Customization of Multi-Agent Systems for Adaptive Resource Management with the Use of Domain Ontologies

George Rzevski, Petr Skobelev, Alexey Zhilyaev, Oleg Lakhin, Igor Mayorov and Elena Simonova

***Abstract*—** Nowadays, the demand for increasing business efficiency is constantly growing, calling for development of resource management systems that can be applied to solve a wide range of complex tasks with minimum costs and within a limited time period. Unfortunately, traditional combinatorial or heuristic methods and tools cannot reach good enough solutions for complex tasks of real-time resource management. For this reason, the paper considers multi-agent technology as the key element of the required solutions. It can help find the balance of interests of all parties involved and flexibly adapt it to unpredictable events, such as, for example, a new order, inaccessible resources, delay of operations, etc. This paper presents the use of ontology for planning, giving the opportunity to develop an ontological enterprise model and a generic multi-agent scheduler, as well as to customize the appropriate requirements for each operation in business processes or technological processes in production applications, project management, supply chains and so on. At the top of the ontology editor is the Semantic Wikipedia, supporting the enterprise knowledge base for the tasks of resource management. The paper also provides examples of applications for supply chain in insurance business and management of maintenance of complex technical objects.

***Keywords*—**adaptability, complex systems, demand-resource networks, multi-agent technology, ontology, manufacturing ontology, real-time scheduling, resource management, self-organizing.

I. Introduction

At the present stage of development of production planning and scheduling systems, there are significant difficulties typical for enterprises operating in conditions of uncertainty. Uncertainties are usually generated by turbulent changes in demand and supply, but also associated with the ramp-up stage of new products, variety of many heterogeneous factors, needs for an individual approach to each order and resource, failure to meet deadlines, continuous change in technology, occurrence of various unpredictable events directly affecting production processes, etc.

The existing enterprise resource planning systems have wide functionality in finance and accounting, but they lack tools for supporting decision-making on real-time resource allocation, scheduling and optimization and reacting to new situations that were not known in advance [1]–[4].

Therefore, maintaining production plans in an adequate state at all levels, taking into account uncertainty of changes, becomes a vitally important task. Modern production systems should respond flexibly and quickly to events, take into account the distributed nature of decision-making and coordination in workshops [5]. The main principles of constructing and implementing such systems are considered in [6]. Solution of such complex and poorly formalized tasks must be performed in real time, therefore the management system should be built on the network-centric principles and be adaptive to changes. Planning systems based on centralized approaches of linear and mathematical programming, neural networks, genetic algorithms and heuristics cannot fully take into account the high dynamics of changes, since it is necessary to continuously adapt schedules and indicators for hundreds and thousands of interrelated products and resources [7]–[11].

Therefore, in order to take into account unforeseen events, reduce and parry uncertainties in orders and resource availability, multi-agent systems become more widely used, in which the schedule is built by self-organization of negotiating agents competing and cooperating on the virtual market – to achieve competitive equilibrium [12].

II. Multi-agent technology for resource scheduling

The developed approach is based on Demand-Resource concept [13], [14]. The agents in Demand-Resource network (DR-network) [15] interact through negotiations using the World Scene, which is constructed as a reflection of the actual situation in the real world, with the current action plan and expected results available to the enterprise agents and agents of integrated units (for example, workshops) at the appropriate levels. The order agent reads out the enterprise structure from the ontology and determines resources relations required for task execution. The agent of the new order, having entered the virtual market, refers to the knowledge base and creates agents of technological processes and then agents of operations,

The paper has been prepared based on the materials of scientific research within the subsidized state theme of the Institute for Control of Complex Systems of the Russian Academy of Sciences for research and development № AAAA-A16-116040410059-7 on the topic: "Research and development of models, methods and algorithms for the construction of shift-daily assignments for product manufacturing in conditions of uncertainty and high dynamics in the production environment". This work was partially supported by the RAS 30 Presidium Program "Theory and technology of multi-level decentralized group management in conditions of conflict and cooperation".

George Rzevski works at the Digital Eco-Systems and Imperial College Incubator, London, UK (e-mail: rzevski@gmail.com).

Petr Skobelev works at the Knowledge Genesis Group and Institute for the Control of Complex Systems of the Russian Academy of Sciences, Samara, Russia (phone: +7929-702-2200; e-mail: petr.skobelev@gmail.com).

Alexey Zhilyaev works at Samara National Research University, Samara, Russia (e-mail: zhilyaev@smartsolutions-123.ru).

Oleg Lakhin works at the Knowledge Genesis Group, Samara, Russia (e-mail: lakhin@smartsolutions-123.ru).

Igor Mayorov works at the Institute for the Control of Complex Systems of the Russian Academy of Sciences, Samara, Russia (e-mail: imayorov@smartsolutions-123.ru).

Elena Simonova works at Samara National Research University, Samara, Russia (e-mail: Simonova.Elena.V@gmail.com).

which send requests for resources - machines, equipment, materials, etc. Agents perform matching according to their individual requirements. Resources (machines, humans, parts, equipment, materials) analyze their local schedules and check for the possible options. The agent of the new order tries to agree with the interfering orders about relocations and the corresponding compensations, and changes local schedules in proactivity mode. In case of a consensus agreement, the new schedule is accepted by all participants, otherwise iteration (waves) of negotiations affect deeper layers of timetables involving chains of distant participants.

