# СЕКЦИЯ 3
# ЗАЩИТА ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И СЕТЕЙ

*УДК 007.681.512.2*

## TECHNOLOGY OF ADAPTIVE INTELLIGENCE MULTIAGENT PROCESSING FOR INFORMATION DEFENSE SYSTEM DESIGN

U.A. VISHNIAKOU, A.H. Al-MASRI, S.K. Al-HAJI

*Belarusian state university of informatics and radioelectronics, Republic of Belarus*

*Submitted 18 March 2019*

**Abstract.** The tendencies of using multi-agent intelligent technologies for information processing are given. The main ideas of building a distributed multi-agent system with distributed knowledge and distributed processing are shown. The directions of multi-agent intelligent systems utilizationin information defense using cloud technologies are presented. Knowledge representation network in the form of object models is proposed, on basic which the intelligent MAS should be constructed. The approaches for adaptive intelligent multi-agent technology in area of information security are introduced.

*Keywords:* multi-agent technologies, distributed knowledge bases, distributed decision-making, cloud environment.

### Introduction

One of the main problems of innovative economy building is intellectualization, the essence of which is to develop effective mechanisms for the formation, publication, updating and mass use of innovative knowledge in management technologies. Among such knowledge are allocated developments in the field of intellectual agents on the basis of semantic Web, Web services and semantic Web services, cloud computing, blockchain technologies [1].

The technology of development and use of multi-agent systems (MAS) and multi-agent management (MAM) is understood like under multi-agent technology. The problems of control and distributed interaction in networks of dynamic systems attract the attention of a large number of researchers. This is due to the widespread use of multi-agent systems in different areas, including automatic adjustment of parameters of neural recognition networks, transport management, distributed sensor networks, control in communication networks, interaction of UAV groups, management of mobile robots, protection of information resources, etc. Distributed MAS are used that perform actions in parallel, for which the task is dividing on parts between several computational threads. Such problems arise not only in computer networks, but also in production networks, service networks, transport and logistics networks. With natural constraints on communication, decentralized strategies are able to effectively solve this type of problem [2, 3].

### The basic of MAC

Multi-agent systems originated at the intersection of system theory and distributed artificial intelligence. Open, active, developing systems are discussed, in which attention is paid to the processes of agents interaction for building systems with new qualities. MAS are built as a union of individual intelligent systems based on knowledge [3]. MAS consists of the following components: a set of agents working with objects; variety of tasks; a space in which there are agents and objects; the set of relations

between agents; many agent actions (operations on objects). Agent management system (AMS) is also an agent that controls access and use of the agent platform [2].

The basis of the organization form of interaction between agents characterized by the combination of their efforts to achieve the goal in the division between their functions, roles and responsibilities is cooperation (C). This can be determined:

K = cooperation + coordination + communication.

It's means the management of the associations between actions under the coordination. Communication between agents depends on the chosen protocol, which is a set of rules that determine how to synthesize meaningful and correct messages.

In the MAC architecture, the main part is the domain-independent core, which includes such components: direct access service (provides direct access to the attributes of agents); message service is responsible for the transmission of messages between agents and kernel systems; agent class library (part of the database) contains the classification of agents in the MAS; agents community, where agents are located (this block provides functions for loading/writing agents and their properties and optimizes the work of agents with resources); ontology is a subject knowledge base containing specific knowledge about objects and environment of functioning, represented in the form of a corresponding semantic network [2].

## The agent structure and their use

The basis of agent structure is the context, or server environment, in which it is executed. Each agent has a fixed identifier-name. In a server environment, you can run not only the source agent, but also a copy of it. Agents are able to create their own copies, sending them to different servers for execution. When the agent arrives on the next server, its code and data are transferred to the new context and erased at the previous location. In the new context the agent can do anything that is not prohibited there. Upon completion of the work in the context the agent may send itself in a different context or upload sender address. Agents can also shut down themselves or at the command of the server, which then moves them from the context to the storage location.

The structure of a typical agent includes inputs (internal parameters of the agent and data on the state of the environment), outputs (parameters affecting the environment and informing the user about the state of the environment and decisions made), the solver – the decision-making procedure. The solver can be a fairly simple algorithm or an element of an artificial intelligence system [3].

Multi-agent systems are used for the development of information and industrial systems. In industry the MAS are used to the solution of management automation of complex systems, for the collection and processing of information in games. Multi-agent technologies are applicable in the management of mobile resources, as well as in such areas as object design, industrial production [2].

In sources [4, 5] MAS deals with the application of automate the construction of intelligent knowledge bases and problem solvers. The resulting model of hybrid knowledge bases, which ensures the compatibility of present knowledge and can be represented in knowledge bases multi-level meta knowledge, to structure the knowledge base according to various criteria and to apply components of knowledge bases again [4]. The agent-oriented model of the hybrid solver allows to build variety of MAS: for production, customer service, construction design [5].

