Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2020.010135
Article

Tech Science Press

Research on Agent-Based Economic Decision Model Systems

Chenxi Liu[1] and Suchun Yang[2,*]

[1]College of Economics and Management, Shandong University of Science and Technology, Qingdao, 266590, China
[2]School of Civil Engineering, Qingdao University of Technology, Qingdao, 266033, China
[*]Corresponding Author: Suchun Yang. Email: ysc116688668@gmail.com

Abstract: Based on an analysis of the development of economic decision support systems, agents are applied to construct intelligent economic decision support systems. This paper proposes a task-oriented agent design concept and designs multiple types of agents to complete the decision-making tasks with the task as the core. The structure of multi-agent based systems is provided, and the concrete realization structure of different types of agents in the system is also provided. Additionally, this study discusses the operational mechanism of the whole system and the cooperation between multiple agents in the system. Finally, these functions are implemented through a combination of VC++ 6.0, multi-threading technology and the expert system tool CLIPS. The economic decision support system combines complex system theory, decision theory, information collection, knowledge discovery technology, economic decision making and simulation technology. It can aid users in making decisions by using communication and cooperation between multiple agents.

Keywords: Decision support system; agent; multiagent system

1 Preface

The socioeconomic system is at its core a person-made system with influences from social, economic, educational, scientific, technological, and ecological environments and encompasses many human activities and many complex factors from the living environment. The fundamental difference between it and a physical system is that there are decision-making links in a social economic system, and people's subjective consciousness has a great impact on the system. Socioeconomics are influenced by many factors such as people, social issues, economics and the natural environment. Additionally, the socioeconomic system is a high-dimensional system with many variables and parameters, and it is affected by the external environment. There are many uncertainties in the system. Finding the solutions to many problems requires the participation of scientists from different fields. Therefore, studies of socioeconomic systems must introduce new ideas from modern systems science, comprehensively absorb the latest achievements from various disciplines, and make full use of computer simulation capabilities.

The decision-making issues confronted in a socioeconomic system are multifaceted. They can be roughly divided into two categories [1]; one category contains structural decision-making problems, and the other category contains semistructured and unstructured decision-making. Semistructured and unstructured problems are often encountered in socioeconomic decision-making. The resolution of such problems often relies on the knowledge and experience of experts and decision makers. The decision-maker's decision-making style also influences the decision-making results to a certain extent. To obtain a more just and objective decision-making plan, the establishment of an intelligent economic decision support system is a good way to study and solve semistructured and unstructured decision-making problems. General economic decision support systems [2] (DSSs) mainly use a quantitative method to model a problem and use the calculation results of numerical models to make decisions. However, these types of DSSs generally have slow knowledge acquisition problems and contain difficult expressions. In recent years, increasingly more artificial intelligence (AI) technologies have been applied to research on and the development of decision support systems (DSSs) in order to improve the processes and mechanisms of problem solving. Alireza et al. [3] ES is a mature field in AI, consisting of a knowledge base, an inference engine, and a database. It uses nonquantitative logic statements to express knowledge, and problem solving is done in an automated way. Combining ES and DSSs can create an intelligent decision support system (IDSS). On this basis, it is possible [4] to combine machine learning with a DSS to construct an IDSS. An IDSS uses a case-based learning strategy, artificial neural networks and other learning methods to make the system automatically acquire certain knowledge [5]. With the deepening of AI research, agent technology has become a key issue in the field of AI research; thus, there are agent-based IDSSs, and agents are increasingly used to construct an IDSS. Therefore, based on the original structure of an intelligent economic decision support system, a multiagent technology is proposed to develop an economic decision support system. Through an agent, theoretical methods, decision theory, information collection and knowledge discovery technology, macroeconomic forecasting and simulations of complex systems are closely combined. Software agents are used to construct economic decision support systems to help users perform economic analyses and make decisions.

2 Agent and Multi Agent Systems

The concept of an agent was originally formed in the field of distributed artificial intelligence, and although there is no unified definition, most researchers believe that [6] an agent is an active entity with knowledge, goals and abilities that can make inference decisions individually or under the guidance of a person. Knowledge is the certain understanding of the agent's description of its own environment and the required solution. An agent also has knowledge that is useful to the agent when interacting with users or other agents, such as knowledge about user characteristics. The goal is the degree to which an agent tries to make an achievement when solving a problem. A goal is achieved by completing a series of tasks. The agent may cooperate with other agents in order to complete the task. An ability is a problem-solving skill possessed by an agent. Agents can change the environment through their own abilities and can continuously learn from the environment in order to improve their ability to solve problems. In general, agents need to have certain characteristics, including activity, autonomy, interaction, sociality, responsiveness, durability, adaptability and even mobility. Some scholars have suggested that an agent should also have real-time capabilities, which are especially important to artificial intelligence researchers. In addition to the above characteristics, an agent should have anthropomorphic characteristics, such as the capability of exhibiting mental states such as knowledge, beliefs, intentions, commitments, and even emotions.

