stems from its reuse of the case base, or experience base associated with a particular application, thus providing an ideal way to make personalised configuration or technical information available to the Internet user.

More recently, product search, product recommendation, product configuration, and product negotiation have been the target of research and commercial activity for applying CBR in e-commerce [53][181]. However, it seems that the mentioned applications and activities are very discrete or isolated from a CBR perspective. The relationship between these applications and the traditional CBR-cycle [315] is also not clear. Therefore, the first goal of this chapter is to propose a unified architecture for a CBR-based e-commerce system which covers almost all mentioned activities and give new insight into the traditional CBR cycle. Then it investigates CBR in intelligent support for e-commerce, product recommendation, product configuration, and product negotiation respectively. This chapter thus gives deep insight into how to use CBR in e-commerce and how to improve the understanding of CBR with its applications in e-commerce. These two sides are complementary to each other. Almost all researchers in this area (e.g. [128][319]) have focused on the first part, while the second part has been basically neglected. Another of the contributions of this chapter is the decomposition of case adaptation into problem adaptation and solution adaptation, which not only improves the understanding of case adaptation in traditional CBR, but also facilitates the refinement of activity of CBR in e-commerce and intelligent support for e-commerce.

The rest of this chapter is organised as follows: Section 6.2 proposes a unified architecture of CBR-based e-commerce systems. Section 6.3 investigates CBR in intelligent support for e-commerce. Section 6.4 looks into product recommendation and Section 6.5 examines product configuration. Section 6.6 examines negotiation with CBR. Section 6.7 ends this chapter with some concluding remarks.

6.2 A Unified Architecture of CBR-based E-commerce Systems

CBR has considerable potential for developing Web-based intelligent systems. Several Web-based systems that use CBR are already in existence [113]. A characteristic of these applications is that they involve implementations of existing CBR technology in a Web context: the client has a remote dialogue through the browser with the CBR application at the server side [319]. Based on this idea, this section introduces a general architecture for Web-based CBR systems after [113], as shown in Fig. 6.2. In this architecture, each client has its own case base (i.e. Client CB)

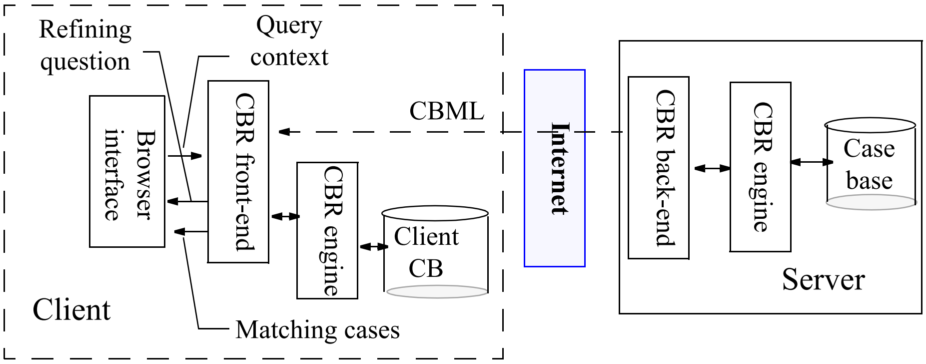


Fig. 6.2 General architecture of Web-based CBR systems after [113]

and a browser based interface at the front-end that connects to the server at the back-end; all the case base processing is performed at the back-end. In the distributed architecture the CBR engine is downloaded to the client side to allow for the later stages of processing to be performed there.

The details of the operation of this Web-based CBR system can be explained in the context of electronic sales. The interface allows the customer to describe his demands p'. This is normalised into a partial case description that is passed to the CBR front-end as a Query context (for brevity, it is still p'). Initially, this will be passed to the CBR back-end to find matching cases (for example, $c = (p, s)$). If too many potential matches are found the CBR engine will identify which feature of the matched cases is the most discriminating. This is then passed to the user interface as a Refining Question, which is still denoted as p'. This process is a typical *demand adaptation*. The response to this request for extra information, which is the result of carrying the demand adaptation, is passed to the back-end as a refined Query context, which is still p' for brevity. This process is continued until such time as the Query context is sufficiently discriminating. At this point, matching cases are passed to the user interface as a *product recommendation*.

In this process, as the Query context is refined, the set of potentially matching cases reduces, for example, $c_1, c_2, \ldots, c_n$ are reduced to $c_1, c_2, \ldots, c_m$, where $m < n$. The advantage of the proposed architecture is that once this set is sufficiently small it can be passed to the front-end where processing can be completed without further interaction across the network. The decision as to when precisely to do this depends on the size of the cases and the response time across the network.

However, from the viewpoint of CBR, this architecture has not realized the main CBR-cycle in an explicit way, because it only stressed case retrieval and problem (demand) adaptation. In what follows, the architecture, in particular the CBR engine, will be improved in order to express how the CBR-cycle corresponds to the activities of real business world.

As already mentioned in Chapter 5, the theoretical foundation of the CBR cycle is based on:

$$\begin{array}{c} P' \\ P' \sim P \\ P \to Q \\ \underline{Q \approx Q'} \\ \therefore Q' \end{array} \tag{1}$$

where P, P' and Q, Q' are composite propositions in a general setting. $\sim$ and $\approx$ are two different similarity metrics and $P \to Q$ is a rule-based representation of a case base, for brevity. One can use other representation models of case bases such as a relational representation as mentioned in Chapter 2.

Similar to the above discussion, the model (1) will be expounded in the context of electronic sales in a revised way, which constitutes the basis for applying CBR in e-commerce or a CBR e-commerce system (CECS). Noted that the aforementioned Web-based CBR system is a concrete realization of a CECS. The customer describes his demand p' to the CECS through its interface. This demand is normalised into an officially structured problem description p' (for brevity). Then the CECS uses its similarity metric mechanism (based on $\sim$) to search and retrieve its case base, which consists of case bases, each of which is denoted $c = (p, q)$, where p is the structured problem description and q is the solution description. In the CECS, problem description and solution description correspond to (customer) demand description and product description respectively. The case search and retrieval process is basically to find out the case set,

$$C(p') = \{c| \ c = (p, q), p \sim p'\} = \{c_1, c_2, \ldots, c_n\} \tag{2}$$

where n is a positive integer number. Usually, $C(p') = \{c_1, c_2, \ldots, c_n\}$ satisfies the following property: For any $i \in \{1, 2, \ldots, n-1\}$ and $c_i = (p_i, q_i)$,

$$s(p_i) \le s(p_{i+1}) \tag{3}$$

where $s(\;\cdot\;)$ is the similarity degree.

If n is reasonably small, then the CECS will directly recommend the product descriptions of $c_1, c_2, \ldots, c_n, q_1, q_2, \ldots, q_n$, through the user interface. This process is usually case reuse. If n is very large[1], the CECS has to recommend the product descriptions of the first m cases in $c_1, c_2, \ldots, c_n$; that is, $q_1, q_2, \ldots, q_m$, to the customer, in order to meet the needs of the customer, where $1 \le m < n$. This process can be called *product recommendation*. More generally, it can be considered as *case re-commendation*, because the CECS usually provides the customer with not only the re-commended products but also the customer demand description.

Case recommendation has not been mentioned in traditional CBR (or CBR-cycle) and thus is a new concept for CBR motivated from CECSs. Based on the above discussion, product (case) recommendation is one process following case retrieval. More specifically, case recommendation is a more general form of case reuse. Therefore, case recommendation is a process necessary for applying CBR in e-commerce. In this way, the traditional CBR cycle has been improved with adding case recommendation to it. Product recommendation will be examined in Section 6.4 in some detail.

After the customer obtains the recommended product descriptions from the CECS, he will evaluate them and then select one of the following:

1. Accept one of the recommended products, q_k, and order it, where $1 \le k \le m$
2. Adjust his demand descriptions p' and then send them to the CECS
3. Refuse the recommended products and leave the CECS.

Among these three cases only the first two require further discussion.

1. For example, if one uses Google to search for “e-commerce”, Google will return more than 3,200,000 Webpages.

For the first case, the deal was successfully done and the CECS routinely retains the successful case $c_k = (p_k, q_k)$ in the case base. At the same time, the CECS has reused the case successfully. This means that the CECS has done *case reuse and case retention,* which are two important components in the traditional CBR cycle.

For the second case, the adjustment of demands is the process of problem adaptation that corresponds to demand adaptation. The big difference between e-commerce and traditional commerce lies really here; that is, it is difficult for a customer in the traditional commerce to adjust his demand when he is in the market. Usually what he can do is to buy what he sees (BWS). However, in e-commerce, a customer has a much broader space for selecting products. In fact, all available products in the Internet might be searched and selected by any customer if he can access the Internet. In this case, he usually adjusts his demands and tries to get more satisfactory products. Therefore, *problem adaptation* or *demand adaptation* is an important part for a CECS. It is also one of the main features of e-commerce differing from traditional commerce.

After having adjusted the demands, the customer then submits it to the CECS. The CECS will do case retrieval and case recommendation once again. Therefore, the problem submission, case retrieval, case recommendation, and problem (demand) adaptation constitutes a cycle. This is the first cycle of the CECS, which differs from the CBR cycle. However, from a theoretical viewpoint, this cycle only realizes the model (1) partially; that is, it realizes the following model:

$$\begin{array}{c} P' \\ P' \sim P \\ \underline{P \rightarrow Q} \\ \therefore Q \end{array} \tag{4}$$

It is obvious that model (4) lacks the beauty of similarity. From a technical viewpoint, the realization of (4) in a CECS is a simplification of real word business scenarios.

Now let us develop the above example naturally: After problem or demand adaptations, the CECS must be aware that the recommended products, for brevity, $q_1, q_2, \ldots, q_m$, cannot meet the demands of the customer completely, although the customer is still interested in meeting his demands with the CECS. What the CECS can do in this case is to change the product descriptions (that is, q'). The process of adjusting the product descriptions is essentially *product* *configuration*. It can be also called *solution adaptation* in the terms of CBR. In the CECS *solution adaptation* can be replaced by *product adaptation*.

In the product adaptation, the CECS uses its similarity metric mechanism (based on $\approx$) to search and retrieve other available products q' rather than that in its own case base. For example, the CECS can use its special search engine to search for the most similar product q'_i to q_i for any $i = 1, 2, \ldots, m$ in the Internet or the case base of its "partner"; that is,

$$C(q') = \{q'_i | i = 1, 2, \ldots, m; \ q'_i \approx q_i\} \tag{5}$$

where, as mentioned, $q_1, q_2, \ldots, q_m$ are prior recommended products. At this point the CECS passed $q'_1, q'_2, \ldots, q'_m$ to the user interface as *recommended products*.

After the customer obtains the recommended product descriptions from the CECS, he will further evaluate the recommended products and select one of the following:

1. Accept one of the recommended products, q'_k, and order it, where $1 \le k \le m$
2. Hope that the CECS performs product adaptation in order to pass him revised recommended products
3. Adjust and then send his demand descriptions p' to the CECS, and at the same time the customer asks the CECS to perform product adaptation in order to pass him revised recommended products
4. Refuse the recommended products and leave the CECS.

Among these four cases only the first three require to study further.

If he selects the first case, then the product adaptation based product recommendation of the CECS is successfully done. The CECS will retain the new case, (p', q'_k) in the case base of the CECS.

However, if the customer selects the second case, then the CECS must perform product adaptation based product recommendation once again until the customer is satisfied or selects the fourth case. This means that the solution adaptation or product adaptation is also a cyclic process. More specifically, case retrieval, case recommendation, and solution (product) adaptation constitute another cycle. This is the second cycle of the CECS, different from the CBR cycle. This cycle also results from the application of CBR in e-commerce. However, from a theoretical viewpoint, this cycle also realizes model (1) partially. In fact, it realizes the following model:

$$\frac{\begin{array}{c} P \\ P \to Q \\ Q \approx Q' \end{array}}{\therefore Q'} \tag{6}$$

It is obvious that model (6) also lacks the beauty of similarity from a theoretical viewpoint. From a technical viewpoint, the realization of (6) in a CECS is a simplification of real word business scenarios. Now, let us turn back to the analysis of the decision making of the customer after he obtains the recommended products.

If he selects the third case, this means that the customer and the CECS are negotiating with each other and trying to reach a compromise. This process is said to be *product negotiation.* This also implies that the case-based product negotiation requires both problem (demand) adaptation and solution (product) adaptation. It is easy to understand that the product negotiation is a cyclic process. Therefore, the integration of problem adaptation and product adaptation is a cyclic process. This is the third cycle of the CECS, which differs from the CBR-cycle. This cycle also results from the application of CBR in e-commerce. Further, from a theoretical viewpoint, this cycle realizes the model (1) completely, because what just described is really the (theoretical) realization of the model (1); that is:

$$\frac{\begin{array}{c} P' \\ P' \sim P \\ P \to Q \\ Q \approx Q' \end{array}}{\therefore Q'} \tag{7}$$

After product negotiation, the customer accepts the recommended product, q', in (p', q') with his revised demand p'. (p', q') is a successful selling case which has not been in the case base. Therefore, the CECS will retain the new case, (p', q'), in the case base of the CECS. Noted that this idea is also different from that in [335], where a retain phase doesn't take place because a successful selling of a product will not lead to an additional product.

So far, this section examined new issues resulted from applying CBR in e-commerce and gave a new insight into the traditional CBR-cycle. The main results can be summarized as follows:

- Decomposition of case adaptation into problem (demand) adaptation and solution (product) adaptation is a necessity for applying CBR in e-commerce
- Solution (product) adaptation is a process following case recommendation providing that the demand adaptation based case recommendation is not successful
- Product negotiation is an integration of problem (demand) negotiation and solution (product) adaptation
- Product adaptation based case recommendation and demand adaptation based case recommendation together can improve the satisfaction of the customer.

In fact, this section has also shown in a unified way why CBR technology can be applied to product recommendation, product configuration, and product negotiation in e-commerce. The correspondence between components of CBR and the mentioned activities can be shown in Fig. 6.3. Applying CBR in e-commerce has been treated here at three levels: logical level, CBR-level, and e-commerce level. This is a new attempt in both CBR and e-commerce.

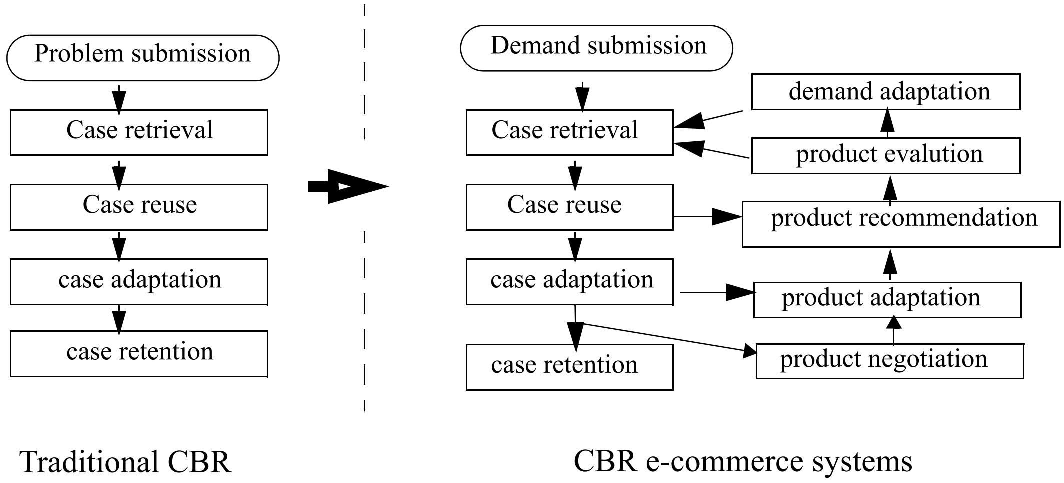


Fig. 6.3 Correspondence between CBR and e-commerce

It should be noted that Wilke etc. [335] also modified the traditional CBR cycle, regarding the different situation in intelligent sales support applications in e-commerce. The goal of their architecture is to expound the sale process with CBR, while the proposed architecture here is not only to model the main activities in e-commerce with CBR, but also to explain why CBR can be applied to product recommendation, product configuration, and product negotiation as well as to give a new insight into traditional CBR; that is, decomposition of case adaptation into problem adaptation and solution adaptation. Only so can case-based product recommendation, case-based product configuration, and case-based product negotiation be treated in a unified way. In other words, case-based product recommendation, case-based product configuration, and case-based product negotiation are all subsystems of the CECS, and intelligent sales support with CBR can be considered a general category for the CECS. Therefore, the following sections will examine intelligent sales support, case-based product recommendation, case-based product configuration, and case-based product negotiation in some detail.

6.3 Intelligent Sales Support with CBR

The notion of intelligent sales support is going to be widely recognised as a new challenge for applying CBR in e-commerce [3]. One of goals using CBR in e-commerce situations is to support a customer with better services and selection facilities [172] (p 91). Another goal of using CBR in e-commerce is to help a producer or sales agent to better classify the customer (or customer segmentation). Today, searching for information or selection of complex products in the WWW is a paintaking task for both customers and business partners, because of rich redundancy of information. The main reason for this well-known deficit in e-commerce is that, compared to usual business procedures, the drawback of today's standard sales solution on the Internet is that they cannot do intelligent sales support [172] (p 93). It is important to put the knowledge and experience of the real sales assistant into the sales support system on the Internet. The system must have enough domain knowledge to be able to aid the customer's search. There is no intelligent support or assistance in the selection of products/services or navigation through complex spaces of available product information or product alternatives on the Internet, although there are some general-purpose search engine available on the Internet such as Google[1] and Openfind[2]. Current product-oriented database search facilities are widely used and organised as being limited in capability for sales support. In order to make these systems customer-friendly, however, new technologies are required to support the user in getting through the vast amount of information [172] (pp 80-90). In what follows, the section investigates intelligent sales support with CBR in e-commerce, in particular in electronic shopping solutions during the retrieval of appropriate products for the customer.

1. http://www.google.com
2. http://www.openfind.com

6.3.1 A General Model for Intelligent Sales Support with CBR

Wilke et al. [335] investigated the sales situation and applied CBR in e-sales. The model of e-sales with CBR is shown in Fig. 6.4. In this model, the selling process via the Internet can be

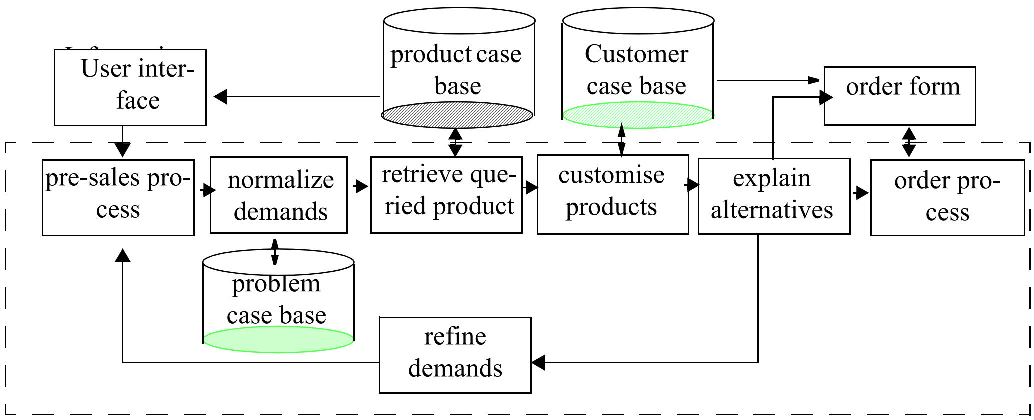


Fig. 6.4 E-sales process with CBR after [53]

classified into the following major steps: The e-sales process starts with a set of demands stated by the customer [53]. After the preliminary presales process [335][286] (p 103), the demands of customers will be translated into a normalised query, which facilitates e-sales processing using WWW techniques. The CBR system will search the product case base to know if there is a problem case that is identical with the normalized query or similar to it. Next, the sales assistant retrieves appropriate product offers using similarity-based reasoning, which is one part of CBR. During a possible adaptation of the retrieved products (synonymous to product configuration, see the later subsection) the retrieved products are tailored based on a case adaptation algorithm to best fit the customers' demands, if necessary. The adapted products are offered to the customer and, at the same time, the difference of the offers and demands is explained to the customers. The customer evaluates these offers during the **revise** phase, which is also a part of CBR, and results in a set of evaluated products. The customer can state that he accepts certain products or parts of the products or he may state that something is not appropriate.

If the adapted products have been accepted by the customer, the information of these products will be processed and stored it into product case base, which is, in essence, the **retain** phase in CBR.

In some cases, the adapted product is not available in the current product case base, and it is basically searched and obtained from other resources on the Web. This means that retain phase can take place because a successful selling of a product leads to an additional product in the product case base, which differs from what is in [167] (p 105). This also means that the product adaptation or product configuration consists of, at least, search. The explanation is in accordance with the sale process in the traditional shop. For example, if a customer visits a car shop, and asks the sales assistant if he can buy a car with property X, the sales assistant at first retrieves all available cars in his shop, and tries to meet the requirement of the customer. If not, the sales assistant may tell the customer "I hope you come here tomorrow, and you can get the required car". After the customer leaves, the sales assistant uses his available resources (e.g. chain partners) to search for the required car. Finally, he may succeed, because a chain partner has one of that kind. Next day, the customer will be informed that a suitable car is available. This step does not occur in the traditional CBR cycle, nor in the model of [167].

In another cases, a new step called *refinement* is introduced. Refinement is, in essence, a kind of adaptation and based on the evaluation given by the customer. Therefore, the adaptation can be considered a cyclic process. The successive refinement of customer's requirement may lead to a cyclic sales process, which is the same as discussed in Section 6.2.

If the customer has obtained a satisfactory result, the overall business process can proceed to the processing of the order.

6.3.2 CBR Supports Sales Staff for Sales Support

Watson and Gardingen [98][319][320] developed a Web-based CBR system, Cool Air, to support engineering sales staff in Western Air, Australia, for selling and installing HVAC[1] systems in 1997. The system has been successfully fielded, and has made impressive revenue [320]. The goals of the system are:

- To reduce the installation specification and quotation time from five days or more to two days
- To reduce the margin of error built in to pricing and thereby produce more competitive quotations, and
- To reduce the burden on head office engineers in checking every detail of every specification.

The system architecture is shown in Fig. 6.5. On the sales staff (client) side a Java applet is used to gather the customer's requirements and send them as an XML query to the server. On the

1. HVAC is a heating, ventilation, and air conditioning system.

server side another Java applet (a servlet) uses this information to query the database, a product case base, to retrieve a set of relevant records. The Java servlet then converts these into XML cases and sends them to the client side applet that uses a nearest neighbour algorithm (NN) to rank the set of cases. This also means that the system uses the XML standard as a communication protocol between client and server side Java applets.

Product cases in this system are represented in XML and stored within a database. Each case (record) comprises 60 fields used for retrieval and many more used to describe the HVAC installations. Case retrieval is a two-stage process. In the first stage the customer's requirements are relaxed through a process of query relaxation. This process takes the original query and relaxes terms in it to ensure that a useful number of records in a broad domain are retrieved from the case base that includes 10k records using SQL retrieval. In the second stage of retrieval the small set of retrieved records are compared by the client-side applet with the original query and similarity degree is calculated using the simple NN algorithm. The key idea is based on the "Many are called but Few are chosen" retrieval algorithm.

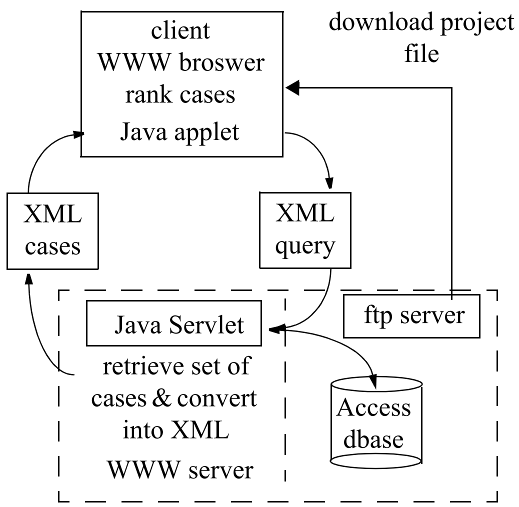


Fig. 6.5 Architecture of CBR system after [320]

6.3.3 CBR-based Catalog in Analog Devices

Analog Devices is one of the major manufacturers and sellers of electronic devices in the USA [172] (p 96). Its online catalog of operational amplifiers is based on CBR, in which the case for an operational amplifier consists of about 40 parameters [172] (p 97). As to the similarity metric, the similarity for all corresponding parameter values are calculated by applying local similarity functions to each pair of corresponding parameters. These local parameter similarities are then used to calculate an overall similarity value for the two devices. The overall similarity is computed as the weighted average of the individual local similarities. It should be noted that the local similarities for discrete and continuous values are calculated in different ways. Discrete similarity metrics are defined by a table which explicitly lists the similarity values for all possible attribute combinations, while continuous similarity metrics are formulated as a function. In many cases, the new CBR system is able to provide customers with information about devices that satisfy their requirements. This reduces the number of calls to Analog Devices' sales support line and enables some engineers to care about more advanced support problem that cannot be solved automatically.

It should be noted that this application does not use case adaptation, although the latter is very important in the CBR-based system. This is because operational amplifiers are unchangeable parts that cannot be reconfigured to the customer's individual demands. Instead of case adaptation, this CBR-based catelog uses *problem adaptation* (or query adaptation) to support intelligent sales as follows [172] (p 99):

The customer enters the parameter values s/he needs into the query form. The CBR system will then retrieve the ten best matches to the request. If the results do not exactly fit the customer's needs, s/he will usually increase the priorities of the parameters that are most important to them. Again, the system displays the ten best matches to the refined query. If the results still do not satisfy the customer, s/he might fill more parameter slots that s/he left empty so far, thus further improving the quality of the returned results. When finally a suitable device has been found, the customer can directly to its detailed data sheet.

Based on above discussion, case adaptation should be therefore decomposed into problem adaptation and solution adaptation. Because a case consists of problem description and solution description, the adaptation in the mentioned CBR catalog system is, in essence, problem adaptation. By problem adaptation, the customer changes or adjusts his problem descriptions to suit the available product (descriptions) if the customer doesn't lost his patience and satisfaction. The philosophy behind problem adaptation is that in many cases, the customer has not crisp but flexible requirements. By solution adaptation, the CBR system tries to change the product configuration in order to meet the crisp requirements of the customers. After this decomposition, it is necessary for a CBR system to look at its running environment to determine if the requirements of the customer are flexible. Further, in practice, both problem adaptation and solution adaptation should be combined in a CBR system to provide powerful intelligent support for e-commerce. Finally, in e-commerce, problem adaptation corresponds to demand adaptation, while solution adaptation corresponds to product adaptation. Demand adaptation and product adaptation will be further examined in the CBR-based negotiation in Section 6.6.

6.3.4 Summary

This section examined the basic framework of intelligent sales support with CBR and reviewed three successful examples of intelligent sales support, which motivated this research to decompose case adaptation into problem adaptation and solution adaptation. It should be noted that compared to other application areas of CBR, intelligent sales support solution for e-commerce is rather new. However, case-based product recommendation, configuration, and negotiation will facilitate research and development of intelligent support for e-commerce, as argued in Section 6.2.

6.4 Product Recommendation

As discussed in Section 6.2, product recommendation is one of most important applications of CBR in e-commerce [3][33][47]. Building effective recommendation systems using case retrieval techniques has drawn some interest in CBR in the past few years [53]. The product recommendation engine represents one of the core parts of a CBR system for e-commerce [3]. It takes inputs as the partial collection being assembled. Its outputs are a subset of the case base with a ranking or rating. This section will look into product recommendation, in particular in case-based product recommendation and collaborative filtering product recommendation. For simplicity, this section refers to e-sales as its real world scenario, without loss of generality.

6.4.1 Case-based Product Recommendation

In product recommendation, a customer is presented with a selection of products from a product catalogue or product case base [33]. Traditionally, there are two approaches to product recommendation [47]: content-based and collaborative (see next subsection). The former is referred to as case-based recommendation [47]. It selects products by matching product descriptions from the catalogue with descriptions of customer preferences and requirements, while the latter is representation-less; that is, content-based approach is based on a more semantically rich representation. The collaborative approach is based on a raw representation of users and assets, while the case-based approach is a representation-based approach. This subsection examines the fundamentals of case-based product recommendation.

The way most case-based product recommendation systems work is as follows [33]: The customer supplies some preferences and demands by filling in an on-screen form. On the basis of the values supplied, the system retrieves and passes one or more product descriptions to the user interface for product recommendation. As mentioned in Section 6.2, case-based product recommendation is a cyclic process, in which the demands of the customer is often adjusted (i.e. demand adaptation). More specifically, it is often the case that the customer will not be immediately satisfied with the recommended products. In most cases, the customer has to return to the on-screen form and adjust the data entered. The adjustment of the demands and product recommendation will be repeated a few times till the deal is done successfully. Based on this discussion, the architecture of a case-based product recommendation system (CPRS), as a subsystem of CECS, is demonstrated in Fig. 6.6. In this architecture, the product case base consists of all prior successful deals; that is, demands and corresponding products. For brevity, this architecture doesn't include product adaptation or product configuration, which will be separately discussed in late section. In what follows, some components will be described in some detail.

- Submit demand. The customer will submit his requirements to the CPRS in a on-screen form. The form will be processed and converted to a normalized demand, which is to be used for the next step
- Retrieve product. The CPRS uses a certain similarity metric mechanism to retrieve the product from the product case base, based on the normalized demand from the customer

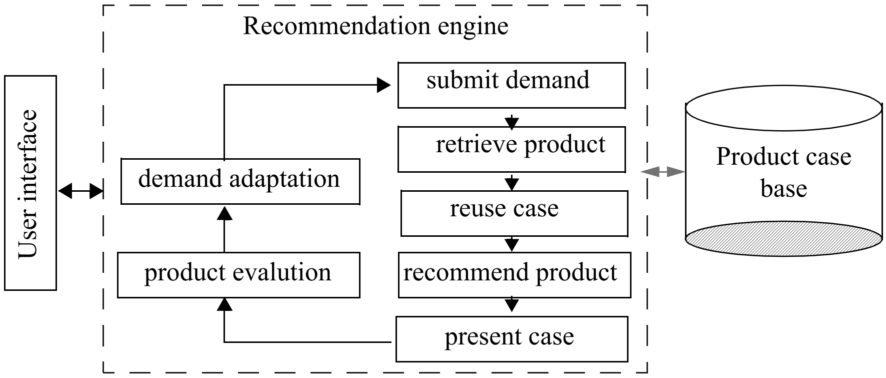


Fig. 6.6 Architecture of a case-based recommendation system

- Reuse case. The CPRS directly passes the retrieved products with the normalized demand to the customer to meet the demand of the customer. However, in e-commerce, a recommendation mechanism is necessary to satisfy the customer, because the number of retrieved cases or products are too large for the customer to know which is the best to meet his own demand
- Recommend product. The CPRS uses special recommendation mechanism to reduce the number of retrieved products and rate them with the importance degree and then pass a fixed number (e.g. 10) of most important products from the retrieved cases to the customer
- Product evaluation. The customer will evaluate the recommended products based on his demand
- Demand adaptation. If the customer doesn't accept any recommended product from the CPRS, then he might adjust his demands and ask the CPRS to try once again. For example, if one wishes to fly from the Gold Coast to Beijing, he has to adjust his demands for travel a few times, because the airport in the Gold Coast is not large so that it is not easy to meet the requirements of each customer satisfactorily.

The CPRS uses a similarity metric mechanism. A customer supplies "ideal" values for some or all of the attributes [33]. The similarity degree between attributes in cases and these ideal values can be computed by local similarity metrics. The degree of similarity of cases in the product case base to the "ideal" case is computed by a global similarity metric (see Section 5.2.). The cases with the highest degrees of similarity to the ideal case are the ones to be recommended to the customer.

It should be noted that this architecture refines the models of case-based recommendation in [3][35][47]**.** It also stresses that product recommendation is a cyclic process. Finally problem adaptation or demand adaptation is first explicitly included in the architecture of case-based recommendation systems.

6.4.2 Collaborative Product Recommendation

As mentioned, another approach to product recommendation in e-commerce is collaborative filtering, which is based on data of users' consumption of assets [47]. In collaborative recommendation systems, customer preferences and requirements are encoded using filters [33]. Filters are absolute: products either satisfy them or they do not. Only products that satisfy the filters are recommended to customers.

The collaborative recommendation process is illustrated in Fig. 6.7, which works with raw data on user's ratings and behaviour and uses this data to produce recommendations.

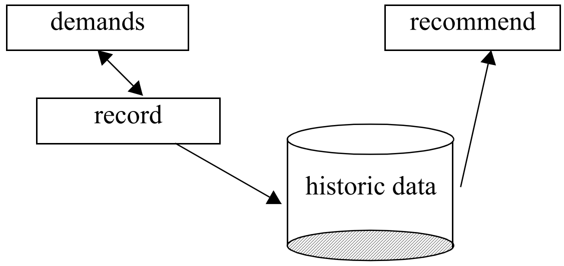


Fig. 6.7 Collaborative product recommendation after [47]

Generally speaking, collaborative recommendation is a three-step procedure:

1. Identify the virtual community associated with a given target user
2. Produce a ranked list of recommendable products
3. Select the top *n* recommendable products as a recommendations, where *n* >1.

Amazon.com uses collaborative recommendation to allow its customers to rate both their purchases and other items in the catalog. Of course, Amazon.com has done more than mentioned. For example, after a customer searches and gets a book, amazon.com lists a few related books for selection by the customer, based on the selling experience "customers who bought this book also bought"[53]. This is a kind of experience-based reasoning or similarity-based reasoning, although Amazon has not declared that it has used CBR in its system.

From a CBR perspective, collaborative recommendation is also a following step after case-retrieval. A search engine such as Google has no special recommendation, because it provides the customer with all retrieved or searched references, and leaves the selection to the customer.

The advantage of case-based recommendation over collaborative commendation is that the set of the recommended products will never be empty, while its disadvantage is that it allows the customer to supply only "ideal" values. Basically speaking, the success of case-based recommendation rests on its ability to order the cases in the product case bases, rather than to filter them [33], Similarity in this kind of system is used to obtain an ordering. The main advantage of collaborative recommendation is that, if enough data is available, good quality recommendations can be produced without requiring representations of the assets being recommended.

6.4.3 Wasabi Personal Shopper- Case-based Recommender Systems

The Wasabi Personal Shopper (WPS) is a case-based recommender system proposed by Burke [35]. WPS provides a conversational interface to a database, based on the principle of CBR. The user examines a suggestion from the system- an item from the catalog- and responds to it with a critique. The system uses the item and the associated critique to formulate a new query returning a new item for consideration. WPS has its roots in a line of FindMe systems. The FindMe technique is one of knowledge-based retrieval. Like other CBR systems, there is a fundamental retrieval mode: Similarity; that is, the user selects a given item from the catalog (called the source) and requests other items similar to it. First, a large set of candidate entities is retrieved from the case base. This set is sorted based on similarity to the source and the top few candidates returned to the user. The architecture of WPS has five basic parts: external information environment, WPS engine, knowledge engineering tools, WPS database, and profiling and reporting part [35].

6.5 Product Configuration with CBR

Product configuration has become an interesting topic of investigation [128], for example, it has been applied to the design of computer systems. Because customisation is one of goals of e-commerce, product configuration will play an important role in some e-business activities such as insurance and travel.

Product configuration is a problem to present the most satisfactory configuration for user requirements while observing the definition of product family [128]. A product family in e-commerce consists of generic products and a variety of parts for implementing the special functions. The goal of product configuration is to arrange products quickly and avoid use of the wrong configurations.

There are many approaches to product configuration such as the constraint-based approach and case-based approach. This section only looks into case-based approach to product configuration, in what follows.

A case-based approach can work without a complete configuration model. Instead, such an approach uses configurations that are similar to user requirements as solutions. Thus, a case-based approach greatly decreases the required effort to obtain and maintain configuration models.

However, a case-based approach sometimes requires a case adaptation phase, in which the similar configurations must be modified to suit the current requirements. Fig. 6.8 is a framework for case-based product configuration. The components in the framework are described below.

- Configuration case base. The configuration case base consists of the prior successful configuration cases, each of which is a pair of customer demands and its corresponding configurations
- Case retrieval. Similar cases are retrieved from the configuration case base in accordance with the similarities between the current demands and a past demand, which was stored as the successful configuration
- Requirement formalization. An object function is dynamically generated by using similar cases to the current demand
- Requirement modification. The well-defined requirement is modified only if there is no configuration which meets the demand of the customer
- Parts database. A parts database contains the definition of a product family. It defines the types of parts, the constraints on parts connectivity, and other kinds of restrictions on the products.

Inakoshi [3] proposed a framework for product configuration that integrates a constraint satisfaction problem (CSP) with CBR, and successfully applied this framework to an online sales system for personal computers.