## Design of MAS

The general methodology of the ascending evolutionary design of MAC can be represented by a chain: <environment – functions OF Mac – role of agents – relations between agents – basic structures of MAC-modification>. It includes the stages of: formulation of purpose (objectives of development) MAC; the identification of core and support functions of agents; clarify the composition of agents and the distribution of tasks among agents, the choice of the architecture of the agents; the provision of basic relationships between agents; determination of possible actions (operations) agents; analysis of real-life, real or anticipated changes in the environment. When designing, the organization of agents can be considered as a set of roles that are in a certain relationship with each other and interact with each other [2].

MAS bottom-up design methodology requires a preliminary task of the initial functions, determining the range of their obligations to each other, the formation of the initial structures and its developing on the basis of the allocated functions and the study of the adequacy of these structures to the nature of the tasks in the selected problem areas.

The technique of top-down design is to determine the social characteristics of MAS on a set of criteria, the construction of the basic types of their organizations, followed by the definition of requirements for the architecture of agents. For artificial social systems and communities, a top-down approach to organizational design is put forward [3].

Agents can be integrated into cloud computing (CC) structures that contain specific functions for problem solving, data processing, and management. They support a natural mix of knowledge-based information and technology and can support the process of logical reasoning (for example, including business regulations). They enable learning and self-improvement at both the infrastructure level (adaptive routing) and the application level (adaptive user interfaces) [6, 7].

There are several international approaches to creating a MAS, the most famous of them are [2]: MASIF (Object Management Group), which is based on the concept of a mobile agent; FIPA (Foundations for Intelligent Physical Agents) specifications based on the intelligence of the agent, as well as standards developed by the research subsection Defense Advanced Research Projects Agency (DARPA), in particular Control of Agent Based Systems. Regarding mobility and intelligence of agents, most experts agree that mobility is the Central characteristic of the agent, intelligence is desirable, but not always strictly required [2, 3].

FIPA's activities include joint research and development by its members of international specifications that will maximize the interaction between agent applications, services and equipment. FIPA specifications focus on enabling intelligent agent communication through standardized agent communication and content languages. Along with the General basics of communication, FIPA also specializes in ontology and negotiation protocols to support interaction in specific application areas (transport support, production, multimedia, networking) [2].

The OMG MASIF standard creates conditions for the migration of mobile agents between MAS via standardized CORBA IDL interfaces. DARPA initiated the work on the distribution of Knowledge Sharing Effort, as a result of which the agent programming languages were divided into syntax, semantics and pragmatics: language KIF (Knowledge Interchange Format) – syntax; Ontolingua – language for defining shared ontology's (semantics); KQML (Knowledge Query and Manipulation Language) – a high-level interaction language (pragmatics). When you create a MAS is also used language of communication between agents – Agent Communication Language (ACL) that specifies the types of messages agents, the content and the ontology. Cooperation between agents is achieved through a set of basic concepts used in communications. The ontology is used as the Application Programming Interface and defines the agent interface.

Within the framework of design MAC direction, architectures, models and software prototypes of several MAS were developed: attack modeling, intrusion detection, intrusion detection training, etc. [8].

## Knowledge representation network in the form of object models

Introduction of concepts of objects-carriers and objects-communicators, structure, operations over them, the recursive description and levels of abstraction allows to define the abstract machine of objects on the basis of which it is possible to create this or that object computing environment with software or hardware implementation. This will create effective implementations for information management and security [7]. Usually the knowledge base (KB) is a set of facts and rules that can be changed during processing. Consider the KB represented by a set of special *n*-objects [9].

By definition, an *n*-object is a pair of $<A_i, f_{i1}>$ two *n*-components. Next, we will use only those objects in which $A = \{0, 1\}$ and $f_{ij}$ is a predicate. Each predicate defines a set of objects such as $P_n$ ($n \geq 2$). Each object in the $P_n$ set has an identifier $U$ denoting its class (class) and name (name), i. e. $U <A_1, \ldots A_n>$ and $<f_{i1}, \ldots f_{in}>$.

Each $f_{ji}$ function from an *n*-object will be treated as a pair of $f_{ij}\ f_{ij}$*, where $f_{ij}$ is an access function for a set of $A_i$ elements, and $f_{ij}$* is a function that assigns a value to the argument xi in the next step of the operation. Suppose that all $f_{ij}$ functions (predicates) in a given n-object depend on one set of arguments $\{x_1, x_2 \ldots x_m\}$. This assumption allows us to call the considered n-object set $P_n$,

with the above properties, the knowledge base, if for any value of the argument only one predicate from the set of predicates is true for each set of argument values. Processing of the object knowledge base (OKB) is performed in the following way:

1. It must set the original values of the $x_1$ argument ... $x_n$. This locks the object in which $f_{i1}$ ($x_1$, $x_2$... $x_n$) = 1. At this step, facts from set $A_2$ ... $A_n$ can be used.

