AGENT BASE APPROACH FOR INTELLIGENT DISTRIBUTTION CONTROL SYSTEMS

Mohammad Sepehri

Saint-Petersburg State Polytechnic University
Distributed Intelligent Systems Dept
Grazhdansky Street, 28
195220 St. Petersburg, Russia
E-mail address: Mo.Sepehri@IKCO.com

Abstract

Global competition and rapidly changing customer requirements are forcing major changes in the production styles, planning, services, configuration of manufacturing organizations and many other areas.
Agents and multi-agent systems are becoming a new way to analyze, design and implement complex (software) systems, specifically, when the design problem is distributed in nature, the development of a solution may benefit from an agent-based approach.
Recently, agent technology has been considered as an important approach for developing industrial distributed systems. Distributed Intelligent Multi-Agent Systems offer modular, flexible, scalable and generals able algorithms and systems solutions for information retrieval, extraction, fusion, and data-driven knowledge discovery using heterogeneous, distributed data and knowledge sources in information rich, open environments. Such systems consist of multiple interacting intelligent software agents.

CONTENTS

1. INTRODUCTION AND CONCEPT DEFINITION 2

2. OVERVIEW MAS TECHNOLOGY 4

3. AGENT BASE APPROACH IN SOCIAL SERVICES IN RUSSIA 7

4. AGENT BASE APPROACH IN MANUFACTURING CONTROL 10

5. CONCLUSIONS 14

REFERENCES 15

1. INTRODUCTION AND CONCEPT DEFINITION

1.1 Agent is a computer system capable of autonomous (capable of acting independently, exhibiting control over their internal state) action in some environment in order to meet its design objectives.

1.2 Intelligent Agents is a computer system capable of flexible autonomous action in some environment.

- **ability to adapt :** problem-solving-rules or algorithms
- **ability to learn :** trial-and-error example and generalization

1.3 Software Agent is a functional software module that is able to execute some predefined class of external tasks and has autonomy during these tasks realization. It reacts on the predefined states of its environment according to acquitted information, its own build-in preferences, and knowledge.

1.4 Agent-Oriented Programming is an approach to building agents, which proposes programming them in terms of mentalist notions such as beliefs, desire and intentions. [M.Wooldridge, 1995]

1.5 Agent-Based Programming is an approach to the building software systems using various agent frames as basic functional components of the designed system architecture (so called MAS architecture).

1.6 Autonomous agent is a system situated within and a part of an environment that senses that environment and acts in it, over time, in pursuit of its own agenda and so as to effect what it senses in the future. [Franklin and Graesser, 1995]

1.7 Intelligent Agents in AI, In the artificial intelligence sense of the term, there are multiple types of agents and sub-agents. **Physical Agents and Temporal Agents .**It is possible to group agents into five classes based on their degree of perceived intelligence and capability:

1.7.1 Simple reflex agents; acts only on the basis of the current percept. The agent function is based on the condition-action rule: If condition then action rules this agent function only succeeds when the environment is fully observable.

1.7.2 Model-based reflex agents; can handle partially observable environments. Its current state is stored inside the agent maintaining some kind of structure which describes the part of the world which cannot be seen. This behavior requires information on how the world behaves and works. This additional information completes the “World View” model.

1.7.3 Goal-based agents; are model-based agents which store information regarding situations that are desirable. This allows the agent a way to choose among multiple possibilities, selecting the one which reaches a goal state.

1.7.4 Utility-based agents; distinguish between goal states and non-goal states. It is possible to define a measure of how desirable a particular state is. This measure can be obtained through the use of a utility function which maps a state to a measure of the utility of the state.

1.7.5 Learning agents; In some literature IAs are also referred to as autonomous intelligent agents, which mean they act independently, and will learn and adapt to changing circumstances. According to Nikola Kasabov.

1.8 Main differences Agents and Objects

- ❖ **Agents are autonomous; decide for themselves**
- ❖ **Agents are smart; flexible(reactive, pro-active, social) behavior**
- ❖ **Agents are active; at least one thread of active control**

1.9 Main differences Agents and Expert Systems

- ❖ **Agents situated in an environment**
- ❖ **Agents act**

Some real-time (typically process control) expert systems are agents.