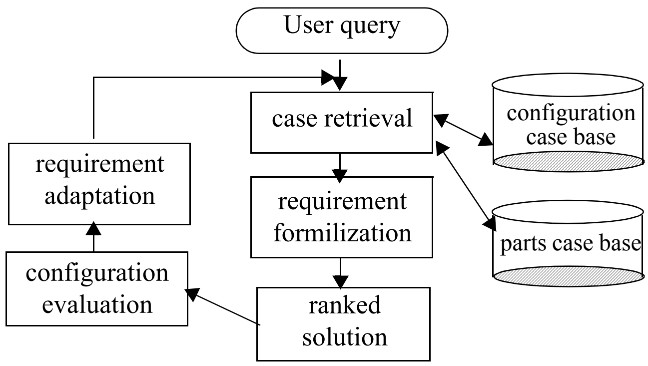


Fig. 6.8 Framework for case-based product configuration based on [3]

However, from a theoretical viewpoint, the proposed framework for CBR-based product configuration is essentially similar to CBR-based product recommendation, although both stem from different real world scenarios; that is, product recommendation and product configuration, because neither CBR-based recommendation systems nor CBR-based configuration systems involves solution adaptation, although both have involved problem adaptation (introduced in

Section 6.2.). Therefore, how to realize product adaptation in CBR-based product configuration is still an open problem.

6.6 Negotiation Using CBR

From a CBR perspective, product configuration is mainly solution adaptation. However, the goal of CBR-based negotiation is the combination of problem adaptation and product adaptation. This combination is necessary for maximizing the customer satisfaction during e-sales that will be examined in what follows. The first part is to examine the CBR-based negotiation and the second part is to introduce an example of a framework for CBR-based negotiation.

6.6.1 CBR-based Negotiation

Negotiation in e-commerce is a process where two parties bargain resources for an intended gain, using tools, and techniques of e-commerce solutions [215][335]. Negotiation is an important part of the selling process on the Internet. In order to support customers in a sufficient way, e-commerce systems should possess the ability of negotiating [335]. However, negotiation is a process with relatively little support to date become of the complexity of the negotiation process, which depends on the complexity of the product or service being negotiated.

The underlying kinds of problem solving strategies divide different approaches for negotiation in e-commerce into two classes: a cooperative approach and a competitive approach [104]. It is assumed that a conflict is in the price of a good with *n* attributes. Competitive negotiation takes place if there is at least a conflict of interests between the buyer and seller. Consequently, there will not be more collaboration than necessary between the buyer and the seller to solve the negotiation problem, although cooperative negotiation tries to get as much collaboration as possible between the two negotiation parties. However, both approaches present two extremes on a continuum of possible underlying problems. In practice, negotiation during the business bargaining process is a compromising negotiation, which lies in between cooperative negotiation and competitive negotiation.

A second criterion to distinguish between different negotiation approaches is the underlying paradigm: the human factor approach, the e-economics/game theory approach, and the computer science approach. The human factor approach provides not much that is directly applicable to e-negotiations. However, it defines the general objective of a satisfactory solution and should, thus, be taken into account. The focus is to manage human factors, like pride, ego, or culture as well as possible, which leads to customer satisfaction if such criteria are fulfilled within a negotiation outcome. The fields of economics and game theory give some valuable insights into the problem. However, these fields are not so applicable in e-commerce, while the computer science approach is most applicable to product search and negotiations in e-commerce.

During the sales process, the customers are navigating through the available products and searching for a product that meets their demands [335]. Some demands are known in advance and additional ones may be discovered during the navigation in the product space. Some demands are *fixed* and must be fulfilled by the product and other demands are more or less *weak or flexible*. Generally speaking, the customer's satisfaction is maximal if the modification of his weak and hard demands is minimal and he finds his product as quickly as possible. The goal of CBR-based negotiation system is to identify these demands in cooperation with the customers and to find a product that fulfils them. During negotiation in e-sales, the CBR-based negotiation system might suggest or even add some new demands or modify some weak demands for the purpose of finding an appropriate product. For configurable products, it is also possible for the CBR-based negotiation system to modify existing products during product (solution) adaptation to meet the customer's demands.

So the task for the CBR-based negotiation system during the negotiation process is the *iterative demand adaptation* and the *iterative product adaptation*. The former is realized by making proposals for adding or changing the demands from the customer, while the latter is done by product adaptation with the goal of finding an agreement point in the multidimensional demand/product space[1]. Therefore the task of the CBR-based negotiation system during the negotiation is, in essence, the iterative adaptation of sales cases or modifications of demands and/or the product. During the negotiation, the customer or the CBR-based negotiation system is allowed to modify the customer demands. If products in the product case base are configurable, it might also be possible to modify the products during negotiation.

1. Noted that the mentioned user demands corresponds to the *problem descriptions*, while the products to the *product descriptions* in the *sale case* in Section 6.4.

6.6.2 A Negotiation Framework with CBR

Zhang and Wong [352] examined a negotiation framework that is called Case-Based Negotiation (CBN). The CBN applies CBR techniques to represent and reuse previous negotiation experiences. The negotiation in the CBN framework involves three fundamental actions:

1. Evaluating the offer from the other agent
2. Defining a negotiation strategy by CBR, and
3. Generating a counter-offer based on proposed strategy.

Given an offer by the opposite agent, if the negotiation agent evaluates it and decides not to accept it, the negotiation agent needs to determine what strategy to follow in the process of generating a counter-offer. A counter-offer is then produced using the proposed concession.

The crucial component in the CBN is a process of defining an appropriate negotiation strategy using CBR techniques. Since the negotiation strategies of each agent are usually hidden to other agents, the agents can only know the results of the negotiation strategies of other agents in the form of offers-counter-offers [352]. During negotiation, a series of offers/counter-offers reflect information related to negotiation strategies. It can be captured as the basis of the previous negotiation experiences. The change from one offer/counter-offer to another offer/counter-offer shows variation of negotiation strategies and information that can be used by agents to perform further negotiation. In the CBN, concessions between offer/counter-offers are used as the basis of negotiation strategy for further negotiation. The concession facilitates reuse and adaptation of previous strategies in similar negotiation contexts. The process for defining a concession by CBR can be composed of three main processes:

1. Retrieving relevant previous negotiation experience
2. Selecting the most matched case based on similarity of retrieved cases and input negotiation context; and
3. Adapting the strategy information in the selected case to propose a concession.

In what follows, how to use CBR to propose appropriate concessions is examined.

Previous negotiation experiences are represented as negotiation cases [352], A negotiation case represents information related to a special buyer or seller in a previous negotiation and captures contextual information and negotiation experience available to the agent. In some detail, a negotiation case can contain the following information:

- Buyer's profile
- Seller's profile
- Product's (e.g. car's) main properties
- Offer made from other agent and concessions used in the previous negotiation session
- Counter-offer made by the agent and concessions used in the previous negotiation session
- Performance information about the feedback of negotiation results.

where, in the context of used car trading, the buyer's and seller's profiles may include name, age, buyer/seller's desire, expected negotiation duration, issues of negotiation (e.g. price, warranty, trade-in, etc.), constraints (e.g. budget), and preferences. The main properties of a used car may include car size, car maker, and car age. It should be noted that the negotiation experience in cases can be extracted from the received offers, the generated counter-offers, and the concessions used in the previous negotiation session.

Rule-based reasoning is applied in the case matching and selection process in the CBN [352]. Case matching is based on fuzzy similarity metrics, which enables partial matching of concession sequences between negotiation cases and input concessions. Further, there are a few heuristics for case selection process. For example,

1. Select negotiation cases which match both buyer's and seller's concessions of input
2. Select negotiation cases after the profile matching.

Once a negotiation case is selected as the most relevant case, which is called candidate concession, to the given input, the CBN takes the concession that was used in previous negotiation and proposes a concession by reusing or adapting it [352]. The case adaptation process is to ensure that the candidate concession fits the current negotiation situation. In the current development, this process checks whether the candidate concession leads to a counter-offer that is over the maximum budget specified in the agent's profile. If so, a set of adaptation rules is applied to modify the candidate concession and submit an appropriate one to the negotiation agent for the generation of a counter-offer. If not, the CBN recommends the candidate concession to the negotiation agent by directly applying the concession without modification.

A Web-based system has been implemented based upon CBN framework and using Java 1.2 and a standard SQL capable relational database management system by Zhang and Wong [352]. It is designed as a distributed three-tier server client architecture and includes a negotiator client, negotiator server, and a negotiation case repository. The negotiator server is the reasoning engine in which the processes of the CBN are performed, including offer evaluation, case retrieval, case matching and reuses as well as counter-offer generation. The negotiator client provides the graphical user interface that is implemented using the Swing API.

It should be noted that an intelligent negotiation agent should be used to negotiate with the customers about their demands and to assist them during the search for an appropriate product [335], which will be examined in the multiagent negotiation is Chapter 7 and Chapter 8.

6.7 Concluding Remarks

This chapter examined applying CBR in e-commerce, in which intelligent sales support, production recommendation, production configuration, and negotiation have been investigated based on the proposed unified architecture for CBR-based e-commerce systems. It stressed that production recommendation is a process that follows case retrieval, while production configuration is a part of case adaptation, and negotiation requires the deeper understanding of case adaptation from a CBR perspective.

This chapter also decomposed case adaptation into problem adaptation and solution adaptation. From the e-commerce viewpoint, customer demand adaptation can be considered as problem adaptation, while product (or goods) adaptation can be taken as solution adaptation. Customer demand adaptation and product adaptation play an important role in e-commerce and lead to two different categories of product recommendation: customer-oriented recommendation and system-based recommendation. Customer demand adaptation is suited to the business situation that the customer can adjust his requirements or problem descriptions with patience. Product adaptation can serve the business situation in which the required product or goods should be tuned to at most satisfy the requirements of the customer. In practice, both problem adaptation and solution adaptation should be combined in a CBR system to provide powerful intelligent support for e-commerce.

Although a wide range of potential applications has been explored, commercial CBR applications in e-commerce have focused for the most part on using case retrieval for decision support [1]. Further, to build up a complete e-commerce framework with a CBR engine as middleware is one of the future challenges. Also, the seamless integration of the CBR process into the intelligent sales process is a promising investigation for the future of e-commerce [172] (p 112). However, it should be noted that in CBR-based e-commerce, CBR plays the role of a “background technology;” that is, it has been *seamlessly* integrated within the existing application systems. Sometimes, customers and/or businessmen are not aware of the fact that they are using CBR techniques. Following this direction, a CBR system should be viewed as a medium to be used in conjunction with other Web-based intelligent system for e-commerce [112].

Beside those mentioned in this chapter, some issues in the following fields can be considered to lead to new advances in CBR in e-commerce: auctions, brokering/negotiation, customer relationship management, and supply chain management (see http://www.ics.uci.edu/~burke/research/cbrec/cfp.html). Further, the effective application of CBR in e-commerce will certainly be facilitated through multiagent systems, because intelligent agents will play an important role in e-commerce, just as human agents have done and are doing in traditional commerce, which will be examined in Chapter 8.

7 Case Based Reasoning in Multiagent Systems

This chapter is the second chapter in Part II of the thesis. It is also the basis for Chapter 9, as shown in the shaded area of Fig. 7.1. This chapter first examines the relationship between case-based reasoning (CBR) systems and multiagent systems (MASs), and proposes knowledge-based models of multiagent CBR systems from both logical and knowledge-based viewpoints. Then this chapter investigates the case base and case retrieval in a distributed setting and examines the integration of case-based reasoning capabilities in a BDI architecture. This chapter also discusses CBR for agent team cooperation. Finally this chapter proposes an agent architecture using CBR to model an agent negotiation strategy.

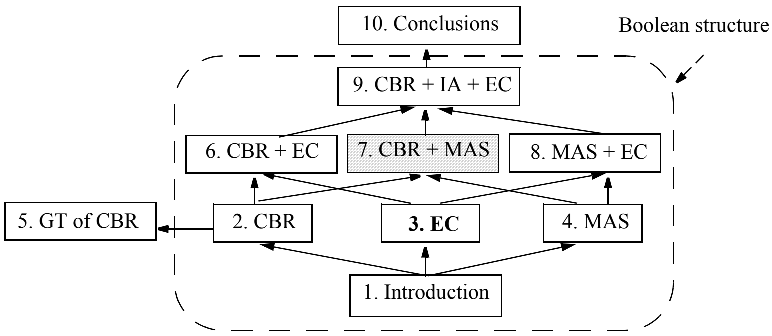


Fig. 7.1 Chapter 7 in the Boolean structure of PhD-thesis

7.1 Introduction

Case-based reasoning (CBR) and multiagent systems (MASs) are two different paradigms in AI. CBR is a reasoning paradigm based on experience-based reasoning or similarity-based reasoning. MAS is a new paradigm to organise AI applications. However, integration of CBR and MASs has drawn increasing attention in the AI community [101][232][237], because CBR offers the multiagent systems paradigm the capability of autonomously learning from experience. Plaza and Ontañón [233] propose a framework for collaboration among agents that use CBR. Prasad [237] discusses issues pertaining to cooperative retrieval and composition of a case in which subcases are distributed across different agents in a MAS. Giampapa and Sycara [101] discuss conversational case-based planning for agent team coordination, in which the acquisition and maintenance of the contextual information that determines the plan requirements is performed by a conversational case-based reasoner, NaCoDAE. NaCoDAE is also used to compositionally generate hierarchical task network plan objectives for the team agents with a MAS. Plaza et al.

[232] investigate cooperation among agents that learn and solve problems using CBR. Further, Olivia et al. [220] describe a framework that integrates CBR capabilities in a BDI architecture.

One of the differences between these investigations is that some of them are in a heterogeneous environment, while the others are in the homogeneous environment. This chapter pursues this advance from a new perspective. That is, it first examines the relationship between CBR systems (CBRSs) and MASs, and proposes knowledge-based models of integrating CBRSs and MASs, which covers almost all attempts that apply CBR in MASs at a high level. Then it discusses how CBR has been applied in MASs at a concrete level. More specifically, this chapter will investigate the case base and case retrieval in a distributed setting, and examine the integration of case-based reasoning capabilities in a BDI architecture. This chapter will also discuss CBR for agent team cooperation. Finall this chapter will propose an agent architecture using CBR to model an agent negotiation strategy

The rest of this chapter is organised as follows: Section 7.2 examines the relationship between CBR and MASs from both a logical viewpoint and a knowledge-based viewpoint and then proposes knowledge-based models for multiagent CBR systems integrating CBRSs and MASs. Section 7.3 investigates the case base and case retrieval in a distributed setting. Section 7.4 discusses CBR for agent team cooperation, which consists of a MAS and an agentified CBR system. Section 7.5 investigates the integration of CBR capabilities in a BDI architecture. Section 7.6 examines how CBR can improve cooperation among the agents within a MAS and how agents use CBR to learn from experience in a medical domain by discussing two cooperative modes for CBR agents within a MAS. Section 7.7 looks into the multiagent negotiation with CBR and the last section ends this chapter with some concluding remarks.

7.2 A Knowledge-based Model of Multiagent CBR Systems

Section 2.3 discussed the relations between expert systems (ESs) and case-based reasoning systems (CBRSs), while Section 4.6 investigated interrelationships of ESs and MASs. Therefore, this section integrates discussions in Section 2.3 and Section 4.6 and examines the relationship between CBRSs and MASs from both a logical viewpoint and a knowledge-based viewpoint and then proposes knowledge-based models of MASs which are based on both CBR systems and knowledge-based systems (see Fig. 7.2).

Section 2.3 concluded that CBRSs can be considered a further development of ESs. Further, both CBRSs and ESs rely on the explicit symbolic representation of experience-based knowledge to solve a new problem [191]. However, ESs use past experience stored in a knowledge base of

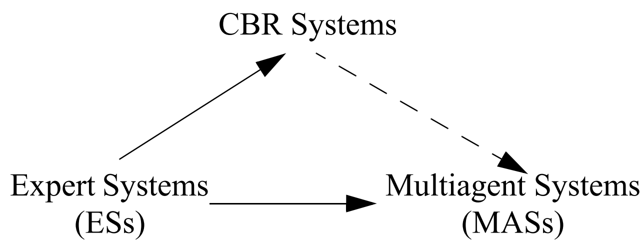


Fig. 7.2 ESs, CBR systems and MASs

generalized heuristics to assist in solving a new problem. They can store the generalized heuristics as rules of thumb or as logical inferences. CBRSs use an abstraction of specific problem-solving experiences to learn to solve a new problem. The representation of specific experiences usually includes the justification of the solution or the requirements of the problem as well as its solution. Moreover, compared to knowledge-based reasoning in ESs, CBRSs stress experience-based reasoning. The case base is an important component in CBRSs, while the knowledge base is one of the main components in ESs. Because similarity-based reasoning is an operational definition of experience-based reasoning, CBR can be considered as a kind of similarity-based reasoning from a logical viewpoint, while the CBRS is still a kind of knowledge-based system from a knowledge-based viewpoint. Therefore, the traditional CBRS can be briefly modelled as:

$$\text{CBRS} = \text{Case base} + \text{CBR engine} \quad (1)$$

where CBR engine denotes the inference engine in CBR system. The CBR engine performs deductive reasoning, in particular similarity-based reasoning, while inference engine in ESs performs traditional deductive reasoning. In this sense, the CBRS is a further development of an ES. This means that the CBRS is similar to the ES from a knowledge-based viewpoint; that is:

$$CBR = CB + CBRE \approx ES = KB + IE \quad (2)$$

where CB denotes the case base in the CBRS, while KB is the knowledge base in the ES. $CBRE$ denotes the CBR engine in the CBR system, while IE is the inference engine of the ES.

Section 4.6 investigated interrelationships of ESs and MASs and showed that high-level intelligence of a system requires a more complex system structure than low-level intelligence does in most cases. The intelligence level of the MAS can be improved through coordination, cooperation, communication, and negotiation among the agents within the MAS, although each of them may be less intelligent than an ES. It thus emphasized that simulation of human intelligence depends not only on the computerized knowledge and reasoning of human experts, to which ESs have paid much attention, but also on cooperation, coordination, communication, and negotiation among the components (agents) within an intelligent system, which MASs have emphasized. The above consideration was summarized in Section 4.6 as the following important relationship between ESs and MASs:

$$MAS = \sum_{1}^{n} A_i + C \approx \sum_{1}^{n} qES_i + C = \sum_{1}^{n} (KB_i + IE_i) + C \tag{3}$$

where A_i is agent i within the MAS, $qES_i = KB_i + IE_i$ is the quasi-ES corresponding to agent i, $i = \{1, 2, \ldots, n\}$. C is the above-mentioned modules for coordination, cooperation, communication, and negotiation among the agents. $\approx$ stands for "is similar to". An concrete example for model (3) is the rule-based multiagent system MAGSY in [92]. Each agent in MAGSY has the problem solving capacity of an expert system and is defined by a triple (F, R, T), where

- F is a set of facts which represent the local knowledge of the agent
- R is a set of rules which define the strategies for the general behaviour of the agent
- T is a set of services which are provided by the agent.

Now taking into account (2), (3) becomes

$$MAS = \sum_{1}^{n} A_i + C \approx \sum_{1}^{n} qCBR_i + C = \sum_{1}^{n} (CB_i + CBRE_i) + C \tag{4}$$

where A_i is still agent i within the MAS, $qCBR_i = CB_i + CBRE_i$ is the quasi-CBRs corresponding to agent i, $i = \{1, 2, \ldots, n\}$.

Therefore, a MAS can be viewed as a kind of CBRSs. Furthermore, it might be practical to simulate each agent within the MAS using CBR technology as much as possible, while one should make good use of MAS technology to deal with coordination, cooperation, communication, and negotiation among the agents in order to improve the intelligence of the MAS.

A concrete example of this model (4) is DistCBR and ColCBR in [232], in which all agents have CBR ability. Another example is a Web-based CBR agent for financial forecasting in [181]. Further, this model is also a more precise form for the multiagent CBR (MAC) system: $M = \{\{A_i, C_i\}\}_{i = 1, \ldots, n}$ proposed by Plaza and Ontañón [233], where M is composed of n agents, and each agent A_i has a case base C_i. Therefore, the above investigation can be considered a generalization of the models of Plaza et al. in [232], and Liu et al. in [181] as well as Plaza and Ontañón [233] for applying CBR in MASs.

The rest of this section will examine the (4) in some more detail:

1. If $CB_i \cap CB_j = \emptyset$ for $i, j = 1, \ldots, n, i \neq j$, then any different two agents A_i and A_j don't share common cases in their own case base. This means that the agents have different experience. This condition sometimes facilitates the corresponding experiments (see [233]) but it might affect the cooperation among agents
2. If $CB_i \cap CB_j = \emptyset$ for $i, j = 1, \ldots, n, i \neq j$ and $CBRE_1 = \ldots = CBRE_n = CBRE$, then the MAS degenerates to the Ensemble CBR system [233] in which CBR agents $A_1, \ldots, A_n$ work with the same CBR method but they have different experience (i.e. different case base $CB_1, \ldots, CB_n$)
3. If $CB_i \cap CB_j = \emptyset$ for $i, j = 1, \ldots, n, i \neq j$ and $CBRE_1 \neq \ldots \neq CBRE_n$, then the MAS is a model for the real world scenario in which CBR agents $A_1, \ldots, A_n$ work with different CBR methods, and they have also different experience (i.e. different case bases $CB_1, \ldots, CB_n$. It should be noted that the different CBR methods result from that CBR is a kind of similarity-based reasoning from a logical viewpoint. Different similarity metrics lead to different CBR

methods or CBR engines. When $n = 2$, agent A_1 and A_2 work with different CBR methods and have different experience to negotiate over a series of negotiation issues (see Section 7.7)

4. If $CB_1 = CB_2 = \ldots = CB_n = CB$ and $CBRE_1 \neq \ldots \neq CBRE_n$, then the CBR agents within the MAS share a common case base, CB, but work with different CBR methods. This case usually happens in the real estate agency in which each CBR agent is a software counterpart of a human agent working in the real estate agency. They share the common resources of properties of houses in the real estate agency. However, they can use different CBR methods to negotiate with the customer over a certain property.

It should be noted that the above discussion is limited to some special cases in multigant CBR systems. In fact, the most general case is where some CBR agents within the MAS share a common case base, while other CBR agents have their own case bases. Some CBR agents like to work with the same CBR method, while other CBR agents work with different CBR methods.

Furthermore, it should be noted this model is homogeneous[1], because each of the agents within the MAS possesses the same ability; that is, CBR. This is not the real case in practice [216]. Therefore, it is necessary to propose the following model, which can be called heterogeneous,

$$MAS = \sum_{1}^{n} A_i + C \approx \sum_{1}^{m} ES_i + \sum_{m+1}^{n} CBR_i + C \tag{5}$$

$$= \sum_{1}^{m} (KB_i + IE_i) + \sum_{m+1}^{n} (CB_i + CBRE_i) + C$$

where, $1 < m < n$. A concrete example of model (5) is CoDiT, a MAS for case-based therapy recommendation in [197] (see Section 7.6). If $m = 1$, then another concrete example of model (5) is the RETSINA multiagent system in [101], in which a conversational case-based reasoner was agentified and inserted (see Section 7.4). This model will be used for integrating CBR and MAS in e-commerce in Chapter 9.

1. This research stresses homogeneous agents have the same knowledge-based reasoning paradigms, while Plaza et al [232] believe that homogeneous agents have the same representation languages so that communication among agents does not require a translation phase.

7.3 Distributed Case Base and Retrieval

In the model (4) of the previous section, the case bases $CB_1, \ldots, CB_n$ may not be situated at a single physical location and may be distributed across the agents A_i, $i = 1, \ldots, n$. For example, in the architecture of distributed CBR in [69] the case bases CB_i are distributed across client nodes and there is also a case base on a central server. How do distributed case bases arise in these MASs [237]? A system that performs rote learning by storing successful problem-solving episodes, where each agent A_i stores its own local case in its case base CB_i, could give rise to such a distributed case base (DCB). However, this may not be the only way, because each of the agents could acquire its own independent problem solving experiences by participating in different teams of agents. Another scenario that one could envisage now is the existence of case bases spreading across the Internet as Doyle et al. did in [69] and CMB, a multiagent CBR system for e-commerce [292]. In this situation, case bases for individual agents may be built independently, without complete knowledge of the kind of problem solving systems in which they are going to participate [236]. Central retrieval queries may not be satisfied by any one case base and may need a composite case derived from different case bases.

Reasoning about cases drawn from a case base that is a component of a DCB presents an agent with additional uncertainties versus single agent CBR systems [236]. As discussed previously, each agent has to rely on its possibly incomplete local view of problem solving to retrieve a local case that best contributes to the overall case. This may lead to the retrieval of subcases that cannot be effectively put together or there may be requirements on the solution that cannot be ascertained until the subcases are aggregated. Thus, Prasad et al. [236] propose the negotiated case retrieval (NCR) strategy that needs the agents to augment their local views with constraining information from other agents to achieve the retrieval and assembly of a better overall case. This strategy involves that each agent asynchronously executes one of the set of possible operations: *initiate* a seed subcase, *extend* an existing partial case, *merge* existing partial cases or *inform* others about a new partial case, as shown in Fig. 7.3.

Initiating a seed subcase involves an agent retrieving a local subcase from its local case base, using the local problem solving state and the relevant portion of the user specification, and forming a seed subcase that can be extended by local cases from other agents to obtain a complete case.

An agent intending to *extend* a subcase from another agent obtains the subcase's relevant feature values that serve as an anchor for the local case retrieval, the result of which is integrated with the corresponding partial case.

Merge is similar to the extend operation. An agent intending to merge one of its chosen partial cases with another agent's partial case obtains the relevant feature values and performs the merge operation.

The *inform* operation involves an agent telling others about the existence of a newly formed partial case that results from the local execution of one of the three previous operators. An *extend* or *merge* operation involves checking for any violations of local constraints by the set of feature values from the non-local partial case and the local case or partial case. Detection of such violations leads to an interaction process among the agents by which they negotiate on conflict resolution alternatives. The negotiation process involves an agent communicating feedback to other agents on the causes and possible resolutions for each of the constraint violations. The receiving agents assimilate this feedback, leading to an enhanced view of the global requirements for future operations. Any subsequent initiate, extend or merge is more likely to avoid the same conflicts.

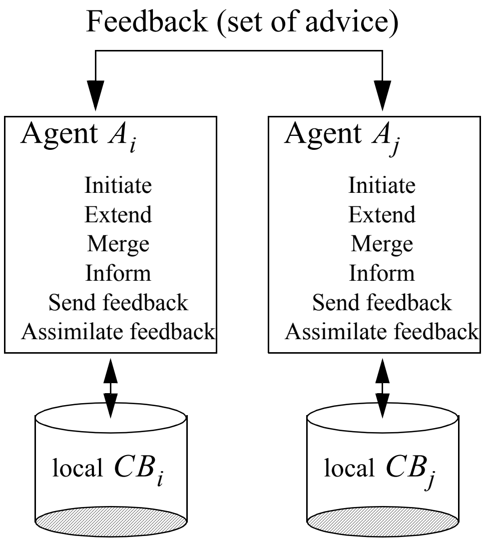


Fig. 7.3 Model of negotiated case retrieval after [236]

7.4 CBR for Agent Team Coordination

As mentioned in Chapter 4, a MAS comprises a group of intelligent agents working together towards a set of common global goals or separate individual goals that may interact. In such a system, each of the agents may not be individually capable of achieving the global goals and/or their goals may have interactions- leading to a need for cooperation among the agents [237]. This section examines the agentification of a CBR system, NaCoDAE, into a multiagent system (MAS), RETSINA (reusable environment for task-structured intelligent network agents), in which the agents may use capability-based or team-oriented agent coordination strategies, for agent team coordination.

RETSINA is a collection of heterogeneous software entities that collaborate with each other to either provide a result or service to other software entities or to an end user [101]. Based on functional viewpoint, RETSINA agents are classified into four types: interface agents, task agents, middle agents, and information agents. RETSINA agents typically use the capability-based coordination technique to task each other, which means that one agent will dynamically discover and interact with other agents based on their capability descriptions. RETSINA agents also support other forms of coordination techniques, such as the team-oriented coordination.

NaCoDAE is a conversational CBR system that helps a user decide a course of action by engaging him in a dialogue in which he must describe the problem or situation [101]. A conversational session begins with the user providing an initial partial description of the problem that he tries to solve. NaCoDAE responds by recommending the ranked solutions from the case base, whose problem descriptions best match the user's problem descriptions, and the ranked questions, which are the unanswered questions in these cases, to the user (interface). After the user obtains these recommendations, he will either refine their problem description by answering selected questions, or accept a solution from the recommended solutions. Therefore, from a viewpoint of CBR, NaCoDAE has performed case retrieval, case recommendation, and problem adaptation that is a part of case adaptation.

NaCoDAE has three features that made it suitable for team co-ordination and interaction with RETSINA agents [101]. First, NaCoDAE can work with partial descriptions of the problem and use them for initiating a dialogue. This could allow one to encode a general strategy of “always knowing the strategy for how to get more information, if nothing else is known". Second, NaCoDAE can continually revise its list of most likely candidate cases, as data is provided to the system by either an agent or the user. This feature leads itself to a form of coherent, compositional and incremental construction of knowledge structures, such as hierarchical task network (HTN) plan objectives and representations of situational or contextual knowledge. This knowledge can be accessed even if time and the lack of specific information do not allow for a description to be completely specified. Third, the cases can be modified to store any type of textual data, including agent capabilities and queries.

After agentification, the NaCoDAE becomes a RETSINA task agent [101] who is situated in the RETSINA community, where there are also Briefing Agents, Matchmakers, MissionAgents, VoiceAgents etc. They work together to perform a certain mission. The agent communication that involves BriefingAgent and NaCoDAE are run in the following way [101]: As the Company Commander speaks, his speech is translated into text by the VoiceAgent. The BriefingAgent receives those textual translations and attempts to match the text of the Commander's speech with the textual answers to questions that were posed by NaCoDAE. If there is a match, then the BriefingAgent will send that answer to NaCoDAE. If NaCoDAE can use that answer to complete a case, then it will return a case to the BriefingAgent; otherwise return a regenerated ranked list of questions and their associated answers. If NaCoDAE's questions contain agent queries, the BriefingAgent will directly query the provider agent if it is known, or first ask either or both of the Matchmakers for the identity of a provider agent, and then contact it. Upon request of the MissionAgents, or upon the completion of a case by NaCoDAE, the BriefingAgent will assemble a shared plan from the case actions and send it to the MissionAgents. During the execution of the scenario, the MissionAgents may also provide the BriefingAgent with updates to their capabilities, which the BriefingAgent can forward to NaCoDAE.

This section examined the CBR system for the team coordination of independent, intelligent software agents. According to Giampapa and Sycara [101], NaCoDAE has demonstrated that its conversational nature is well-suited for agent information gathering domains.

7.5 Case-based BDI Agents

Integrating CBR capabilities in a BDI architecture is another attempt to integrate CBR and MASs. This section will examine a framework that integrates CBR capabilities in a BDI architecture as well as its application to the design of Web information retrieval proposed by Olivia et al. [220].

BDI structure mainly consists of five factors: beliefs, desires, intentions, goals, and plans, which constitute the mental state of a BDI agent [32] (p 47), as shown in Fig. 7.4:

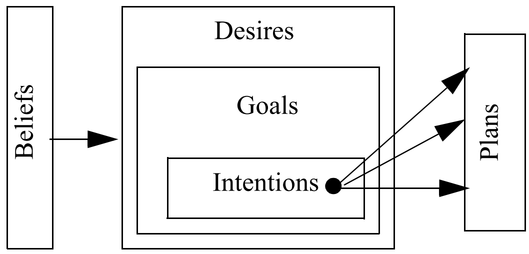


Fig. 7.4 BDI structure based on Brenner [32]

- Beliefs contain the fundamental views of an agent with regard to its environment. An agent uses them to express its expectations of the possible future states
- Desires are derived directly from the beliefs. They contains the agent's judgements of future situations
- Intentions are a subset of the goals. If an agent decides to follow a specific goal, this goal becomes an intention
- Goals represent that subset of the agent's desires on whose fulfilment it could act. In contrast to its desires, an agent's goals must be realistic and must not conflict with each other
- Plans combine the agent's intentions into consistent units.

BDI agents have been widely used in relatively complex and dynamically changing environments [32]. Olivia et al. [220] proposes a CBR-BDI agent architecture for information retrieval (IR) on the WWW in order to improve the performance of currently deployed Web IR systems in terms of search efficiency and resource discovery in well-demarcated domains. Web CBR-BDI agents are designed to locate and extract information from homepages of academic staff members with particular research interests. The CBR-BDI architecture has the following main components: the case memory, the domain-specific knowledge base, and the CBR-BDI interpreter, as shown in Fig. 7.5 In what follows, the first two mentioned components will be examined in some detail.

- The domain-specific knowledge base is implemented in a form of concept hierarchy, which is collection of keywords representing broad areas of expertise (concepts). A collection of keywords (sub-concepts) representing specific sub-area of expertise is also attached to these concepts. Concepts are mapped to specific university academic entities on a well-demarcated application domain such as Australian universities. The concept hierarchy plays an important role in focusing on the search process in the start-up of the Web CBR-BDI agent where no previous cases are stored, and when the similarity-based mechanism does not target any particular case in the case memory. Furthermore, it helps the system to identify related research interest if it fails to retrieve an exact match information. For example, if no exact match for research interest in mobile agents is found, the system is able to retrieve academic homepages with relevant research interest such as intelligent agents and autonomous agents

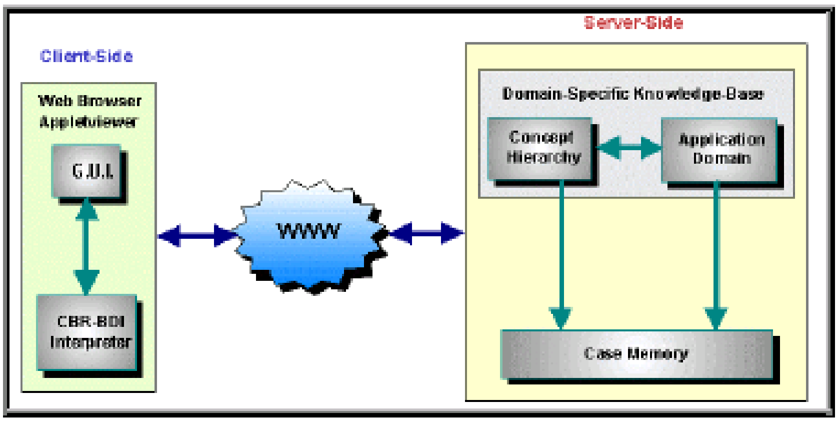


Fig. 7.5 A CBR-BDI architecture after [220]

- Case memory. Cases stored in the case memory are constructed in terms of belief, desire, intention, outcome status, and outcome URL. More specifically, belief is the university domain to be searched. Desire is a sub-concept/concept that represents the specific/similar research interest being searched. Intention is the focused search to concept-related academic entities. Outcome status is either a successful or unsuccessful case. Outcome URL is the URL staff link directory academic entity.

The overall process is run as follows: The end-user is presented with the GUI where he can specify sub-concepts or research interests associated with a given concept/domain of knowledge, together with the universities of interest. Web CBR-BDI agents are triggered by pressing the start button. The first objective of the agent is to perform a standard CBR analysis of the input problem description. The input problem description is constructed by the combination of an end-user's selected university domain (belief), and subconcept/specific research interest (desire). The similarity-based mechanism serves to find the most similar cases with the input problem description.

Given that similar cases are sorted by outcome status (found/not found), the Web CBR-BDI agent first scans the most promising URLs (outcome status = found), and leaves for the last stages of the search the less promising ones (negative cases).

In the case where the similarity-based mechanism retrieves similar cases, the case memory may lead directly to a promising URL from where to initiate either a depth-first or breadth-first search, instead of traversing exhaustively the sub-webs of a particular university.

The results obtained from the CBR analysis drive the Web traversal of the agent to retrieve the desired information.

7.6 Cooperative CBR Agents in MAS

As mentioned in previous Chapter 4 and Section 7.4, cooperation is an important characteristic in MASs. An agent with "perfect" knowledge and "complete" capabilities for a given task has no need to require the cooperation of other agents [232]. However, a regular agent is less intelligent than an expert as discussed in Chapter 4, he can't have "perfect" knowledge and "complete" capabilities for a given task. Even an expert in a society can't say that he has "perfect" knowledge and "complete" capabilities for a given task in his professional field. Therefore, it is necessary for an agent to cooperate with other agents within the MAS to perform a given task. This section examines how CBR can improve cooperation among the agents within a MAS and how agents use CBR to learn from experience in a medical domain, which used to be an important application field of expert systems such MYCIN [287]. The real world scenario is CoDiT.

CoDiT is a MAS, wherein agents use CBR to recommend therapy for diabetic patients [197]. CoDiT consists of a few agents that perform CBR and are able to communicate and cooperate for recommendation of a therapy[1]. Each agent, as a software counterpart of a human doctor, has a case base with data of the patients of a specific M.D.; moreover, legal and deontological reasons prevent that patient data could be centralised since only the patient's doctor is entitled to have that

1. For more information see http://www.iiia.csic.es/Projects/smach

data. Thus, this scenario fits the MAS approach since resources are distributed but some doctors (or their agents) could also be interested in the case of a patient that is unknown to them but stored in some other doctor's case base. Further, the diabetes therapy CBR agents in the MAS are *peer* agents, since each agent is capable of solving the whole task alone (recommending a therapy) using the resources available in its case base. However, it is obvious that in such a scenario the agents should exchange patient data (maintaining anonymity for legal and deontological reasons) in order to improve their performance.

