Sub-section 3. Informatics, computer science and automation

Golub O. T.
Senior teacher, National Technical University of Ukraine "Kyiv Polytechnic Institute"

METHODS OF KNOWLEDGE REPRESENTATION IN DECISION SUPPORT SYSTEMS FOR INTELLECTUAL GAMES

Nowadays intellectual games have found their application in such areas as economy, sociology, politics etc. Urgency of decision support systems (DSS) for intellectual games is in giving the user of game environment an instrument which let him digress from the computation mechanics in decision support, concentrate at strategy realization, decrease an amount of unrecorded factors. It refines the made decisions and reduces the player loading. The aim of the article is a realization of modern information systems design patterns and software design technologies in DMSS design for intellectual games.

To attain the aim the analyses of features of intellectual game information interaction, the definition of model of knowledge representation in a system, the modeling of current situation analyses and direct decision support should be made. A model of retrospective analysis the task of which is in statistic data processing, adjustment of relevant and creation of new knowledge should also be designed.

Features of intellectual games information interaction.

Intellectual game is a game where players employ their intelligence and/or knowledge. We mean the games in which a player employs his/her analytical ability. Knowledge needed for playing a player obtains from rules and his/her own playing experience. Such games absolutely differ from those in which a player should simply answer some questions. The thing is that players in such games compete in ability to find a right outlet in a dynamic situation and not only check their knowledge. The examples of such games are a psychological team game Mafia or the game Sport Poker which has a wide strategic potential and a psychological situation analyses.

A specificity of intellectual games is that they act real-time. Each game has certain rules which efficiently determine an action sequence of playing parties if events develop incrementally, or limit only a diversity of pending actions of players and specify effect of interaction with the system without affecting a sequence and procedure of players. Technological side of event tracking is easily implemented by technology EventListener, which checks the system on beginning of according event and if an event appears a business-logics of processing occurs. And example of successful realization of it is Parsley Framework and Adobe Flex 4.0 [1]. Such combination of technologies makes it possible to create a connection between a source and an events handler even in a case of many to many interaction.

While interacting with a playing area in some moment a player has a big amount of possible influences on the area and such its components as other players. An influence is not limited by a certain atomic operation and in some circumstances combines factors of several actions in a time or a certain sequence of actions. Such sets of processes of interaction are more effective then common sequences of actions appearing and sometimes even provoke some chain processes.

Accounting different influences caused by a player it should also be mentioned

a planning of area objects. For instance a model of a player is one of the system essences. It represents a set of attributes and facts that characterize a player, possible methods of influence at playing area and its feedback. According to a certain situation and a running status of a player its model characteristics differ. The same happens with possible actions of a player itself: a set of possibilities widens, present possibilities differ, some of them become inaccessible, the other elements of the system become diverse. Implementation of models of such essences easily happens in practice by an object oriented principle of design the most known of which are the technologies .Net and Java [2].

In modern games one of the models of knowledge representation which have some advantages as well as disadvantages are usually used. For more effective realization of the game and approaching it to reality, we propose a combined model of knowledge representation in DMSS.

Let's consider the most accepted models of knowledge representation. The *System of products* is an easily used model. It is used as a mechanism of formal logic as well as a part of different areas of informal logic. But while the number of rules grows, the ties structure complexification occurs and the probability of antagonisms appears. To make an instrument of tracing and solving them is an uncommon task. It's reasonable to use the model in the vein of intellectual game rules organization which a priori has no contradictions as they are constructed without being two-valued. The model is also used for creating the individual rules of a player which depict the strategy of his behavior in some terms.

The *Semantic network* has a possibility of data representation in a close to natural way. It has a broad spectrum of ties between atomic data, improves the tracing of cause-effect relations. But a disassembling exaggerates the work with multiplex essences. An appropriate area of the model resort is a multiplex interaction inside the system while describing direct user actions, feedback, ability to work. The model of knowledge representation is built on the base of semantic network by broadening its possibilities of representing the essences. Semantic networks have a unique possibility of tracing the active actions packages connected to some object. That is why while making an analysis of the data given by package UserTracking which fixes the actions of players we can use adaptive models of probability estimation which are an important component of the model of current and strategic decision support in turn.

The Frames are exploited specially for maintaining and handling the uncommon complex objects. By their nature they are similar to the classes in object-oriented paradigm of software development which is now the most effective method of development of complex system and of their embodiment in a software code. That is why the structure of frame systems is clear for software specialists. The mentioned characteristics of objects representation is completed by knowledge model. The combining of multiplex essences which have precise structure of advanced ties mechanisms let us construct the model of knowledge representation similar to the structure of natural language and let us implement the mechanisms of accurate and deep analyses of informational interaction. The designing of such models for certain knowledge domain is quite difficult in practice as a specialist has to build into the system a specificity which may be unseen at once. Gaming images are interpreted intuitively by a person while a computer has only data and no intelligence. That is why the intellectualization of the system depends on the skills of its developers, notably on the efficiency of game designing, realization of its components and informational interaction

between them.

Taking into account weak sides of each model of knowledge representation it is reasonable to exploit a combined model which is an integration of product system for making rules, frames for constructing multiplex essences, semantic networks for representing ties and interaction between the essences [3].

The *combined model* can be described like this. Objects like the model of player, elements and rules of playing environment, events which they design are separate essences or attributes of the essences. Possible ties are a direct action and a back action of interacting essences.

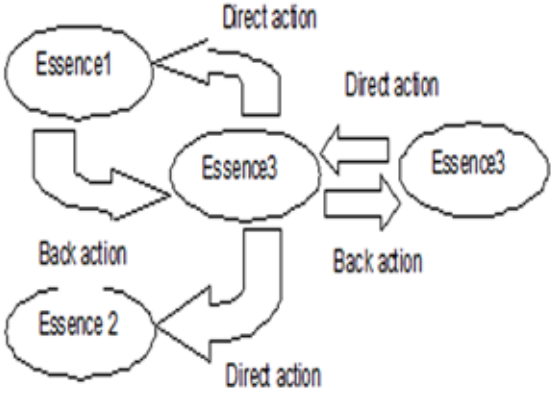

Figure 1. Combined model

Conclusions

The article is devoted to the analyses of specificity of informational interaction in intellectual games. At the base of the analyses it is built a combined model of knowledge representation which adequately depicts the characteristics of the system of decision support for intellectual games and is easy-to-use in a software code.

References

1. Efraim Turban, Jay E. Aronson, Ting-Peng Liang Decision Support Systems and Intelligent Systems (7th Edition). - 2004. – 368 p.
2. Rajendra Akerkar, Priti Sajja Knowledge-Based Systems. -arXiv:0911.4729v1 24 2009. - p.1-18.
3. Gordon S., Novak Jr. TMYCIN System Expert Tool. – 1996. – 217 p.