1.10 Inter-agent Communication; effective communication requires:

- ❖ **Shared Knowledge of syntax**
- ❖ **Shared understanding of semantics and pragmatics means of exchanging sentences (or even signs) to communicate**

1.11 Modes of Interaction,

- ❖ **Independence**(no interaction)
- ❖ **Simple Collaboration**(compatible goals, sufficient resources, insufficient skills)
- ❖ **Obstruction**(incompatible goals, insufficient resources, sufficient skills)
- ❖ **Coordinated collaboration**(compatible goals, insufficient resources, insufficient skills)
- ❖ **Pure individual competition**(incompatible goals, sufficient resources, sufficient skills)
- ❖ **Pure collective competition**(incompatible goals, sufficient resources, insufficient skills)
- ❖ **Individual conflict over resources** (incompatible goals, insufficient resources, sufficient skills)
- ❖ **Collective conflict over resources** (incompatible goals, insufficient resources, insufficient skills)

2. OVERVIEW MAS TECHNOLOGY

2.1 Multi agents systems; In artificial intelligence research, agent-based systems technology has been hailed as a new paradigm for conceptualizing,

designing, and implementing software systems.
A multi-agent system is a group of agents that interact to solve problems that are beyond the individual capabilities. Multi-agent system design is more complicated than a single agent design. It presents many new cooperation issues. This presents us…

- ❖ **Task allocation:** how to assign responsibility to a single agent?
- ❖ **Resolving Conflicts:** How to resolve conflicting knowledge, actions and goals among agents.
- ❖ **Communication:** How to understand each other?

2.2 Potential of Multi agents systems; Open information systems will contain multiple autonomous agents or agents acting on behalf of autonomous users or entities solution of complex problems require the services of multiple agents with diverse capabilities and needs e.g., the mediator-based approach to information systems Multi-agent systems can support distributed collaborative problem solving by agent collections that dynamically organize themselves
Multi-agent systems support a modular, extensible approach to design of complex information systems.

2.3 Challenge Multi agents systems; How can multi agent systems generate useful behaviors?

- ❖ **Inter-agent communication (of knowledge, intentions, beliefs).**
- ❖ **Inter-agent collaboration (e.g., through negotiation among self-interested rational agents).**
- ❖ **Coordination and Control in multi-agent systems.**

2.4 Why MAS Are Different? In general agent-oriented programming is the next step of object-oriented programming (OOP), which integrates achievements of last decades in artificial intelligence, parallel computing and telecommunications. Comparing with objects in OOP,

2.4.1 Conventional Software ; It is the example of a large program which calls subprograms (methods) sequentially one by one Even with progress of object-oriented programming (OOP) there is still pre-organized structure of source code lines which give direct instructions to computer what to do, when and how. But all these lines have “no ideas” what they are doing and for what reason. To change this structure you will always need programmers.

2.4.2 Multi-Agent Software; It is a self-organized network of agents (software objects) constantly working in parallel building or revising links. No one knows which agent will call which – it depends. Every agent is autonomous and struggles to achieve own goal. Autonomy means that agent cannot be invoked as a subprogram (method), it only decides what to do and when basing on knowledge of domain. It looks like clearly a work of swarm of bees or a colony of ants.

2.5 Trust and Reputation; Advanced Agent Technologies supporting decision making in complex domains. Improving Robustness and Fault Tolerance by Assessing the Trustworthiness of Information and Sources As decision-makers, software agents construct beliefs about themselves, others, and the environment. These beliefs form the basis for decisions and actions.
The concept of multi-dimensional trust is developed to help achieve goals when the domain problem can be expressed using a set of constraints.

2.5 Multi agents system Applications ; It is possible to group into:

- Aircraft maintenance
- Electronic-government services
- Military defining
- Wireless collaboration and communications
- Military logistics planning
- Supply-chain management
- Joint mission planning
- Financial portfolio management
- many other areas

2.6 Application of distributed agent-based systems; Application of distributed agent-based systems often categorized as Distributed Artificial Intelligence (DAI) or Multi-Agent Systems (MAS), provide decision support and operation control in complex domains where decision makers must accomplish goals despite environments characterized by high volatility, information overload, and inherent uncertainty.

3. AGENT BASE APPROACH IN SOCIAL SERVICES IN RUSSIA

This section presents multi-agent system for social services based on social passport and smart cards of citizens. It describes developed approach based on agents and ontology, architecture of the system and its specific features. It is shown that application of multi-agent technology can bring high value and clear benefits for clients in full scale regional e-government systems.

Problem: This is a serious problem for social services of each region which give support to their citizens.