The main CBR task involved in CoDiT is *retrieve* and *reuse*, as shown in Fig. 7.6. There is also an automatic **retain** task (not shown in the figure) that incorporates a solved problem into the agent's episodic memory. Generally speaking, the main retrieve task can be decomposed into three subtasks: identify, search, and select (also see [1]). The **identify** task has a method that constructs a perspective on the patient; then task **search** retrieves from the case base those cases

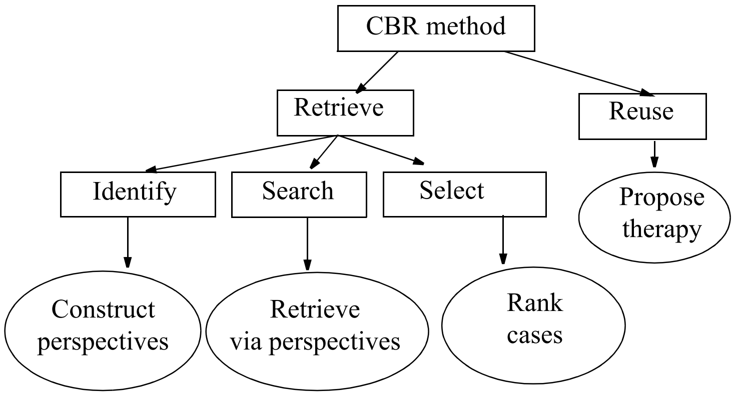


Fig. 7.6 Task decomposition of CBR for diabetes therapy after [197]

that satisfy the model built by the perspective. Next, the **select** task has a method that constructs a preference model of the retrieved cases from domain-specific knowledge. Finally, **reuse** is a task that takes the most preferred case and adapts its solution (therapy) to the current patient; the adaptation method uses domain-specific knowledge and if a most preferred case cannot be adapted then tries to adapt the next-preferred case. This means that case adaptation has been used here as a part of case reuse. In what follows, the rest of this section examines how the methods used in the retrieve task can incorporate communication and cooperation with other agents in order to find out relevant cases for other agents.

A cooperation mode establishes how two agents must behave to accomplish a particular task. However, when an agent can opt for more than acquaintance to cooperate in solving a specific (sub)task, then different co-operation strategies can be established for each cooperation mode depending on different criteria followed by the agent to solve such (sub)task. For instance, depending on how the set of helper agents chosen to cooperate is constructed and how this set is sorted to be traversed in search of a competent agent. In this way, a cooperation mode can determine how two agents cooperate whereas a cooperation strategy settles how more than two agents do.

Therefore, the term cooperative CBR groups together the set of cooperation modes and cooperation strategies that can be deployed by some CBR agents wherein each CBR agent has its own case base.

Cooperation among CBR agents can be thought as an extension of agents' set of precedents; that is, an expansion of the individual memory of a CBR agent to the memories of some CBR agents. For instance, in CoDiT the retrieve task incorporates cooperation with other agents in order to find relevant cases known for other agents- i.e. to find the patient record most relevant to the current problem.

Two cooperation modes between CBR agents were proposed in [197]: Distributed Case-Based Reasoning (DistCBR) and Collective Case-based Reasoning (ColCBR). The DistCBR cooperation mode is a class of cooperation protocols where a CBR agent A_{orig} is able to ask one or several other CBR agents $\{A_1, \ldots, A_n\}$ to solve a problem on its behalf, and the ColCBR cooperation mode is a class of cooperation protocols where a CBR agent A_i is able to send a specific CBR method to one or several CBR agents $\{A_1, \ldots, A_n\}$ that are capable of using that method with their case base to solve the task at hand [232]. Therefore, the DistCBR cooperation mode enables an agent to share experiential knowledge acquired by an acquaintance by means of particular problem solving methods, while the ColCBR cooperation mode allows a couple of CBR agents to share experiential knowledge. Both DistCBR and ColCBR are based on solving the retrieve task reusing the experiential knowledge (in form of cases) of other CBR agents:

- DistCBR. An agent (the originator) delegates the retrieve task to another agent (the helper) indicating the helper's CBR method to solve such task. In this sense, the CBR process is distributed since every agent works using its own method of solving problems [232]

- ColCBR. An agent (the originator) forwards the retrieve task and the PSM (problem solving method) of that task to an acquaintance (the helper). That is to say, the originator, in addition to the task, also conveys the PSM to solve that task. In this sense, the originator is using the memory of the other agents as an extension of its own- as a collective memory- by means of being able to impose to other agents the use of the CBR method of the originator [232].

In both cooperation modes helper's experiential knowledge is shared and then reused by the originator [197]. However, while the DistCBR cooperation mode also allows helper's problem solving knowledge to be shared and reused by the originator, using the ColCBR cooperation mode the PSM sent by the originator is shared by the helper to retrieve the most relevant case(s) that will be later reused by the originator. From an authority point of view, it can be said that using DistCBR the originator delegates authority to the helper to solve the task in hand. On the contrary, using ColCBR the originator maintains the authority, since it has fully control over the PSM applied, merely using the experiential knowledge of the helper.

The following actions are performed by two CBR agents whilst cooperating using the DistCBR cooperation mode [197]:

1. The originator asks the helper to solve (delegates) the retrieved task indicating which helper's problem solving method must be applied to solve such task
2. On receipt of the task, the helper retrieves the most relevant precedent(s) using its corresponding retrieval method (as indicated by the originator)
3. Thereafter, the helper refers the available precedent(s) back to the originator which will have been inferred using its own (helper's) PSM.

The ColCBR cooperation mode implies the following actions to be carried out between two CBR agents [197]:

1. The originator sends the retrieve task to be solved and a originator's retrieval method to be applied to solve such task together to the helper
2. On receipt of the task and the PSM, the helper retrieves the most relevant precedent(s) using the PSM received
3. Thereafter, the helper refers the available precedent(s) back to the originator which will have been inferred using the originator's PSM method.

This section discussed two different cooperative modes for CBR agents within a MAS. The above discussion allows us to exemplify the sharing and reuse of problem solving knowledge and experiential knowledge (in the form of cases in CBR) among agents within a MAS.

It should be noted that from a viewpoint of pure CBR, CoDiT is a case-based recommendation system for a medical domain, in particular for diabetic patients. The basic difference of CoDiT from other case-based recommendation systems mentioned in Chapter 6 is that CoDiT is placed in multiagent settings. Because of communication and cooperation, case-based recommendation systems become more complex in multiagent settings.

7.7 Applying CBR to Multiagent Negotiation

As mentioned in Chapter 6, an intelligent agent should be used to negotiate with the customers (or customer agents) for their demands and to assist them during the search for an appropriate product. This section will examine how to use CBR to automating negotiation in a multiagent setting.

7.7.1 Introduction

Negotiation in MASs is one of the main research lines in MASs and has been studied from many different points of view such as game theory, artificial intelligence, and CBR in Chapter 6. In the area of CBR, Sycara presents a model of negotiation that combines CBR and optimization of multi-attribute utilities of intelligent agents. She provides a model of goal conflict resolution through negotiation implemented in the PERSUADER system that resolves labour disputes. Matoes [199] employs CBR to determine in each step of the negotiation the best performance of the agent by selecting the weighted proposal combinations and the parameters associated with a set of tactics. Recent growing interest in intelligent agents in e-commerce has given more importance to the problem of automated negotiation. Intelligent agents negotiate to coordinate their activities and come to mutual agreement, in particular in the e-bargaining process (see Chapter 8). In many cases, automated negotiation requires different behaviours of intelligent agents for different negotiation situations [199]. The rest of this section will first look at the negotiation in a real estate agency and negotiation strategies. Then it will present an agent architecture using CBR to model an agent negotiation strategy.

7.7.2 Negotiation in a Real Estate Agency

In a real estate agency there is a set of properties that need to be sold. In this domain there are two main players: seller agents and buyer agents. The seller agent acts on behalf of the interests of the real estate agency, while the buyer agent represents the interests of a customer. The seller agent needs to sell a house with maximal profit for the agency, while the buyer agent wants to buy a house for his buyer with specific features and minimal price. This is an obvious conflict of interest that is usually resolved by a negotiation.

The seller agent has complete information of all the properties about houses on sale at the real estate agency. However, in some cases the buyer agent does not have a clear opinion on his preference on the negotiation issues. During the negotiation, the seller agent usually includes new negotiation issues to enrich the description of a house. Then the buyer uses this new information to compare and discriminate better among the different offers made by the seller agent. Thus, the buyer agent tries to obtain a complete description of the properties, negotiating over the set of negotiation issues mentioned before. Usually the agents try to adjust either the issues related to the description of the house and later the price issues. They negotiate until they obtain an agreement, in this case a property that satisfied both sides, if any exists, or one of them withdraws.

7.7.3 Negotiation Strategies

The negotiation strategies are defined based upon the knowledge, past experience, and information available to the negotiation agents [352]. The aim of a negotiation strategy is to determine the best courses of action to reach an agreement [199]. When agent a receives an offer from agent b, it becomes the last element in the current negotiation thread between the agents. If the offer is unsatisfactory to a, the agent a generates a counter-offer. In generating its counter-offer, a may use the information of mental state and different weighted combinations of tactics for each of the negotiation issues. The negotiation issues in the real estate agency mainly include the features of a house, for instance, surface, district, number of rooms, floor number, garage, price, brightness, number of bathrooms, elevator, and address [199].

Most systems use a number of predefined negotiation strategies to generate counter-offers. For example, negotiation in Market Maker [209] allows the agents to use three predefined negotiation strategies: anxious, cool-headed, and frugal corresponding to linear, quadratic, and exponential functions in the generation of offers/count-offers. The users need to decide which strategy the agents will follow during negotiation. However, negotiation strategies can also be acquired from previous negotiation cases or experiences based on CBR in the CBN (case-based negotiation) framework [352]. The CBN agents revise and adapt negotiation strategies in each decision-making episode of the negotiation process [352].

7.7.4 Case-based Agent Architecture

As mentioned in previous chapters, CBR has received a lot of attention over the last few years, and has been employed with good results in many areas [199][317] including negotiation in e-commerce. The case-based negotiation agent uses CBR technology to perform negotiation on behalf of either seller or buyer or broker; that is, he will assess at the similarity of the current negotiation to previous negotiation cases kept in the negotiation case base. The successful negotiation cases that the case-based negotiation agent performed are kept in the negotiation case base for *reuse* in later negotiation case retrieval. The case-based negotiation agent can use the fuzzy rule-based adaptation to adapt the most similar negotiation case to the current negotiation situation. The architecture of the case-based negotiation agent is shown in Fig. 7.7. Some components will be discussed in more detail.

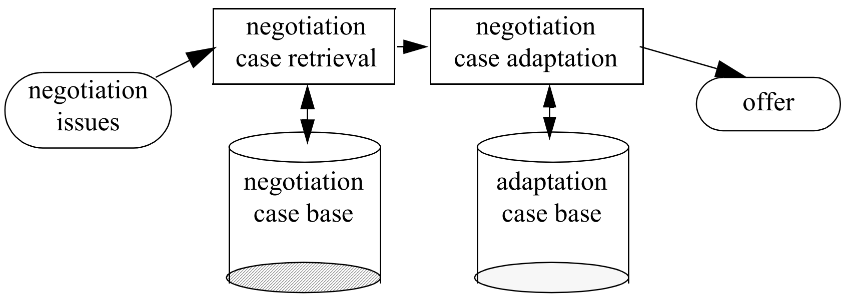


Fig. 7.7 An architecture of a case-based negotiation agent

- Negotiation issues are similar to the current problem in the traditional CBR model. It is also the requirements of a customer. For example, during the negotiation in a real estate agency [199], the seller agent and buyer agent will negotiate over the following negotiation issues for a house: price, number of rooms, garage, floor number, number of bathrooms, address, surface, district, furnished or unfurnished

- A negotiation case in the negotiation case base can be considered as a prior successful negotiation process, which mainly includes the sequence of offer and counter-offer and the eventual successful offer based on the initial negotiation issues. It also provides detailed negotiation context and decisions made in previous negotiations
- The negotiation case retrieval is executed concurrently with the other activities of a case-base negotiation agent. When an agent sends an offer, it immediately begins to retrieve those negotiation cases that are most similar to the current negotiation cases from its negotiation case base. When it receives a counter-offer corresponding to its offer it is incorporated into the negotiation thread and used to finally select the most similar negotiation case from those that were obtained in the meantime
- Once a negotiation case is selected as the most relevant negotiation case to the current negotiation issues, the case-based negotiation agent might revise or adapt the negotiation case in order to meet the changing count-offer from his counterpart. The negotiation case adaptation can depend on a set of fuzzy adaptation rules, which represent conditions of the environment in which the negotiation acts and determines variations in the value of the parameters of the negotiation issues and negotiation tactics (also see [199]). In general, these fuzzy rules follow the following classical form:

$$Rule_i\text{: IF } x_1 \text{ is } A_{i1} \text{ and } \dots \text{ and } x_n \text{ is } A_{in} \text{ Then } y \text{ is } B_i \tag{6}$$

where $x_1, \dots, x_n$ and y are the feature variables, $A_{i1}, \dots, A_{in}$, and B_i are linguistic labels of the variables $x_1, \dots, x_n, y$ which are in the universe of discourse $U_1, \dots, U_n, V$ of the variables. An example of linguistic labels might be: {excellent, good, not satisfactory, bad}. These linguistic labels are characterised by their membership functions $\mu_{A_{ij}}: U_j \to [0, 1]$, $j = 1, \dots, n$; $B_i: V \to [0, 1]$.

7.8 Concluding Remarks

This chapter examined the relationship between CBR and MASs from both a logical viewpoint and a knowledge-based viewpoint and then proposed knowledge-based models of a multiagent CBR system integrating CBR systems and knowledge-based systems, which basically cover almost all attempts that have applied CBR in MASs in a homogeneous or heterogeneous setting. The key idea behind these models are that CBR systems can be considered as a further development of expert systems (ESs), and the integration of CBR systems and MASs should take into account cooperation, coordination, communication, and negotiation, in order to model the social function of individual intelligence. Then this chapter investigated the case base and case retrieval in a distributed setting and examined the integration of case-based reasoning capabilities in a BDI architecture.

Cooperation is an important characteristic in MASs. This chapter discussed CBR for agent team cooperation, which consists of a MAS and an agentified CBR system, and examined how CBR can improve cooperation among the agents within a MAS and how agents use CBR to learn from experience in a medical domain by discussing two cooperative modes for CBR agents within a MAS.

Negotiation is another important characteristic in MASs. This chapter looked into CBR-based negotiation in a real estate agency and negotiation strategies. Then it proposed an agent architecture using CBR to model an agent negotiation strategy.

It should be noted that research and development of multiagent CBR systems is still in its infancy, although some advances in this field has been reported or appeared in the past few years. Further, the studies are basically in a homogeneous multiagent setting. Therefore, there are a lot of issues in the future study of multiagent CBR systems. For example, the proposed models ((4) and (5)) require further investigation at a more detailed level. Negotiation is a general concept in a multiagent setting. In fact, its special forms are auction, brokering, and mediation, which are all important for commerce and business. Therefore, how to apply CBR in auction, brokering, and mediation in a multiagent e-commerce setting is a big issue for intelligent e-commerce, which will be examined in Chapter 9.

8 Multiagent E-Commerce

This chapter is the final chapter in Part II of the thesis. It is also the basis for Chapter 9, as shown in the shaded area of Fig. 8.1. This chapter will examine multiagent e-commerce. Then it examines multiagent-based negotiation. Further it classifies multiagent-based negotiation into multiagent-based auction, multiagent-based mediation, and multiagent-based brokerage, and gives a brief survey of related works in each. Then it investigates multiagent brokerage and argues that bargaining and compromise play an important role in brokerage. Finally, this chapter proposes an architecture of a multiagent-based intelligent broker for the bargaining process.

The exploration here differs from the available approaches in several ways. First it emphasizes that compromise and bargaining are the central activities in the negotiation process. Second, negotiation is a very general concept in Web Intelligence. Three special cases of negotiation are mediation, auction and brokerage. Third, compromise[1] is the necessary condition for negotiation. Finally, the difference between negotiation, auction, mediation, and brokerage can be classified based on the strength of compromise and bargaining.

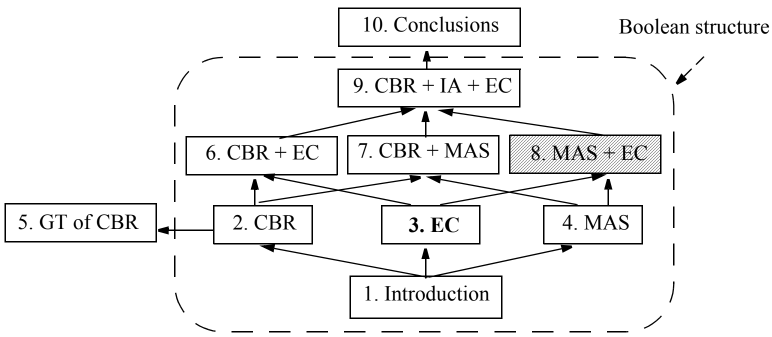


Fig. 8.1 Chapter 8 in the Boolean structure of PhD-thesis

8.1 Introduction

The revolution of the Internet and the WWW has changed traditional commercial activities such as shopping, brokerage, negotiation, and retailing. Customers can purchase a large selection of merchandise items from an ever-increasing number of Internet stores [179]. Basically speaking, there are two forms of e-commerce applications [53]: ones that simply put existing products and

1. Although Davis and Smith mentioned compromise in [59]: “cooperation as a form of compromise between potentially conflicting desires,” they still focused on cooperation rather than compromise

means of selling online, and others that create new ways of selling online using intelligent techniques. The first category is a natural mapping from traditional commerce, while the latter can be considered as an intelligent transformation from traditional commerce to intelligent e-commerce, which involves the birth of new business processes made possible by the Internet and new technology to make it successful. Applying intelligent agents in e-commerce belongs to the latter category [292].

Applying intelligent agents or multiagent systems (MASs) in e-commerce can be considered as multiagent-based e-commerce[1], which has been among the most rapidly growing areas of research and development in information technology in the last few years [142][292][294][344]. Recently, agent technologies have been applied to e-commerce to improve search effectiveness and reduce transaction costs. Many research studies or commercial projects on intelligent agents for e-commerce have been undertaken such as Jango, AuctionBot, BargainFinder, and Market Maker [32][188][209][104]. Intelligent agents are rapidly gaining popularity in e-commerce [332], in which agents have been playing the roles of buyers, sellers, intermediaries, and information providers [183].

Automation of negotiation, which corresponds to negotiation-based e-commerce, has received a great deal of attention from the MAS community [268], because such endeavours have the important potential for significantly reducing negotiation time and removing some of the reticence of humans to engage in negotiation and then facilitating the intelligent negotiation agents that are able to perform negotiation on behalf of users [183][187]. Furthermore, auction, mediation, and brokerage can be considered as three concrete forms of negotiation [292]. These attempts correspond to the following three different kind of e-commerce: auction-based e-commerce, bargaining-based e-commerce, and mediation-based e-commerce [292]. There are a number of studies on multiagent-based negotiation, mediation [209], auctions [352], brokerage [286], and the bargaining process [85], although there are few studies that examine their interrelations. For example, Maes et al. have done considerable research on mediation-based e-commerce using intelligent agents [188], which has also been drawn increasing interest in European countries [67].

1. For brevity, “multiagent” stands for “multiagent-based”.

This chapter will first reviews intelligent agents in e-commerce or agent-based e-commerce. Then it examines multiagent negotiation, which is the core of multiagent negotiation-based e-commerce. Further it classifies multiagent negotiation into multiagent auction, multiagent mediation, and multiagent brokerage, and gives a brief survey of related works in each. Then it investigates multiagent brokerage and argues that bargaining and compromise play an important role in brokerage. Based on the characteristics of buyer agents, seller agents, and brokers, this chapter proposes an architecture of a multiagent-based intelligent broker for the bargaining process.

The rest of this chapter is organised as follows: Section 8.2 reviews agent-based e-commerce. Section 8.3 examines multiagent negotiation. Section 8.4 investigates auction-based e-commerce. Section 8.5 examines mediation-based e-commerce. Section 8.6 studies brokerage-based e-commerce. Section 8.7 proposes an architecture of a multiagent-based intelligent broker for the bargaining process. Section 8.4 ends this chapter with some concluding remarks.

8.2 Agent-based E-commerce

Intelligent agent technology has been applied to the e-commerce domain. This section will review intelligent agents in e-commerce and examine multiagent-based e-commerce.

8.2.1 Intelligent Agents in E-Commerce

Intelligent agents are rapidly gaining popularity in e-commerce [135][332]. There have been various intelligent agents available for e-commerce activities such as process management, information mining, knowledge management, decision making, and so on [86][141][203]. For example, agents can search for information about products of interest to the user, compare prices and features, negotiate for a fair price and even, if authorized by its user, make a purchase, and authorize payment through a credit card or digital cash provider. There are opportunities for retail and financial institutions to make use of their agents to provide personalized service to customers, to collect and make use of detailed customer information, and to develop products and services which reflect the interests of customers searching their Websites. As security techniques become more highly developed and customers gain more confidence in them, agents will frequently be authorized to make purchases for the user.

8.2.2 Intelligent Agents as Personal Assistants

The intelligent agent, as a personal assistant, is collaborating with the user in the same work environment [189]. The assistant becomes gradually more effective as it learns the user's interests, habits, and preferences. Notice that the agent is not necessarily an interface between the computer and the user. In fact, the most successful interface agents are those that do not prohibit the user from taking actions and fulfilling tasks personally.

Intelligent agents assist users in a couple of different ways: they hide the complexity of difficult tasks, they perform tasks on the user's behalf, they can train or teach the user, they help different users collaborate, and they monitor events and procedures, and so on [189][332].

8.2.3 Intelligent Information Agents

The explosive growth of information available on the Web makes information search and filtering an early application domain of intelligent agents [178], A typical approach is to use keyword matching to locate a document or to measure the relevance of a document. Many AI techniques such as rules, best-first search, and genetic algorithms have been used to search information intelligently and effectively. Based on the intelligent agent technology, information agents also attempt to facilitate information search and filtering.

Information agents will be continuously active, proactively trying to gather information even without the user's explicit command. An information agent must have modules that search information resources, collect search results, and present the results [32]. The information agents must be capable of interacting with other agents at the social or communicative level [150].

8.2.4 Multiagent-based E-Commerce

Just as human agents have played a critical role in traditional commerce, as the software counterpart of human agents, intelligent agents will also be playing an important role in e-commerce. However, just as a single agent can play an important role in a special organisation of current commerce only if he can cooperate with other agents, an intelligent agent has to cooperate with other agents in order to play an active role in e-commerce; that is, multiagent-based e-commerce, which is denoted as multiagent e-commerce, for brevity.

Multiagent e-commerce is any attempt to apply multiagent technology to e-commerce [269][292]. Recently, multiagent e-commerce has drawn increasing attention, promising a

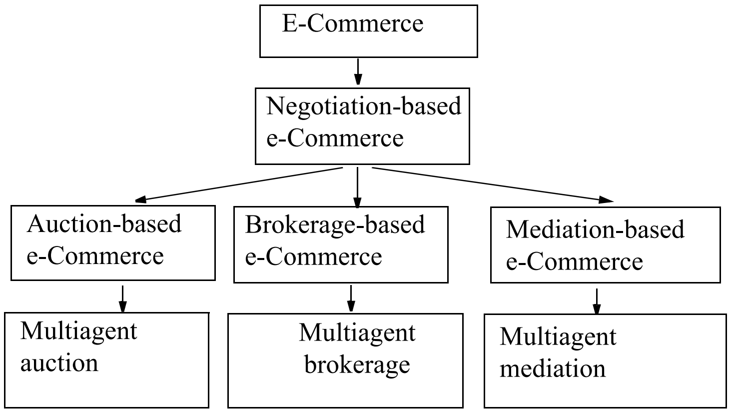


Fig. 8.2 Multiagent e-commerce

revolution in the way we conduct some of the most important activities in traditional commerce, negotiation, auction, mediation, and brokerage. Negotiation-based e-commerce, auction-based e-commerce, mediation-based e-commerce, and brokerage-based e-commerce can be considered as the important aspects of intelligent e-commerce systems. All four involve AI technologies. How to use intelligent agents and MASs in negotiation-based e-commerce, auction-based e-commerce, mediation-based e-commerce, and brokerage-based e-commerce has become a central issue in multiagent e-commerce, which are respectively realized by multiagent auction systems, multiagent mediation systems, and multiagent brokerage systems [294]. All these systems can be considered a specialisation of multiagent negotiation systems, as shown in Fig. 8.2. The following sections will be devoted to the techniques behind these mentioned systems.

8.3 Multiagent Negotiation

Negotiation has long been recognised as a central topic in distributed AI (DAI) and MAS [66] (p 70) since it became a metaphor for distributed problem solving in MAS in 1983 [59]. Initially it focused primarily on negotiation as collaborative, distributed problem solving, as a means towards improving coordination of multiple agents working together on a common task [59][238]. As e-commerce became increasingly important, the work expanded to encompass situations with agents representing individuals or businesses with potentially conflicting interests. This can be considered as automated negotiation [136], or multiagent negotiation or negotiation in e-commerce [294], in which intelligent agents bargain for goods and services on behalf of some end-users.

8.3.1 Characteristics of Negotiation

Negotiation in e-commerce is a decision making process by which two or more parties multilaterally bargain resources for mutual intended gain, using the tools and techniques of e-commerce [12][26][177][292]. Negotiation basically consists of a negotiation protocol, negotiation strategies, negotiation issues, and negotiation processing [288]. While negotiation protocol comprises the rules (i.e. legitimate actions) of the negotiation, negotiation strategies or tactics define how to win the negotiation.

According to Adam et al. [2] (pp 107-8), there are four main dimensions that affect the design and operation of a multiagent negotiation system: ability of negotiation agents, autonomy of agents, number of parties (agents) involved, and the number of negotiation issues. The dimension of the ability of negotiation agents ranges from no bargaining to a bargain for everything. Autonomy of agents could range from full autonomy where the agent conducts the negotiation without any human intervention to an advisor system that operates in a helper mode to a human negotiator. The negotiation process can involve only two parties (e.g. buyer and seller) or multiple parties (e.g. buyer, seller, and broker). In terms of the number of negotiation issues, negotiations can vary from single issue (e.g. price) to multiple issues (e.g. integration bargaining) [2].

More generally, Lumuscio et al. [183] propose a classification scheme for negotiation in e-commerce, in which they list the following main parameters for characterizing negotiation:

- Cardinality of the negotiation
  - negotiation domain: single issue or multiple issues
  - interactions: 1 : 1, m : 1, and m : n [313]
- Agent characteristics (also see Chapter 4)
  - role, i.e. buyer, seller, auctioneer, mediator, coordinator [219], or broker [85]
  - rationality: perfect or bounded
  - knowledge
  - commitment
  - social behaviour, i.e. cooperation, coordination
  - bidding strategy
- Event parameters

  - bid validity
  - bid visibility
  - clearing schedule and timeouts
  - quote schedule
- Information parameters
  - price quotes
  - transaction history
- Allocation parameters, which only is applied in m : 1 and m : n cases, for example, in auction.

The following sections will use this classification scheme to examine the relationship between negotiation, auction, mediation, and brokerage.

8.3.2 Automated Negotiation in E-Commerce

Automated negotiation has been of particular interest due to the relevant role that negotiation plays among trading agents at the activity of auction, mediation or brokering [15][81][93][225]. Matos in [199] presents two types of agent architecture: one based on CBR and another based on fuzzy logic, to model an agent negotiation strategy. At each step of the negotiation process these architectures fix the weighted combination of tactics to employ and the parameter values related to these tactics. When an agent is provided with the case-based architecture, it uses previous knowledge and information of the environment state to change its negotiation behaviour. On the other hand, when provided with a fuzzy architecture it employs a set of fuzzy rules to determine the values of parameters of the negotiation model.

In the buying and selling environment, negotiation agents need to manage their own negotiation strategies during the whole negotiation process. Current e-commerce trading systems which look at e-negotiation usually use a lot of predefined negotiation strategies [104][188]. For example, Market Maker, the successor of Kasbah, is an example of e-commerce negotiation systems [62], and assists the negotiations between buyers and sellers by providing agents that can autonomously negotiate and make the best possible deal on the user's behalf and allows the agents to use predefined negotiation strategies in the generation of offers/counter-offers. The user needs to decide which negotiation strategy his agent should follow during negotiation.

To achieve an agreement through negotiation, the negotiation agent can use various negotiation strategies available. Basically, according to the changing situation, they may [177]

- relax the soft constraints of the subgoal
- change the values of the properties in the bid
- further decompose the sub-goal into a set of sub-goals that make it easier for the seller / buyer agent to be satisfied.

In order to deal with inherent complexity and changing world information of the real-world transactions, the negotiation could also use the corresponding promising techniques such as multi-objective decision analysis and multi-attribute utility theory (MAUT), distributed constraint satisfaction, conjoint analysis, and machine learning [105][352].

8.3.3 Integrative Negotiation

Integrative negotiation and distributive negotiation are two types discussed in [105][292]. The former is the decision-making process of resolving a conflict involving two or more parties over multiple interdependent, but non-mutually exclusive goals. Desired retail merchant-customer relationships and interactions can be described in terms of integrative negotiation - the cooperative process of resolving multiple interdependent, but non-mutually exclusive goals [105]. In essence, integrative negotiation is a “win-win” type of negotiation, while distributive negotiation is a “win-lose” type of negotiation [105]. From a merchant’s perspective, integrative negotiation is about tailoring its offerings to each customer’s individual needs resulting in greater customer satisfaction. From a customer’s perspective, integrative negotiation is about conversing with retailers to help compare merchant offerings across their full range of value resulting in mutually rewarding and hassle-free shopping experiences. Therefore, an integrative negotiation through the space of merchant offerings can help maximize goals of consumer-owned shopping agents and merchant-owned sales agents across each product’s full range of value.

8.3.4 Summary

Multiagent negotiation is one of the main research activities in multiagent e-commerce [199][249][352], because negotiation is the common basis for auction, mediation, and brokerage in commerce. In other words, auction, mediation, and bargaining are more concrete forms of negotiation. The following sections will examine multiagent auction, multiagent mediation, and multiagent brokerage.

It should be noted that practical work on negotiation goes in two directions: negotiation support systems (NSSs) and Web-based negotiation support tools [2] (pp 107-8). NSSs are aimed to assist optimal agreement among human negotiators. The NSS by Rangaswamy and Shell uses computers initially to collect, refine, formalize rather vague human preferences and analyse the offers and suggest better solutions. INSPIRE© is a Web-based negotiation support tool developed by the InterNeg Group, Carleton University and Concordia University[1]. It can be used to study, teach, simulate, and facilitate real life situations. Although it focuses on cross-cultural international negotiations, both of those systems can be used for online negotiations.

8.4 Auction-based E-Commerce

Auction, mediation, and brokering are three key concepts for commerce. In fact, they are also major functions of an intermediary in business, although the essential part in all of them is negotiation. Therefore, e-commerce can be classified using these three concepts:

- Auction-based e-commerce
- Mediation-based e-commerce
- Brokerage-based e-commerce.

The first has been intensively studied at Michigan University, while the second has been carefully studied at MIT. Now it has been considered as the mainstream research of agent mediated e-commerce in European countries [66][67][269]. In what follows, these three kinds of e-commerce will be examined in some more detail.

8.4.1 Characteristics of Auction

Auction is a trade type that involves a seller (agent), many potential buyers (or buyer agents) [143], and an auctioneer governing the auction. The seller basically doesn't participate in auctioning but tells the auctioneer what the reserved price of the product is. The buyers bid sequentially to compete for the product to be sold. The main auction rule is that a bid is required to be higher than the last bid. During the final stage of the auction, the auctioneer indicates that he

1. See http://interneg.org/inspire/

is willing to accept the highest bid. The highest bidder expresses his wish to accept the auctioneer's offer [241].

Auction is a popular model of negotiation for open multilateral bidding [208][228]. Auctions are a simple form of negotiation to implement, due to their well-predefined rules and only involve single negotiation issue. Auctions are thus a paradigm used in many automated negotiation systems [219], and have become increasingly important part both for business transactions and for consumer purchasing in e-commerce [238].

The relationship between auction and negotiation can be verified based on the classification scheme mentioned previously, which formats auction as follows:

- Cardinality of the auction
  - negotiation domain: single issue; that is, price
  - interactions: m : 1; that is, where many bidders and one auctioneer (or seller)
- Agent characteristics
  - Role, i.e. auctioneer and bidders
  - Rationality: Bounded computation
  - Knowledge: Bidders have the basic knowledge about the goods, while the auctioneers have more knowledge about the goods. From the knowledge-based viewpoint, they have their own knowledge base about the knowledge of the goods and the knowledge about auction strategy
  - Commitment: Various levels of commitment can be present. For example, after having made an offer, agents might be obliged to stop bidding for similar goods until an acceptance or counter offer is received
  - Social behaviour, i.e. cooperation, coordination, and communication. The bidders can cooperate, coordinate, and communicate with other bidders to improve the bidding strategy
  - Bidding strategy: Which is the important part for the bidders to propose the bidding prices
- Event parameters
  - Bid validity: The bidders often have to offer the bids at an appropriate time and the bids must satisfy some constraints on their value [183]. For example, a bid is required to be higher than the last posted

  - Bid visibility: At auction, the bid is always visible to other bidders and the auctioneer
  - Clearing schedule and timeouts: These depend on which kind of auctions has been used
  - Quote schedule: N/A
- Information parameters
  - Price quotes, which is the reserved price in the auction
  - Transaction history: From the case-based viewpoint, this is stored in the case base
- Allocation parameters. The allocation governs the winner of an auction when more than one agent has shown an interest in the good [183]. The M-th and (M+1)-th price allocation policies cover most auction scenarios, where M is the number of received bids.

The most popular auctions are English auctions, Dutch auctions, first-price sealed bid auctions, and second-price sealed bid auctions (also caled Vickrey auctions) [228][267]. In an English auction, the auctioneer offers a good for sale and the bidders bid the price they are willing to pay [238]. Each bid announced must be greater than the previous bid, and then the item is sold to the highest bidder. In a Dutch auction, the process runs in reverse. The auctioneer announces a proposed price, and bidders can accept it if they choose. As time progresses, the auctioneer decreases the proposed price until a bidder accepts. In a Vickrey auction, bidders place their bids in a sealed envelope and submit them to a trusted third party or auctioneer. At a certain time, the auctioneer opens the envelope, and the item is sold to the bidder at the second highest price.

8.4.2 Electronic Auction

This subsection looks at the open-bids auction [241], which follows an offer which was posted and published at the auction's start. The auction is open for a limited time interval. A bid is required to be higher than the last posted. During the final stage of the auction, the auctioneer indicates that he is willing to accept the highest bid. The highest bidder expresses his wish to accept the auctioneer's offer. The highest bidder buys if its bid is greater or equal to the reserved price.

The above open-bids auction can be transformed into e-auction or Web-based auction; that is, a set of users subscribed to an e-auction website forms a community. A set of trade-objects is published for sale in e-auction. A subcommunity is formed when (1) a vendor offers a trade-object with a reserved price, the minimum price that the vendor is committed to the auctioneer, and (2) an auctioneer takes charge of the offer and sets the auction rules.

Once the two roles (i.e. vendor, auctioneer) are fulfilled, the auctioneer opens the auction and bidders can start bidding, shown in Fig. 8.3. Numerous *policy rules* for e-auction have been developed as in the traditional auction process. For example, the auctioneer and the vendor cannot bid; only subscribed users can bid. From the viewpoint of agent technology, e-auction bidders are mobile, adaptive, and autonomous.

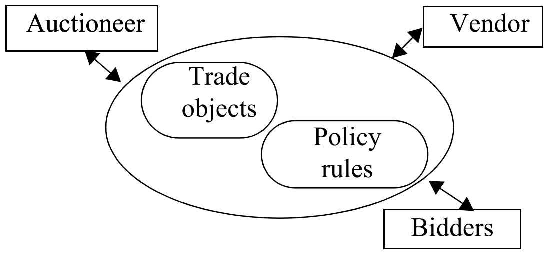


Fig. 8.3 E -auction environment after [241]

The dramatic development of the Internet has led to a plethora of e-auctions on the Internet, which offer electronic implementations of traditional auctions; that is, they offer integration of the bidding process with contracting, payments and delivery [300] (p 37). The sources of income for the auction provider are in selling the technology platform, in transaction fees and in advertising. Benefits for suppliers and buyers are increased efficiency and time savings, no need for physical transport until the deal has been established. Because of the lower cost, it becomes feasible to offer small quantities of low value, e.g. surplus goods for sale. Sources of income for supplier are in reduced surplus stock, better utilization of production capacity, and lower sales overheads. Sources of income for buyers are in reduced purchasing overhead cost and reduced cost of goods or services purchased. Examples for B2B electronic auctions are Infomar and FastParts[1].