Usually, researchers study agents within the realm of multiagent systems [7]. An agent combines reasoning and knowledge representation and plays an important role in creating intelligent systems and simulating intelligent behavior [8]. A single agent is mainly used to simulate intelligent human behavior, while a multiagent system has the ultimate goal of simulating human social systems. Multiagent systems form through communication and cooperation between multiple agents [9]. Therefore, the real insight gained from researching agents is that they can constitute a multiagent system. A multiagent system (MAS) consisting of multiagents forms a unified and coordinated whole through mutual cooperation and cooperative communication between the agents. Their construction involves the analysis and discussion of issues such as mutual cooperation, collaboration, negotiation, and competition among multiple agents and the complex interaction patterns associated with them. Compared with a single agent, a MAS has the following characteristics [10]: Each member agent has incomplete information and problem solving abilities, there are no global controls, data are scattered or distributed, and the calculations are asynchronous, concurrent or parallel. Multiagents with different knowledge, goals, and abilities can solve problems beyond the single agent through communication and coordination.

At present, researchers attach great importance to studies of the theory, structure and language of agents. In terms of application, the use of an agent-based method to study decision support systems is still in its infancy. Poley et al. [11] proposed an agent-based decision support system and a life cycle method and established a framework for an agent-based decision support system. Wang [12] combined knowledge discovery, knowledge analysis and group decision support technology to construct an intelligent agent-assisted decision support system. Tung Bui of the University of Hawaii [13] proposed a framework for an agent-based decision support system. Based on the characteristics of the agent, the agent was classified, and different agents were designed to complete different decision tasks. The development of the whole system was divided into two stages, the micro level and the macro level. At the micro level, each agent concentrates on completing its specific tasks. At the macro level, there are considerations on how to implement cooperation between agents to support the whole process of decision making. Ching-Shen James DONG [14] and Grace SauLan LOO [15] used agents to construct a Web-based DSS. In the system, four kinds of agents, namely, a user interface agent, a user agent, a DSS component agent and a DSS component mapping agent, were designed. Hu Daiping used an agent to design and implement a macroeconomic decision support system. An application produced by Wang [16] of the City University of Hong Kong and other application-constrained negotiation agents built a DSS decision-making model.

3 System Design

3.1 System Structure

General intelligent economic decision-making systems apply knowledge engineering and rely on artificial intelligence principles and methods to assist in decision-making based on knowledge and reasoning. Knowledge engineering applications mainly solve problems related to knowledge representation, knowledge acquisition paths, knowledge reasoning and human-machine interactions in the decision system. The basic structure of an intelligent economic decision system [15] is shown in Fig. 1.