Solution: To solve problem, they have developed a multi-agent system. Citizens can access the system via the Internet, Internet-kiosks which last version includes services of a cash-machine.

3.1 Ontology, Social Passports and Smart Cards of Citizen; The citizen of Samara Region has the right to obtain a social smart card free of charge. It is called “smart”. The smart card becomes the key to access those data bases. Part of the provided data is obligatory and needs to be confirmed with originals of certain documents (e.g., passport number, etc.) Obviously it is very unlikely that all these data will be accumulated in one data base. Instead at this stage the system uses ontology to integrate information about a person. Their ontology of social sphere, defined in the form of semantic networks, acts as a metadata and contains data about location of different information on a certain person and its format.

For implementation of modern e-Government concepts, The agents uninterruptedly work for the good of a man, ensuring his or her social support, taking care of his or her health and education, work and security, culture and sports.

- **Agent of a person -** acts for and on behalf of a person,
- **Agent of a social law -** the agent of a law scans citizens social passports and finds out relevant law
- **Agent of a social officer –** Its task is to find citizens who require social support
- **Agent of a social organization -** acts for and on behalf of an organization officer

3.2 Real Time Knowledge Base, In their work system users and software agents may use ontology-oriented knowledge bases. We know that Modern methods of ontology representation as the mostly wide-spread well specified and open for extension. System use ontological model for representation of laws in social sphere. The OWL standard defines the format of ontology representation in the form of an XML file based on RDFS scheme. As an example let’s use regional law №122-GD “On governmental support of citizens with children”. It defines several benefits for families with 3 or more children, including “30% discount from statute-established public utilities fee”. The following is a description of the benefit:

```
<owl:Class rdf:ID="30_percent_discount_for_public_utilities ">
<!—- Restriction defining the document which provides benefits (defined
directly without a link) -->
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="http://www.kg.ru/lawsBase.owl#byLaws"/>
<owl:hasValue>
<laws:Regulatory_document rdf:ID="Statute_122-GD">
<rdfs:comment rdf:datatype="http://www.w3.org/2008/XMLSchema#string"
>STATUTE №122-GD “On governmental support of citizens with children
"</rdfs:comment>
</laws:Regulatory_document>
</owl:hasValue>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf
rdf:resource="http://www.kg.ru/lawsBase.owl#Public_utilities"/>
<!—- Restriction on categories of citizens who use the benefit (defined by a
link to the class, defined later) -->
<rdfs:subClassOf>
<owl:Restriction>
<owl:allValuesFrom>
<owl:Class rdf:ID="Families_with_3_or_more_children"/>
</owl:allValuesFrom>
<owl:onProperty
rdf:resource="http://www.kg.ru/lawsBase.owl#forCategories"/>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:comment rdf:datatype="http://www.w3.org/2008/XMLSchema#string"
>30% discount from statute-established public utilities fee</rdfs:comment>
</owl:Class>
```

It illustrates the use of ontology language in description of social laws. It is worthy of note that description possibilities may be widened due to use of mechanisms of inheritance, import and definition of equivalency of different components. With the developed description and standard OWL ontology analysis tools it is possible to make conclusions on possibility of granting benefits to particular people. It was also essential to implement tools to access personal data in external information systems in order for the system to function.

3.3 Technical Architecture

The developed system historically was formed on the basis of quite a number of existing data bases of Ministry of Social and Humanitarian Development of the Samara region. The software was developed on three-level architecture with the use of J2EE technologies which allow creating platform-independent applications, including:

- ❖ subsystem of personal interaction with citizens with the use of internet kiosks;
- ❖ subsystem of decision making support for social sphere ministries, namely Ministry of Social Protection, Public Health, Culture and Education;
- ❖ Internet portal for integration of social sphere resources, which are organizations, authorized by the above-mentioned ministries.

Originally the system was hosted on one central server of the Ministry, which was accessed both from the local network by the visitors of the Ministry and its social officers and from the internet by the public. All data bases were also located on one server, which contained information on approximately 1.5 million People. The development of the system a new goal was set: to create a distributed P2P system architecture capabilities of transparent user access to all nodes of the system from any geographic location. The system is organized as Service-Oriented Architecture which has nested nodes, but all of them, being autonomous parts of the system in the whole. The most important step to construction of the distributed network was creation of a single repository of all ontology, which is replicated and updated on all servers and ensures common knowledge for each remote server. This step illustrates structure of a fully distributed system, being developed now, which includes Ministry of Humanitarian and Social Development…

3.3 Key Stages of the System

- ❖ With the pilot system (2001) at the first stage the Samara Region Administration became the winner of “Electronic Russia”, a national-wide projects contest.