Some websites are running English auctions, such as Auction sales (http:// www.auctionsales.com) [238]. These auction sites offer a bidding agent which bids on your behalf. The user enters the maximum he is willing to pay, and it places the lowest possible bid on the website. If all bidders in an auction use such an agent, the auction becomes a Vickrey style

1. http://www.fastparts.com

auction instead of an English auction. The sale is made at second price plus the minimum bid increment.

There are automated, semi-automated, and manual auctions currently online [228]. The auctioning is restricted to price. Auctions do not function very well in cases where the buyer needs a customised product or service, because they reduce the negotiation to a single negotiation issue: price [313]. Another problem is that bids in auctions are usually binding and therefore the user has to trust his agent.

8.4.3 Multiagent-based Auctions

E-auction has also been improved using MAS techniques [66]. Examples for multiagent auction are ONSALE®, AuctionBot, and Fishmarket.

ONSALE® is an e-auction site where people submit bids on products according to the rules of the auction [332]. There is an opportunity for intelligent agents to participate in this e-auction. Prototype marketplaces have been developed where potential buyer agents compete against each other using game theoretic strategies to outwit the other bidders.

AuctionBot is a general-purpose e-auction server [188], and provides an automated auction house for experimentation with bidding algorithms. Its users create new auctions by choosing an auction from the auction types and then specifying its parameters such as clearing times and method for resolving tie bids. Buyers (or buyer agents) and sellers (or seller agents) can then bid according to the auction’s multilateral distributive negotiation protocols.

The Spanish Fishmarket provides a sophisticated multiagent platform for an electronic auction [18] (p 156). It follows the different scenes that a traditional fish auction involves, as registration of the sellers goods in the auction house by a seller manager, register of the buyers by an admitter agent, bid for goods directed by a seller manager. The Fishmarket defines an electronic institution that is managed by a central agent. All the interactions between the other agents and the institution are managed by this manager agent. Each agent is given an identification and an interaction protocol that defines which *ilocutions* can be used and its meaning.

8.5 Mediation-based E-Commerce

Mediation-based e-commerce stems from the work in MIT [104]. Recently it has attracted increasing interest in Europe [66][67]. Dignum et al. [66] use agent-mediated e-commerce to cover all attempts of applying intelligent agents or MASs in e-commerce. However, mediation is only one of main business role of intermediaries in commerce, as discussed in Section 8.3. The automated mediation based on MASs can only be considered some of all attempts of multiagent e-commerce. This section argues that mediation is one special case of negotiation in e-commerce. Therefore, it is better to use mediation-based e-commerce instead of agent-mediated e-commerce. More specifically, mediation-based e-commerce is a special form of negotiation-based e-commerce. It emphasizes the role of agent's mediation in e-commerce, and is realized with MAS technology.

8.5.1 Mediation and Intermediatory

An dictionary definition[1] states that to *mediate* is to "arrange (an agreement) by talking to two separate people or groups involved in a disagreement, or to arrange a connection between two things, people or groups." Thus, from a commerce viewpoint, a mediator is an independent intermediary mediating the interests of two opponents (e.g. buyers and sellers), which is very common in real life business interactions [292]. For e-commerce, mediation is even more important [80].

Mediation can also be verified based on the classification scheme for negotiation mentioned previously [183] as follows:

- Cardinality of the auction
  - negotiation domain: single issue or more issues
  - interactions: 2 : 1; that is, a mediator basically interacts with two buyer agents and seller agents in e-commerce
- Agent characteristics
  - Role, i.e. mediator
  - Rationality: Bounded computation
  - Knowledge: Mediators have the basic knowledge about how to arrange the buyer agents and the seller agent to reach a deal
  - Commitment: N/A

1. Cambridge International Dictionary of English, Cambridge University Press, 1995.

  - Social behaviour, i.e. cooperation, coordination, the mediator can coordinate the buyer agent and seller agent to resolve the disagreement
  - bidding strategy: N/A
- Event parameters
  - Bid validity: N/A
  - Bid visibility: N/A
  - Clearing schedule and timeouts: N/A
  - Quote schedule: N/A
- Information parameters
  - Price quotes: N/A
  - Transaction history: From the case-based viewpoint, mediator has also rich experience in mediation, which is stored in the mediation case base
- Allocation parameters. N/A.

Based on the above discussion, the mentioned classification scheme for negotiation is less important for modelling mediation than auction. Further, from a viewpoint of e-commerce, the mediators have some different tasks from those of auctioneers. For example, according to Wierderhold [333], mediators are modules occupying an explicit, active layer between the user and applications and the data resources. They will be accessed by application programs residing in the user workstations. Mediators form a distinct middle layer, making the user applications independent of data resources. Further, in most cases, negotiation is not the main function of the mediator [264].

A mediator may provide any of the following main forms of assistance in the resolution of traditional industrial conflict [254] (p 61):

1. Reducing irrationality
2. Reducing non-rationality, by main interventions that enable the parties to clarify their intentions and their expected gains and costs
3. Exploring alternative solutions
4. Facilitating (constructive) communication between opposing parties
5. Regulating the costs of conflict

6. Establishing and reinforcing norms and rules of procedure.

It should be noted that not only the auctioneer but also the mediator belong to the categories of intermediary, which has also drawn increasing attention in e-commerce. An electronic intermediary is a business entity that performs at least one intermediation function [264]. Intermediation functions are those which help or completely enable a buyer and a seller to complete a transaction. Mediated transactions use an outside third party to give some assistance to at least one party (sometimes both) in at least one commercial function. It has been argued that in the perfect electronic market, buyers and sellers will be able to contact each other in a direct, frictionless manner, thereby “eliminating the middleman”. However, evidence in the marketplace demonstrates that at least for some time to come, the role of intermediaries is becoming increasingly important, in particular with the heavy information overload on the Internet. The major functions of an online intermediary are searching, trust, aggregation, as an infomediary, negotiation etc. [264][332].

8.5.2 Multiagent-based Mediation

Mediators, as special intelligent agents, have been proposed to optimize the whole buying experience and revolutionize commerce [209]. The personalized, continuously running, autonomous nature of agents make them well-suited for mediating consumer behaviors involving information filtering and retrieval, personalized evaluations, and complex coordinations when certain prespecified conditions apply [188].

Guttman et al. believe that it is useful to explore agents as mediators in e-commerce in the context of the traditional marketing consumer buying behaviour (CBB) research with concepts from software agents to accommodate e-markets [104][105]. The CBB consists of six fundamental stages which guide buying behaviour of customers:

1. Need identification or recognition: the customer becomes aware of some unmet need. Within this stage, the buyer can be stimulated through product information
2. Product brokering: the retrieval of information to help determine what to say. This includes the evaluation of product alternatives based on buyer-provided criteria
3. Merchant brokering: to determine who to buy from. This includes the evaluation of merchant alternatives based on buyer-provided criteria (e.g. price, warranty, availability, delivery time, reputation, etc.)

4. Negotiation: how to determine the terms of the transaction
5. Purchase and delivery: the signal of termination of the negotiation
6. Service and evaluation: product service.

These six stages also elucidate where agent technologies apply to the customer shopping experience and allow us to more formally categorize existing agent-mediated e-commerce system [259][209]. Based on this model, intensive research and development of agent-mediated e-commerce systems has been done to automate one or some stages of the model. Guttman et al. [105] survey some of these systems, as shown in Table 8.1. The following will look at Tete-a-Tete and Market Maker for some detail.

Tete-a-Tete is a multiagent electronic marketplace, and engages consumer-owned shopping agents and merchant-owned sales agents integrative negotiations to maximize their owner's individual needs [209]. Tete-a-Tete sales agents automate the negotiation process for merchants. Shopping agents, on the other hand, actively assist shoppers during negotiations by providing a level of decision support to help them decide which merchant offering best meets their needs. This decision support is based on multi-attribute utility theory (MAUT). Tete-a-Tete's MAUT mechanism provides a real-time, utilitarian shopping experience.

Table 8.1 the On-line shopping framework with representative examples of agent mediation

Stage in BBM	Person alogic	Firefly	Bargain Finder	Jango	Market Maker	Auction -Bot	Tete-a-Tete
1. need identification							
2. product brokering	x	x					x
3. merchant brokering			x	x			x
4. negotiation					x	x	x
5. payment & delivery							
6. service & evaluation							

Market Maker is also a multiagent electronic marketplace where agents buy and sell to one another on behalf of consumers [188][209]. The consumer must decide whether it is a buying agent or selling agent. In the Market Maker environment, a selling agent is analogous to a classified ad. A user creating a new selling agent describes the item the agent is to sell. For a buying agent, the user specifies the values for a list of parameters: Sell by, desired price, and lowest acceptable price. Selling agents are proactive. Basically, they go into the marketplace, contact buying agents, and negotiate with them to find the best deal. A selling agent is autonomous in that, once released into the marketplace, it negotiates and makes decisions on its own, without requiring consumer intervention. Nonetheless, the consumer has high-level control of the agent's behaviour because in creating a new selling agent, the user sets several parameters to guide it; that is, desired date to sell the item by, desired price, and lowest acceptable price [209]. The user always has final control. When a selling agent reaches an agreement with a buying agent, the respective users may want to give an OK before the agents "shake hands" on the deal. The agent has a negotiating strategy, which can be chosen from the negotiation strategies predefined in Market Maker.

The MIT Media Lab's multiagent-based mediation systems are already creating new markets (e.g. low-cost consumer-to-consumer and refurbished goods) and reducing transaction costs in a variety of business models [209].

However, it is significant to develop a MAS to cover more stages in the CBB model. The CASBA project is developing an e-marketplace to improve the quality of existing e-commerce services, which is achieved through using the intelligent agent technology, enabling the market framework to offer timesaving automation of auctions and flexibility through negotiations among agents [312]. The goal of CASBA is to support all stages of the mentioned CBB model (at least to some extent for each stage) with the main part being the stage 3 and 4. The CASBA architecture is shown in Fig. 8.4.

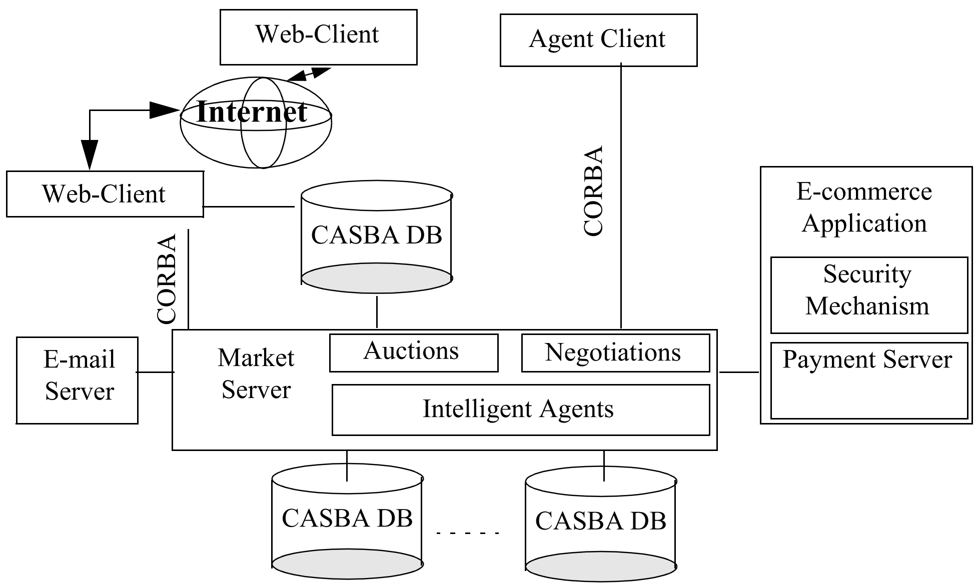


Fig. 8.4 CASBA Architecture after [312]

It should be noted that the CASBA is not merely a multiagent mediation system but a hybrid MAS, because it includes the functions of negotiation, auction, mediation, and brokerage (i.e. merchant brokering).

8.5.3 Summary

Just as more and more economic activities were delegated to specialists in the last few years, most market transactions in the future will also mediated by specialists, though most likely in the form of intelligent agents [310]. That is, intelligent agents will play a more important role in market transactions, in particular in automating the mentioned CBB model. However, in order to let a MAS support all stages of the CBB model, it is necessary to apply hybrid techniques to MASs, like CASBA has attempted. This topic is of practical significance, although it is beyond the focus of this chapter.

8.6 Brokerage-based E-Commerce

With the rapid development of e-commerce, selling direct to customer, for example, selling air tickets direct to customer through online booking, will remove some intermediaries. This is so called “disintermediation”. However, new types of intermediaries will emerge [53]; that is, the software counterpart of traditional intermediaries, or intelligent intermediaries will play an important role in e-commerce. For example, personalogic.com is brokering for product recommendation and selection. This section will examine brokerage-based e-commerce and highlight multiagent brokerage.

In the research of multiagent e-commerce there are three popular terms; that is, auction, mediation, and brokerage. But there is less research on their relationships so far. This section will thus argue that auction, mediation, and brokerage are three special kinds of negotiation, and investigate their relationships, their applications, their features and differences in bargaining and compromise. First of all, this section will examine the characteristics of brokerage and discuss the relationships between auction, mediation, and brokerage.

8.6.1 Characteristics of Brokerage

Brokerage is another trade type that involves many buyers, many sellers, and a broker, which can be considered as a concrete form of negotiation. A typical example is the real estate broker. Both buyers and sellers submit their requests to the broker. The broker tries to match the requests through bargaining and compromise with buyers and sellers. From a viewpoint of business history, brokerage is one of the main trading transactions in traditional business activities [85]. It still plays an important role in consumer purchasing and decisions as well as commercial transactions [179].

Brokerage can also be verified based on the classification scheme for negotiation mentioned previously [183] as follows:

- Cardinality of the auction
  - Negotiation domain: single issue or multi-issues
  - Interactions: 2 : 1; that is, the broker will interact with the buyer agent and the seller agent respectively
- Agent characteristics
  - Role, i.e. broker
  - Rationality: Bounded computation
  - Knowledge: The buyer agent and the seller agent have the basic knowledge about the goods, while the brokers have more knowledge about the goods, because the broker is in essence the agent of both the buyer agent and the seller agent [292]. From a knowledge-based viewpoint, they have their own knowledge base about the knowledge of the goods and the knowledge about bargaining strategies
  - Commitment: Various levels of commitment can be present. For example, after having made a bargaining round[1], the broker or the buyer agent or the seller agent might be obliged to stop bargaining for similar goods until an acceptance or counter offer is received
  - Social behaviour, i.e. cooperation, coordination. The broker can coordinate the buyer agent and the seller agent. The buyer agent and the seller agent can cooperate with each other to improve the bargaining strategy
  - Bargaining strategy, which is the important part for the broker, while bidding strategy is the important part for the bidders to propose the bidding prices in auction
- Event parameters
  - Bid validity: N/A

1. The simple cycle of offer and counter offer is a bargaining round.

  - Bid visibility: N/A
  - Clearing schedule and timeouts: N/A
  - Quote schedule: N/A
- Information parameters
  - Price quotes, which are complex compared with the reserved price in auction, see the discussion of compromise space and bargaining
  - Transaction history: From the case-based viewpoint, this is stored in the case base
- Allocation parameters. N/A.

From the above discussion, one can also see that brokered systems often involve three conflicting parties (buyer agent, seller agent, and broker) rather than two parties (buyer agent and auctioneer) of online auctions. Although mediated systems also involve three parties (mediator, buyer agent, and seller), it has only two real conflicting or competing parties (buyer agent and seller agent). The mentioned classification scheme for negotiation is less important for modelling brokerage than auction. In fact, it seems that the mentioned classification scheme for negotiation is most appropriate for classifying auction. From a viewpoint of agent-based e-commerce, brokers have some different tasks from those of auctioneers and mediators. For example, brokers pay more attention to compromise, and proactively bargain with buyer agents and seller agents during the brokerage, which plays a lesser role in mediation and auction, because a broker can be defined to be a mediator performing the negotiation function [264]. Further, the bargaining process is the main trading transaction in traditional business activities [253][263]. It still plays a central role in current business activities, although the latter has evolved into a complex hierarchical organization. This is the reason why brokerage is more complicated than auction and mediation.

While full-blown automated bargaining has been widely recognised as an important function of an on-line intermediary [264], it is very difficult to actually implement automated bargaining into an on-line intermediary for all the reasons it is very difficult to implement automated bargaining directly into buyers and sellers. However, single-attribute intermediation, over price alone, in the marketplaces of many buyers and sellers, is a form of an on-line auction, and can be implemented for all the same reasons on-line auctions are currently feasible and successful.

While auction theory is well understood and well grounded, there is no coherent body of "brokerage theory" to provide the same base for brokers [264], because:

- Traditional bargaining theory tools do not apply
- Tightly linked buyer-seller-broker strategies are complex
- Dynamic time dimensions of the system make solution difficult.

8.6.2 Agents and Brokers in the Bargaining Process

This subsection first considers the goals and characteristics of a seller agent, a buyer agent, and a broker in the bargaining process, which essentially determine their behaviours and roles in the brokering (Fig. 8.5). Then it differentiates brokers, seller agents, and buyer agents from a functionality viewpoint.

On behalf of a seller, the seller agent's primary goals are long term profitability through selling as many products as possible to as many buyers or buyer agents as possible for as much money as possible with transaction costs as low as possible [86][105]. As a representative of a buyer, the buyer agent's primary goals are to have the buyer's special needs satisfied through the purchase of well-suited products from appropriate sellers or seller agents for as little money and transaction cost as possible. It is obvious that the primary goal of a seller agent is opposite to that of a potential buyer agent and vice versa. The main goal of a broker is to resolve this contradiction between the seller agent and buyer agent to get the satisfaction of both of them through a bargaining process. In other words, the primary goal of a broker is to make best use of his available resources and information to help maximize both of these goals and at the same time to get as much money as possible (not only the surcharges from both buyer agents and seller agents) through bargaining processes. Brokers thus play a central role in bargaining processes.

From a seller agent's perspective [86], the bargaining process is that the broker tailors the seller's offerings to each buyer or buyer agent's individual needs resulting in greater satisfaction. In a buyer or a buyer agent's opinion, the bargaining process is about the buyer's broker

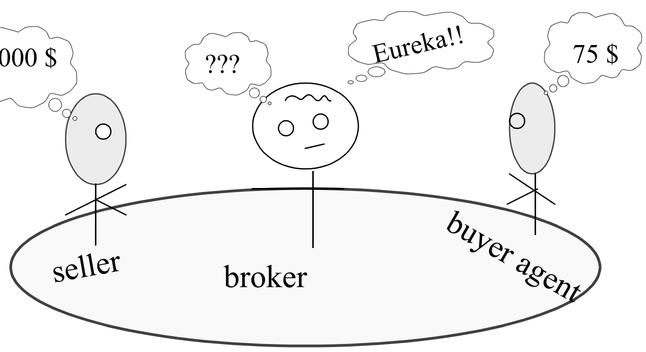


Fig. 8.5 Agents and broker in the bargaining process after [86]

conversing with seller agents to help compare their offerings across their full range of value resulting in mutual reward. Therefore, not only seller agents but also buyer agents actively communicate and negotiate with the broker in order to get maximal profits. However, from a broker's perspective, the bargaining process is similar to a battlefield, on which the broker should win both seller agents and buyer agents. In order to realize his goal, the broker should have lots of product information from the seller agents [119]. The broker should have also lots of information about the buyer agents' needs. Therefore, the necessary condition for a successful broker is to have two sets of customers, i.e. seller agents and buyer agents, and their information. But this is still not sufficient. The broker should use various reasoning methods or tricks to make both seller agents and buyer agents feel they have obtained satisfaction because of his bargaining [40][125][298]. The broker should further segment the goods information from the seller agents and the needs information from the buyer agents and use appropriate matching methods to provide each buyer with appropriate goods. Finally, the broker should actively communicate, coordinate, cooperate, and negotiate with both seller agents and buyer agents [119].

8.6.3 Principles of Bargaining and Compromise

This subsection will examine the role of bargaining and compromise in negotiation, in particular in brokerage, and also investigate the relationship between bargaining and compromise. First of all, it begins with the definitions of bargaining and negotiation.

To bargain: to negotiate over the terms of a purchase, agreement or contact... to establish an agreement between parties settling what each shall give and take or perform and receive in a transaction between them [254].

To negotiate: to deal or bargain with another or others... to confer with another so as to arrive at the settlement of some matter.

Based the above definitions, bargaining and negotiation are defined in nearly equivalent fashion [254] (p 2). However, in practice and also in context of this work, negotiation is a general concept, while bargaining is a concrete action in negotiation. In other words, the bargaining is the main activity in the negotiation process.

According to Rubin and Brown [254], bargaining relationships have the following characteristics:

1. At least two parties are involved
2. The parties have a conflicting interest with respect to one or more different issues
3. Regardless of the existence of prior experience or acquaintance with one another, the parties are at least temporarily joined together in a special kind of voluntary relationship [254]. The most important word in the above statement is the word *voluntary*. For bargaining to exist, the parties must believe they are participants by choice rather than by compulsion. Each is thus confronted with two important and related kinds of choices
4. At least some degree of commonality of interest for bargaining to occur, although their interests are partly in conflict [254] (p 10)
5. The activity in the relationship concerns (a) the division or exchange of one or more specific resources and/or (b) the resolution of one or more intangible issues among the parties or among those whom they represent
6. The activity usually involves the representation of demands or offers by one party, evaluation of these by the other, followed by concessions and counter-offers. The activity is thus sequential rather than simultaneous [254] (p 14).

In what follows, the subsection looks at the role of bargaining in auction, mediation, and brokerage.

For auction, the auctioneer and bidders have conflicting interests, because the auctioneer is responsible for the seller. However, the auctioneer does not take part in the bargaining among the bidders. The real conflicting interests happen among the bidders, not between the auctioneer and bidders.

Similar to auctioneers, the mediator does not require bargaining with the buyer agents or seller agents, because the mediator and buyer agents or seller agents do not directly constitute the conflicting parties. The main task of mediators by mediation is to arrange that the buyer agents and seller agents bargain in order to reach an agreement. However, different from the auctioneer and the mediator, the broker requires bargaining directly with buyer agents and seller agents respectively in order to gain not only the surcharge but also the bargaining profit from the buyer agent and seller agent.

Besides bargaining, another important concept for negotiation is compromise. Generally speaking, to *compromise* is to reach agreement in an argument in which the conflicting parties reduce their demands in order to agree. In practice, if the conflicting parties do not have any intention to compromise in the negotiation process, then the negotiation is difficult to proceed. Thus, compromise (the mentioned concession) is the necessary condition or preparation for any negotiation. Further, any bargaining requires the compromise of the conflicting parties, while any compromise of the conflicting parties might lead to further bargaining[1]. Negotiation can thus be considered as a sequential series of bargaining and compromise; that is:

$$\text{Negotiation} = \text{bargaining} \rightarrow \text{compromise} \ldots \text{bargaining} \rightarrow \text{compromise} \tag{1}$$

The mediator and broker have the same task to arrange the parties involved in the disagreement to express their compromise in order to let the negotiation go on, and therefore they facilitate the compromise of the parties involved in the negotiation process. Different from the mediator, the broker is usually one of the conflicting parties, because, he usually bargains with buyer agents and seller agents respectively. Therefore, the mediator only arranges the conflicting parties (buyer agents and seller agents) to prepare compromise over the negotiation issues, while the broker himself must prepare to compromise over the negotiation issue with buyer agents and seller agents respectively. Further, different from the mediator and the broker, the auctioneer does not even require to arrange the bidders (it is difficult to say bidders are the conflicting party with the auctioneer) to prepare compromise. The relationship between compromise and bargaining for negotiation, auction, mediation, and brokerage is now summarized in Table 8.2.

Table 8.2 Compromise and bargaining in negotiation

	Compromise	**Bargaining**
Negotiation	necessary	main activity
Auction	not necessary	not necessary
Mediation	not necessary	not necessary
Brokerage	necessary	necessary

It should be noted that with the refinement of business activities and development of science and technology, the border between agents and brokers becomes fuzzy. The activities for bargaining and compromise have become less and less with the selection space of customers becoming larger and larger. Some functions of a traditional broker have also been delegated by other special agents, by whom there are no bargaining and compromise any more. For example, in

1. This also means that a bargaining process is a compromise process.

current society, an insurance broker finds the information and presents the alternatives to his customers. He has neither intention nor time to bargain with his customers. What the customer should do is to select the satisfactory one from the insurance items that meets his requirements. However, brokerage is still an important business activity in negotiation of wholesale businesses. Further, a thorough investigation into brokerage can facilitate bargaining and compromise, although the latter has not drawn enough attention in e-commerce. In what follows, the subsection proposes a formal model of compromise and bargaining in negotiation from a viewpoint of brokerage[1], in which bargaining and compromise are necessary.

Assume that the negotiation is a n-lateral (bilateral, trilateral, ...), m-attribute negotiation (this is more formal for modelling compromise and bargaining). For brevity, the negotiation involves three agents ($n = 3$); that is, a broker, a buyer agent, and a seller agent, and m different attributes (which correspond to negotiation issues), $a_1, \ldots, a_m$[2]. For example, in real estate brokerage, the negotiation involves three agents: real estate broker agent, buyer agent, and seller agent. The real estate negotiation usually involves the following issues: the number of rooms, the convenience of the transport, and price, etc. For each j, the domain of a_j is U_j; for brevity[3], $U_j = R^1$. Then an offer (or counter offer) can be defined as $o = (a_1, \ldots, a_m), a_j \in U_j$ and can be autonomously generated by the agent's strategy [39].

For the negotiation issue j, the agent i has a *compromise subspace*, which is a subset of U_j, denoted as C_{ij}, and must be satisfied during the negotiation. Therefore, for each agent i, its compromise space C_i is

$$C_i = \{(a_1, \ldots, a_m) | a_j \in C_{ij}, j = 1, 2, \ldots, m\} \tag{2}$$

1. The brokerage can be made over a set of issues, instead of the single-issue price found in most auctions

2. a_j is also denoted for the corresponding attribute value

3. Negotiation in brokerage usually ranges over a number of quantitative (i.e. price) and qualitative (i.e. nature of the contract) issues. Each successful negotiation requires a lot of such issues to be resolved to satisfy three parties. Agents may be required to make trade-offs between issues (e.g. faster completion time for lower quality) in order to an agreement.

In other words, the compromise space is the collection of the compromises arranged for each negotiation issue.

For agent i, offer o is called a *feasible offer*, if it satisfies $o \in C_i$. According to the previous discussion, it is easy to come into the following result:

Proposition: The necessary condition for performing bargaining (negotiation) is

$$(C_1 \cap C_2 \cap \ldots \cap C_{n \angle 1} \cap C_n) \neq \varnothing \tag{3}$$

This is the formal representation of characteristic 4 of the above mentioned bargaining.

For example, in real estate, the price p and age of the house a are two bargaining issues. The compromise space of the real estate broker k, the buyer agent b the seller agent s is shown in Table 8.3.

Table 8.3 Compromise space of agents -I

	price($)	**age of house(years)**
buyer agent	150, 000-200, 000	6-10
broker	100,000-500,000	3-15
seller agent	180,000-210,0000	4-12

According to the above proposition, bargaining between the broker, the buyer agent, and the seller agent is possible.

In fact, every agent likes to obtain the highest profit or least loss by the bargaining or negotiation. Thus, it is important for each agent to understand what the real compromise space of other conflicting agents is by bargaining. This understanding is a process of *compressing* the compromise space of the conflicting agents. Therefore, the bargaining process is a process of narrowing the compromise space of the conflicting agents[1].

For example, after a bargaining round between broker with the seller agent and the buyer agent the compromise space become the following, as shown in Table 8.4.

With the ongoing bargaining, each agent will first compromise in some secondary important negotiation issues, and then in primary negotiation issues such as price in the mentioned example. However, in order to obtain the highest profit, the broker, buyer agent, and seller agent comply

1. It should be noted that by bargaining, one can not tell the truth or real compromise space to his conflicting agents. The tricky reasoning is also necessary for any bargaining [84].

with the following compromise principle, given the constant surcharges for the successful deal.

Table 8.4 Compromise space of agents-II

	price($)	**age of house(years)**
buyer agent	170, 000-190, 000	6-8
broker	140,000-300,000	5-12
seller agent	180,000-200,0000	4-9

For brevity, it involves only price as negotiation issue,

For broker, the principle of bargaining is to make the bid price of the buyer agent as high as possible and to make the asking price of the seller agent as low as possible.

For buyer agent, the principle of bargaining is to make the offer price of the broker as low as possible

For seller agent, the principle of bargaining is to make the offer price of the broker as high as possible.

Which negotiation issue is more important in the bargaining process, and whether the offer can be accepted, and what counter-offer should be submitted? All these depend on *offer evaluation*.

The offer evaluation can be based on multi-attribute utility theory (MAUT), which is a useful tool for making decisions involving multiple interdependent objectives based on uncertainty and preference (utility) analysis [39][66], p. 16). In order to do so, agent i must further take into account the preferences for each negotiation issue $j \in \{1, \ldots, m\}$ under bargaining:

- A weight w_{ij}, which is the relative importance of the issue j in the issues in question for agent i, where $\sum_j w_{ij} = 1$
- A scoring function $f_{ij}: C_{ij} \rightarrow [0, 1]$, which was assigned a value by the agent i for issue j and every value in the compromise subspace C_{ij}, the higher is the score, the better is the agent's utility or the agent's preference.

Therefore, the preference or utility of offer $o = (a_1, \ldots, a_m)$ in the compromise space for agent i is:

$$f_i(o) = \sum_j w_{ij} f_{ij}(a_j) \tag{4}$$

After generating an offer, agent i will decide on submitting it upon comparing its utility to the one associated with the previously received counter-offer. The one with highest utility will prevail.

It should be noted that Segev and Beam [264] investigate the effect of search costs and brokerage costs on the performance of the broker's optimal strategy for the case of M buyers and N sellers based on probability, where the negotiation attribute is single; that is, the price attribute of the product or service. Cardoso and Oliveria [39] propose a negotiation model based on MAUT, in which they discuss multilateral negotiations over a set of negotiation issues. It seems that compromise and bargaining have not yet played a role in the mentioned works.

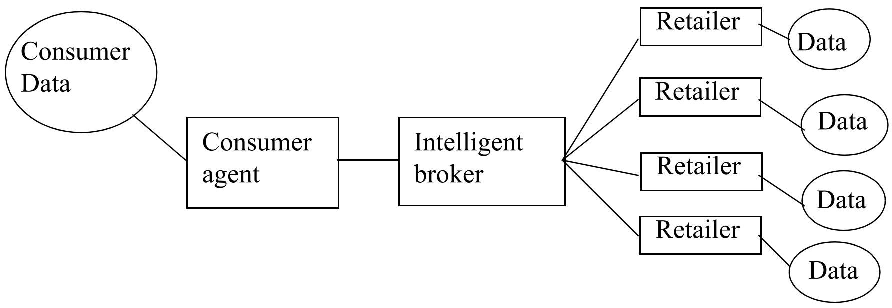
8.6.4 Electronic Brokerage

An electronic brokerage is an attempt to automate the traditional brokering process whereby human seller agents, buyer agents, and a broker bargain resources for mutual intended gain to finalise a deal, using the tools and techniques of e-commerce [86]. With the rapid development of the Internet and WWW, the electronic brokerage market is growing exponentially[1] based upon the promise of speed, convenience, and cost-effective access to markets. Further, most brokers on the Internet concentrate on the aggregation of information from underlying electronic catalogs [26]. Anderson Consulting's Bargainfinder and Netbo's Jango are some of the most well known examples for brokers supporting dynamic data gathering. Guttman et al. [188][104] analyse seven brokerage services and show which phases they support based on MAS technology (also see Section 8.5). Electronic brokerage is regarded as a core functionality in overcoming many current limitations of e-commerce [26].

8.6.5 Multiagent Brokerage

Multiagent technology has been applied to electronic brokerage [57][264]. The multi-agent brokerage framework is similar to the e-commerce framework from a viewpoint of the functional relationship of the constituents, because agents share their information and cooperate with other agents to achieve a global goal through the communication channel, dialogue method, and control

1. http://www.zonaresearch.com/info/press/99-sep21.htm



. *Consumer Agent*

- Monitors a consumer's data for unusual conditions
- Negotiates for products
- Negotiates for data exchange
- Provides needs and preference data to intermediary
- Maintain levels of privacy
- Provides payment authorization.

. *Broker: Intelligent broker*

- Use consumer needs and preference data to search for products
- Negotiates for products
- Negotiates for information exchange between consumer and business
- Maintains levels of privacy for consumer
- Provides appropriate marketing to consumer for business

Fig. 8.6 An intelligent broker in a marketplace after [332]

mechanism. Many researchers have developed electronic brokerage frameworks based on multiagent technology [264]. CDNOW®[1] is an example of an intelligent broker who shops for music CDs and finds the consumers' choice at the lowest price [332]. This broker is a natural application for intelligent agents, and in a business with large volumes of transactions is absolutely necessary. The retailer agents may also play the role of infomediary, negotiating for the exchange of products and services as well as data.

Finnie, Sun, and Weber [85][86] propose a framework for a broker-centred multiagent brokering processes (BCMB). The original ideas behind this is that the work of a human broker, for example, searching for requirement information of buyers and supply information of sellers, matchmaking and bargaining, should be done by a few intelligent agents in a MAS in a cooperative way. Further, as opposed to other multiagent brokerage systems, in BCMB the broker's main task is bargaining in the e-marketplace on behalf of both agents of the seller and the buyer. The rest of this section will examine intelligent buyer agents, intelligent brokers, and intelligent seller agents in the brokering process from the viewpoint of MASs, which is the preparation for examining the framework of multiagent brokerage.

In the bargaining process of the multiagent brokerage, similar to human agents, buyer agents and seller agents[2] should have a certain level of autonomous agency. For example, buyer agents

1. See URL: http://www.cdnow.com/

act independently of a buyer, but take into consideration the buyer's requests (e.g., gather and filter new data based on their buyer requests) [34]. They should also have certain adaptive behaviours to improve their understanding of the user's (i.e. seller's and buyer's) desires and intentions. Further, they should have cooperative ability to work with other buyer agents or seller agents to some extent. However, intelligent seller agents or buyer agents do not necessarily have the ability of mobility, proactivity, and reasoning in some cases. In contrast, the intelligent broker as a smarter intelligent agent, should have very strong mobility, proactivity and reasoning ability, so that it can search information from all available Web servers proactively, and select the most useful information, which might meet the offers of a seller agent or the needs of a buyer agent. The intelligent broker then uses as many different reasoning methods as possible such as case-based reasoning (CBR), abductive CBR [293], model-based reasoning (MBR) and rule-based reasoning (RBR) [34] and reasoning with trick [285], to negotiate with the buyer agents and seller agents, if necessary.

8.7 An Architecture of Multiagent Brokerage

As already mentioned, brokering, in particular bargaining, is one of the important business activities (also see [84][85]), in which the buyer agents, seller agents, and the broker are main players. The broker plays a central role in the bargaining process. This section views the broker involved in the bargaining process as a business agent and proposes an architecture to model it using MAS technology, which is an intelligent broker-centred MAS (IBCMAS) for automating brokerage in e-commerce.

Using MAS technology, one could develop a broker-centred MAS to assist the broker to work in the bargaining process [84][85]. The system architecture (Fig. 8.7) mainly consists of three multiagent subsystems [287]: intelligent buyer agent subsystem, intelligent seller agent subsystem, and intelligent broker. While the intelligent seller agent subsystem consists of all available intelligent seller agents on-line, the intelligent buyer agent subsystem comprises all available buyer agents[1] on-line. The intelligent broker is also a MAS.

2. From this section on, buyer (or seller) agents are intelligent buyer (or seller) agents respectively.

1. For convenience, x agent stands for intelligent x agent, where x is i.e. buyer or seller, etc.

In this system, the buyer agents, seller agents, and intelligent broker are all proactive. The seller agent tries to cooperate and negotiate with the intelligent broker to find a most satisfactory deal to sell the goods of his seller. The buyer agent also tries to cooperate and negotiate with the intelligent broker to get what his buyer needs with least price. The intelligent broker proactively searches all available information about the supply and demand of goods on the e-market [188].

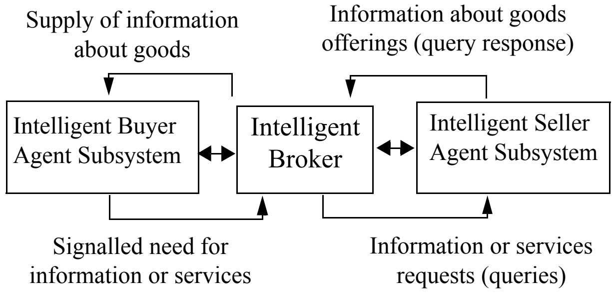


Fig. 8.7 An architecture for a broker-centered MAS

Every intelligent buyer and seller agent in this system is semi-autonomous, in that, once entering into the e-brokering process, the intelligent buyer (seller) agent bargains and makes decisions on his own, without requiring his buyer's (seller's) intervention. The intelligent broker autonomously bargains with the intelligent buyer agents or the intelligent seller agents in order to successfully achieve a deal.