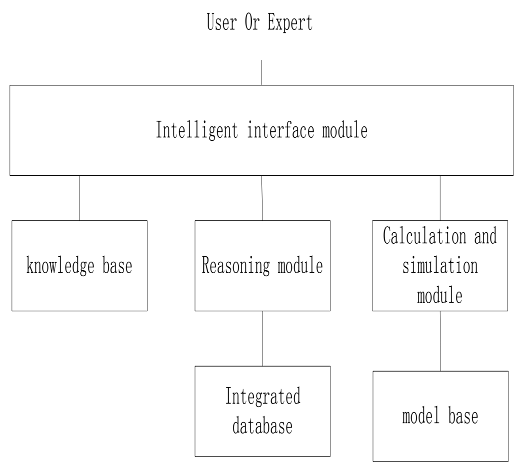

Figure 1: The basic structure of intelligent economic decision-making

The intelligent interface module is a vehicle for operators, administrators, experts, and decision makers to interact with the system. This module is responsible for inputting raw data and outputting analytical data, decision model solution results, and other information, as well as providing query services and accumulating knowledge in the knowledge base [17]. The reasoning module is the core of the system. It includes three submodules with various functions, which are exact problem solving, fuzzy decision support and complex problem solving. The exact problem solving module mainly solves structural decision problems; the fuzzy decision support module mainly focuses on uncertain, fuzzy and incomplete factors, performs inexact reasoning and problem solving, and generates a fuzzy decision model for the purpose of assisting in decision making [18]. The complex problem solving module is proposed to solve major problems in socioeconomic systems. It adopts methods such as prototype simplification to deal with complex problems. The computer simulation module simulates the strategic decisions of the socioeconomic system through quantitative calculations and the application of system dynamics and other models. The knowledge base is a collection of data that express information based on the characteristics of socioeconomic strategy issues, using production rules and fuzzy production rules. The integrated database stores relevant data for the socioeconomic system. Its management is the responsibility of the database management system. The model library stores a repository of strategic models, tactical models, and transactional operational models. The process of implementing decision-making assistance commences roughly as follows: First, a qualitative analysis is carried out using national policies, guidelines, and expert knowledge and experience extracted from the knowledge base. Then, the knowledge base, database, and model library are used to generate the inference model and to carry out quantitative analyses of intelligent reasoning and scheme simulations to complete the decision task.

In accordance with the basic structure of general intelligent economic decision-making, the structure of a multiagent economic decision support system using agent technology is shown in Fig. 2.

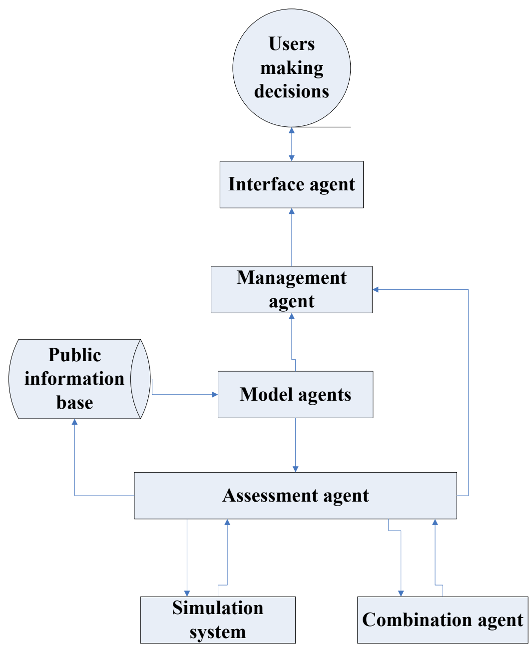

Figure 2: The multiagent decision support system structure

As shown in Fig. 2, the user interacts with the interface agent, which records and stores the user's identity in its own knowledge base. Then, the interface agent processes the user's questions and interfaces with the central control agent. The central control agent decomposes the problem into several subproblems, chooses the model agent needed to solve the subproblems, and receives a decision from the model agent based on its own knowledge. The decision results are then sent to the evaluation agent. The evaluation agent makes an evaluation according to specific principles, and then the decision-making scheme is simulated, and the simulation results are sent to the combination agent. A composite agent will receive a new decision-making scheme from several schemes obtained from the evaluation agent based on a specific combination method and send the scheme to the evaluation agent. After several evaluation cycles, a satisfactory solution is finally obtained and returned to the user.

To assist users in complex operations such as decision support, the multiagent economic decision support system designs a variety of agents with different functions to complete different subtasks. The agents are:

1) Interface Agent. The interface agent undertakes human-machine interface management, including customization of the human-machine interface under the different operating states of the system. On the one hand, the user makes a decision request to the system through the interface agent and inputs the relevant information according to the specific interface; on the other hand, the system also provides the user with visual decision conclusions and various types of auxiliary decision information through the interface agent or requests additional information from the user to complete the task.

2) Management Agent. The functions of the management agent mainly include ① management of other agents within the entire system, including the identification, type and function of various agents. ② The planning and coordination of tasks. User application problems are further decomposed according to requests from the interface agent, and a task is assigned to the relevant model agent. ③ The transfer model agent provides information, models and other support for problem-solving decisions. ④ Submission of the conclusions of the problem-solving agent to the interface agent.

3) Model Agent. After accepting a decision task, the model agent obtains the required information from the self-contained knowledge base and carries out the model's organizational reasoning through its own inference engine to generate the specific solution that the model requires. The solution is obtained by solving the model, and the result of the decision is sent to the evaluation agent.

4) Evaluation Agent. The evaluation agent performs evaluations according to certain principles and then simulates the decision plan and sends the simulation results to the combination agent.