- ❖ The second stage (2003 – 2005) developed its own specialized software on a large scale.
- ❖ At present time the third stage of the system creation is being completed, replication and implementation of the system are being carried out.

3.4 Next Steps and Development Plan

3.4.1 The system development in new domains of the social sphere:

- ❖ System development in new social spheres: healthcare, civil registry office, public utilities, education, culture, etc.;
- ❖ Integration of social passport with medical insurance certificate, patronymic certificate, etc;
- ❖ Transfer of social payments and a possibility to use these money in supermarkets, for other expenses;

3.4.2 Develop a toolkit, architecture and functionality of the system:

- ❖ Transfer to a fully distributed SOA system architecture with P2P interaction of servers;
- ❖ Electronic protocol of interactions between ministries, organizations and citizens;
- ❖ extension of agents functionality which pro-actively to implement a wider range of citizens' demands and needs;
- ❖ creation of multi-user, distributed ontologies and solving problems of their collaborative building and using;

4. AGENT BASE APPROACH IN MANUFACTURING CONTROL

Manufacturing scheduling and control is a difficult problem, particularly when it takes place in an open, dynamic environment.

In a manufacturing system, rarely do things go as expected. The set of things to do is generally dynamic. A task can take more time than anticipated or less time than anticipated, and tasks can arrive early or late.

Because of its highly combinatorial aspects, its dynamic nature and its practical interest for manufacturing systems, the scheduling problem has been widely studied in the literature by various methods:

Heuristics, constraint propagation techniques, constraint satisfaction problem formalism, simulated annealing, Taboo search, genetic algorithms, neural networks, etc.

Agent technology has recently been used in attempts to resolve this problem.

4.1 Agents and controllers

The comparing becomes important when developing an agent-based method for solving control problems. Several similarities and differences between these two concepts can be noticed.

4.1.1 Similarities between agents and controllers are:

- ❖ **Objective:** Both a controller and an agent have objectives that determine their behavior. A controller resembles a reactive agent.
- ❖ **Situatedness:** Both an agent and a controller interact with some "surrounding" by sensing and acting.

4.1.2 Differences between agents and controllers are:

- ❖ **Pro-active:** A controller is usually designed to operate continuously during the whole operating time of a control system. An agent, however, decides whether it wants to gets operational and Produce actions. It might not produce actions during the hole operating time of the system.
- ❖ **Thread of control:** Matters related to the thread of control are a major concern in MAS, whereas this is generally not addressed during the design of controllers. Agent based systems are typically structured and implemented in ways that make it impossible to guarantee that hard real-time bounds will be met.

4.2 Agents are not (yet) common in control engineering; because the field of multi-agent systems is relatively new and Control theory has a strong mathematical foundation, whereas the field of multi-agent system mainly is focused on abstract descriptions of system. Still there are examples in which MAS has been used in control engineering.

4.2.1 Applications are described by Mac-Leod & Stothert. They describe the problem of controlling a **mine refrigeration system**. To model the refrigeration system as a cooperating multi-agent system and results in a planning mechanism that launches specialized agents to suit operating conditions and operator demand." The application of MacLeod and & Stothert clearly is of the kind "constructing a control system by using agents".

4.2.2 Lygeros, Godbole and Sastry describe an application which could be described as "constructing controllers as architectures of agents". They consider problems that consist of a **large number of agents** and that have to make efficient use of a **scarce resource**. Examples of such problems include highway systems, where vehicles can be viewed as agents competing for scarce highway space-time and air traffic management systems where aircraft compete for air space and runway space.

4.2.3 MacKenzie describes an agent-based method for designing controllers for (mobile) robots. This method is based on the behavior-based robotics paradigm: the controller of the robot consists of a set of concurrently operating agents. MacKenzie defines an agent as a distinct entity capable of exhibiting a behavioral response to stimulus. Furthermore, in order to build hierarchical controllers, he defines an **"assemblage agent"**. An assemblage agent is a society of coordinated agents and can be part of some other assemblage agent.

4.3 Controller-agent

Multi-controllers consist of several locally operating controllers. A controller-agent is a largely autonomous, locally operating controller that consists of a control algorithm (in the form of an update and a calculate function), an operating regime characterization, initialization and finalization functions and an interface to coordinate its behavior in order to handle dependencies among controller-agents.