As shown in Fig. 8.8, the intelligent broker is mainly comprised of a buyer (seller)

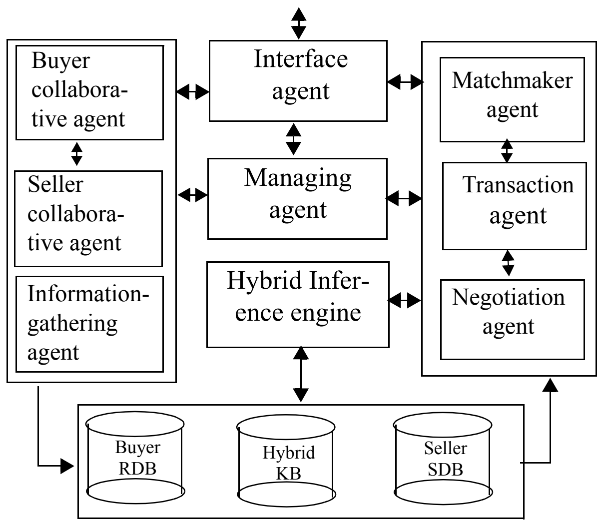


Fig. 8.8 An intelligent broker

collaborative agent, an information-gathering agent, an interface agent, a managing agent, a matchmaker agent, a transaction agent, a negotiation agent, a hybrid inference engine, and a hybrid knowledge base (Hybrid KB), a buyer request database (buyer RDB), and a seller supply database (seller SDB).

- The buyer (seller) collaborative agent [32] is an agent that proactively interacts and cooperates with the buyer (seller) agents and gets the supply-demand information and, at the same time, special information about the buyers and sellers, and then stores it in Buyer RDB or Seller SDB respectively, if necessary. In certain cases, the buyer (seller) collaborative agent decides if the related buyer agent and seller agent can directly contact each other in order to reach a deal
- The information-gathering agent [85] is a mobile agent that proactively roams around the main search engines in the Internet such as Excite and Yahoo. It interacts and collaborates with them in order to search and analyse the required market information indirectly from individual Web-sites [186][207] and then puts it in the corresponding data or knowledge bases
- The interface agent interacts with buyer/seller agents, buyer/seller collaborative agents, the managing agent, etc. to transfer the transaction message
- The managing agent plays a central rule in this subsystem. His main task is to decide which agent should do what and how to deal with an important transaction. Another task is to coordinate the tasks among all other agents [84]
- The matchmaker agent searches related databases and matches a request for goods from a buyer agent and a supply of those goods from a seller agent using appropriate matching algorithms [32]. It also matches the goods-requesting buyer agent and goods-supplying seller agent and then sends the matched information to the interface agent or buyer/seller collaborative agent after cost analysis by the transaction cost analysis agent
- The negotiation agent is a mobile and proactive agent that performs not only integrative but also distributive negotiation strategies during negotiation with the buyer agent and seller agent. Because business negotiation is complex and difficult in some cases, the intelligence of the negotiation agent lies in that he can change his negotiation strategies instantly according to the changing available (information) resources or cases. He prepares necessary compromise under bargaining. Thus, the negotiation agent may use all available human inference methods such as case-based reasoning (CBR), model-based reasoning (MBR), rule-based reasoning (RBR), fuzzy reasoning, and even reasoning with trick in different cases, if necessary. He can thus deceive or mislead the buyer/seller agents in a certain case [125]

- The transaction analysis agent analyses and computes every transaction cost of deals [259], which the negotiation agent and the matchmaker agent suggest, and then submits the recommendations to the negotiation agent and the matchmaker agent or the managing agent, if necessary
- The hybrid inference engine, containing a massive "reasoning kit", provides the agents, in particular, the negotiation agent, with the appropriate hybrid mechanism, so that the negotiation agent can easily perform a certain reasoning whenever it thinks necessary [85]
- Buyer RDB (seller SDB) stores the request (supply) information about goods provided by the buyer (seller) collaborative agent through the buyer (seller) agent
- Hybrid KB consists of knowledge extracted from buyer RDB and seller SDB. It is also comprised of the (rule, model, case, fuzzy, trick) knowledge about the market and negotiation. Hybrid KB and hybrid inference engines cooperate to facilitate all of the agents in the intelligent broker subsystem sharing the information in the concerning data or knowledge bases and making decisions in bargaining processes.

The previous section and this section investigated broker and multiagent brokerage and their goals, functionalities, architectures, and interrelationships with negotiation, auction, and mediation. It further examined the principle of bargaining and compromise in negotiation, in particular in brokerage. It then considered a broker involved in the brokering process as a business agent and proposed an architecture for multiagent brokerage.

8.8 Concluding Remarks

This chapter first reviewed intelligent agents in e-commerce; that is, agent-based e-commerce. Then it examined multiagent negotiation, which is the core of multiagent negotiation-based e-commerce. Further it classified multiagent-based e-commerce into multiagent-based auction, multiagent-based mediation, and multiagent-based brokerage and gave a brief survey of related works in each. Then it investigated multiagent brokerage. The main idea behind it is that the work of human mediators, auctioneers, and brokers such as searching the information of customers, matchmaking and brokering should be done by a few intelligent agents in a cooperative way. Bargaining and compromise play an important role in negotiation, in particular in brokerage.

Based on the characteristics of buyer agents, seller agents, and brokers, this chapter proposed an architecture of a multiagent-based intelligent broker for the brokering process and argued that such an architecture is an abstraction of human agents and brokers working in bargaining processes of brokerage. These approaches will facilitate research and development of multiagent e-commerce.

It should be noted that the multiagent e-commerce is far away from a real world one, because several features such as human behaviour are not considered. Human behaviour in real world commerce negotiation involves difficult points such as

- multiple attribute negotiation
- similar product suggestion
- correlated product suggestion
- learning or the experience of previous negotiations
- tricky (deceptional) reasoning in negotiation
- bargaining and compromise in negotiation.

To our knowledge, the first four features have been mentioned and examined in [62], and only the first of the above features has been incorporated by many e-commerce negotiation systems so far, and the rest has not been touched in multiagent negotiation.

In order to implement the mentioned intelligent broker, it is of significance to further investigate the inference methods, compromise strategies, and bargaining strategies of intelligent agents and intelligent brokers in brokering processes. It is also of significance to further examine communication and interoperability among the agents and brokers in the mentioned systems.

Part - III: Integration

Part III is the integration of three mentioned aspects under one roof. It consists of only Chapter 9, which is at fourth level, as shown in the shaded area of Fig. III

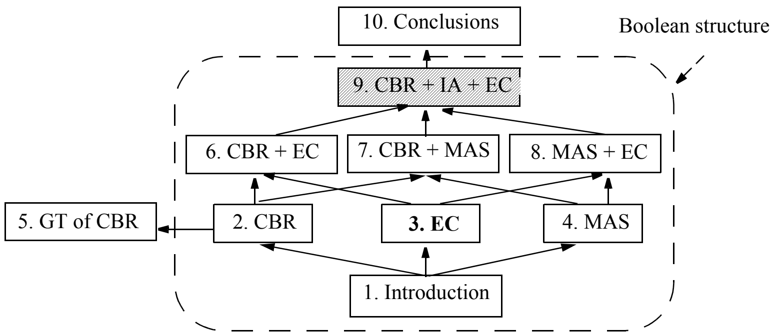


Fig. III. Part III in the Boolean structure of PhD-thesis

9 Integration of CBR and Multiagent Systems in E-Commerce

This chapter belongs to the Part III of the thesis, as shown in the shaded area of Fig. 9.1. It first examines parsimony of intelligence and artificial redundancy in MASs, which are practical strategies for implementing any MAS as well as intelligent systems. Then it presents CMB, which is a system integrating case-based reasoning (CBR) and multiagent brokerage. The key idea in CMB is that some agents have CBR ability while other agents have not, based on the parsimony principle of intelligence in MASs. Finally it investigates the analysis and design for implementing the proposed CMB.

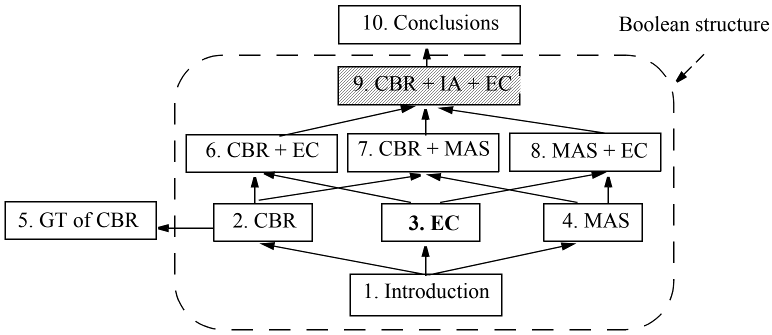


Fig. 9.1 Chapter 9 in the Boolean structure of PhD-thesis

9.1 Introduction

As discussed, applying CBR and MASs in e-commerce has drawn increasing attention in the CBR community. Chapter 8 (also see [85][84]) proposed a framework for a broker-centred multiagent brokerage. The original idea behind this is that the work of a human broker, for example, searching requirement information of buyers and supply information of sellers, matchmaking, and bargaining should be done by a few intelligent agents in a MAS in a cooperative way. Further, as opposed to other multiagent brokerage systems, the proposed broker's main task is bargaining in the e-brokerage place on behalf of both agents of the seller and the buyer. However, multiagent brokerage systems are still in a very early stage, although there are more and more brokerage services available on the Internet. One of the goals in this chapter is to extend the research in Chapter 8 using CBR and propose a framework for CMB, a CBR system for multiagent brokerage, which integrates CBR, multiagent systems, and brokerage under one roof.

It is not realistic to exploit all the facilities, theories, and techniques that this thesis offers with the available demonstration applications. However, this chapter tries to realize the framework of some theories and techniques in a unified way; that is, integrating CBR, MAS and e-commerce. At the same time it investigates the techniques available for implementing CMB.

Agent technology has also been applied to assist in decision-making. For example, Ba et al. [8] developed a client-broker-server architecture for Internet decision support through the coordination of interface agents, gateway agents, and information retrieval agents. Some also report the customer agents and vendor agents and their relationship [178]. However, the classification of customer and vendor is too simple to cover various trade types. It is useful to develop a more complicated framework that covers a spectrum of trade types. This chapter will examine these new advances of intelligent agents in e-commerce.

The rest of this chapter is organised as follows: Section 9.2 examines parsimony of intelligence and artificial redundancy in MASs. Section 9.3 examines case-based reasoning (CBR) for multiagent e-commerce. Section 9.4 investigates the architecture of CMB. Section 9.5 and Section 9.6 look into the analysis and design for implementing the proposed CMB. This chapter concludes with an overview of key research challenges and also attempts to extrapolate in future work.

9.2 Parsimony of Intelligence in Multiagent Systems

As mentioned in Chapter 7, there are a few multiagent CBR systems integrating CBR and MAS. In such systems, every agent possesses CBR ability. However, the people in any social system usually have different abilities. Therefore, a knowledge-based model of multiagent CBR systems is proposed in Section 7.2, in which a multiagent CBR system basically includes some agents with CBR ability and other agents without CBR ability. However, why some agents in the MAS do not require CBR ability is still an open problem. This section will examine the parsimony of intelligence in MASs to answer this question in a more general sense.

9.2.1 Intelligence of An Expert System

Although the concept of the intelligence of a system goes back to the 1950s [306], the definition of intelligence of the system can be regarded as [290]:

"Any software system is intelligent if it helps us work more efficiently and effectively, or if it helps us to understand our intelligence better, or it liberates our intelligence to some extent."

Therefore, the goal of Artificial Intelligence (AI) has always been to understand our intelligence better and to make artificial systems intelligent. As mentioned in Chapter 2, expert systems (ESs) used to be one of the most successful applications in AI as early as 1990s. Because an ES mainly consists of knowledge base and inference engine, the intelligence in the ES mainly depends on knowledge base and inference engine, which can be expressed in a short form:

$$\text{Int(ES)} = \text{int(Knowledge base + inference engine)} \qquad (1)$$

More specifically, the intelligence of an ES results from the quality of knowledge in the knowledge base and the power of inference engine.

It should be noted that computerization of knowledge is one of most significant contributions of ESs to computer science, which leads to knowledge engineering and knowledge base systems (KBSs). However, it is obvious that (1) is too simple to be understood that ESs can replace human experts.

9.2.2 Intelligence of Multiagent Systems

As mentioned in Chapter 4, multiagent systems (MASs) are among the most rapidly growing areas in AI communities with the rapid development of the Internet and WWW. The further reason for interest in MAS is that MASs can integrate distributed AI (DAI) and symbolic AI from a viewpoint of AI. A MAS is a group of agents that work together to find answers to problems that are beyond the individual capabilities or knowledge of each agents. Further, cooperation, collaboration, communication, and negotiation (for short, C^3N) are most important features of MASs, which are all social behaviours. Communication is fundamental for any social behaviour of human beings. Therefore, the individual intelligence depends on not only the intelligence of an individual agent but also the social behaviour of the individual agent; that is:

$$\text{int(MAS)} = \sum \text{int(Ai)} + \text{int}(\text{C}^3\text{N}) \qquad (2)$$

In other words, C^3N plays an important role in development or implementation of intelligence of the MAS; knowledge and reasoning alone are not enough to simulate the human intelligence. The key idea behind this is that the development of intelligence of a human being depends on not only his own knowledge or reasoning but also on his social behaviours in the society where he lives. Therefore, the simulation of human intelligence depends not only on the computerized knowledge and reasoning of human experts but also on the simulation of their social behaviour [294].

At least, the following aspects of intelligence of human beings have been discussed from a MAS viewpoint [125][294]:

- Individual reasoning

  Rational: deductive, non-monotonic reasoning, inductive reasoning, etc.

  Irrational: reasoning with tricks, deception, etc.

- Social intelligence: cooperation, collaboration, communication, negotiation, bargaining, autonomy, mobility, and so on.

The characteristics of agents are also tied to their intrinsic and extrinsic properties [125] such as: lifespan, level of cognition, construction, mobility, adaptability, modelling, locality, social autonomy, sociability, friendliness, and interaction. However, the characteristics of agents are too complex and many to be understood. In other words, there are so many properties of intelligence. Which are most important? For the Internet world, a mobile and autonomous agent might be most important, because the evolution from an ES to an autonomous and mobile agent is similar to that from bullet to missile [288]. This is the reason why MASs or mobile MASs have drawn increasing attention in AI fields.

There is a further question: Can you implement a MAS with all mentioned intelligent behaviours? The answer is certainly not at this stage of research. However, how to realize a MAS with as many aspects of intelligence as possible is a practical issue, which require us to discuss the parsimony principle of intelligence.

9.2.3 Parsimony Principle of Intelligence

Generally speaking, the *principle of parsimony* is a criterion for deciding among scientific theories or explanations. One should always choose the simplest explanation of a phenomenon, the one that requires the fewest leaps of logic (see http://www.wmg.org.uk/mcn/glossary/principleofparsimony.html).

The principle of parsimony is not a new idea in software engineering. In fact, it is usually used in the requirement analysis or modelling the components in a software system (also see [336]). We usually prefer to be parsimonious to meet the requirement of the customer when we develop software systems for him owing to the effect of enlarging errors in the late stages of software development. Furthermore, in many practical system models, we also prefer the simplest one that uses the least number of components and the simplest system structure that describes the real world scenario adequately.

However, many researchers are trying to build a MAS with as many intelligent behaviours as possible, based on the previous discussion. This disagrees with the principle of parsimony.

The principle of parsimony is related to the optimization principle and specialization principle, which are essential parts of mathematics and computer science. Optimization principle is a set of principles that optimize the process, model or system architecture, while specification principle is a set of principles that refine the modelling of a system in a special domain. For example, code optimization can simplify procedure and code, based on mathematical logic. Specialization principle can be considered as an academic principle, although in engineering one also uses integration (or synthesis) principle; however, the latter is usually based on specialization principle. Further, specialization is the necessary premise for optimization. “Divide and conquer” is an ordinary methodology in AI, in particular for problem reduction. “Divide” in consideration is, in essence, a concrete action of specification. “Conquer” is a concrete action of solving the divided and simpler problems. Optimization is a further action after “conquering” the divided problem to find the optimal solutions. Optimization based on the principle of parsimony is of practical significance, because one always tries to obtain the optimal solution to the problem based on simplest methods or least resources. Therefore, principle of parsimony is important for modelling and implementing any intelligent system. The concrete application of the principle of parsimony for research and development of MASs is parsimony principle of intelligence in MASs.

Parsimony of intelligence in MASs is to use minimal agents, minimal intelligent properties of these and the most concise system structure to implement a MAS that can meet the requirements of the customer satisfactorily. The minimal agents limit the number of agents within the MAS, because the redundancy of agents within a system usually makes the performance of the system less effective.

Minimal intelligent properties limit the number of intelligent properties of an individual agent, because any intelligent agent can only do well what is most appropriate to him. The agent with a lot of intelligent properties usually brings about system complexity. This parsimony of intelligent properties resulted from the lessons in research and development of expert systems (ESs), in which there is too much promise and too few realizations [287].

The most concise system structure simplifies the system architecture and the relationship between agents with the MAS.

Now, the subsection will propose some heuristic strategies for parsimony of intelligence in the MASs. From a pragmatic viewpoint, the following heuristic strategies for parsimony of intelligence are constructive:

- To analyse the real world problem and evaluate which intelligent behaviours of the human agents in consideration have to be involved in the MAS
- To minimize the intelligent behaviours which must be automated in the MAS
- To minimize the number of agents to as few as possible
- To simplify the system architecture as much as possible (also see [229]).

All these heuristic strategies are based on component-based technology, object-oriented technology in software engineering, also on the art of problem solving in mathematics.

It should be noted that researchers always attempt to get an optimal solution to a real-world problem, while ordinary people like a satisfactory product. Few believe that Microsoft products are the optimal solution to solve the problems in the “office”. However, almost everyone has to believe that they are satisfactory. Knowing the optimal solution is an important step to obtain a satisfactory solution. That is, after knowing the parsimony of intelligence in MASs, one should examine the artificial redundancy of intelligence that is necessary for implementing a satisfactory multiagent system.

9.2.4 Artificial Redundancy of Intelligence

From a graphical viewpoint, the optimal solution and satisfactory solution to the real world problem correspond to the simple path and the basic path respectively, because every basic path is a simple path [282] (p 227), as shown in Fig. 9.2.

The reverse is, however, not valid, because there is redundancy of edge (s) in a simple path, if the sample path is changed into a basic path. This case is similar to the transformation from a

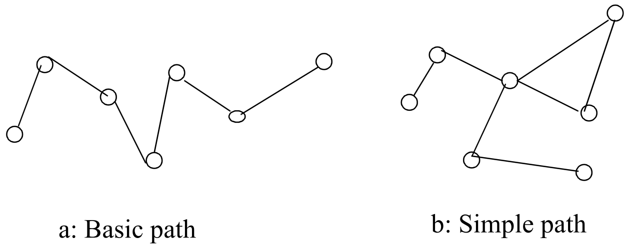


Fig. 9.2 Basic path and simple path

satisfactory solution to an optimal solution, in which some “redundancy” should be removed. On the other side, in order to make a system practically satisfactory, some redundancy is also necessary. For example, in normal situations one does not need an extra tyre when he drives his car. However, one should carry an extra tyre in the car. This extra tyre is an artificial redundancy. In fact, in modern communications systems, artificial redundancy is already added to the encoding of messages in order to reduce errors in message transmission (see http://www.dromo.com/fusionanomaly/claudeshannon.html). Artificial redundancy has also been used to improve the performance of artificial neural networks [206]. Artificial redundancy in MASs should include the necessary redundancy in:

- The number of agents
- The number of intelligent properties
- The system structure.

That is, one should loosen the constraints of minimal agents, minimal intelligent properties, and the most concise system structure and allow a few redundant agents and additional intelligent properties of agents existing in the MAS, according to the practical requirements. Therefore, artificial redundancy is a necessary complementary part for parsimony of intelligence in MAS.

It should be noted that artificial redundancy usually affects intelligent systems negatively, although knowledge redundancy is a necessary premise for development of intelligence in human beings.

9.2.5 Summary

This section examined the principles of parsimony, parsimony of intelligence, and artificial redundancy in MASs, which all are the necessary methods or strategies for implementing any MAS. Parsimony of intelligence in MASs is a pragmatic strategy for avoiding re-occurrence of the lessons of ESs in MASs, while artificial redundancy can satisfactorily meet the requirements of customers. For implementing a MAS, it is necessary to first examine the parsimony of intelligence in the system and then apply artificial redundancy to provide a satisfactory and user-friendly environment for the customers.

9.3 CBR for Multiagent E-Commerce

Chapter 8 examined multiagent e-commerce, in particular multiagent negotiation, which is classified into multiagent auction, multiagent mediation, and multiagent brokerage. This section will examine the application of case-based reasoning (CBR) to these categories. Before going into the details, the section, first of all, takes a closer look at CBR and proposes some new insight into it.

9.3.1 New Insight into Case-based Reasoning

There are many AI technologies such as neural networks, fuzzy logic, knowledge-based technology, data mining, and knowledge discovery available to facilitate multiagent e-commerce systems [294]. CBR is one of them. CBR is a reasoning paradigm based on previous cases or experiences [152]. In other words, CBR can be considered as a kind of experience-based reasoning; that is:

$$\text{CBR} := \text{Experience-based reasoning} \tag{3}$$

Furthermore, similarity-based reasoning is a special form of experience-based reasoning, because "Two cars with similar quality features have similar prices" is an important experience principle in business activities. Therefore, Eq.3 can be specialized as

$$\text{CBR} := \text{Similarity-based reasoning} \tag{4}$$

Similarity-based reasoning can be formalized as the following reasoning model Eq.5:

$$\begin{array}{c} P \to Q \\ \underline{P'} \\ \therefore\ Q' \end{array} \tag{5}$$

where P, P', Q, and Q' represent compound propositions, Q and Q' are similar in the sense of similarity. This is a theoretical foundation for CBR, in particular for case retrieval, case building, and case adaptation [291].

There has been an important influence of knowledge base systems (KBS) on CBR systems in most CBR literature [1][151][315]. For example, the case base in the CBR system can be considered as a variant of the knowledge base in KBSs. Experience plays an important role in CBR systems, just as knowledge does in KBSs. Based on this idea, the section proposes an knowledge-based model for CBR systems, as shown in Fig. 9.3.

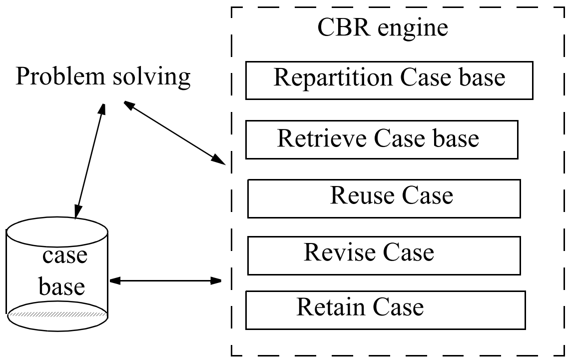


Fig. 9.3 Knowledge-based model of CBR

In this model, similar to the inference engine in expert systems [115], a CBR engine in CBR systems is the reasoning mechanism for performing similarity-based reasoning. However, we ignore working memories including the user interface in the figure. In fact, it is important that in the user interface, the user should know what the problems are, etc. The major part of the CBR system is the case base and the CBR engine. In terms of the CBR system, the case base consists of the past encountered problems and corresponding solutions. The CBR engine consists of all the processes that build and manipulate the case base to derive information requested by the user based CBR. At least the CBR engine consists of the following processes [88][316], as shown in Fig. 9.3:

- Repartition the case base, as required
- Retrieve the most similar cases in the case base
- Reuse the retrieved cases to attempt to give the solution to the problem(s)
- Revise the retrieved cases, and
- Retain the new case as a part of a new case base.

A new problem is matched against the encountered problems in cases stored in the case base after the case base was built. One or more similar cases, in each of them there is similar problem to the new problem, are retrieved from the case base [91]. A solution suggested by these cases is reused and tested for success. If necessary, the retrieved case(s) will probably be revised to produce a new case which can then be retained in the case base. If a new case is saved in the case base, the case base will be repartitioned in some situation [88].

For example, suppose we have a case base of second-hand cars C. Suppose also that every car (case) has exactly seven attributes and the problem description attributes are the first six, i.e. $P=$ {*year, power, mileage, make, model, body shape*} and the solution description attribute is the last one, i.e. $Q=$ {*price*} and the range of the year, power, mileage, and price are numerical and the range of other attributes can be considered as qualitative, such as “bad”, “good”, and “excellent”. The global similarity can be defined by aggregation of local similarities S_i, $i = \{1, 2, \ldots, 6\}$ for each problem description attributes. Now Peter wishes to buy a second hand car with problem descriptions (car features) $P_0 = \{p_1, \ldots, p_6\}$ using the mentioned CBR system provided by the dealer. He will retrieve the case base of second-hand cars C and measures which car(s) have the similar features to his requirements. For example, there are three cars which are retrieved, and are similar to what he requires. Then he considers the price of them, and chooses the retrieved car with the lowest price. He perhaps asks the seller to revise the service of the car if he buys, for example, the seller should provide 1 year’s guarantee instead of six months guarantee. After the deal is done, the dealer will retain this new selling case to the case base of second-hand cars according to certain rules for inserting a new case into the case base [88].

9.3.2 CBR for Multiagent Negotiation

Negotiation plays an essential part of most B2B transactions. Responsibility for much of negotiation will be delegated to software agents [78]. These agents will monitor other trade agents continuously, watching for potential opportunities. They will be able to enter into negotiation with many potential trade partners at once, reaching an acceptable deal and setting up a contract in a matter of milliseconds.

Three main approaches for negotiation strategies are currently being explored [78]: the rule-based approach, the adaptive behaviour approach, and the game theoretic approach. The first two approaches can be implemented by CBR techniques from a knowledge-based viewpoint, because CBR can be considered as a generalized form of rule-based approach. CBR can also be considered as a kind of the adaptive behaviour approach, because case adaptation is one important part in the CBR process.

In what follows, the subsection will examine the integration of CBR and multiagent systems for e-negotiation.

As discussed in Chapter 8, automated negotiation plays an important role in e-commerce. Matos and Sierra propose a case-based agent architecture to model an agent negotiation strategy in MAS [199]. At each step of the negotiation process the architecture determines the weighted combination of tactics to be employed and the parameter values related to these tactics. Matos and Sierra believe that when the agent is provided with a case-based architecture, it uses previous knowledge and information of the environment state to change its negotiation behaviour.

Similar to Mates and Sierra, Zhang and Wong use CBR techniques to acquire negotiation strategies from previous negotiation experiences [352]. They develop a case-based negotiation (CBN) agent for used car trading as a test-bed. The CBN agent can perform either as a car buyer agent or car seller agent. The agents revise and adapt negotiation strategies to propose an appropriate negotiation strategy in each decision-making episode of the negotiation process. The negotiation strategies are based upon the knowledge, past experience, and information available to the negotiating agents. Therefore, the negotiation in CBN involves three main actions:

1. Evaluating the offer from other agents
2. Defining an episode strategy by CBR, and
3. Generating a counteroffer based on proposed negotiation strategy.

The crucial component in the CBN is the process of defining an appropriate strategy using CBR techniques, which can be viewed as a process of proposing a concession by reusing previous negotiation experiences in the CBN framework. The CBR process for defining a concession, as shown in Fig. 9.4, is composed of:

- Retrieving relevant previous negotiation experience, which is viewed as a case

- Selecting a most matched case based on similarity or retrieved cases and input negotiation context, and
- Adapting the negotiation strategy information in the selected case to propose a concession.

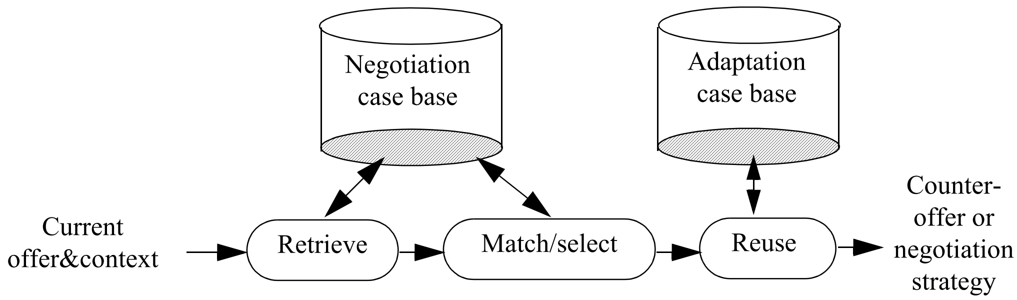


Fig. 9.4 The CBR process of defining negotiation strategy in CBN after [352]

where the negotiation case base consists of a number of previous negotiation cases, each of which provides detailed negotiation context and decisions related to a specific agent (e.g. seller agent or buyer agent) in the previous negotiations.

It should be noted that once a number of similar cases are retrieved, the CBN needs to select the most relevant case for the current negotiation episode [352]. Similarity measurement is a crucial issue for the reuse/adaptation of previous negotiation experiences in the CBN. Rule-based reasoning is applied in the case matching and selection.

Furthermore, one of the most important tasks for the CBR agent during the negotiation process is the iterative adaptation of user demands and the iterative adaptation of products. The former is realized by making proposals for adding or changing these demands, while the latter is done by product configuration with the goal of finding an agreement point in the multidimensional demand/product space [335]. It should be noted that the mentioned user demands correspond to the *problem descriptions*, while the products here correspond to the *product descriptions* in the *sales case* (see Section 6.3). Therefore the task of the sales agent during the CBR based negotiation is, in essence, the iterative adaptation of sales cases and adaptation of demands and/or the product. During the CBR based negotiation, the customer or the sales agent are allowed to modify the customer demands only. If products in the product case base are configurable, it might also be possible to modify the products themselves and the demands during negotiation.

However, the above-mentioned framework is essentially a single CBR agent system, although in the context they use either CBN agents or negotiating agents (e.g. seller and buyer), this is also valid for the case-based negotiating agents in [199]. Further, Zhang and Wong [352] follow a traditional idea that a human agent corresponds to a software agent, which is sometimes invalid in multiagent systems. In other words, it is more pragmatic for a human agent to correspond to a few software agents, because the agents in the MAS constitute a community, in which different agents should play different roles. Therefore, it is significant to examine MASs with different kinds of agents which will be discussed in Section 9.4.

9.3.3 CBR for Multiagent Mediation

Sycara developed a framework for intelligent computer-supported conflict resolution through negotiation/mediation[1] with CBR. The model integrates CBR and decision theoretic techniques to provide enhanced conflict resolution and negotiation support in group problem solving settings. This model has been implemented in the PERSUADER, a multiagent system operates in the domain of labour management disputes. The PERSUADER, acting as a mediator, facilitates the disputants' problem solving so that a mutually agreed upon settlement can be achieved. The PERSUADER evaluates and generates potential agreements and then proposes them to involved negotiation agents. The PERSUADER embodies a general negotiation model that handles multi-agent, multi-issue, single or repeated encounters based on an integration of CBR and multi-attribute utility theory.

9.3.4 CBR for Multiagent Auction

As discussed in the previous chapter, auctions are an attractive domain of interest for AI researchers, because many e-auctions in the Internet such as AuctionBot, Onsale, eBay, and many others have established auctioning as one of mainstream forms of e-commerce. Thus, multiagent auctions appear as a convenient mechanism for automated trading and automated negotiation, because of the simplicity of their conventions for interaction when multi-party negotiations are involved. Further, e-auctions may successfully reduce storage, delivery or clearing house costs in many markets [198]. This popularity has spawned AI research and development in auction

1. See http://www-2.cs.cmu.edu/afs/cs/user/katia/www/persuader.html

servers as well as in trading agents. Moreover, auctions are not only employed in Web-based trading, but also as one of the most prevalent coordination mechanisms for agent-mediated resource allocation problems.

In the work of Martín and Plaza [198], the bidding mechanism for auctions is based on downward-bidding protocol (DBP). In these auctions, the auctioneer opens a new bidding round to auction a good among a group of agents, he starts quoting offers downward from the chosen good's starting price. For each price bid, three situations might arise during the open round:

1. Several buyers submit their bids at the current price. In this case, a collision comes about, the good is not sold to any buyer, and the auctioneer restarts the round at a higher price
2. Only one buyer submits a bid at the current price. The good is sold to this buyer whenever his credit can support his bid. Whenever there is an unsupported bid the round is restarted by the auctioneer at a higher price, the unsuccessful bidder is punished with a fine, and he is expelled out from the auction room; and
3. No buyer submits a bid at the current price. If the reserve price has not been reached yet, the auctioneer quotes a new price which is obtained by decreasing the current price according to the price step.

If the reserve price is reached, the auctioneer declares the good withdrawn and closes the round.

CoDit is a multiagent system consisting of a group of agents that perform CBR and are able to communicate and cooperate for therapy recommendation in diabetic patients [198]. Each agent has a case base with data of the patients. The basic task of the diabetes therapy CBR agents is *retrieve* and *reuse. Retrieve* is a task that identify, search, and select from the case base those cases that satisfy the model built by the perspective, while *reuse* is a task that takes the most preferred case and adapts its solution (therapy) to the symptom of the current patient.

It should be noted that the above-mentioned framework is essentially a multiagent system consisting of CBR agents; that is, no other kind of agents exist in the multiagent system. This is at least not satisfactory in an agent community with different strengths and weaknesses. Therefore, it is of practical significance to examine MASs with different kinds of agents.

9.4 CMB: Integration of CBR and Multiagent Brokerage

Chapter 8 proposed a framework for broker-centred multiagent bargaining processes. The key idea behind it differs from research in other frameworks, for example [352] and [232], in that the proposed framework stresses that:

- The work of a human broker should be done cooperatively by a few intelligent agents in a MAS
- The broker is the agent of both the seller agent and buyer agent.

However, there are some drawbacks in that framework. For example, one drawback is that we have not gone into which special intelligent features that the agents within the system should possess. Another drawback is that we have also not paid attention to the principle of intelligence parsimony in multiagent systems (see Section 9.2). This section will resolve these drawbacks by providing a new framework of CMB (CBR system for multiagent brokerage), which is a multiagent system integrating CBR and e-brokerage. Therefore, the investigation here is the further development of applying CBR to multiagent negotiation in Chapter 7 and multiagent brokerage in Chapter 8.

The goal of CMB is to automate brokerage in e-commerce through integrating CBR and MAS to assist brokers to work in e-brokerage and bargaining processes. Taking into account the above discussion, the key ideas of CMB are that:

- Only some agents in CMB (e.g.negotiation agent, seller agent, etc.) have CBR ability
- If an agent has CBR ability, then its basic architecture consists of its own case base and CBR engine
- The system performance of CMB depends on not only the intelligence of its individual agent but also the cooperation, coordination, communication, and negotiation with other agents [85][84].

Therefore, the feature of CBR in CMB lies in that some of the agents in CMB have CBR ability. This is different from CHROMA [232] and CBN [352], in which every agent has CBR ability. The reason is that CMB follows the principle of intelligence parsimony; that is, the requirement of certain intelligent ability for an agent depends on what it "really requires" when it does a certain work on behalf of a human agent. In what follows, the section goes into the details of the framework for CMB.

The CMB is going to implement three different perspectives: the one of buyers, the other of sellers as well as the one of the broker. From a viewpoint of the buyer agent, CMB is his intelligent agent and must act as a decision support system (DSS), helping and advising him where and how to buy the items he is looking for [313]. In such a way, the buyer agent must be able to find several sellers for each item taking into account of his buyer's preferences and has the ability to assist his buyer with deciding which goods best fit his buyer requirements. Learning is an important feature of this agent since it must remember past experience and buyers' preference. CBR is a kind of experience-based reasoning, learning and reasoning from experience, which is concreted with similarity-based reasoning. Therefore, the realization of such a buyer agent should be CBR-enabled.

From the viewpoint of the seller agent, CMB is also his agent and is able to build customer profiles and to segment its customer group to optimise the process of acquiring customers [323]. The seller agent could search for the prices, payments, and available products of the seller's competitors. The experience of selling goods and attracting customers are also very important for seller agents so that they should be CBR-enabled.

In fact, in physical commerce, the buyer or seller are able to learn by their own business experiences, which companies are reliable, where to look for personalized goods or services and how to handle the negotiation process [313]. This also means that it is necessary for both buyer agents and seller agents to be CBR-based.

From the viewpoint of the broker, CMB should be his own agent. However, the broker agent has a lot of tasks involving attracting buyer agents, seller agents, segmenting customer groups, and searching for the information about requirements from the buyer agents and the suppliers from the seller agents. In particular, one of main tasks of the broker agent in CMB is to perform bargaining and compromise, so that the broker agent should be able to know where and when it should perform the compromise in bargaining or brokering. Further, the experience of bargaining and compromise is most important for a successful broker. Therefore, the tasks of a human broker should be decomposed and then do by some intelligent agents within the CMB.