5) Combination Agent. After the combination agent removes some decision-making schemes from the evaluation agent, it obtains a new decision-making scheme through a specific combination method and sends the scheme back to the evaluation agent.

In a multiagent system, each agent will communicate and cooperate with other agents. The cooperative agents in a multiagent system are divided into two categories. The first category includes agents that can cooperate with all other agents. The evaluation agent and management agent in this system belong to this type. The model agent, the combination agent, and the interface agent in this system belong to the second category only if they are associated with a certain class or a certain class of agents. The system also has a database and a public information base, and various types of agents have their own knowledge base.

3.2 The System Decision Process

The process of making decisions by the multiagent economic decision support system commences as follows:

1) The decision-making user first interacts with the interface agent. The interface agent records the user's identity and stores it in its own knowledge base. As the interaction agent interacts with the user step by step, the user attempts to solve the problem in the human-machine cooperation environment. In the process of interacting with the user, the interface agent can ascertain the specific understanding mode employed by the user and the problem-solving mode that the user likes and store and form simple user personalization;

2) The interface agent interacts with the management agent in an attempt to understand the problem, decomposes the original problem into multiple subproblems with the help of the management agent, and simultaneously activates the corresponding model agent;

3) After accepting a decision task, the model agent obtains the required information from its own model knowledge base and model method library and performs the model's organizational reasoning through its own inference engine to generate the specific solution model required. It solves the problem by executing the model and finally makes the decision. If it is necessary to cooperate with other model agents, the basic information from the model agent is obtained from the management agent. The cooperation request is sent through the communication component, and the cooperation result is obtained through the communication component. After obtaining the decision result, the result is sent to the evaluation agent;

4) The evaluation agent makes an evaluation according to certain principles and then simulates the decision plan and sends the simulation result to the combination agent;

5) After the combination agent removes some decision-making schemes from the evaluation agent, it obtains a new decision-making scheme through a specific combination method and sends the scheme to the evaluation agent;

6) The final decision plan is sent by the evaluation agent to the management agent, and the management agent submits the final decision plan to the interface agent and provides the user with a final visual decision through the interface agent.

The decision maker chooses the final decision scheme according to the decision format and the evaluation simulation results provided by the system and combines it with their own experience. If the user does not receive a satisfactory decision plan, a new round of decision-making will be performed with the new information until a satisfactory plan is obtained.

4. System Implementation

4.1 Agent Structure Design

In the process of designing the agent, there are some methods for designing the structure of the agent. The agent-centered basic model consists of an event processing system, a method set, and an internal state set. The event perceptron senses the external environment and reacts to it and to its internal state. The event processing system is the core part of the agent. The method set is the means by which the event can be completed. The internal state set represents the current state of the agent. The target-based agent design method, the target, the mental state, knowledge and other attributes, along with communication, reasoning decision, transaction processing, self-learning and interaction abilities, are means for the target to pursue the agent, and event processing is a means for the agent to achieve its goals. The various attributes and behaviors of the agent serve to provide the target service.

In the target-based approach to target design, the goal of the agent is to complete one or more tasks. The method used to complete the task is transaction processing. Therefore, the object-oriented agent design method and the event-based agent design method operate at different levels of problem analysis. The goal is a more abstract concept, and the realization of the goal requires a variety of tasks to be achieved. Event processing operates within a concrete implementation layer; the task is within the application layer, and the target is within the abstraction layer. In this sense, the object-oriented agent design is more suitable for the theoretical framework of agents, which can be used to guide the theoretical development of agents. The agent design method based on event processing is more suitable for the specific program implementation of agents, which involves completing some specific event processing step of the task. The application of the agents in various fields connects the two methods.

The object-oriented agent design method can be used to design the theoretical framework for agents, just as artificial intelligence research is a method of finding general problems from the beginning and is used to construct general programs. To date, the use of specific knowledge from narrower fields is the same as building very specialized programs. Since the agent is applied to different fields and the tasks vary among fields, the methods for dealing with the problems are different. Even the knowledge- and transaction-processing methods used in the different phases of the same task in the same field are different. Agents with the same structure may be easier to design and easier to operate but may not be suitable for specific applications. Because the unified structure should consider all situations, it will inevitably appear large and complete, but this is not the case for every situation, which will affect the performance of the agent and thus the performance of the entire agent-based system. The event-based design method uses event processing as the core of the agent, which limits the ability of the agent to a certain extent, because an agent needs to handle multiple events in order to complete a certain task.