4.4 Some application using agent-based design method for multi controller system Controller-agent

- ❖ Design and implementation of an intelligent room thermostat **(Assen, Van Breemen & De Vries)**
- ❖ Design and (partial) implementation of a safeguard module for an industrial manipulator **(K¨upers)**
- ❖ Design and implementation of a controller for the double beam experimental setup **(De Kruif)**
- ❖ Design and implementation of a behavior-based controller for the autonomous mobile robot Arty **(Balkema,Boer, Dertien et)**
- ❖ Controller design and implementation for a placement module of the Fast Component Mounter **(Jansen, Lubek& Seters)**
- ❖ Design and simulation of the water vessel problem **(Van Breemen & DeVries)**
- ❖ Modeling of the manufacturing process of corrugated cardboard process **(Van Breemen, De Vries & Striper)**
- ❖ Design and simulation of a controller for the manufacturing process of corrugated cardboard**(Striper)**

4.5 Manufacturing Planning, Scheduling and Control

Manufacturing control relates to strategies and algorithms for operating a manufacturing plant, taking into account both the present and past observed states of the manufacturing plant, as well as the demand from the market.

The manufacturing control problem can be considered at two levels:

- **Low-level,** the individual manufacturing resources are to be controlled to deliver unit-processes expected by the high-level control functions.
- **High-level** manufacturing control is concerned with coordinating the available manufacturing resources to make the desired numbers of types of products.

In agent-based manufacturing systems, agent technology is usually applied to high-level manufacturing control, but can also be applied at the lower level (Brennan et al 1997; Wang et al 1998).

Shaw may have been **the first** to propose using agents in manufacturing scheduling and factory control. He proposed that a manufacturing cell could subcontract work to other cells through a bidding mechanism (Shaw and Whinston 1983; Shaw 1988).

YAMS (Yet Another Manufacturing System) (Parunak 1987) was another of the earliest agent-based manufacturing systems, wherein each factory and factory component is represented as an **agent**. Each agent has a collection of plans, representing its capabilities.

4.6 Some projects using agent technology for manufacturing planning, scheduling and execution control

- Using autonomous agents to represent physical entities, processes and operations **(Parunak et al ITI, U of Cincinnati)**
- Using simulated annealing to search problem space **(Sprumont & Muller U. of Neuchatel)**
- Design each joint of a robot is modeled as an agent **(Overgaard et al Odense U)**
- Partial Dynamic Control Hierarchy (PDCH); Using agents to model IEC-1499 Functional Blocks**(Brennan et al 1997, Wang et al U of Calgary)**
- Emergent Behavior in Manufacturing Control; Proactive Disturbance Handling; Hot Pluggable Agents**(Bruckner et al Daimler-Benz AG, KULeuven)**
- Implemented as CORBA objects communicating through ILU object environment **(Barber et al U of Texas at Austin)**

4.7 Intelligent controllers in complex control systems

For the demand of large scaled and complex control systems in industry process, we define project to put forward a general framework of agent-based distributed intelligent control systems (ADICS), which combines the natural distribution of distributed control systems, and constructs a multi-agent system (MAS) by taking agents as intelligent controllers. In distributed intelligent manufacturing systems, agents can be used to:

- **Encapsulate existing software systems to resolve legacy problems and integrate manufacturing enterprises**
- **Represent manufacturing resources**
- **Model special services in manufacturing systems,**
- **Incorporate a whole scheduler or planner and control**

5. CONCLUSIONS

5.1 Prospects

Currently, there is an interest from both industry and universities into design methods for embedded systems. The main focus is co-design of both software and hardware aspects of a systems. Often, these embedded systems are complex systems, as they consist of multiple functional components on both the software and hardware level.
The concept of an agent should be used more in these design methods.

5.2 Suggestions for future research

The most exist agent-based framework is aimed at developing discrete-time controllers that run on single processors. In order to have a more general framework, the following options for future research are suggested:

- **Discrete-event system approach;** control systems may contain sensors and actuators that operate on an event-driven basis. In order to build such control systems, a discrete time approach is not sufficient, or even feasible. A discrete-event system approach may provide the theoretical basis for developing a design framework in which both periodic and periodic components can be specified.
- **Heterogeneous hardware;** To avoid the difficulties of a multiple (heterogeneous) processor system typically exist framework run on a single processor .A multiple processor system requires the distribution of the software components over the processors.