Based on the above discussion, CMB is a broker-centred MAS, as shown in Fig. 9.5, consisting of three multiagent subsystems [85][84]: intelligent buyer agent subsystem, intelligent seller agent subsystem, and intelligent broker. While the intelligent seller agent subsystem consists of all available intelligent seller agents on-line, the intelligent buyer agent subsystem comprises all available buyer agents[1] on-line. The intelligent broker is also a MAS.

In CMB, the customer can create his own agents: either buyer agents or seller agents. Furthermore, the buyer agents, seller agents, and the intelligent broker are all proactive. The seller agent tries to cooperate and negotiate with the intelligent broker to find a most satisfactory deal to sell the goods of his seller. The buyer agent also tries to cooperate and negotiate with the intelligent broker to get what his buyer needs at lowest price. The intelligent broker proactively searches all available information about the request and supply of goods on the e-market [84].

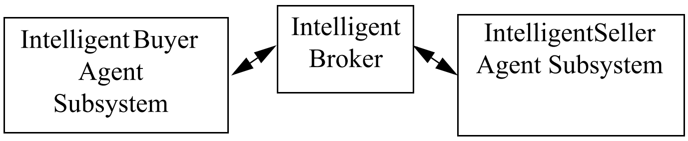


Fig. 9.5 Architecture of CMB

Every intelligent buyer and seller agent in CMB is semi-autonomous, in that, once entering into the electronic bargaining process, the intelligent buyer (seller) agent negotiates and makes decisions on his own, without requiring his buyer's (seller's) intervention.

As mentioned previously, business experience plays an important role in commercial activity. Therefore, buyer agents and seller agents should have CBR ability to make decisions during the bargaining with the intelligent broker. Usually, they retrieve or revise the related business information and adapt the bargaining strategies in order to get most profits. For example, Peter asks his buyer agent to buy a second-hand car using CMB. The buyer agent will use the CBR subsystem provided by CMB for each agent to retrieve the information of second-hand cars from the case base of second-hand cars based on its past experience, according to the requirements of Peter. Then he will use CBR to recommend a possible solution, i.e. a second-hand car to Peter. If Peter is not satisfied with the recommended car completely, the buyer agent has to bargain with the intelligent broker for revising the attributes of the car such as revising the post-sale services. If

1. For convenience, x agent stands for intelligent x agent, where x is i.e. buyer or seller, etc.

Peter accepts the recommended car, then the buyer agent will retain the new case to the case base and the intelligent broker also save the new case into his own case base of second-hand cars, because this is a successful case.

As shown in Fig. 9.6, the intelligent broker comprises a buyer (seller) collaborative agent, an information-gathering agent, an interface agent, a managing agent, a matchmaker agent, a transaction agent, a negotiation agent, a buyer (request) database (BDB), and a seller (supply) database (SDB). The key insight behind is that the work of the human broker should be modelled by the activities of a few agents within a MAS in a cooperative way. The business experience can be treated with CBR. In what follows, only the mentioned agents will be discussed in detail (also see [85][84] for detail):

- The buyer (seller) collaborative agent is a software agent that proactively cooperates with the buyer (seller) agents to get the request/supply information and special information about the buyers and sellers, and then save it into BDB or SDB respectively, if necessary. In certain cases, the buyer (seller) collaborative agent decides if a buyer agent and a seller agent can directly contact each other in order to reach a deal
- The information-gathering agent [85] is a mobile agent that proactively roams around the Internet. It interacts and collaborates with the main search engines in the Internet such as Excite and Yahoo in order to search and analyse the required market information indirectly

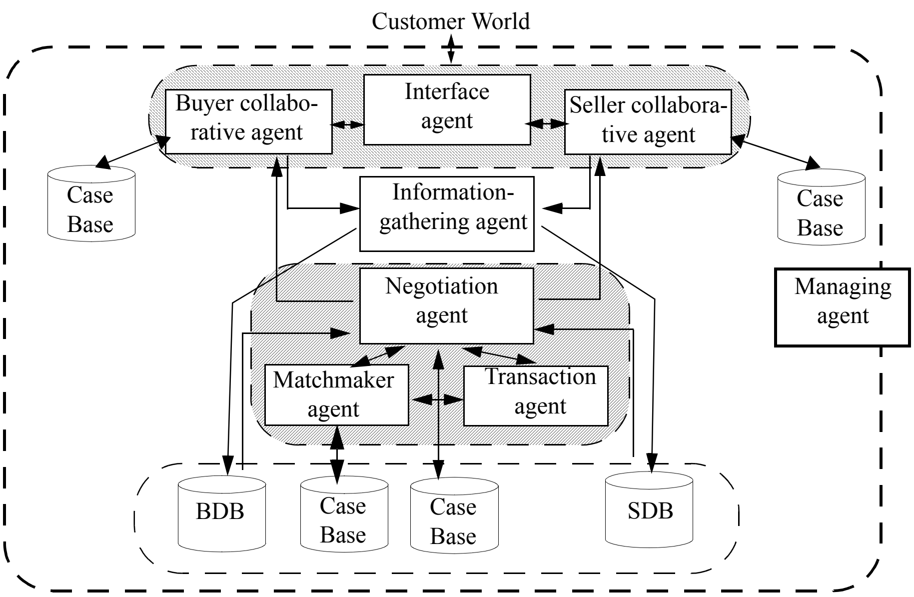


Fig. 9.6 Intelligent broker (Subsystem) based on [85]

from individual Web sites [186][207], and then puts it in the corresponding data or knowledge bases. For example, the information-gathering agent uses search engines to search all available information of second-hand cars and then save it to SDB

- The matchmaker agent searches the database and matches a request of goods from a buyer agent and a supply of those goods from a seller agent using appropriate matching algorithms. Because similarity-based matching is a basic technique for information matching and it is also a kind of experience-based reasoning, the matchmaker should have the CBR ability during performing the matching, the CBR ability here is basically limited to similarity-based retrieval or similarity-based matching. It also matches the goods-requesting buyer agent and the goods-supplying seller agent and then sends the matched information to the interface agent or buyer (seller) collaborative agent after transaction analysis by the transaction agent. For example, the matchmaker agent retrieves the case base of second-hand cars and tries to know which seller agent's car matches the requirements of Peter's buyer agent
- The negotiation agent is an autonomous and proactive agent that performs not only integrative but also distributive negotiation strategies (see [105]) during negotiation with the buyer agent and seller agent. Because business negotiation is complex and difficult in some cases, the intelligence of the negotiation agent lies in that he can change his negotiation strategies according to the changing available (information) resources. Thus, the negotiation agent should be adaptive and may use a range of available AI technologies. Adaptation is usually based on experience. Further negotiation experience plays a pivotal role for the negotiation agent during the bargaining. Therefore, the negotiation agent should have CBR ability during the negotiation. For example, the negotiation agent uses not only the mentioned case base of second-hand car transactions, but also the case base for the preference of buyer (agent) and seller (agent) when bargaining with the buyer agent and seller agent, because similar preferences of the customers usually lead to similar solutions
- The managing agent is responsible for the management of the running agents and the e-brokerage place
- The transaction analysis agent analyses and computes every transaction of deals, which the negotiation agent and the matchmaker agent suggest, and then submits the recommendations to the negotiation agent and the matchmaker agent or the managing agent, if necessary

- The hybrid inference engine, containing a massive "reasoning kit", provides the agents, in particular, negotiation agent with the appropriate hybrid mechanism, so as to that the negotiation agent can easily perform a certain reasoning that he thinks necessary in the necessary case [176]. For example, the "reasoning kit" has the reasoning mechanisms for performing CBR, deductive reasoning, abductive reasoning, and inductive reasoning as well as fuzzy reasoning
- Buyer RDB (seller SDB) stores the request (supply) information about goods provided by the buyer (seller) collaborative agent through buyer (seller) agent in a certain form
- Hybrid KB consists of knowledge extracted from buyer RDB and seller SDB. It is also comprised of the (rule-based, model-based, case-based, fuzzy, trick) knowledge about the market and negotiation. Hybrid KB and hybrid inference engines cooperate to facilitate all of the agents in the intelligent broker subsystem to share the information in the data warehouse and make decision in bargaining process.

It should be noted that autonomy and mobility of agents are the most important features different from other stationary intelligent systems. There are many studies of these [125]. Autonomous and mobile agents are well suited for e-commerce [162]. A commercial transaction may require real-time access to remote resources, such as stock quotes and perhaps even agent-to-agent negotiation. Different agents have different goals and implement and exercise different strategies to accomplish them. We envision agents embodying the intentions of their creators, acting and negotiating on their behalf. Autonomous and mobile agent technology is a very appealing solution for this kind of problem, because mobile agents can roam through the Internet and perform specific delegated tasks at a designated server site in the Internet.

Telescript was the first commercial platform for mobile agent systems and also lead to many of the recent systems [307], for example, many "spies", one kind of brothers of mobile agents, are busy in the Internet on behalf of a certain "human agency" [313]. The mobility of agents is motivated by the desire of utilizing efficiently geographically distributed services [307]. Agents are dispatched to the remote server site where they perform the necessary interactions locally. After completing the task at one site they return with the results or continue their journey to visit other sites as well, as required. Furthermore, in contrast to other forms of mobile code, i.e. applets, mobile agents are more flexible and autonomous in deciding where and when to move.

9.5 Analysis and Design of CMB

The previous chapters and sections proposed many models, architectures, and frameworks around the CMB from either a knowledge-based viewpoint, or a logical viewpoint or an e-commerce viewpoint. All those proposed models, architectures, and frameworks are independent of any existing programming languages and software development tools or platforms. However, in order to make the proposed architecture of CMB into a software system, this section will investigate the analysis and design of CMB, which is the important premise for implementing CMB.

9.5.1 Related Work

In order to provide the analysis and design of CMB, it is necessary to review some research studies which are related to CBR systems, MASs or CBR-based multiagent systems, although we have not found any system which is very similar to what proposed above; that is, it is an attempt to integrate CBR and multiagent systems in electronic brokerage.

CBN is a Web-based system based on CBN framework (Section 9.3.2), which was designed as a distributed three-tier server client architecture: negotiation client, negotiation server, and negotiation case repository [352]. The negotiator client provides the graphical user interface (GUI) that is implemented using the Swing API. The negotiator server is the reasoning engine which offers negotiation evaluation, negotiation case retrieval, negotiation case matching and reuse as well as counter-offer generation. For example, it applies various similarity assessments to find the best matched negotiation case from a pool of relevant negotiation cases retrieved from the negotiation case base and adapts best-matched negotiation case to obtain episodic strategy to generate counter-offer. CBN was developed using Java 1.2 and a standard SQL capable relational database management system.

CASBA is being developed as an e-marketplace to improve the quality of existing e-commerce services, by introducing a higher flexibility and automating trading processes [312]. This is achieved through applying intelligent agent technology to enable the market framework to offer automated auctions and negotiation among the agents. CASBA has an open architecture utilising technology like CORBA, Java, and Javascript. Interfaces to e-commerce components such as payment servers allow the CASBA server to be integrated into existing e-commerce solutions. The e-mail server is used to inform subscribed users about new products or auctions on the market. However, CBR does not play any role in CASBA.

As discussed in Section 4.7, coordination, cooperation, and communication are main features of MASs. Thus, the efficient coordination of agents is very important in multiagent e-commerce systems. Contract Net Protocol[1] has been the most commonly used for coordinating agents in negotiation [170]. This communication protocol concerns how contract managers announce the tasks to other agents, how potential contractors return bids to the manager, and how the manager then awards the contract. This protocol can be of significance in some cases.

9.5.2 Interface Agent of CMB

The Interface agent, a front-end of the e-brokerage, consists of a user-friendly platform located on the server. The interface agent helps the customers to build either a buyer agent or a seller agent or a broker agent, as required, in order to realize the real brokerage in the e-brokerage environment. The combination of MASs and WWW technologies provides the means to develop high level interfaces, which can support process interaction among Internet users and remote MAS application [313]. The current WWW tools and techniques give strong support to reduce the efforts of developing and using interface agent and to create better communication channels among the agents during the e-brokerage.

Therefore, the interface agent in CMB should possess the ability of information navigation to help the user or seller or buyer agent to concrete the domain to which information belong. For example, in electronic marketplaces, Market Maker, users can create their own agents with the intelligent interface agents and delegate business tasks such as buying, selling, searching items to them. These agents are able to negotiate with each other in order to perform delegated tasks with highest profit [58].

In addition, the e-mail server and email clients are also the necessary component in the interface agent and every buyer agent and seller agent, which can be used to inform the agents about the new buyer agents and seller agents and the brokerage information or notification to the

1. See Smith, R.: The contract net protocol: higher-level communication and control in a distributed problem solver, *IEEE Transactions on Computer* 29 (1980) 1104-113.

agents in the e-brokerage place. The e-mail clients are also a necessary means for communication between buyer agents and seller agents, among buyer agents or seller agents.

9.5.3 Development of CMB with Design Patterns

Design patterns are often partitioned into three kinds: conceptual patterns, design patterns, and programming patterns [103][247]: A conceptual pattern is described by terms and concepts from a particular application domain. A design pattern is described in terms of software constructs, such as objects, classes, inheritance, and operators. It elaborates a conceptual pattern by specifying a software implementation. Programming pattern is expressed in programming language elements. Programming patterns commit a design pattern to a language specific implementation. Macros are typically used to implement programming patterns.

Design patterns for coordination, coordination patterns, are a recently emerging concept [111]. An appropriate coordination pattern must be selected to satisfy the interactive behaviours required of the system. Hayden et al. propose a number of coordination patterns, grouped into four basic architectural styles. Hierarchical, federated, peer-to-peer, and agent-pair patterns. In hierarchical patterns, top-down control is imposed by agents in a supervisory or managerial role. In federated patterns, an umbrella system provides overall coordination that the agents submit to. In peer-to-peer patterns, individual agents are responsible for managing coordination and potential conflicts with others. Finally, agent-pair patterns describe one-to-one interaction.

For example, the following is a sample pattern:

- **Pattern Name**: pnBroker
- **Intention**: The broker bargains with seller agent and buyer agent respectively in order to perform a brokerage
- **Motivation**: Instead of searching ability, the main task of a broker is bargaining and compromise
- **Applicability**: If many buyer agents and seller agents exist in an e-brokerage place, this pattern is applicable.
- **Structure**: See Fig. 9.6
- **Participants**: An intelligent broker, many seller agents, and many buyer agents
- **Collaborations**: Limited

- **Consequences**: The deal is done either successfully or unsuccessfully
- **Implementation**: N/A
- **Known Uses**: The Object Management Groups's (OMG) Object Request Broker (ORB) and remote procedure call (RPC), which provides location transparency is an example of a broker architecture, although they are not agent systems
- **Related patterns**: pnBuyerAgent, pnSellerAgent.

9.5.4 Applying Game Technology to CMB

Some work [175][160][356][357] in computer games, for example, bridge on-line[1], provides functionality that meets some requirements of CMB to some extent. Like brokerage, bidding in bridge is a process of compromising and bargaining, which also includes tricks. Therefore, CMB can borrow the architecture given in [175], shown as in Fig. 9.7.

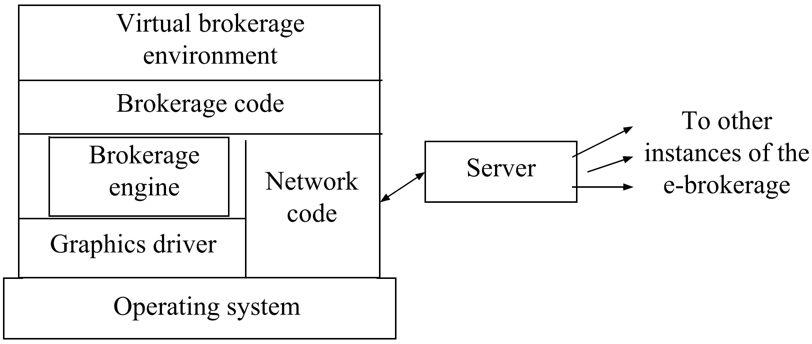


Fig. 9.7 The alternative architecture of CMB based on [175]

At the top level is the virtual brokerage environment with which the intelligent buyer agent and seller agent as well as broker agent interact. They come in a wide range of appearances and rules of interaction. They often use a blackboard to interact following the rules of interaction. For example, the broker agent can not tell either the buyer agent or seller agent his real price during the bargaining.

The level below is the brokerage code, which handles most of the basic mechanisms of brokerage itself such as display parameters, networking, and the base-or atomic-level actions for agents' behaviours. The brokerage or bargaining algorithms are also realized in the internal part at this level.

The brokerage engine (based on game engines in [175]), a generic architecture for e-brokering that integrates multiagent systems and CBR in e-commerce, provides a platform for rendering

1. See http://au.games.yahoo.com

multiple views and coordinating real and simulated scenes as well as supporting multiagent interaction.

The networking code allows several users/agents in remote locations to explore and interact in the same virtual brokerage environment and the graphics drivers translate generic requests from the brokerage engine to the underlying graphics library [175].

One computer acts as the server for the virtual brokerage environment, while the others support the individual buyer agents or seller agents. The server maintains information on whichever virtual brokerage environment it is supporting at the time. It communicates with the buyer agents or seller agents to maintain global information about shared brokerage environments, agent's interaction, and synchronization information.

There are editors available for users to create their own view in the virtual brokerage environment (VBE)[160]; that is, each buyer or seller agent has a different (his own) window into their shared e-brokerage environment. In the VBE, each agent has its personalized head icon.

Finally, during the e-brokerage, each agent can use the *bargaining board to* bargain brokerage items, for example, price. The bargaining board will be automatically displayed in the window available to the agent, when the CMB asks this agent to bargain. The agent (e.g. seller agent) will fill in the acceptable compromise space/subspaces (for example price interval), which depends on the compromise space. The latter is given by the intelligent broker agent and has been displayed in the bargaining board in different colours, which can be read by the (human) seller agent. This is also valid for the bargaining of the buyer agent.

9.6 Implementing CMB

This section describes the implementation of CMB prototype systems for simulating activity of an intelligent broker that bargains with buyer agents and seller agents in the real estate scenarios.

Visual Basic (VB) programming language has been used to implement a prototype of the CBR system at present. Its user interface consists of the following commands: Repartition case base, Retrieve case, Reuse case, Revise case, Retain case, as shown in Fig. 9.8, to implement the knowledge-based model of CBR in Section 9.3.1. The Retrieve case is realized with the following subsystem, with which the user provides his/her requirements, as shown in Fig. 9.9. The Repartition case base, Reuse case, Revise case, Retain case are mainly realized by the subsystem, as shown in Fig. 9.10. The OLE DB (Object Linking and Embedding Database) and ADO

(ActiveX Data Objects) [351] (pp 672) are used to access the case base, perform similarity-based

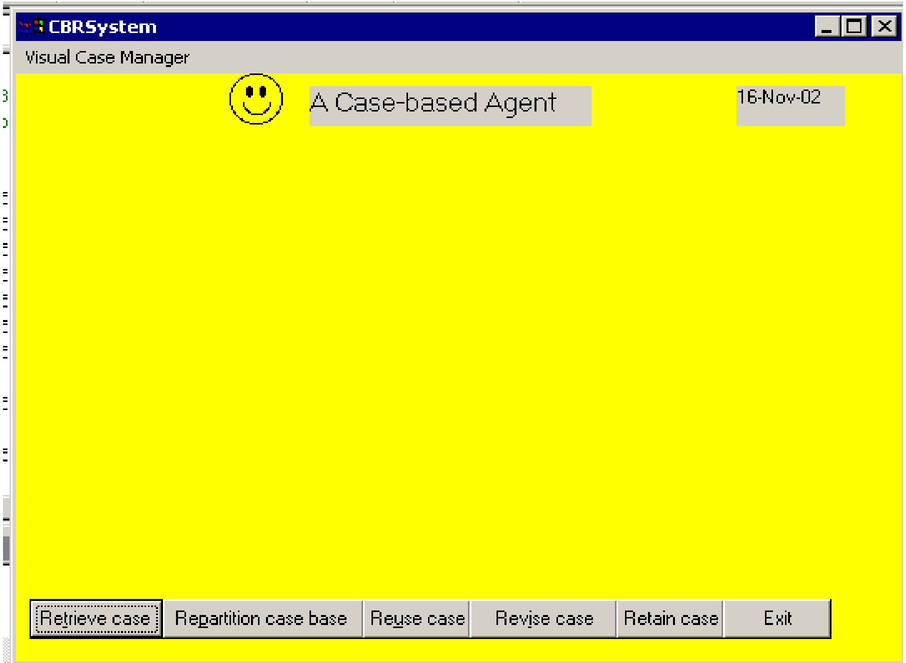


Fig. 9.8 A prototype of a CBR system

retrieval and get the retrieved case, as shown in Fig. 9.11.

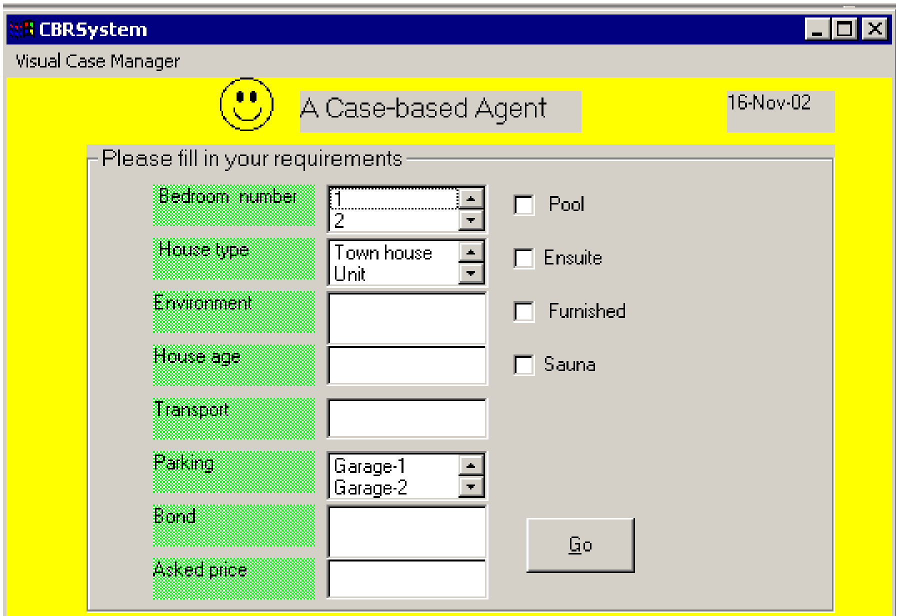


Fig. 9.9 Retrieving case

Furthermore, Visual J++ [83] will be used to build CMB, a CBR system for multiagent brokerage as a prototype of a distributed CBR-based multiagent system. In CMB, the customer can decide on his own agent as either a seller agent or buyer agent or broker, which is the same as that in Market Maker at MIT. Then both the buyer agent and seller agent are a micro-CBR system, its architecture similar to that just mentioned. The architecture of the micro-CBR system will also be used for the negotiation agent and information gathering agent, because they are CBR-enabled.

There are many intelligent properties of agents available in CMB. After selecting his agent, the customer can select further properties of the agent he likes, which will be given in a property

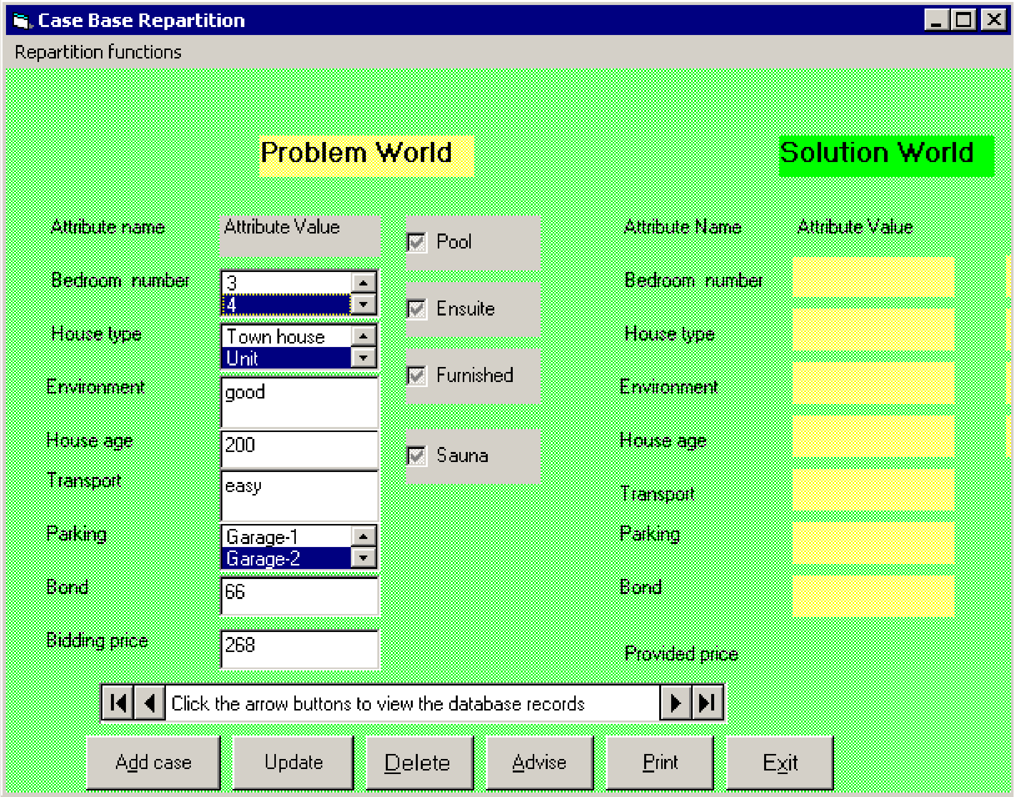


Fig. 9.10 A case administrator

list, which is similar to the property list in current programming environments such as Java development environment. In this way, the customer becomes the real centre from the user interface. However, for the intelligent broker, we should predefine the intelligent properties which the intelligent broker should have.

It should be noted that the CMB architecture is a work in process. As of this writing, only the CMB engine is in its form. The CBR agent for players in multiagent brokerage is completely developed. Other parts for mobility and autonomy of agents with CMB are still under development. We are also aware that more work needs to be done, especially regarding compromise logic and bargaining strategies under uncertainty and trick.

9.7 Concluding Remarks

This chapter first examined the principle of intelligence parsimony and artificial redundancy of intelligence in MASs. Then it proposed a framework of CMB, a CBR system for multiagent brokerage, which integrates CBR, MAS, and brokerage. It also briefly introduced some new insight into CBR. The key insight is that an efficient way for applying CBR in e-commerce is through intelligent agents or multiagent systems. Based on the parsimony of intelligence and artificial redundancy in MASs, this chapter concluded that any practical MAS should satisfy:

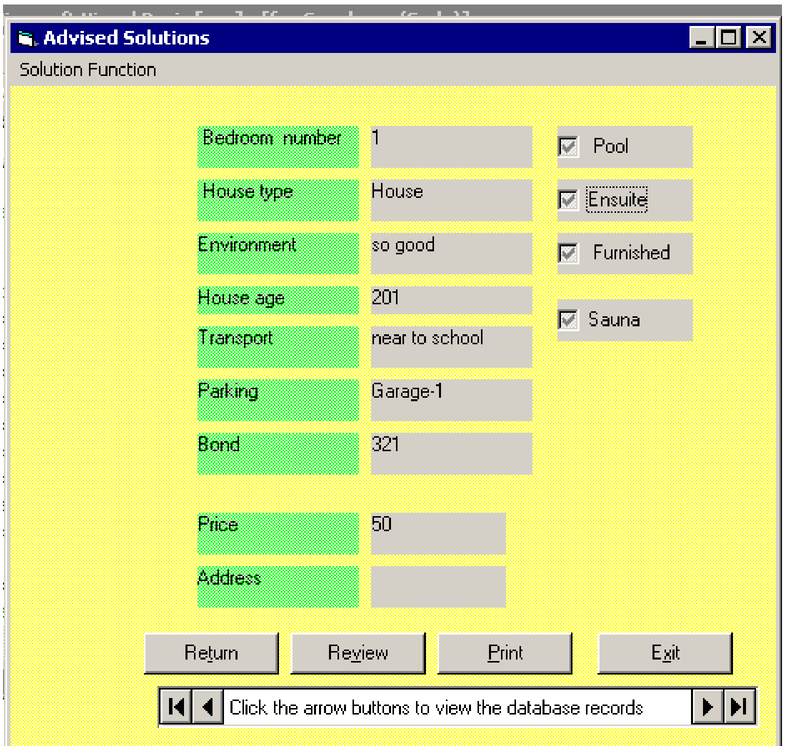


Fig. 9.11 A advised solution

- Only some necessary agents in the MAS have CBR ability
- Only some necessary agents are mobile
- Only some necessary agents are autonomous.

It should be noted that e-commerce applications are just about to leave their infancy According to Henry Lieberman and Sybil Shearin[1], most e-commerce sites today are little more than electronic catalogs of product offerings. Consumer input is limited to requirements questionnaires, search engines, and accepting or rejecting particular offerings. But in complex purchases, such as real estate, cars, and computers, it is often difficult to specify exactly what you want, and priorities and preferences often change in the process of exploration.

Furthermore, the potential of the Internet for truly transforming commerce is largely unrealized to date. A human buyer is still responsible for collecting and interpreting information on merchants and products, making decisions on merchants and products and finally entering purchase and payment information. However, intelligent agent technology can be used to

1. see URL:http://www.media.mit.edu/research/sig.php?type=sig&id=7

automate some of the most time-consuming stages of the business process, for example, information brokering and product brokering. Therefore, using intelligent techniques for e-commerce has become an interesting topic for research and development of intelligent e-commerce [66][67]. Integration of CBR and multiagent systems in e-commerce is just such an attempt. However, how to model experience in e-commerce in particular, in mediation, negotiation, and brokerage is a big issue in this direction. Therefore, in future work we will further study real world scenarios in the multiagent brokerage field, in particular to incorporate the real brokerage firm to apply the proposed integrated models of CBR and MASs to improve the on-line brokerage.

The proposed design and analysis lead to a lot of new issues in order to implement the architecture of CMB, for the effectiveness, efficiency, and security of the system architecture. As future work, we will implement a prototype e-brokerage environment based on the architecture of CMB, to evaluate the chosen software tools and technology and as a means to look for the human brokerage partners who have interest in transferring the traditional brokerage into e-brokerage using intelligent agents and CBR technology. We will also explore some implementation details such as the communication, bargaining, and compromise between buyer agents and seller agents during the brokerage.

10 Concluding Remarks

This chapter is the final of chapter of the thesis, which is also outside the Boolean structure of the thesis, as shown in the shaded area of Fig. 10.1.

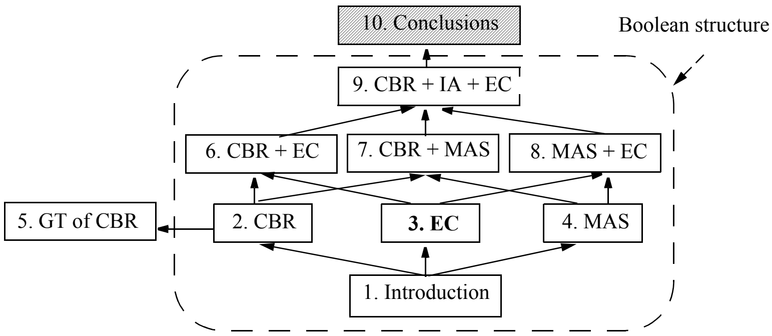


Fig. 10.1 Chapter 10 is outside the Boolean structure of PhD-thesis

The revolution of the Internet and the WWW has changed traditional commercial activities such as shopping, negotiation, and brokerage. Applying intelligent techniques, most of them from artificial intelligence (AI), in e-commerce has drawn increasing attention in both academic fields and general society, because the intense competition among Internet-based business to acquire customers and retain the existing ones has made intelligent techniques an indispensable part of e-commerce [294]. Case-based reasoning (CBR) systems, multiagent systems (MASs), and expert systems (ESs) can be considered as some of intelligent techniques for e-commerce. This thesis first examined CBR, intelligent agents, MASs, and e-commerce, then it investigated the interrelationships between them from a logical viewpoint, a knowledge-based viewpoint, and a business viewpoint. Finally it integrated them under one roof, and proposed the architecture of CMB that is a system of integrating CBR and MAS in e-commerce, in particular in e-brokerage, which is a special case of e-negotiation. In other words, the preceding investigation considered CBR in e-commerce in a number of different contexts. In the most specific view, it considered a systematic approach to theoretical case-based reasoning. More generally, it offered an approach to case-based reasoning and a view of CBR, which is considered as an intelligent technique in Web Intelligence. In the most general view, it considered CBR in the context of AI problem solving techniques. The following sections will consider the originalities and contributions offered by this thesis in each of these contexts, reviewing in the process the central themes of the thesis. More specifically, this thesis has made some contributions to CBR, knowledge management and experience management, AI and MASs, Information Systems, Web Intelligence as well Boolean Algebra. In what follows, these contributions in the mentioned fields will be summarized in some detail.

10.1 Contributions to Case-based Reasoning

This thesis developed a general theory of case-based reasoning (see Chapter 5). More specifically, it explored similarity relations, similarity measures, similarity metrics, and distance functions in a unified way and built an important relationship between similarity metrics and Euclidean metrics. It examined similarity measures, similarity metrics and distance functions in a novel way and extended the concept of similarity given by Zadeh, examined similarity relations, fuzzy similarity relations, similarity metrics, and their relationships. It thus proposed six different types of similarity relations and corresponding similarity metrics, each of which corresponds to some cases in nature and society. The core idea behind this is the integration of similarity relations and metrics into similarity metrics, based on investigation of similarity relations and traditional metrics used in CBR. Another original idea is to understand the relation between transitivity in fuzzy similarity relations and the triangle inequality in the plane from a new viewpoint, which has been ignored by other researchers.

Then it provided a theoretical formalization for building case bases with three novel algorithms based on similarity relations and fuzzy similarity relations. It also proposed a R^5 model for case based reasoning. Furthermore it examined abductive CBR and deductive CBR and proposed a unified model for integrating abductive CBR and deductive CBR. Finally it proposed rule-based models for case retrieval based on similarity relations, fuzzy similarity relations, and similarity metrics, and fuzzy rule-based models for case retrieval based on the composite rule of inference of Zadeh [355]. Besides, the thesis also proposed a theoretical foundation for case adaptation and a cyclic case adaptation model, which is a kind of similarity-based reasoning (see Chapter 2).

The developed general theory of case-based reasoning filled the gap that CBR has a lack of theoretical foundation or formal methods, and moved CBR towards a firm theoretical foundation, of which similarity or similarity-based reasoning is at the heart, just as relations are at the heart of relational database [108]

10.2 Contributions to Knowledge Management and Experience Management

This thesis examined the relationship between knowledge and experience, between knowledge-based reasoning and experience-based reasoning, and between knowledge-based systems and experience-based systems. It then showed that knowledge is a concrete form of experience and similarity-based reasoning is an operational definition of experience-based reasoning. If perception can be considered as a new direction of artificial intelligence [350], then experience management or experience-based systems are a promising research field of artificial intelligence[1] and Web Intelligence

How to obtain the right knowledge in the right place at the right time is also a big issue for e-commerce with the increasing information overload in the Internet. This thesis (see Chapter 3) examined information overload and information brokerage with models.

Because of research and development of CBR we can say that we are already in the early stages of an experience management revolution. In the near future, we will have the intelligent experience management systems to produce, manage, and process experience in a domain.

10.3 Contributions to E-commerce and E-business

This thesis (see Chapter 3) discussed the evolution from traditional commerce to e-commerce, and examined three kinds of chains in e-commerce: value chains, supply chains, and agent chains. It showed that the linear structure of the traditional value chain, supply chain, and agent chain can be replaced by the most complex structure; that is, a complete graph-based structure, because the dramatic development of the Internet and WWW makes communication free of the time constraints and distance essentially zero. This thesis (see Chapter 3) also discussed transaction-based e-commerce: B2B e-commerce, B2C e-commerce, and C2C e-commerce with models and examples and argued that C2C e-commerce is also an important supplement to the major forms of e-commerce: B2B e-commerce and B2C e-commerce.

This thesis (see Chapter 6) proposed a unified architecture for a CBR-based e-commerce system which covers almost all mentioned activities and give new insight into the traditional CBR cycle. Then it investigates CBR in intelligent support for e-commerce, product recommendation, product configuration, and product negotiation respectively. Another of the contributions of this

1. http://em2002.aifb.uni-karlsruhe.de/call_for_papers.htm

thesis (see Chapter 6) is the decomposition of case adaptation into problem adaptation and solution adaptation, which not only improves the understanding of case adaptation in traditional CBR, but also facilitates the refinement of activity of CBR in e-commerce and intelligent support for e-commerce.