In view of the above reasoning, this paper adopts the task-oriented agent design method and the agent's task as the core to design the agent structure. The managing agent manages the agent, the model agent mainly executes modeling and prediction operations, and the interactive agent remotely calls the model agent. The running modes of multiple agents are shown in Fig. 3.

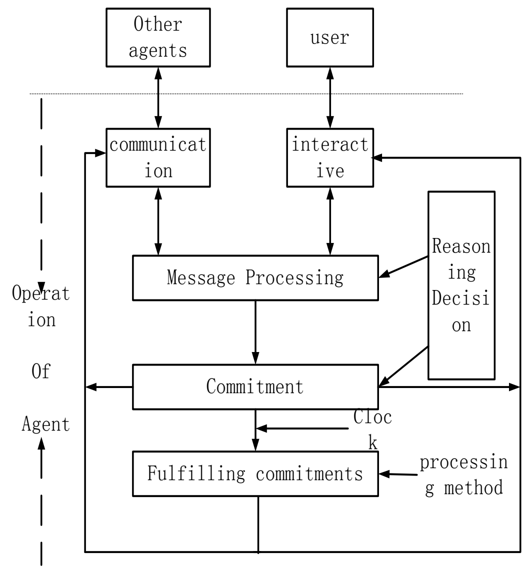

Figure 3: The operation mode of the agent

Agents in multiagent systems constantly receive messages from users or agents in other systems, process messages, make promises according to their own capabilities and statuses, notify the message senders of the promises made, and then use transaction-processing methods to execute expired promises to process transactions.

4.2 Agent Implementation

1) The interface agent continuously interacts with users during the whole system's running process to help the system understand user decision problems and ascertains how the users understand the problems. The interface agent has its own knowledge database (user feature information base and problem domain knowledge base). The user feature information repository stores the different hobbies of different users in order to understand and solve problems. To understand the problems provided by users and learn user characteristics, the interface agent ascertains the users' characteristics and preferences through a learning algorithm via the user feature information base. The knowledge base in the problem domain stores expertise on issues in a particular field. The functional components use different treatments for different problems faced by the same user by means of problem domain knowledge, which is responsible for correctly understanding the problem. The communication component is responsible for the information transfer of the agent. The specific structural design of the interface agent is shown in Fig. 4.

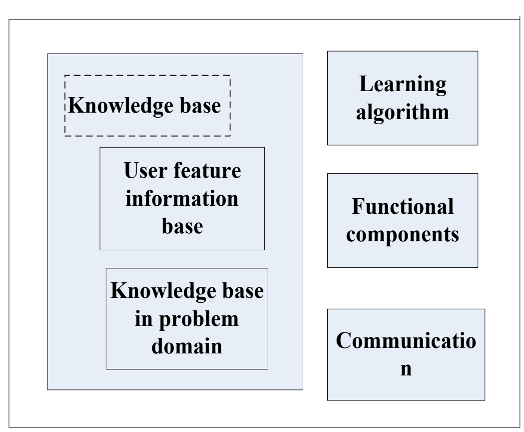

Figure 4: The structure diagram of the interface agent

The administrator is responsible for the management of all representatives of the system. Its tasks are to manage the agents in the system, to facilitate cooperation between the agents and to distribute the work. It has its own knowledge base, which mainly includes the knowledge of assignments and basic information on the different types of agents. The functional elements are primarily responsible for completing the addition, updating and deletion of the agents functions and the delegation functions. The specific structure of the management factor is shown in Fig. 5.