In overall conclusion; The agent-based design helps the designer to solve complex control problems, by offering concepts to structure the problem and to organize the solution.

It is essential to create a control system for the economical operation.

We discussed that Intelligent Multi-Agent Theory can create an intelligent control system for Dynamic environment.

New intelligent control systems are very different from previous control approaches. It combines optimized computation with automatic control and introduces a multi-agent, intelligent control system to improve total efficiency.

In new intelligent distribution control system agent base system, agents with different objectives all work together with mutual coordination and consultation to improve operations.

Combining multi-agent theory and optimization to create a more intelligent control system will greatly improve efficiency, increasing the Reliability of Control Systems with Agent Technology and reduce the operating costs.

REFERENCES

[1] Albert van Breemen, Theo de Vries (2000).An Agent-Based Framework for Designing Multi-Controller Systems. In proceedings of the Fifth International Conference on The Practical Applications of Intelligent Agents and Multi-Agent Technology,

[2] Darell Dean Zimbelman. (1981).Computerized Control of an Open-channel Water distribution System. Doctor’s thesis. Arizona State University,

[3] Gerhard Weiss. (1999). Multiagent Systems – A Modern Approach to Distributed Modern Approach to Artificial Intelligence. The MIT Press,

[4] Hyuckchul Jung. (2003). Confilict Resolution Strategies and Their Performance Models for Large-scale Multi-Agent Systems. Doctor’s thesis. University of Southern California,

[5] Juichi Kosakaya, Katsunori Yamaoka, Ryuji Sugita. (2000). Cooperative Multi-Agent Intelligent Field Terminals for Distributed Control Systems. IEICE Trans. Fundamentals, Vol.E83-A, No.11,

[6] Leo A.Stankevich. (1999). the Cognitive Approach to Design of Soccer Agent. RoboCup-99 Team Descriptions, Simulation League, Team Ploytech100, 103-109,

[7] Michael Wooldridge, Nicholas R. Jennings, David Kinny. (2000). the Gaia Methodology for Agent-Oriented Analysis and Design. Autonomous Agents and Multi-Agent Systems, 3, 285-312,

[8] V.A. Vittikh, P.O. Skobelev. Multi-Agent Interaction Models for Building

Supply and Demand Matchings Networks in Open Systems. // Automatics and Telemechanics, №1, 2003, pp. 177-185.
[9] The OWL Web Ontology Language http://www.w3.org/TR/2004/REC-owl-features-20040210/
[10] RDF Vocabulary Description Language 1.0: RDF Schema, W3C Working Draft 23 January 2003 http://www.w3.org/TR/rdf-schema/
[11] V.A. Vittikh, D.V. Volkhontsev, E.A. Grizanko, A.N. Nikitin, G.D. Svetkina, P.O. Skobelev. Social Passport as a Distributed Knowledge and Data Base. // Proceedings of the V International Conference “Complex Systems: Control and Modeling Problems” (Samara, June 17-22, 2003). – Samara: SSC of RAS, 2003, pp. 386-390.
[12] Artificial Intelligence, http://www.knowclub.com

[13] V.A. Vittikh, G.D. Svetkina, P.O. Skobelev, D.V. Volkhontsev, E.A. Grizanko, A.N. Nikitin, O.L Surnin, M.A.Shamashov. Development of the First Stage of the Region Management System with the Use of Multi-Agent Technologies // Proceedings of the VI International Conference “Complex Systems: Control and Modeling Problems” (Samara, June 14-17, 2004). – Samara: SSC of RAS, 2004, pp. 346-351.

[14] K.A. Titov, V.A. Vittikh. Reforming of the Government Management System: Key Principles, Results and Next Steps. // Proceedings of the VI International Conference “Complex Systems: Control and Modeling Problems” – Samara, SSC of RAS, 2005, pp. 227 – 232.

[15] Intelligent Agents and Multi Agent Systems
http://www.cs.iastate.edu/~honavar/
http://www.elsevier.com/locate/cogsys

[16] Multi Agent Systems
http://www.cs.cmu.edu/afs/cs/usr/pstone/public/papers/97MAS-survey

[17] Agent-Based Manufacturing Systems
http://imsg.enme.ucalgary.ca/

[18] Multi-Agent System of Samara Region Social Services
www.magenta-technology.com

[19] Multi agent Systems Katia P. Sycara http://www.aaai.org