This thesis also proposed a framework of CMB, a CBR system for multiagent brokerage, which integrates CBR, intelligent agents and brokerage. The key insight is that an efficient way for applying CBR in e-commerce is through intelligent agents or multiagent systems. Based on the parsimony principles of intelligence and artificial redundancy in MASs, the thesis argued that any practical MASs should satisfy:

- Only some necessary agents in the MAS have CBR ability
- Only some necessary agents are mobile
- Only some necessary agents are autonomous.

This thesis also (in Chapter 8) discussed the historic/social evolution of agents and brokers and their relationship in bargaining processes using an agent and broker chain.

10.4 Contributions to Web Intelligence

As many people believed, the most important application field of AI in the 1980's was expert systems, while from the middle of the 1990's the most important application field of AI is the intelligent techniques for Web systems, around the Internet. This is Web intelligence. Web Intelligence (WI) is a new direction for scientific research and development that explores the fundamental roles as well as practical impacts of Artificial Intelligence (AI) and advanced Information Technology (IT) on the next generation of Web-empowered products, systems, services, and activities (see http://kis.maebashi-it.ac.jp/wi01/). Web intelligence has become a lovely word for international conferences or workshops or journals or books[1].

WI is, at the moment, related to Web information systems environments and foundations, human-media interaction, Web information management, Web information retrieval, Web agents, Web mining and farming, and emerging Web-based applications (also see http://kis.maebashi-

1. The author used www.openfind.com to search for "Web Intelligence" and obtained 18,100 websites on 25th September 2002.

it.ac.jp/wi01/). Intelligent techniques in e-commerce are an important component of Web Intelligence.

Generally speaking, almost all contributions of this thesis can be considered as one part of WI. However, the most important contribution of this thesis to WI is its investigation into intelligent techniques in e-commerce, which is mainly included in its Chapter 8. More specifically, this thesis (see Chapter 8) first reviewed intelligent agents in e-commerce; that is, agent-based e-commerce. Then it examined multiagent negotiation, which is the core of multiagent negotiation-based e-commerce. Further it classified multiagent-based e-commerce into multiagent-based auction, multiagent-based mediation, and multiagent-based brokerage and gave a brief survey of related works in each. Then it investigated multiagent brokerage. The main idea behind it is that the work of human mediators, auctioneers, and brokers such as searching for the customer information, matchmaking, and brokering should be done by a few intelligent agents in a cooperative way. The thesis also argued that bargaining and compromise play an important role in negotiation, in particular in brokerage.

Based on the characteristics of buyer agents, seller agents, and brokers, this thesis proposed an architecture of a multiagent-based intelligent broker for the brokering process and argued that such an architecture is an abstraction of human agents and brokers working in bargaining processes of brokerage.

It should be noted that the new face of old expert systems is that the knowledge in the knowledge base is all the available information on the Internet. The inference engine is all the tools for accessing and processing the information on the Internet such as browsers, search engines or metasearch engines, and information agents. This is a discentralized inference engine, in contrast to the central inference engine in a traditional expert system. Nobody can control all possible tools for accessing and processing the information on the Internet.

10.5 Contributions to Artificial Intelligence

This thesis examined the relationship of rule-based ESs (RBESs) and CBRSs (see Chapter 2). It also investigated the relationship of CBR, traditional (deductive) reasoning, and fuzzy reasoning, and argued that similarity-based reasoning is the common theoretical basis of CBR, fuzzy reasoning, analogical reasoning (AR), and model-based reasoning (MBR), while deduction is the theoretical basis of similarity-based reasoning. It showed that CBR is a process reasoning, in which a traditional reasoning paradigm plays a pivotal role in each stage of the process (see Chapter 2).

From a methodological viewpoint, the research and development of the thesis can be justified on three different grounds: from the point of view of general AI, from the most specific of multiagent systems, and from the approaches related to e-commerce [150] (p 174). From the viewpoint of AI, at least negotiation can be pointed out which is relevant. AI may provide relevant methodological and technical solution to automated negotiation. From an viewpoint of MASs, auctions are a conveniently scalable problem inside a general program of investigation in multiagent system

From a viewpoint of distributed artificial intelligence, collaboration, cooperation, communication, and negotiation are main features of MASs [85], influenced by the work of Davis and Smith in 1983 [59]. However, multiagent negotiation requires the automation of business negotiation using MASs. In order to do so, this thesis also examined the relationship between bargaining and compromise in negotiation, and argued that bargaining and compromise play an important role in negotiation and in particular in brokerage. The further understanding of bargaining and compromise is a necessary condition for automating negotiation and brokerage.

10.6 Contributons to Multiagent Systems

This thesis showed that high-level intelligence of a system requires a more complex system structure than low-level intelligence does in most cases (see Chapter 6). The intelligence level of the MAS can be improved through coordination, cooperation, communication, and negotiation among the agents within the MAS, although each of them may be less intelligent than an ES. The thesis thus emphasized that simulation of human intelligence depends not only on the computerized knowledge and reasoning of human experts, to which ESs have paid much attention, but also on cooperation, coordination, and communication among the components (agents) within an intelligent system, which MASs have emphasized. Therefore, ES technology and MAS technology complement each other and their integration will facilitate the research and development of intelligent systems.

This thesis also proposed knowledge-based models for integrating CBR systems and MASs. Finally, this thesis (see Chapter 9) also examined the principle of intelligence parsimony and artificial redundancy of intelligence in MASs and proposed heuristic strategies for parsimony of intelligence in development of MASs.

10.7 Contributions to Information Systems

Information systems is an applied discipline that studies the processes of the creation, operation, and social contexts and consequences of systems that manipulate information. The creation and operation of such systems requires the sub-processes of systems analysis, design, development, and management which are bracketed at the beginning by social context and at completion by social consequences [358]. Information systems (IS) play a fundamental and ever-expanding role in all business organisations [276].

Because e-commerce, artificial intelligence, expert systems, system analysis are important components of Information systems, the contributions of this thesis to e-commerce, e-business, artificial intelligence, and Web intelligence are also important contributions to information systems. For example, intelligent techniques for e-commerce, examined in this thesis, will facilitate research and development of information systems for e-commerce or Web intelligence.

10.8 Contributions to Boolean Algebra

As is well known, Boolean algebra has wide applications to science and technology. For example, two concrete examples of Boolean algebra are switching circuits and logic gates. This thesis first applied Boolean structure to its general architecture and integrated CBR, MAS and EC under one roof, which can easily examine relationships between CBR and MAS and E-commerce from a logical viewpoint. Therefore, applying a Boolean structure to the general architecture of the thesis is of methodological significance.

The thesis also used Boolean algebra to examine the relationship of coordination, cooperation, and communication, and showed that the simplest category is MASs without any of these three "C"s, which can be referred to the classical knowledge-based systems (KBSs), while the most complicated category is the MASs with all these three "C"s. Further, this model also paves the way from KBSs to MASs (see Section 4.5.4)

10.9 Future Work

It should be noted that e-commerce applications are just about to leave their infancy According to Henry Lieberman and Sybil Shearin[1], most e-commerce sites today are little more than electronic catalogs of product offerings. Consumer input is limited to requirements questionnaires, search engines, and accepting or rejecting particular offerings. But in complex purchases, such as real estate, cars, or computers, it is often difficult to specify exactly what you want, and priorities and preferences often change in the process of exploration.

Furthermore, the potential of the Internet for truly transforming commerce is largely not realized to date. A human buyer is still responsible for collecting and interpreting information on merchants and products, making decisions on merchants and products and finally entering purchase and payment information. However, intelligent agent technologies can be used to automate several of the most time consuming stages of the business process for example, information brokering and product brokering. Therefore using intelligent techniques in e-commerce has become an interesting topic for research and development of e-commerce. Integration of CBR and multiagent systems in e-commerce is just such an attempt. However, how to formalise or computerize experience in e-commerce, in particular in mediation, negotiation, and brokerage is a big issue. Therefore in future work we will further study real world scenarios in the brokerage field, in particular to incorporate with a real brokerage firm to apply the proposed integrated models of CBR and MASs to improve the on-line brokerage.

In our view, CMB is at the very beginning of research into integration of CBR, multiagent systems in e-commerce. What we have attempted is to fuse CBR and MASs in e-commerce to develop new technical possibilities of intelligent e-commerce. Our research on software is still in its early infancy, but we hope to be able to provide a first “proof of concept” for intelligent e-commerce. In future work we intend to explore applications of the proposed approaches

The design and analysis for implementing CMB in Chapter 9 leads to a lot of new issues in order to implement the architecture of CMB, for the effectiveness, efficiency, and security of the system architecture. As future work, we will implement a prototype e-brokerage place based on the architecture of CMB, to evaluate the chosen software tools and technology and as a means to

1. see URL:http://www.media.mit.edu/research/sig.php?type=sig&id=7

look for the human brokerage partners who have interest in transferring the traditional brokerage into e-brokerage using intelligent agents and CBR technology. We will also explore some implementation details such as the communication between buyer agents and seller agents during the brokerage.

Bibliography

[1] Aamodt A, Plaza E. Case-based reasoning: Foundational issues, methodological Variations, and system approaches, *Artificial Intelligence Communications*, IOS Press, 7(1) 1994. pp 39-59

[2] Adam NR, Dogramaci O, Gangopadhyay A, Yesha, Y. *Electronic Commerce: Technical, Business, and legal Issues*. New Jersey: Prentice Hall PTR, Prentice-Hall Inc. 1999

[3] Aguzzoli S, Avesani P, Massa P. Compositional CBR via collaborative filtering. URL: http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws3/ws3toc1.pdf. 2001

[4] Albus JS. The engineering of mind. *Information Sciences*, 117 (1999) pp 1-18

[5] Allen BP. Case-based Reasoning: Business Applications, *Comm of The ACM,* 37 (3) 1994. pp 40-42

[6] Alonso G, Fiedler U, Hagen C, Lazcano A, Schuldt H, Weiler N. WISE: Business to Business E-Commerce. In: *Proc. of 9th Intl Workshop on Research Issues on Data Engineering: Information Technology for Virtual Enterprises (RIDE-VE'99)*, Sydney, Australia, March 23-24 1999. pp132-139

[7] Atkinson C. Component-Based Product Line Engineering with UML 1/e Addison Wesley, 2002. 528 p

[8] Ba S, Kalakota R, Whinston AB. Using client-broker-server architecture for Intrenet decision support. *Decision Support Systems,* 19, 2000. 171-192.

[9] Bailey J. Active databases and agent systems- A comparison. In Sellis T (ed.): *Proc. of the 2nd Intl. Workshop on rules in database Systems*. LNCS 975, Athens, Greece, 1995. pp 342-256

[10] Baldwin JF, et al. *Fril- fuzzy and Evidential Reasoning in Artificial intelligence*. Research studies press LTD. Taunton, Somerset, England, 1995

[11] Baral C. Abductive reasoning through filtering, *Artificial Intelligence,* 120 2000. pp 1-28

[12] Barbuceanu M, Lo WK. Multi-attribute utility theoretic negotiation for electronic commerce. In: Dignum, F. and Cortés, U.(eds): *Agent Mediated Electronic Commerce III: Current Issues in Agent-based Electronic Commerce Systems.* Berlin: Springer-Verlag 2001, 15-30.

[13] Bartsch-Spörl B, Lenz M, Hübner A. Case-Based Reasoning - Survey and future directions. In Puppe F.(ed): *XPS-99: Knowledge-Based Systems: Survey and Future Directions: Proceedings of the Fifth Biannual German Conference on Knowledge-Based Systems, Würzburg, Germany, 1999. LNCS 1570.* Berlin: Springer-Verlag, 1999. pp 67-89

[14] Batlogg J, Feuerhelm D, Pfannenschwarz B. Ein Information Broker für wierderverwendbare Lernmodule. URL: http://www.cooperation-management.de/english/research/ 2000

[15] Beam C, Segev A, Bichler M, Krishnan. On negotiations and deal making in electronic markets. *Information Systems Frontiers*, 1 (3) 1999. pp 241-258

[16] Beer M, Akram M, Babbs M. Building an agent architecture for an occupational therapy Internet school. In: Arabnia HR (ed.): *Proc Int Conf on Artificial Intelligence* (IC-AI'2000), Las Vegas, Nevada, USA June 26-29. CSREA Press, 2000. pp 197-203

[17] Beer M, d'Inverno M, Luck M, Jennings NR, Preist C, Schroeder M. Negotiation in Multi-Agent Systems. *Knowledge Engineering Review,* 14 (3) (1999) pp 285-289

[18] Béjar J, Cortés U. Agent strategies on DPB auction tournaments. In: Dignum F, Cortés U (eds): *Agent Mediated Electronic Commerce III: Current Issues in Agent-based Electronic Commerce Systems.* Berlin: Springer-Verlag 2001. pp 155-172

[19] Benjamins VR, at al. IBROW3- An intelligent brokering service fro knowledge-component reuse on the World-Wide Web. URL: http://www.swi.psy.uva.nl/projects/IBROW3/docs/

[20] Bento C, Costa E. A similarity metric for retrieval of cases imperfectly described and explained. In: Richter, Wess, et al. (eds): *Proc. of First European Workshop on CBR EWCBR'93*, 1993. pp 8-13

[21] Berghel H. Cyberspace 2000: Dealing with information overload. *Comm of the ACM*, 40 (2) 1997

[22] Bergmann R, Wilke W, Vollrath I, Wess S. Integrating general knowledge with object-oriented case representation and reasoning. In: Burkhard HD, Lenz M (Eds): *4th German Workshop: Case-Based Reasoning - System Development and Evaluation, Informatik-Berichte* Nr. 55, Humboldt-Universität Berlin, 1996. pp120-127. URL: http://wwwagr.informatik.uni-kl.de/~vollrath/Publications.html.

[23] Bergmann R. On the use of taxonomies for representing case features and local similarity measures. URL: www.cbr-web.org/documents/bergmann_tax-sim.pdf. 1998

[24] Bergmann R, Stahl S. Similarity measures for object-oriented case representations. In: *ProcEuropean Workshop on Case-Based Reasoning, EWCBR'98*, 1998. pp 25-36

[25] Bezdek JC, Harris JD. Fuzzy partitions and relations: An axiomatic basis for clustering. *Fuzzy Sets and Systems* 1 (1978) pp 111-127

[26] Bichler M, Segev A. A brokerage framework for Internet commerce, *Distributed and Parallel Databases 7* 1999. pp 133-148

[27] Böhme G. *Fuzzy Logik*, Springer; 1993, 320 p

[28] Bonissone PP, Ayub S. Similarity measures for case-based reasoning systems. In: Meunier B, et al. (eds): IPMU'92-advanced methods in AI: *4th Int Conf on Information Processing and Management of Uncertainty in KBS*, Palma de Mallorca, July 1992 Proceedings, Spain. LNAI 682, Spring Verlag; 1992. pp 483-487

[29] Bosc P, Pivert O. Imprecise data management and flexible querying in databases. In: Yager RR, Zadeh LA (eds): *Fuzzy Sets, Neural Networks, and Soft Computing*. Van Nostrand Reimhold, 1994. pp 368-376

[30] Bosak J. XML, Java, and the future of the Web. URL: http://webreview.com/wr/pub/97/12/19/xml/index.htm 1997

[31] Brazier F, Keplicz BD, Jennings NR, Treur J. Formal specification of multiagent systems: A real-world case, URL: ftp://ftp.elec.qmw.ac.uk/pub/isag/distributed-a...ICMAS95c.ps.Z 1995

[32] Brenner W, Zarnekow R, Wittig H. *Intelligent Software Agents: Foundation and Applications*, Berlin: Springer-Verlag; 1998

[33] Bridge D. Production recommendation systems: A new direction. URL: http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws3/ws3toc2.pdf 2001

[34] Bui T, Lee J. An agent-based framework for building decision support systems. *Decision Support Systems,* 25 (1999) pp 225-237

[35] Burke R. The Wasabi personal shopper: A case-based recommender system. In: *Proc 16th National Conference on Artificial Intelligence*, 1999. pp 844-849

[36] Burkhard HD, Richter MM. On the notion of similarity in case based reasoning and fuzzy theory. In: Pal SK, Dillon TS, Yeung DS (eds) *Soft Computing in Case Based Reasoning*. London: Springer, 2001. pp 28-45

[37] Busetta P, et al. Jack intelligent agents- Components for intelligent agents in Java, *AgentLink Newsletter*, URL: http://citeseer.nj.nec.com/busetta99jack.html 1999

[38] Caglayan A, Harrison C. *Agent Sourcebook.* John Wiley & Sons Inc. 1997

[39] Cardoso HL, Oliveira E. A platform for electronic commerce with adaptive agents. In: Dignum F, Cortés U (eds) *Agent Mediated Electronic Commerce III: Current Issues in Agent-based Electronic Commerce Systems.* LNAI 1991, Berlin: Springer-Verlag; 2001. pp 96-107

[40] Castelfranchi C, Tan Y (eds) *Trust and Deception in Virtual Societies*, Dordrecht, The Netherlands: Kluwer Academic Publishers; 2001. 257 p

[41] Cercone N, McCalla G (eds): *The Knowledge Fronter*: Essays in the Representation of Knowledge. New York: Springer-Verlag; 1987

[42] Chircu AM, Kauffman RJ. Reintermediation strategies in business-to-business electronic commerce. *Int J of Electronic Commerce*, 4 (4) 2000. pp 7-42

[43] Chorafas DN. *Agent Technology Handbook*. New York: McGraw-Hill; 1997

[44] Chrysanthis PK, Znati T, Banerjee S, Chang SK. Establishing virtual enterprises by means of mobile agents. In: *Proc. of 9th Intl Workshop on Research Issues on Data Engineering: Information Technology for Virtual Enterprises (RIDE-VE'99)*, Sydney, Australia, March 23-24, 1999. pp 116-123

[45] Chun I, Lee J, Lee E. I-SEE: An intelligent search agent for EC. *Int J. of Electronic Commerce*, 4 (2) 2000. 83-98

[46] Ciampolini A, Lamma E, Mello P, Stefanelli C. Abductive coordination for logic agents. ACM Symposium on Applied Computing (SAC'99) San Antonio, Texas, 1999. pp 134-140

[47] Clerkin P, Hayes C, Cunningham P. Automated case generation for recommender systems using knowldge discovery techniques. URL: http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws3/ws3toc3.pdf 2001

[48] Comer DE, Stevens DL. I*nternetworking with TCP/IP Vol. III: Client-server programming and applications.* Englewood Cliffs, NJ: Prentice Hall; 1993

[49] Console L, Theseider Dupre' D, Torasso P. On the relationship between abduction and deduction. *J. Logic Comput.* 1 (5) 1991. pp 661-690

[50] Cox MT, et al. Toward agent-based mixed-initiative interfaces. In: H.R. Arabnia (ed.) *ProcInt Conf on Artificial Intelligence* (IC-AI'2000), Las Vegas, Nevada, USA, CSREA Press; June 26-29, 2000. pp 309-315

[51] Cranefield SJS, Diaz A, Purvis MK. Planning and matchmaking for the interoperation of information processing agents. The Information Science discussion Paper Series, University of Otago, New Zealand, 97/01, 1997

[52] Cranefield S, Purvis M, Nowostawski. Is it an ontology or an abstract syntax? modelling objects, knowledge and agent messages. The Information Science discussion Paper Series, University of Otago, New Zealand, 2000/08, 2000

[53] Cunningham P. Intelligent systems in electronic commerce. Invited talk on ICCBR-99, Germany; 1999. URL: http://www.iccbr.org/iccbr99/ 1999

[54] Cunningham P, Bergmann R, et al. Intelligent support for online sales: the Websell experience. URL: http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws3/ws3toc6.pdf 2001

[55] Cukier K. Europe 'plagued by e-commerce confusion'. *CommunicationsWeek*, Issue 220, 1 March 1999

[56] Curran T, Keller G. *SAP R/3 Business Blueprint: Understanding the business process reference model*. Saddle River, NJ.: Prentice Hall PTR; 1998

[57] Damiani E, Khosla R. A human centered agent-based architecture for electronic brokerage. In: *ACM Symposium on Applied Computing*, San Antonio, Texas, USA, 1999. pp 243-249.

[58] Dastani M, Jacobs N, Jonke r CM, Treuer J. Modeling user preferences and mediating agents in electronic commerce. In: Dignum F, Sierra C (eds) *Agent Mediated Electronic Commerce: The European Agent Link Perspective.* LNAI 1991, Berlin: Springer-Verlag; 2001. pp 163-93

[59] Davis R, Smith RG. Negotiation as a metaphor for distributed problem solving. *Artificial Intelligence,* 20, 1983. pp 63-109

[60] Decker K, Sycara K, Williamson M. Middle-agents for the Internet. In: *Proc 15th Int Joint Conf on Artificial Intelligence*. Nagoya, Japan, August 1997.

[61] Decker K, Williamson M, Sycara K. Matchmaking and brokering. URL: http://www.cs.cmu.edu/~softagents/p...icmas96-matchbroker.ps.gz 1996

[62] De Paula GE, Ramos FS, Ramalbo GL. Bilateral negotiation model for agent-mediated electronic commerce. In: Dignum F, Cortés U (eds) *Agent Mediated Electronic Commerce III: Current Issues in Agent-based Electronic Commerce Systems.* Berlin: Springer-Verlag; 2001. pp 1-14

[63] DePree JD, Swartz CW. *Introduction to Real Analysis*. New York: John Wiley & Sons; 1988

[64] Devedzic V. A survey of modern knowledge modelling techniques. *Expert Systems with Applications* 17 1999. pp 275-294

[65] Diaz A, Cranefield SJS, Purvis MK. Planning and matchmaking in a multi-agent system for software integration. The Information Science Discussion Paper series, University of Otago, New Zealand, 97/06, 1997

[66] Dignum F, Cortés U (eds) *Agent Mediated Electronic Commerce III: Current Issues in Agent-based Electronic Commerce Systems.* Berlin: Springer-Verlag; 2001

[67] Dignum F, Sierra C (eds) *Agent Mediated Electronic Commerce: The European Agent Link Perspective.* Berlin: Springer-Verlag; 2001

[68] Doran JE, Franklin S, Jennings NR, Norman TJ. On cooperation in multi-agent systems. *The Knowledge Engineering Review,* 12 (3) 1997, pp 309-314

[69] Doyle M, Ferrario MA, Hayes C, Cunningham P, Smyth B. CBR Net: Smart technology over a network. Dept. of Computer Science, Trinity College Dublin, External Techn. Rep. TCD-CS-1998-07. 1998

[70] Dreyfus HL, Dreyfus SE. *Mind Over Machine*. New York: The Free Press, A Division of Macmillan, Inc. 1986

[71] Dubitzky W, Azuaje F. A Genetic algorithm and growing cell structure approach to learning case retrieval structures. In: Pal SK, Dillon TS, Yeung DS (eds) *Soft Computing in Case Based Reasoning*. London: Springer; 2001. pp 115-46

[72] Dubois D, Esteva F, Garcia P, Godo L, de Màntaras RL, Prade H. Case-based Reasoning: A Fuzzy Approach. In: Ralescu AL, Shanahan JG (eds) *Fuzzy Logic in Artificial Intelligence*, IJCAI'97 Workshop, Berlin: Springer-Verlag; 1999. pp 79-90

[73] Dubois D, Prade H. Similarity-Based Approximate Reasoning. Computational Intelligence: Imitating life, New York: IEEE Press; 1994. pp 69-80

[74] Dubois D, Esteva F, Garcia P, Godo L, de Màntaras RL, Prade H. Fuzzy set-based models in case-based reasoning, IIIA Research Report 97-09, 1997

[75] Dubois D, Esteva F, Garcia P, Godo L, de Màntaras RL, Prade H. Fuzzy modelling of case-based reasoning and decision. In: Leake DB, Plaza E (eds): *Case-based reasoning research and development (ICCBR-97)* LNAI 1266, Berlin: Springer; 1997. pp 599-610

[76] Errico B, Jurisica I. Adaptive agent-based systems for the Web: An application to the NECTAR project, *AAAI Spring Symposium on Intelligent Agents in Cyberspace,* Stanford University, March 22 - 24, Menlo Park, CA. AAAI Press; 1999. pp 75-83

[77] EXPO 2000. URL: http://www.practical-applications.co.uk/Expo2000/ 1999

[78] Eymann T, Padovan B, Schoder D. The catallaxy as a new paradigm for the design of information systems. In: Shi Z, Faltings B, Musen M (eds) *Proc of Int Conf on Intelligent Information Processing (IIP 2000)* (in conjunction with the 16th World Computer Congress), Beijing; 21-25 August 2000. pp 348-355

[79] Falasconi S, et al. Using Ontologies in multiagent systems. URL: http://ksi.cpsc.ucalgary.ca/KAW/KAW96/falasconi/index.html 1996

[80] Fankhauser P, Tesch T. Agents, a broker, and Lies. In: *Proc 9th Int Workshop on Research Issues on Data Engineering: Information Technology for Virtual Enterprises (RIDE-VE'99)*, Sydney, Australia; March 23-24 1999. pp 56-63

[81] Faratin P, Sierra C, Jennings NR. Using similarity criteria to make negotiation trade-offs. In: *Proc 4th Int Conf on Multi-Agent Systems* (ICMAS-2000), Boston, USA; 2000. pp 119-126

[82] Faratin P, Sierra C, Jennings NR. Negotiation decision functions for autonomous Agents. *Int J of Robotics and Autonomous Systems,* 24 (1998) pp 159-182

[83] Farrell J, Gosselin D (1999) *Java^{TM} Programming with Microsft® Visual J++®* 6.0. Ontario, Canada: Course Technology; 1999

[84] Finnie G, Sun Z, Weber K. An intelligent broker-centered multiagent architecture for bargaining processes. In: Arabnia HR (ed.) *Proc Int Conf on Artificial Intelligence* (IC-AI'2000), Las Vegas, Nevada, USA, CSREA Press; June 26-29 2000. pp 175-182

[85] Finnie, G, Sun Z, Weber K. A multiagent-based intelligent broker architecture for bargaining process. In: Kowalczyk R, Lee M (eds*) AIEC'99 Proceedings,* CSIRO, Sydney, Australia; December 6 1999. pp 47-56

[86] Finnie, G, Sun Z, Weber K. Multiagent-based electronic bargaining processes. In: Naghdy F, Tianfield H, et al. (eds) *Proc of the International ICSC Symposia on Intelligent Systems & Applications*. December 11-15 Wollongong, Australia, ICSC Academic Press, Canada/Switzerland; 2000. pp 535-541

[87] Finnie G, Sun Z. Similarity and metrics in case-based reasoning. *Int J of Intelligent System,* 17(3) 2002. pp 273-287

[88] Finnie G, Sun Z. R^5 model for case-based reasoning. *Knowledge-based Systems*, 16 (1) 2003. pp 59-65

[89] Finnie, G, Sun, Z. A knowledge-based model of multiagent CBR systems. Accepted for publication at Int Conf on Computational Intelligence for Modelling, Control and Automation (CIMCA'2003), Vienna, Austria, 2002

[90] Finnie G and Sun Z**.** A logical foundation for the CBR cycle. Submitted to *Int J of Intelligent Systems,* 2002

[91] Finnie G, Wittig G. Intelligent support for internet marketing with case based reasoning 2nd Annual CollECTeR Conference on Electronic Commerce, Sydney, September 1998. pp 6-14

[92] Fischer, K. The rule-based multi-agent system MAGSY. URL: ftp://ftp.dfki.uni-sb.de/pub/MAGSY/Papers/CKBS92.ps.gz 1993

[93] Fischer M. Characterising simple negotiation as distributed agent-based theorem proving: a preliminary report. In: *Proc 4th Int Conf on Multi-Agent Systems* (ICMAS-2000), Boston, USA; 2000. pp 127-135

[94] Flash P. Abduction and Induction in AI. URL: http://www.cogs.susx.ac.uk/ecai98/twbrowse/W19-bod; 1998

[95] Flores-Mendez RA. Towards a standardization of multi-agent system frameworks. URL: http://www.cpsc.ucalgary.ca./~robertof/index.html# , 1999

[96] Franklin S, Graesser A. Is it an agent, or just a program?: A Taxonomy for autonomous agents. In: *Proc. 3rd Intl Workshop on Agent Theories, Architectures, and languages* in 1996; Intelligent Agents III, Springer-Verlag; 1997. 21-35 URL: ftp://ftp.msci.memphis.edu/comp/caat/agentprog.ps.z

[97] Franklin S. Coordination without communication. URL: http://www.msci.memphis.edu/~franklin/coord.html 2000.

[98] Gardingen D, Watson I. A web based CBR system for heating ventilation and air conditioning systems sales support. *Knowledge-based Systems*, 12 (5-6) 1999. pp 207-214

[99] Gates B. *Business @ the Speed of Thought; Using a Digital Nervous System*. New York: Warner Books; 1999

[100] Genesereth M R, Ketchpel SP. Software agents. *Comm of The ACM*, 37 (7) July 1994. pp 48-53

[101] Giampapa JA, Sycara K. Conversational case-based planning for agent team coordination. In: *Case-Based Reasoning Research and Development: Proc 4th Int Conf on Case-Based Reasoning (*ICCBR 2001). Berlin Heidelberg: Springer-Verlag, LNAI 2080, 2001. pp 189-203

[102] Glover EJ, Lawrence S, Gordon MD, Birmingham WP, Giles CL. Web Search-Your way. *Comm of the ACM*, 44 (12) 2001. pp 96-102

[103] Goldsmith SY, Spires SV, Phillips LR. Object frameworks for agent system development, URL: http://aisl.sandia.gov/Papers/aaai-agent-tools.pdf 1998

[104] Guttman RH, Moukas AG, Maes P. Agent-mentioned electronic commerce: A survey. *Knowledge Engineering Review*, 13(2) 1998. pp 147-159. URL: http://ecommerce.media.mit.edu

[105] Guttman RH, Maes P. Agent-mediated integrative negotiation for retail electronic commerce. URL: http://ecommerce.media.mit.edu/ 1998

[106] Hait FB. A multiagent system integrating a learning process applied to financial market. In: Arabnia HR (ed.) *Proc Int Conf on Artificial Intelligence* (IC-AI'2000) Vol. II, Las Vegas, Nevada, USA, CSREA Press; June 26-29, 2000. pp 681- 90

[107] Hájek P, Havránek T, Jirousek R. *Uncertain Information Processing in expert systems*. CRC Press, Inc. 1992

[108] Hall C, O'Donnell J. *Discrete Mathematics Using a Computer*. London: Springer-Verlag; 2000

[109] Hammond K. *Case-based Planning: Viewing Planning as a Memory Task*. Cambridge, MA: Academic Press; 1989

[110] Han KS, Noh MH. Critical failure factors that discourage the growth of electronic commerce. *Int J. of Electronic Commerce*, 4 (2) 2000. pp 25-43

[111] Hayden S, Carrick C, Yang Q. A catalog of agent coordination patterns. In: Proc. *3rd Int Conf on Autonoumous Agents*. Seattle WA, USA, 1999. pp 412-413

[112] Hayes C, Cunningham P. Shaping a CBR view with XML. In: *Proc. of Int Conf on CBR (ICCBR'99*) Munich, Germany, July 2000. pp 468-81

[113] Hayes C, Cunningham P, Doyle M. Distributed CBR using XML. In: *Workshop for Intelligent Systems and Electronic Commerce* (within German Conference on Artificial Intelligence (KI'98)). September 15-17 1998

[114] Hayes-Roth F. Expert systems. In: Shapiro SC (ed.) *Encyclopedia of Artificial Intelligence*, Vol.1-2, New York: Wiley and Sons; 1992. pp 287-298

[115] Hayes-Roth F. Rule-based systems. In: Shapiro SC (ed.) *Encyclopedia of Artificial Intelligence,* Vol.1-2, New York: Wiley and Sons; 1992. pp 963-973

[116] Hayes-Roth F, Jacobstein N. The state of knowledge-based systems. *Comm of The ACM,* 37 (3) March 1994. pp 27-39

[117] http://www.igec.umbc.edu/

[118] Hedberg S. Agents for sale: first wave of intelligent agents go commercial. *IEEE Expert,* 11 (6) December 1996. p 16

[119] Hermans B. Desperately seeking: Helping hands and human touch. URL: http://www.firsmonday.dk/issues/issue3-11/hermans/ 1999

[120] Hermans B. Intelligent software agents on the internet. URL: http://www.hermans.org/agents/index.html 2000

[121] Hoffner Y. Supporting contract match-making. In: *Proc 9th Int Workshop on Research Issues on Data Engineering: Information Technology for Virtual Enterprises (RIDE-VE'99)*, Sydney, Australia; March 23-24 1999. pp 64-71

[122] Holland HJ. *Adaptation in Natural and Artificial Systems*- an introductory Analysis with applications to biology, control, and artificial intelligence. The MIT Press; 1995

[123] Holmback H, Greaves M, Bradshaw J. A pragmatic principle for agent communication. In: Etzioni, O. et al (eds), Proc 3rd Int Conf on Autonomous Agents, ACM Press, 1999. pp 368-369

[124] Hüllermeier E, Dubois D, Prade H. Formalizing case based inference using fuzzy rules. In: Pal SK, Dillon TS, Yeung DS (eds) *Soft Computing in Case Based Reasoning*. London: Springer; 2001. pp 47-72

[125] Huhns MN, Singh MP (eds) *Readings in Agents*. San Francisco: Morgen Kaufmann Publishers, Inc. 1998

[126] Hunt J. Evolutionary case based design. In: Watson [316]

[127] IBROW3: An intelligent brokering service for knowledge-component reuse on the World-Wide Web. URL: http://www.swi.psy.uva.nl/projects/IBROW3/ 1999

[128] Inakoshi H, Okamoto S, Ohta Y, Yugami N. Effective decision support for product configuration by using CBR. URL: http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws3/ws3toc4.pdf

[129] Jacas J, Valverde L. On fuzzy relations, metrics and cluster analysis. URL: http://dmi.uib.es/people/valverde/gran1/GRAN1.html, 1996

[130] Jacobson I, Griss M, Jonsson P. *Software Reuse Architecture, Process and Organization for Business Success*. New York: ACM Press; 1997

[131] Jaczynski, T.: An object-oriented framework for the design and implementation of case-based reasoners. CBR, Newletter, 1998

[132] Jain AK, Apraricio IV, M, Singh MP. Agents for process coherence in virtual enterprises. *Comm of the ACM*, 42 (3) 1999. pp 62- 69

[133] Jennings K. ARCHON: Cooperating agents for industrial process control. URL: http://.ecs.soton.ac.uk/~nrj/archon/test_1.html 1996

[134] Jennings NR. Coordination techniques for distributed Artificial Intelligence. In: O'Hare GMP, Jennings NR (eds), *Foundations of Distributed Artificial Intelligence.* Wiley; 1996. pp 187-210

[135] Jennings NR, et al. ADEPT: Managing business processes using intelligent agents. http://www.csd.abdn.ac.uk/tnorman/publications/bcs1996.pdf 1996.