2. For each object all values of $f_{ij}$ functions (predicates) are calculated (for all $j \geq 2$), which generate values $x_1$, $x_2$... $x_n$ in the next step.

The object-logical model of knowledge representation has two levels:

– the first level includes an object-oriented language of knowledge description;

– the second level is a system that includes an object base (OKB), an object network and a logical subsystem. OKB has descriptions of all classes of objects. An object-logical network combines the capabilities of functional and semantic networks, and can include objects and relationships for some application area.

Classes, name, value, or set of values are connected to each network object. Relations / functions calculate or correct the values of their arguments and try to join object nodes, otherwise object nodes, provided that the arguments are identical.

Such local processes will be initiated (in other words, will start, will run) at any change of object nodes and will be executed until the complete stabilization of the network is achieved. Many rules are connected to any node of the object network. These rules are triggered when the object node is modified.

## Problem solving with object networks

The following problems can be formulated and solved on the basis of the presented object-logical network, using knowledge from monograph [9]:

1. Design objects with the required properties, characteristics, constraints, and so on, using the original objects from the BR;

2. Test the object and its components to find possible failures in areas such as computers, telecommunication system and etc.;

3. To control complex objects;

4. To simulate a certain action in the technical and social systems;

5. To carry out examination of some problems and to make a decision in the conditions of incomplete or contradictory information.

Addressing these issues will include the following steps:

1. Identifying the problem when using the high-level object logic language (HOLL);

2. Translation of the applied description of the problem into an internal form and creation (on its basis) of a working network of objects using OKB.

3. Transform the resulting object-logical network by calculating the appropriate algorithm.

## Adaptive intelligent MAC concept

The report presents the following solutions for adaptive intelligent multi-agent information security technology (AI MAT ID):

− the subsystem of analysis of information security of the protected object, implementing the audit procedures for the state of protection of the object;

− attack detection system, which includes agents of workstations, servers, routers and subnets and allows to draw a conclusion about the presence and variety of attacks;

− dynamic knowledge base of is agents, including ontologies, formal and heuristic rules, BS interface;

− the method of making a joint decision by agents, which allows to assess the state of information security of the protected object on the basis of a dynamic knowledge base formed from various sources;

− multi-agent attack counter subsystem, which allows to accumulate knowledge and train a multi-agent system for further detection of new threats;

− subsystem for evaluating the effectiveness of the proposed methods of protection.

The architecture of MAS is built using this technology and includes a knowledge base in the form of rules of production, the mechanism of inference, receptors and effectors of the agent, the module of communication with other agents. In relation to the task of security analysis, agents transmit facts about external influences to the knowledge base. As a result of the logical output, a solution is developed, which is transmitted to the processor about changes in the external environment. Different types of agents can be used for distributed problem solving: agent-subordinator, multiple agents of executors, agent-integrator. Agents can be interconnected as a multi-level architecture, which can be horizontal or vertical.

## Conclusion

1. The direction of intelligent MAS construction the further development of models, methods, architecture and software to solve the problem of adaptive information security system design.

2. Knowledge representation network in the form of object models is proposed, on basic which the intelligent MAS shell be construct.

3. The approaches for adaptive intelligent multi-agent technology (AI MAT) in information security area are introduced.

## References

1. Vishniakou U.A. Information management and security: methods, models, software and hardware solutions. Monograph. Minsk. MIU, 2014.
2. Leyton-Brown K, Shoham. Y Multiagent Systems: Algorithmic, Game-Theoretic and Logical Foundations London: Cambridge University Press. 2009.
3. Rzevski G. // Proc. of 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD2012). Kyoto, Japan, August 8–10. 2012. P. 434–437.
4. Davydenko I.T. Models, methods and means of development of generated knowledge bases on the basis of semantic compatibility of reusable components: autoref. of PhD thesis in tech. sciences Minsk, 2018.
5. Shunkevich D.V. Agent-oriented problem solvers of intelligent systems component: autoref. of PhD thesis in tech. sciences Minsk, 2018.
6. Vishniakou U.A. Information security in corporate systems, electronic commerce and cloud computing: methods, models, hard-software tools Minsk, Bestprint, 2016.
7. Fingar P. Cloud computing – the business platform of the XXI century M.: Aquamarine Book, 2011.
8. Kotenko I.V., Yusupov R.M. // Bulletin of the Russian Academy of Sciences. 2007. Vol. 77. No. 4. P. 323–333.
9. Vishnyakou U. A., German O.V. Models and Tools of Logical Inference Systems. Moscow: Nauka, 1999.