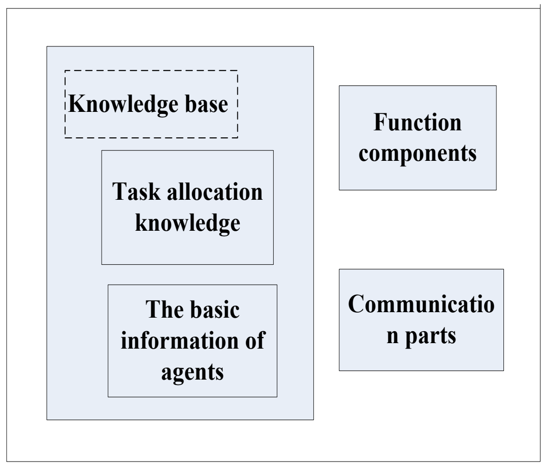

Figure 5: The structure diagram of the management agent

The model agent finds the solution to the problem or subproblem. The tasks are different because of the problems to be solved. The basic knowledge of the model is contained in the knowledge storage model. The model's method inventory can be used to solve the model when the model agent cooperates with other model agents to successfully solve the problem. The basic information for the model agent is obtained from the management agent and is stored in its partner information database, and the partner information database is first searched when cooperation is needed in the future. When the external environment changes, the learning algorithm learns and completes the model agent's revision of the target and knowledge together with the functional components. In addition to completing the revision of the target and knowledge, the functional component also completes the task of assigning the parameters required by the model agent. The specific structure of the model agent is shown in Fig. 6.

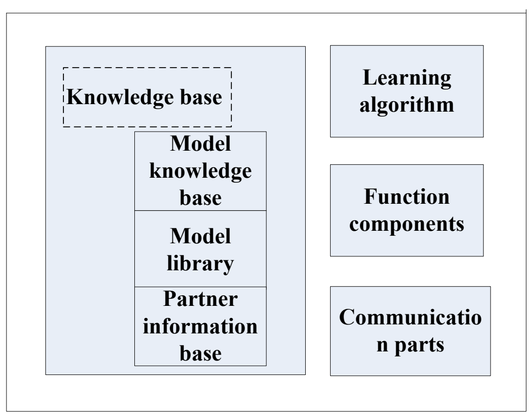

Figure 6: Structural diagram of the model agent

The evaluation agent performs preliminary analysis and evaluation of the results obtained by the model agent. Its tasks are to conduct a preliminary analysis of the results, obtain the decision-making plan from the results and evaluate the plan. Its knowledge base stores relevant decision-making knowledge for use in decision-making. The evaluation library stores effective evaluation criteria for the program or direct results, which is the basis for the evaluation of the results. The learning algorithm learns during the decision-making process, and the specific work of the functional components to complete the decision-making and evaluation. Its specific structure is shown in Fig. 7.

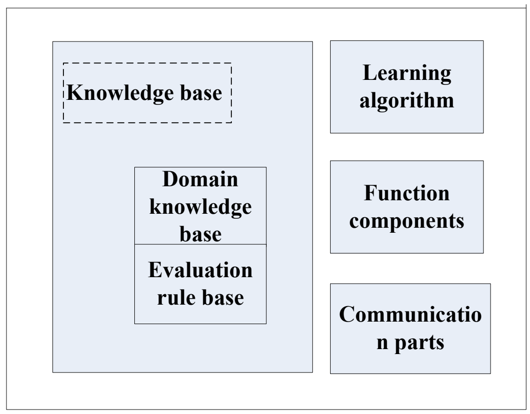

Figure 7: Structural diagram of the evaluation agent

The combination agent is responsible for the integration of the results. Its tasks are to comprehensively integrate the results of the evaluations of the evaluation agent and store the final result of the successful solution in the system itself. Its knowledge includes the integrated information base and the completed task library. The integrated information repository stores information about the result integration. All the successful problem solving and result information of the task library storage system is completed. The learning algorithm learns the experience of each successful integration. The functional component completes the integration of the specific results and the storage of the completed task information. Its specific structure is shown in Fig. 8.

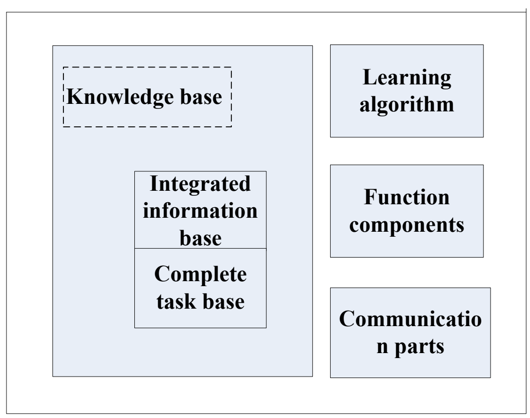

Figure 8: Structural diagram of the combination agent

In a multiagent system, there are cooperation modes among multiple agents. At present, the existing cooperation methods are put in place by the communication components between the agents. There are two main types of communication methods [19], namely, the blackboard mode and the message/dialog mode. Using the blackboard mode means that there is a blackboard in the system that all agents know about. Each agent writes the information that needs to be exchanged on the blackboard for other agents to read. The message/conversation mode causes a direct exchange of messages between two agents that need to cooperate. This is the interaction between the two agents. There is also a way of transmitting broadcast messages. Broadcasting is a form of communication in which a message is sent to a group of agents, which represents an interaction between more than two agents.