[136] Jennings NR, Parsons S, Sierra C, Faratin P. Automated Negotiation. In: *Proc. 5th Int. Conf. on the Practical Application of Intelligent Agents and Multi- Agent Systems (PAAM-2000)*, Manchester, UK, 2000. pp 23-30

[137] Jennings NR, Varga LZ, Aarnts RP, Fuchs J, Skarek P. Transforming standalone expert systems into a community of cooperating agents. URL: ftp://ftp.elec.qmw.ac.uk/pub/isag/distri...ENG-APPL-AI-6-4.ps.Z 1993

[138] Jennings NR, Wooldridge MJ. *Agent-oriented software engineering.* In: Bradshaw J (ed.), *Handbook of Agent Technology*, AAAI/MIT Press; 2000

[139] Jennings NR. On agent-based software engineering. *Artificial Intelligence*, 117, 2000. pp 277-296

[140] Jennings NR, Wooldridge MJ. Applications of intelligent agents. In: Jennings NR, Wooldridge MJ (eds) *Agent Technology: Foundations, Applications, and Markets*. Berlin: Springer-Verlag; 1998. pp 3-28

[141] Jonker CM, Treur J. A re-usable broker agent architecture with dynamic maintenance capabilities. In: Etzioni O, et al (eds), *Proc 3rd Int Conf on Autonomous Agents*, ACM Press; 1999. pp 376-377

[142] Joshi, A., Singh, M.P.: Multiagent systems on the Net. *Comm of The ACM*, 42(3) 1999. 39-40

[143] Jung JJ, Jo GS. Brokerage between buyer and seller agents using constraint satisfaction problem models. *Decision Support Systems,* 28 2000. pp 293-304

[144] Jurisica I. TA3: Theory, Implementation, and Applications of Similarity-based Retrieval for Case-based Reasoning, PhD thesis, Department of Computer Science, University of Toronto, Toronto, Canada, 1998

[145] Kalakota R, Whinston AB. *Frontiers of Electronic Commerce*. Reasing, Massachusetts: Addisson-Wesley Publishing Company, Inc. 1996

[146] Kalakota R, Whinston AB. *Electronic Commerce: A manager guide*. Reasing, Massachusetts: Addisson-Wesley Longman, Inc. 1997

[147] Khan AS, Hofmann A. Acquiring adaptation knowledge for CBR with MIKAS. In: Stumptner M, Corbett D, Brooks M (eds), *AI 2001: Advances in Artificial Intelligence*. LNAI 2256. Berlin: Springer-Verlag; 2001. pp 201-212

[148] Kindler H, Densow D, Fischer B, Fliedner TM. Mapping laboratory medicine onto the select and test model to facilitate knowledge-based report generation in laboratory medicine. In: Barahona P, et al. (eds), *LNAI* 934. Berlin: Springer; 1995. pp 265-275

[149] Kim SK, Shin KH, Yoo CW. A design of the mobile agent languages using URN for effective mobility extension. In: Arabnia HR (ed.), *Proc Inter Conf on Artificial Intelligence* (IC-AI'2000) Vol. II, Las Vegas, Nevada, USA, CSREA Press; June 26-29 2000. pp 581-586

[150] Klusch M (ed.). *Intelligent Information agents, agent-based information discovery and management on the internet*. Berlin: Springer-Verlag; 1999

[151] Kolodner JL (ed.). Special issue on Case-Based Reasoning. *Machine Learning,* 10(3) 1993

[152] Kolodner J. *Case-based Reasoning*. San Mateo: Morgan-Kaufmann Publishers; 1993. 668 p

[153] Koning JL. Designing and Testing Negotiation protocols for electronic commerce applications. In: Dignum F, Sierra C (eds), *Agent Mediated Electronic Commerce: The European Agent Link Perspective.* Berlin: Springer-Verlag; 2001. pp 34-60

[154] Kosko B. *Neural Networks and Fuzzy Systems: a dynamical systems approach to machine intelligence*. Prentice Hall, Inc. 1992

[155] Kosiur D. *Understanding Electronic Commerce*. Microsoft Press; 1997

[156] Kotz D, Gray RS. Mobile agents and the future of the Internet. *ACM Operating Systems Review*, 33 (3) 1999. pp 7-13

[157] Kou W, Yesha Y (eds): *Electronic Commerce Technology Trends, Challenges and opportunities*. Carlsbad, USA: IBM Press; 2000

[158] Kowalski G. *Information Retrieval Systems: Theory and Implementation.* Boston/Dordrecht/London: Kluwer Academic Publishers; 1997

[159] Lai H. An object-oriented architecture for intelligent virtual receptionists. *Int J of Electronic Commerce*, 4 (3) 2000. pp 69-86

[160] Laird JE. Research in human-level AI using computer games. *Comm of The ACM,* 45 (1) 2002. pp 32-35

[161] Lang S. *Complex Analysis* (2nd ed.). New York: Springer-Verlag; 1985

[162] Lange DB, Oshima M. Dispatch your agents; shut off your machine. *Comm of The ACM,* 42 (3) 1999. pp 88-89

[163] Larson K, Sandholm T. Bargaining with limited computation: Deliberation equilibrium. *Artificial Intelligence*, 132 2001. pp 183-217

[164] Lawrence S, Giles CL. Tips for searching the Web URL: http://www.neci.nec.com/~lawrence/searchtips.htm

[165] Leake DB. Focusing construction and selection of abductive hypotheses. In: *Proc 11th Int Joint Conf on Artificial Intelligence*; 1993. pp 24-29

[166] Leake DB. Abduction, experience, and goals: a model of everyday abductive explanations. *J of Experimental and Theoretical Artificial Intelligence*, 7 1995. pp 407-428

[167] Leake D. *Case-Based Reasoning: Experiences, Lessons & Future Direction*. Menlo Park, California: AAAI Press / MIT Press; 1996. 420 p

[168] Leake DB, Plaza E (eds): *Case-Based Reasoning Research and Development*. *Proc 2nd Int Conf on CBR, ICCBR-97*. Providence, USA, Jul. 1997, Springer; 1997

[169] Leake DB, Wilson DC. When experience is wrong: Examining CBR for changing tasks and enviornments. In: *Proc ICCBR-99, LNAI 1650*. Berlin: Springer Verlag; 1999. pp 218-232

[170] Lee KJ, Chang YS, Lee JK. Time-bound negotiation framework for electronic commerce agents. *Decision Support Systems,* 28 2000. pp 319-31

[171] Lenz M. Experiences from deploying CBR applications in electronic commerce. URL: http://wwwagr.informatik.uni-kl.de/lsa/gwcbr99/ 1999

[172] Lenz M, Bartsch-Spörl B, Burkhard HD, Wess S (eds): *Case-Based Reasoning Technology, from Foundations to Applications*, Berlin: Springer; 1998. 405 p

[173] Levine MM. A brief history of information brokering. URL: http://www.asis.org/Bullten/Feb-95/ 1995

[174] Lewis WJ. *Data Warehousing and E-Commerce*. Upper Saddle River, NJ: Prentice Hall; 2001

[175] Lewis M, Jacobson J. Game engines in scientific research. *Comm of The ACM,* 45 (1) 2002. pp 27-31

[176] Li LLX. Knowledge-based Problem Solving: an Approach to Health Assessment. *Expert System with Applications,* 16 1999. pp 33-42

[177] Li Y, et al. Decision support system for virtual enterprise creation. In: Arabnia HR (ed.). *Proc Inter Conf on Artificial Intelligence* (IC-AI'2000), Vol. II, Las Vegas, Nevada, USA, CSREA Press. June 26-29 2000. pp 807-13

[178] Liang TP, Huang JS. A framework for applying intelligent agents to support electronic trading. *Decision Support Systems,* 28 2000. pp 305-17

[179] Liang TP, Donng HS. Effect of bargaining in electronic commerce. *Int J of Electronic Commerce*, 4 (3) 2000. pp 23-43

[180] Lister K, Sterling L. Agents in a multi-cultural world: towards ontological reconciliation. In: Stumptner M, Corbett D, Brooks M (eds), *AI 2001: Advances in Artificial Intelligence*. LNAI 2256. Berlin: Springer-Verlag; 2001. pp.321-332

[181] Liu JNK, Leung TTS. A Web-based CBR agent for financial forecasting. URL: http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws5/ws5toc11.pdf. 2001

[182] Liu ZQ. Fuzzy logic based neural network for case based reasoning. In: Pal SK, Dillon TS, Yeung DS (eds): *Soft Computing in Case Based Reasoning.* London: Springer; 2001. pp 213-240

[183] Lomuscio A, Wooldridge M, Jennings NR. Negotiation in e-commerce. In: Dignum F, Sierra C (eds): *Agent Mediated Electronic Commerce: The European Agent Link Perspective.* Berlin: Springer-Verlag; 2001. pp 19-33

[184] Luger G F, Stubblefield WA. *Artificial Intelligence*. Addison Wesley Longman, Inc. 1998

[185] Lyons DJ. Brokers tuning in to intelligent workstations. *PC Week*, 4 (40) 1987. p 187

[186] Ma G, Xu J, Shi C. On the mental state model of agent. *J of Software,* 10 (4) 1999

[187] Maamar Z, Dorion E, Daigle C. Toward virtual marketplaces for e-commerce support. *Comm of the ACM,* 44 (12) 2001. pp 35-38

[188] Maes P, Guttman RH, Moukas AG. Agents that buy and sell. *Comm of The ACM*, 42 (3) 1999. pp 81-91

[189] Maes P. Agents that reduce work and information overload. *Comm of The ACM*, 37 (7) July 1994. pp 31-40

[190] Magnani L. *Abduction, Reason, and Science, Processes of Discovery and Explanation*. New York: Kluwer Academic/Plenum Publishers; 2001. 205 p

[191] Maher ML, Garza AGdS. Case-based reasoning in design. URL: http://www.computer.muni.cz/pubs/expert/feat4-97.htm. 1997

[192] Maher ML, Pu P. I*ssues and Applications of Case-based Reasoning in Desig*n. Lawrence Earlbaum Associates, 1997

[193] Mandry T, Pernul G, Roehm AW. Mobile agents in electronic markets: Opportunities, Risks, agent protection. *IJEC* 5 (2) 2001. pp 47-60

[194] Marir F, Watson I. Case-Based Reasoning: A Categorised Bibliography. *The Knowledge Engineering Review,* 9 (3) 1994. available URL: http://online.loyno.edu/cisa494/papers/Marir.html

[195] Marling C, Petot G, Sterling L. Integrating case-based and rule-based reasoning to meet multiple design constraints. *Computational Intelligence,* 15 (3), 1999. pp 308-332

[196] Martin DL, Cheyer AL, Moran DB. The open agent architecture: A Framework for building distributed software systems. *Applied Artificial Intelligence*, 13, 1999. pp 91-128

[197] Martín FJ, Plaza E, Arcos JL. Knowledge and experience reuse through communication among competent (peer) agents. *Int J of Software Engineering and Knowledge Engineering,* 9 (3) 1999. pp 319-41

[198] Martín F, Plaza E. Auction-based retrieval. In *Proc of 2nd Congrés Català d'Intel.ligència Artificial*, Girona, Spain. pp 136-145. URL: http://www.iiia.csic.es/Projects/FedLearn/CoopCBR.html

[199] Matos N, Sierra C. Evolutionary computing and negotiating agents. In: Noriega P, Sierra C (eds): *Proc 1st Int Workshop on Agent Mediated Electronic Trading* (AMET-98). Berlin: Springer-Verlag; 1999. pp 126-50

[200] Matskin M, et al. Agora: a multi-agent platform and its implementation. In: Arabnia HR (ed.): *Proc Int Conf on Artificial Intelligence* (IC-AI'2000), Las Vegas, Nevada, USA, CSREA Press, June 26-29 2000. pp 549-55

[201] Matthias K (ed.): *Intelligent Information Agents: Agent-based information Discovery and Management on the Internet*. Berlin: Springer; 1999

[202] McCartney R, Sanders KE. The case for cases: A call for purity in case-based reasoning. In: *Proc AAAI Symposium on Case-based Reasoning*; 1990. pp 12-16

[203] McDonald T. When will AI get down to business? URL: http://www.ecommercetimes.com/perl/printer16430/ 2002

[204] McGoveen G. Information nobodies. *On The Internet*, 5 (5) 1999. pp 7, 13

[205] McSherry D. Automating case selection in the construction of a case library. *Knowledge-based systems*, 13 2000. pp 133-40

[206] Medler DA, Dawson MR. Using redundancy to improve the performance of artificial neural networks. URL: http://www.neuro.mcw.edu/~dmedler/papers/ai94.pdf 1994

[207] Merwe Jvd, von Solms SH. Electronic commerce with secure intelligent trade agents. *Computer & Security,* 17 (5) 1998. pp 435-446

[208] Monderer D, Tennenholtz M. Optimal auctions revisited. *Artificial Intelligence,* 120 2000. pp 29-42

[209] Moukas A, Zacharia G, Guttman R, Maes P. Agent-mediated electronic commerce: An MIT media laboratory perspective. *Int J of Electronic Commerce,* 4 (3) 2000. pp 5-21

[210] Mueller JP, Pischel M. Doing business in the information marketplace. In: Etzioni O, et al (eds): *Proc 3rd Int Conf. on Autonomous Agents.* ACM Press; 1999. pp 139-147

[211] The National Office for the Information Economy, E-commerce beyond 2000, Final report, the Commonwealth Department of Communications, Information Technology and the Arts, Canberra, 2000

[212] Newell A, Simon HA. Computer science as empirical inquiry: Symbols and search, 1975 Turing Award Lecture. In: ACM Press Anthology Series: *ACM Turing Award Lectures The First Twenty Years* (1966 to 1985), New York: ACM Press; 1987. pp 287-317

[213] Nguyen NB, Ho TB. A mixed similarity measure in near-linear computational complexity for distance-based methods. In: Zighed DA, Komorowski J, Zytkow J (eds): *Principles of Data Mining and Knowledge Discovery (PKDD* 2000), LNAI 1910, Berlin: Springer Verlag; 2000. pp 211-220

[214] Nilsson NJ. *Artificial Intelligence: A New Synthesis*. San Francisco, California: Morgan Kaufmann Publishers, Inc. 1998. 513 p.

[215] Novais P, Brito L, Neves J. Agreement in virtual marketplaces with CBR-supported negotiation. In: *Proc. PAAM 2000- 5th Int Conf on the Practical Application of Intelligent Agents and Multiagents*. Manchester; 2000. pp 203-206

[216] Nwana HS, Ndunu DT. A brief Introduction to software agent technology. In [140], pp 29-47

[217] Nwana HS, Ndunu DT, Lee LC, Collis JC. ZEUS: A toolkit for building distributed multiagent systems. *Applied Artificial Intelligence,* 13 1999. pp 129-185

[218] Odell J, Parunak HVD, Bauer B. Extending UML for agents. URL: http://www.jamesodell.com/ExtendingUML.pdf

[219] Oliveira E, Rocha AP. Agents advanced features for negotiation in electronic commerce and virtual organisations formation process. In: Dignum F, Sierra C (eds): *Agent Mediated Electronic Commerce: The European Agent Link Perspective.* LNAI 1991, Berlin: Springer-Verlag, 2001. pp 78-97

[220] Olivia C, Chang CF, Enguis CF, Ghose AK. Case-based BDI agents: An effective approach for intelligent search on the World Wide Web, *AAAI Spring Symposium on Intelligent Agents in Cyberspace,* Stanford University, March 22 - 24, Menlo Park, CA. AAAI Press; 1999. pp 20-27

[221] Osborne HR, Bridge DG. A formal analysis of case base retrieval. In: Leake DB, Plaza E (eds): Case-based reasoning research and development (ICCBR-97), LNAI 1266, Berlin: Springer, 1997. pp 235-44. URL: http://citeseer.nj.nec.com/osborne97formal.html 1997

[222] Ovchinnikov S. Similarity relations, fuzzy partitions, and fuzzy orderings. *Fuzzy Sets and Systems,* 40, 1991. pp 107-26

[223] Pal SK, Dillon TS, Yeung DS (eds): *Soft Computing in Case Based Reasoning*. London: Springer; 2001

[224] Papaioannou T. Mobile Information Agents for Cyberspace- State of the Art and Visions. In: *Proc Cooperating Information Agents* (CIA-2000), 2000. URL: http://www.agents.umbc.edu/agentnews/5/19/

[225] Paurobally S, Cunningham J. Specifying the process and states of negotiation. In: Dignum F, Sierra C (eds) *Agent Mediated Electronic Commerce: The European Agent Link Perspective.* Berlin: Springer-Verlag; 2001. pp 61-77

[226] Parsons S. Current Approaches to Handling Imperfect information in Data and Knowledge bases, I*EEE Trans. knowl. & data engi*, 8 (3), 1996

[227] Patil B. Agents and agent technologies: An overview. URL: http://www.darmstadt.gmd.de/~patil/agent_report, 1998

[228] Paurobally S, Cunningham J. Specifying the processes and states of negotiation. In: Dignum F, Sierra C (eds): *Agent Mediated Electronic Commerce: The European Agent Link Perspective.* LNAI 1991, Berlin: Springer-Verlag; 2001. pp 61-77

[229] Perry DE, Wolf AL. Foundations for the study of software architecture. *ACM SIGSOFT Software Engineering Notes* 17 (4) 1992. pp 40-52. URL: http://citeseer.nj.nec.com/perry92foundation.html.

[230] Petrie CJ. Agent-based engineering, the web, and intelligence. *IEEE Intelligent Systems,* 11 (6) 1996. pp 24-29

[231] Plaza E, Esteva F, Garcia P, Godo L, López de Màntaras R. A logical approach to case-based reasoning using fuzzy similarity relations. *Info Scie,* 106, 1996. 105-122. URL: http://www.iiia.csic.es

[232] Plaza E, Arcos JL, Martin F. Cooperative case-based reasoning. In: *Distributed Artificial Intelligence meets Machine Learning,* LNAI 1221. Springer Verlag; 1997. pp 180-201

[233] Plaza E, Ontañón S. Ensemble case-based reasoning: Collaboration policies for multiagent cooperative CBR. In: *Case-Based Reasoning Research and Development: ICCBR 2001*. LNAI 2080. Springer-Verlag; 2001. pp 437-451

[234] Poole D, Goebel R, Aleliunas R. Thoerest: A logical reasoning system for defaults and diagnosis. In: Cercone N, McCalla G (eds): *The Knowledge Frontier. Essays in the Representation of Knowledge.* New York: Springer-Verlag; 1987. pp 331-52

[235] Portinale L, Torasso P. ADAPtER: An integrated diagnostic system combining case-based and abductive reasoning. In: *Proc 1st Int Conf on Case-Based Reasoning* - ICCBR95. Sesimbra, LNAI 1010. Springer-Verlag; 1995. pp 277-288; URL: http://citeseer.nj.nec.com/portinale95adapter.html

[236] Prasad MVN, Lesser VR, Lander SE. Retrieval and reasoning in distributed case bases. *J of Visual Communication and Image Representation,* 7 (1), 1996. pp 74-87

[237] Prasad MVN. Distributed case-based reasoning. In: *Proc 4th Int Conf on Multi-Agent Systems (ICMAS-2000)*, Boston, USA; 2000. pp 222-229

[238] Preist C, Bartolini C, Phillips I. Algorithm design for agents which participate in multiple simultaneous auctions: In: Dignum F, Cortés U (eds): *Agent Mediated Electronic Commerce III: Current Issues in Agent-based Electronic Commerce Systems.* Berlin: Springer-Verlag; 2001. pp 139-154

[239] Pressman RS. *Software Engineering: A practitioner's approach* (5 edn). McGraw-Hill, Boston, Singapore; 2001

[240] Purvis M, Cranefield SJS, Diaz A, et al. The NZDIS project: an agent-based distributed information systems architecture. The Information Science Discussion Paper series, University of Otago, New Zealand; 97/01, 1997

[241] Rakotonirainy A, Seng Wai Loke, Zaslavsky A. Towards multiagent support for open mobile virtual communities. In: Arabnia HR (ed.) *Proc Int Conf on Artificial Intelligence* (IC-AI'2000). Las Vegas, Nevada, USA, CSREA Press; June 26-29 2000. pp 127-33

[242] Reddy R. To dream the possible dream. *Comm of the ACM,* 39 (5) 1996. pp 105-112

[243] Reeves S, Clarke M. *Logic for computer science*. Wokingham, England: Addison-Wesley Publishing Company; 1990

[244] Rich E, Knight K. *Artificial Intelligence (2nd edn.).* New York: McGraw-Hill, Inc. 1991

[245] Richter MM. CBR: Past, present, future a personal view. Invited talk at *ICCBR01*; URL: http://wwwagr.informatik.uni-kl.de/richter/ 2001

[246] Riechen D (ed.): Special Issue: Intelligent Agents. *Comm of The ACM*, 37 (7) July 1994

[247] Riehle D, Zuellighoven H. Understanding and using patterns in software development. *Theory and Practice of Object Systems,* 2 (1) 1996. pp 3-13

[248] Riesbeck CK. What next? the future of case-based reasoning in post-modern AI. In: Leake DB, [167], 1996. pp 371-88

[249] Robinson WN, Volkov V. Supporting the negotiation life cycle, Comm of the ACM, 41 (5) 1998. pp 95-102

[250] Roesler M, Hawkins DT. Intelligent Agents: Software Servants for an Electronic Information World. *Online* 18 (4), 1994. pp 18-28

[251] Rosenzweig B, Silvestrova E. Oracle PL/SQL. Upper Saddle River, NJ: Prentice-Hall PTR; 2000. pp 447

[252] Ross KA, Wright CRB. *Discrete Mathematics (*2nd edn.) Englewood Cliffs, New Jersey: Prentice Hall; 1988

[253] Roth AE. *Axiomatic Models of Bargaining*. Berlin/New York: Springer; 1979

[254] Rubin JZ, Brown BR. *The social psychology of bargaining and negotiation.* San Diego, New York: Academic Press, Inc. 1975

[255] Rudin W. *Real complex analysis*. New York: McGraw-Hill; 1987

[256] Russell S, Norvig P. *Artificial Intelligence: A modern approach*. Upper Saddle River, New jersey: Prentice Hall; 1995

[257] Salz-Trautman P. DT issues e-commerce card. *CommunicationsWeek*, 217, Jan 1999

[258] Sanders KE, Kettler BP, Hendler JA. The case for structure-based representations. In: *Proc 1st Int Conf on Case-Based Reasoning*, Portugal; 1995

[259] Sarkar M, Butler B, Steinfield C. Cybermediaries in electronic marketplaces: toward theory building. *J Business Research*, 41, 1998. pp 215-221

[260] Schank RC. Dynamic Memory: *A Theory of Reminding and Learning in Computers and People*. New York: Cambridge University Press; 1982

[261] Schank RC, Riesbeck C. *Inside Case-Based Reasoning*. Hillsdale, NJ: Lawrence Erlbaum Associates; 1989

[262] Schmitt S, Maximini R, et al. Implementation of a product customization module for CBR systems in electronic commerce environments. URL: http://wwwagr.informatik.uni-kl.de/~gwcbr2k/program.html, 2000

[263] Schneider GP, Perry JT. *Electronic Commerce.* Course Technology; 2000

[264] Segev A, Beam C. Brokering strategies in electronic commerce markets. In: *Proc ACM Conf on Electronic Commerce* (EC'99), Denver, Colorado: ACM Press; 1999 pp 167-76

[265] Sharl A. *Evolutionary Web Development*, London/Berlin: Springer; 2000

[266] Schillo M, Büeckert HJ, Fischer K, Klusch M. Towards a definition of robustness for market-style open multi-agent systems. URL: http://www.millo.de/virtosphere/data/publications/articles/Schillo+.Robustness.AA01.pdf 2001

[267] Schneider GP, Perry JT. *Electronic Commerce* (2nd edn). Canada: Course Technology; 2001. 478 p

[268] Sierra C, Faratin P, Jennings NR. Deliberative automated negotiators using fuzzy similarities. In: *Proc EUSFLAT-ESTYLF Joint Conf on Fuzzy Logic*. Palma de Mallorca, Spain; 1999. pp 155-158

[269] Sierra C, Dignum F. Agent-mediated electronic commerce: Scientific and technological roadmap. In: Dignum F, Sierra C (eds): *Agent Mediated Electronic Commerce: The European Agent Link Perspective.* LNAI 1991 Berlin: Springer-Verlag; 2001. pp 1-18

[270] Simoudis E. Using case-based retrieval for customer technical support. *IEEE Expert,* 7 (5) 1992. pp 7-12

[271] Singh MP. Towards a formal theory of communication for multiagent systems. In: *Proc IJCAI'91*; 1991. URL: http://www.csc.ncsu.edu/faculty/mpsingh/papers/

[272] Smyth B, Cunningham P (eds): *Advances in Case-based Reasoning*. LNAI 1488. Berlin: Springer-Verlag; 1998

[273] Smith I, Faltings B (eds): *Advances in Case-based Reasoning.* LNAI 1168. Springer-Verlag; 1996

[274] Smyth B, Keane MT. Adaptation-guided retrieval: Questioning the similarity assumption in reasoning. *Artificial Intelligence,* 102 (2) 1998. pp 249-293

[275] Slonim TY, Schneider M. Design issues in fuzzy case-based reasoning. *Fuzzy Sets and Systems,* 117 2001. pp 251-267

[276] Stair RM, Reynolds G. *Principles of Information Systems*. Cambridge, MA: Course Technology; 1999

[277] Stanfill C, Waltz D. Toward memory-based reasoning. *Comm of the ACM,* 29 (12) 1986 pp 1213-1228

[278] Sun R. Robust reasoning: integrating rule-based and similarity-based reasoning. *Artificial Intelligence,* 75, 1995. pp 241-295

[279] Sun Z. Logics and their applications to computer science. *J of Hebei University* 11 (2), 1991. pp 93-98

[280] Sun Z, Tang W, Lu P. Approximate matching & approximate reasoning in expert systems. *J of Hebei University* 13 (2), 1993. pp 54-59

[281] Sun Z. AI development and exploration to some AI issues. *J of Hebei University*, 14 (1) 1994. pp105-112.

[282] Sun Z, Xiao J. *Essentials of Discrete Mathematics, Problems and Solutions.* Baoding: Hebei University Press; 1994

[283] Sun Z. Soft computing and artificial intelligence. In: *Proc GCI' 96*, Wetzlar, Germany; 7-8 April; pp 31-35, 1996

[284] Sun Z. Fuzzy logic and artificial intelligence. Working paper. Dept. of Operations Research and Business Information Technology, RWTH Aachen Germany; 1996

[285] Sun, Z. and Weber K. Logic with Trick. In: Lasker GE (ed.): *Advances in Artificial Intelligence and Engineering Cybernetics*, Volume IV: Systems Logic & Neural Networks. The International Institute for Advanced Studies in Systems Research and Cybernetics; 1998. pp 44-49

[286] Sun Z, Finnie G. Applying intelligent agent technology to investment information brokering. In: *Proc Int Conf on Advanced Investment Technology.* Gold Coast, Australia; 19-21 December 1999. pp 341-351

[287] Sun Z, Finnie G. ES = MAS ?, In Shi, Z. Faltings, B. and Musen, M. (eds) *Proc Conf on Intelligent Information Processing (IIP 2000)* (within 16th World Computer Congress), Beijing, China; 21-25 August 2000. Electronics Industry Press; pp 541-48

[288] Sun Z. Towards understanding CBR and intelligent agents in e-commerce. Presentation at School of Information Technology, Bond University, 02 August 2000

[289] Sun Z, Finnie G, Weber K. Integration of abductive CBR and deductive CBR, In: *Proc 10th IEEE Int Conf on Fuzzy Systems (FUZZ-IEEE 2001)*, Melbourne, Australia; 2-5 December 2001. pp 1432-35

[290] Sun, Z. Parsimony of intelligence in multiagent systems. Seminar Presentation, School of Information Technology, Bond University, 01 August 2001

[291] Sun Z, Finnie G. Rule-based models for case retrievals. In: Baba N, Jain LC, Howlett RJ (eds): *Knowledge-based Intelligent Information Engineering Systems & Allied Technologies (KES'01). Frontiers in Artificial Intelligence and Application,* Vol. 69. Amsterdam: IOS Press; 2001. pp 1511-1515

[292] Sun Z, Finnie G. Multiagent brokerage with CBR. In: *Proc Australasian Conf on Information System (ACIS'01)*. Coffs Harbour, Australia; 4-7 December 2001. pp 635-644

[293] Sun Z, Finnie G, Weber K. Case base building based on similarity relations. Submitted to *Inform. Scie.* 2000; TR02-12. School of Information Technology, Bond University, 2002

[294] Sun Z, Finnie G. Intelligent techniques in e-commerce: A CBR approach. In: Faucher C, Jain L, Ichalkaranje N (eds): *Innovations in Knowledge Engineering*. Berlin: Springer-Verlag; 2002. in press

[295] Sun Z, Finnie G. Fuzzy rule-based models for case retrieval. *Int J of Engineering Intelligent Systems,* 2002. in press

[296] Sycara K. Machine learning for intelligent support of conflict resolution. *Decision Support Systems*, 10 1993. pp 121-136

[297] Tai H, Kosaka K. These Java mobile agents move comfortably throughout the Internet. *Comm of The ACM*, 42 (3) 1999. pp 100-101

[298] Tan YH, Thoen W. Toward a generic model of thrust for electronic commerce. *Int J of Electronic Commerce,* 5(2) 2001. pp 61-74

[299] Thuraisingham, B. *Web Data Management and Electronic Commerce*. Boca Ratton, Florida: CRC Press; 2000

[300] Timmers P. *Electronic Commerce: strategies and models for business-to-business trading*. John Wiley & Sons Ltd; 1999

[301] Tomasic A, Gravano L. Data structures for efficient broker implementation. *ACM Trans. on Inform. Sys.,* 15 (3) July 1997. pp 223-253

[302] Torasso P, Console L, Portinale L, Theseider D. On the role of abduction. *ACM Computing Surveys*, 27 (3) 1995. pp 353-355

[303] Trepper C. *E-Commerce Strategies*. Washington: Miscrosoft Press; 2000

[304] Turban E. Intelligent Agents in Electronic Commerce, Invited lecture at AIT'99 workshops, Gold Coast, Australia; 19 December 1999

[305] Turban E, Lee J, King D, Chung, HM. *Electronic Commerce, A Managerial Perspective*. Upper Saddle River, HJ: Prentice Hall; 2000

[306] Turing AM. Computing machinery and intelligence. *Mind*, 59, 1950. pp 433-460

[307] Uhrmacher AM, Tyschler P, Tyschler D. Modelling and simulation of mobile agents. *Future Generation Computer Systems*, 17 2000. pp 107-118

[308] Ullman JD, Widsom J. *A First Course in Database Systems*. New Jersey: Prentice Hall; 1997

[309] Ünal A. Electronic commerce and multi-enterprise supply/value/business chains. *Information Sciences* 127 2000. pp 63-68

[310] Varian HR. The computer-mediated economy. *Comm of the ACM*, 44 (3) 2001. pp 93

[311] Veloso M, Aamodt A (eds): *Case-Based Reasoning Research and Development. Proc ICCBR-95*, Sesimbra, Portugal, Oct. 1995. Springer; 1995

[312] Vetter M, Pitsch S. Towards a flexible trading process over the Internet. In: Dignum F, Sierra C (eds): *Agent Mediated Electronic Commerce: The European Agent Link Perspective*. LNAI 1991. Berlin: Springer-Verlag; 2001. pp 148-162

[313] Viamonte MJ, Ramos C. A model for an electronic market place. In: Dignum F, Sierra C (eds): *Agent Mediated Electronic Commerce: The European Agent Link Perspective*. LNAI 1991. Berlin: Springer-Verlag; 2001. pp 115-125

[314] Voss A. Towards a methodology for case adaptation. In: Wahlster W (ed.): *Proc 12th European Conf on Artificial Intelligence (ECAI'96)*, John Wiley and Sons; 1996. pp 147-51

[315] Watson I. An introduction to case-based reasoning. In: Waston I (ed.): *Progress in Case-based reasoning*. Berlin: Springer; 1995. pp 3-16

[316] Watson I (ed.): *Progress in Case-based reasoning*. Berlin: Springer; 1995

[317] Watson I. *Applying Case-Based Reasoning: Techniques for Enterprise Systems*. San Francisco, California: Morgan Kaufmann Publishers; 1997

[318] Watson I. CBR is a methodology not a technology. In: Miles R, Moulton M, Bramer M (eds): *Research & Development in Expert Systems* XV. London: Springer; 1998. pp 213-223

[319] Watson I, Gardingen D. A distributed case-based reasoning application for engineering sales support. In: *Proc 16th Int Joint Conf on Artificial Intelligence (IJCAI-99), Vol. 1*, Morgen Kaufmann Publishers; 1999. pp 600-605

[320] Watson I. Lessions learned during HVAC installation. URL: http://www.cs.auckland.ac.nz/~ian/papers/cbr/lessons.pdf. 2000

[321] Watson M. *Intelligent Java™ Applications for the Internet and Intranets*. San Francisco, California: Morgan Kaufmann Publishers; 1997

[322] Wayner P. *Agent Unleashed, A public domain look at agent technology*. Boston: AP Professional; 1995

[323] Weber K, Sun Z. Fuzzy stochastic dynamic programming for marketing decision support. *Int J of Intelligent Systems,* 15 (8) 2000. pp 763-783

[324] Weber K, Sun Z. Fuzzy stochastic multistage decision process with implicitly given termination time. In: Hampel R, Wagenknecht M, Chaker N (eds): Fuzzy Control Application - Theory and Practice. Heidelberg: Physica-Verlag, 2000. pp 318-328

[325] Weber K Sun J and Sun Z. A multiagent framework for integrating revenue-creating control processes. In: *Proc of AGIFORS Reservations & Yield Management Study Group Annual Meeting*, Berlin, Germany; 16-19 April; http://www.agifors.org/rymsg/ 2002

[326] Wei B, Pan Y. Multiagent dynamic coordination for collaborative design. *J of Computer Aided Design & Computer Graphics* (in Chinese), 11 (5) 1999. pp 474-476

[327] Weiss G. Adaptation and learning in multiagent systems: Some remarks and a bibliography. In: Weiss G, Sen S (eds): *LNAI 1042*, Berlin: Springer; 1996. pp 1-21

[328] Weiss G (ed.): *Multiagent Systems: A modern approach to Distributed Artificial Intelligence*. Cambridge, Massachusetts, USA: MIT Press; 1999

[329] Wenig RG http://members.aol.com/rgwenig/defknow.htm

[330] Wellman MP. A market-oriented programming environment and its application to distributed multicommodity flow problems. *J of Artificial Intelligence Research*. 1 August 1993. http://ic-www.arc.nasa.gov/ic/jair-www/volume1/wellman93a.ps

[331] Wess S, Althoff K, Richter MM (eds): *Topics in Case-Based Reasoning. 1st European Workshop on CBR (EWCBR-93).* Kaiserslautern, Germany; 1993, Springer-Verlag; 1994

[332] Whitaker B. Intelligent agents for electronic commerce. URL: http://www.ncr.com/journal/whit/ 1999

[333] Wierderhold G. Mediators in the architecture of future information systems. In: Huhns MN, Singh MP (eds)*: Readings in Agents, see* [125], pp 185-196

[334] Wilke W. Case-based reasoning and electronic commerce. Invited Talk at ICCBR97, 1997

[335] Wilke W, Lenz M, Wess S. Intelligent Sales Support with CBR. In: Lenz M, et al. (eds): *Case Based Reasoning Technology: from foundations to applications*. Berlin: Springer LNCS 1400. 1998. pp 91-113

[336] Winikoff M, Padgham L, Harland J. Simplifying development of intelligent agents. In: Stumptner M, Corbett D, Brooks M (eds): AI 2001: *Advances in Artificial Intelligence.* LNAI 2256. Berlin: Springer-Verlag; 2001. pp 557-568

[337] Wirth N. *Algorithms + Data Structures = Programs*. Prentice-Hall; 1976

[338] Wong D, Paciorek N, Moore D. Java-based mobile agents. *Comm of The ACM,* 42 (3) 1999. pp 92-102

[339] Wooldrige MJ, Jennings NR (eds): *Intelligent Agents*, *ECAI-94 workshop on agent theories, architectures, and languages*. Amsterdam, Ausgust 1994. Springer-Verlag; 1994

[340] Wooldridge M, Jennings, NR. Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10 (2) 1995. pp 115-152

[341] Wu X, Zou Y. *Techniques of Expert Systems* (in Chinese), Beijing: Electronic Industry Press, 1988

[342] Yager RR, Zadeh LA (eds): *Fuzzy Sets, Neural Networks, and Soft Computing*. Van Nostrand Reimhold; 1994

[343] Yager RR. Case based reasoning, fuzzy systems modelling and solution composition. In: Leake DB, Plaza E (eds): Case-based reasoning research and development (ICCBR-97) LNAI 1266, Berlin: Springer; 1997. pp 633-642

[344] Yager RR. Fuzzy methods in e-commerce. In: *Proc 18th Int Conf of the North American Fuzzy Information Processing Society*- NAFIPS'99, June 1999. pp 5-11

[345] Yager RR. Targeted e-commerce marketing using fuzzy intelligent agents. *IEEE Intelligent Systems and their Applications*, 15 (6) 2000. pp 42-45

[346] Yamamoto G, Nakamura Y. Architecture and performance evaluation of a massive multiagent system. In: Etzioni O, et al. (eds): *Proc of the 3rd Intl Conf. on Autonomous Agents*. ACM Press; 1999. pp 319-325

[347] Yen J. Fuzzy Logic-A modern perspective. *Trans. on knowl. & data engi.* 11 (1) Jan 1999

[348] Zadeh LA. Similarity relations and fuzzy orderings. *Inf Sci,* 3, 1971. pp 177-200

[349] Zadeh LA. Discussion: Probability theory and fuzzy logic are complementary rather than competitive. *Technometrics*, 37 (3) 1996. pp 271-276

[350] Zadeh LA. A new direction in ai--toward a computational theory of perceptions. URL: http://buffy.eecs.berkeley.edu/IRO/Summary/02abstracts/zadeh.4.html 2002

[351] Zak D. programming with Miscrosoft® Visual Basic® 6.0. Course Technology; 1999

[352] Zhang D, Wong W. Using CBR for adaptive negotiation. In: Shi Z, Faltings B, Musen M (eds): *Proc Conf on Intelligent Information Processing (IIP2000)*, Aug. 21-25, 2000 Beijing. pp 428-37

[353] Zhang L, et al. Research on Internet-based software agent discovering and gathering training resource information for courseware design. *J. Computer R & D*, 36 (4) 1999

[354] Zhang Z, Yang Q. Feature weight maintenance in case bases using introspective learning. *J of Intelligent Information Systems*, 16 2001 pp 95-116

[355] Zimmermann HJ. *Fuzzy Set Theory and its Application*. Boston/Dordrecht/London: Kluwer Academic Publishers; 1991

[356] Bylund M, Espinoza F. Testing and demonstrating context-aware services with Quake III, *Comm of The ACM*, 45 (1) 2002. pp 46-48

[357] Kaminka GA, Veloso MM, et al. GameBots: A flexible test bed for multiagent team research, *Comm of The ACM*, 45(1) 2002 pp 43-45

[358] http://www.cs.usyd.edu.au/jonpat/Information_Systems/IS_propaganda/IS_mission_statement.html