In the literature [20], it is proposed to use a cooperative agent that is responsible for the cooperation between multiple agents. Based on this concept, the cooperation between model agents is carried out. As mentioned earlier in this paper, the initial cooperation between the model agents is coordinated through the management agent. After successful cooperation, the cooperation records are saved. In the future, cooperation for similar situations requires only that the cooperation records be directly located. The communication between agents in the whole system occurs through direct message exchange.

4.3 Using Multithreading to Implement an Agent

SHOHAM [21] first proposed agent-oriented programming (AOP) and developed the corresponding language. However, the AOP language is far less mature and universal than object-oriented programming languages (such as C++ and Java). Generally, mature programming languages and techniques are often used when implementing agents. To this end, this paper proposes a method combining the VC++ platform and the expert system tool CLIPS to design and develop an agent running in a Windows NT environment. In multithreaded applications, all threads in a process are parallel. Because the communication between threads is easier than that between processes, multithreads are used to institute the parallel operations of all functional modules. In multiagent decision support systems, the threads include:

1) Main threads: Threads that automatically create and start user interfaces. An agent's human-machine interface is used to accept users' operations and display the necessary information.

2) On the one hand, communication threads are used to establish direct communication with other agents, and on the other hand, they are used to send and receive mailbox letters. Both communication methods use the KQML format language to transmit communication content.

3) Reasoning decision threads infer by calling the functions provided by the CLIPS Dynamic Link Library (DLL) and then make decisions with the agent.

4) Transaction threads are composed of transaction processing methods or models, which are created when needed.

5) Learning threads are composed of learning methods, which are created by the main threads when agents are learning.

These threads communicate with each other using the Post-ThreadMessage function. These threads can also create more threads while in operation.

Users can control these threads directly through an agent's user interface or indirectly control an agent's operation by means of communication through an interactive agent.

The database system uses the Access desktop database, and the database files store messages, mail, internal statuses, promises, the results of implementation and other elements. Then, the system uses BDE to run the database, uses TTable and TDBGrid to display the data list, and uses TDBChart to display the data curve. Every agent is created based on the objector's idea. The same type of agent is a class in VC ++. The inheritance and transformation of the class into an objector are used to carry out the inheritance and overload of the factor. Interagency communication takes place directly through the messaging mechanism in the Windows system. Behavior functions that can be performed in parallel within the factor are performed with the yarn as the basic unit. The Dynamic Link Library (DLL) provided by CLIPS is activated using the call mechanism of VC ++ and the Windows Dynamic Link Library.

5 Conclusion

Socioeconomic systems are important and typical complex systems. In the decision-making process, the system is influenced by many external factors and people's subjective consciousness. Therefore, to obtain correct and accurate decisions, it is necessary to establish an intelligent economic decision support system. An agent is an active entity with knowledge, goals and abilities that can make inference decisions independently or under human guidance. It combines reasoning with knowledge representation and plays an important role in creating intelligent systems and simulating intelligent behavior. Therefore, on the basis of the basic structure of general intelligent systems supporting economic decisions, this document builds an economic intelligent decision support system using agent technology, suggests a work-oriented agent design concept, and creates many different types of factors with tasks as the core. The specific structure and methods of application of different types of agents are documented. A system of support for multistakeholder economic decisions is proposed with the guidance of theory and methods that address complexity issues, and mature theories and methods are incorporated. Using a computer system, decision-making users are connected on the basis of expert knowledge through an agent, thus achieving an interaction between a multidisciplinary theoretical approach and the knowledge and experience of experts and computers.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] T. Logenthiran, D. Srinivasan and A. M. Khambadkone, "Multi-agent system for energy resource scheduling of integrated microgrids in a distributed system," *Electric Power Systems Research*, vol. 81, no. 1, pp. 138–148, 2011.

[2] J. E. Hernández, J. Mula, R. Poler and A. C. Lyons, "Collaborative planning in multi-tier supply chains supported by a negotiation-based mechanism and multi-agent system," *Group Decision & Negotiation*, vol. 23, no. 2, pp. 235–269, 2014.

[3] J. Alireza, A. M. Masrah Azrifah, S. Nasir and S. Hasan, "A multi-agent supply chain recommendation and negotiation framework," *Advanced Materials Research*, vol. 716, pp. 527–532, 2013.

[4] R. J. Valdivieso-Sarabia, J. M. García-Chamizo and M. Nieto-Hidalgo, "Distributed power management system with dynamic load management based on multi-agent system for smart grid," *Ubiquitous Computing and Ambient Intelligence, Personalisation and User Adapted Services,* vol. 8867, pp. 349–356, 2014.

[5] J. Lagorse, D. Paire and A. Miraoui, "A multi-agent system for energy management of distributed power sources," *Renewable Energy*, vol. 35, no. 1, pp. 174-182, 2010.

[6] Q. B. Le, S. J. Park, P. L. G. Vlek and A. B. Cremers, "Land-use dynamic simulator (ludas): A multi-agent system model for simulating spatio-temporal dynamics of coupled human–landscape system. i. structure and theoretical specification," *Ecological Informatics*, vol. 3, no. 2, pp. 135–153, 2008.

[7] E. L. Karfopoulos and N. D. Hatziargyriou, "A multi-agent system for controlled charging of a large population of electric vehicles," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1196–1204, 2013.

[8] H. Lee, P. Mihailescu and J. W. Shepherdson, "Anomaly management scheme for a multi-agent system," 2012.

[9] J. J. Gomezsanz, R. Fuentes, J. Pavón and I. Garcíamagariño, "INGENIAS development kit: A visual multi-agent system development environment," in *Proc. of the Seventh AAMAS*, pp. 1675–1676, 2008.

[10] Q. B. Le, S. J. Park and P. L. G. Vlek, "Land use dynamic simulator (ludas): A multi-agent system model for simulating spatio-temporal dynamics of coupled human–landscape system: 2. scenario-based application for impact assessment of land-use policies," *Ecological Informatics*, vol. 5, no. 3, pp. 203–221, 2010.

[11] M. J. Poley, K. I. Edelenbos, M. Mosseveld, M. A. M. V. Wijk, P. M. H. Maureen *et al.,* "Cost consequences of implementing an electronic decision support system for ordering laboratory tests in primary care: Evidence from a controlled prospective study in the Netherlands," *Clinical Chemistry*, vol. 53, no. 2, pp. 213–219, 2007.

[12] M. Rashidi, "Decision support system for remediation of concrete bridges," *Genome Research*, vol. 18, no. 5, pp. 830–838, 2013.

[13] S. Y. Chou and Y. H. Chang, "A decision support system for supplier selection based on a strategy-aligned fuzzy smart approach," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2241–2253, 2008.

[14] U. Cebeci, "Fuzzy ahp-based decision support system for selecting erp systems in textile industry by using balanced scorecard," *Expert Systems with Applications*, vol. 36, no. 5, pp. 8900–8909, 2009.

[15] J. Ma, J. Lu and G. Zhang, "Decider: A fuzzy multi-criteria group decision support system," *Knowledge-Based Systems*, vol. 23, no. 1, pp. 23–31, 2010.

[16] P. Jankowski and L. Richard, "Integration of gis-based suitability analysis and multicriteria evaluation in a spatial decision support system for route selection," *Environment and Planning B: Planning and Design*, vol. 21, no. 3, pp. 323–340, 1994.

[17] J. Perez, F. J. Cabrerizo and E. Herrera-Viedma, "A mobile decision support system for dynamic group decision-making problems," *IEEE Transactions on Systems Man and Cybernetics-Part A Systems and Humans*, vol. 40, no. 6, pp. 1244–1256, 2010.

[18] Y. K. Juan, P. Gao and J. Wang, "A hybrid decision support system for sustainable office building renovation and energy performance improvement," *Energy and Buildings*, vol. 42, no. 3, pp. 290–297, 2010.

[19] A. Phdungsilp, "Integrated energy and carbon modeling with a decision support system: Policy scenarios for low-carbon city development in Bangkok," *Energy Policy*, vol. 38, no. 9, pp. 4808-4817, 2010.

[20] V. Mijovic, N. Tomasevi, V. Janev, M. Stanojevic and S. Vranes, "Ontology enabled decision support system for emergency management at airports," in *Proc. of the 7th International Conference on Semantic Systems*, pp. 163–166, 2011.

[21] T. Zhang, G. Zhang, J. Ma and J. Lu, "Power distribution system planning evaluation by a fuzzy multi-criteria group decision support system," *International Journal of Computational Intelligence Systems*, vol. 3, no. 4, pp. 474–485, 2010.