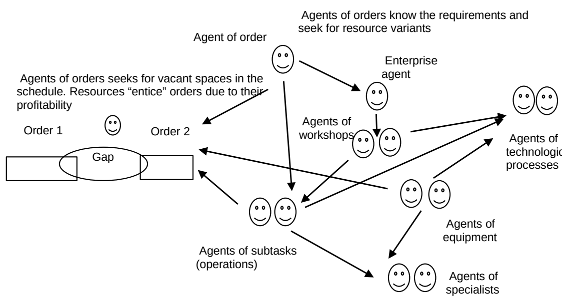


Fig. 1 the DR-world scene

Such a distributed approach makes it possible to increase the complexity of the tasks to be solved, extending the problem statement and introducing new agents, for example, to take into account equipment repairs, transportation of products, etc.

III. THE METHOD OF ADAPTIVE SCHEDULING

The formalized problem statement is based on searching for a consensus between agents in DRN virtual market and can be formulated as following [12].

Let us assume that all agents of demands and resources have their own goals, criteria, preferences and constraints (e.g., due date, cost, risk, priority, required equipment type or worker qualification). The importance of each criterion can be represented by weight coefficients in a linear combination of criteria for the given situation in scheduling, but can change during the schedule forming or execution.

Let us introduce the Satisfaction function for each agent, which will show deviation of the current value ("virtual value") of this function from the given ideal value by any of the criteria for the current step of finding scheduling solution for this agent.

Let each demand j have several individual criteria x_i and suggested ideal values x_{ij}^{id}. For each agent of demand j normalized function is calculated by component i, given for example as a linear function $f_{ij}^{task}(x_i\text{-}x_{ij}^{id})$. In most cases this function has bell form with maximum in the point of suggested ideal value. As a summary value of the result for each demand, the sum of virtual values for each criteria i with the given weight coefficients α_{ij}^{task} is estimated.

By the proper selection of signs and form of the function, the goal of each agent can be reformulated as increasing (maximizing) of value y_j^{task} of demand j (upper index task means that the values belong to the demand agents):

$$y_j^{task} = \sum_i \alpha_{ij}^{task} \cdot f_{ij}^{task}(x_i - x_{ij}^{id}),$$

where $\forall j$ weight coefficients are normalized: $\sum_i \alpha_{ij}^{task} = 1$.

The problem of finding the states x_{ij}^{*} of agents of demands j, maximizing the total value of all orders can be formulated as the following:

$$y^{task} = \sum_j \beta_j^{task} y_j^{task} = \sum_j \beta_j^{task} \sum_i \alpha_{ij}^{task} f_{ij}^{task}\left(x_i - x_{ij}^{id}\right) \quad (1)$$

$$y^{task^*} = \max_{x_i}(y^{task})$$

where β_j^{task} is demand weight that allows to set and dynamically change the priorities showing importance of criteria.

Similarly, the value function can be given for the resources by criteria z_k, with function $f_{kl}^{res}(z_k\text{-}z_{kl}^{id})$, weight α_{kl}^{res} of criterion k for resource l, and resource value β_l^{res} for the system (which is similar for weight for demand agents function):

$$y^{res} = \sum_l \beta_l^{res} \cdot y_l^{res} = \sum_l \beta_l^{res} \sum_k \alpha_{kl}^{res} \cdot f_{kl}^{res}\left(z_k - z_{kl}^{id}\right) \quad (2)$$

$$y^{res*} = \max_{z_k}(y^{res})$$

$$z_k \in D^K,\ x_i \in D^I \quad \forall i, k, I = Dim(D^I), K = Dim(D^K) \quad (3)$$

Variables x and z belong to some areas of the space of criteria for demands and resources, I and K are dimensions of the corresponding criteria spaces, upper index res means that the values belong to resource agents.

Thus in Demand-Resource network the optimization problem is formulated as solving (1) - (3).

The developed method is based on DRN concept, where agents operate in the virtual market (VM) and continuously try to improve their individual satisfaction functions (that reflect their given multi-criteria objectives) with the use of bonus-penalty functions (Fig. 2).

Agent is trying to maximize Satisfaction function using available virtual money: better position – bigger bonus – more money on the agent's virtual account. The bigger the sum of money – the more options for schedule reconstruction.

Bonus-Penalty function defines bonus-penalty to reflect changes in Satisfaction function and defines elasticity for trade-offs in conflict resolution.

Protocols of solving conflicts are based on Contract Net modifications - each task can be allocated in open slot, shifted, dropped, divided or swapped.

However, in case of a conflict, one agent can suggest compensation for another agent and the second agent will accept proposal if only the sum of Satisfaction function changes will not worsen.

Different protocols can define different methods of conflict solving.

The core part of the developed method can be identified as the following:

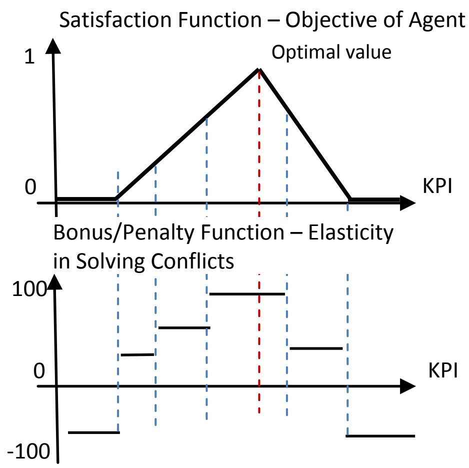


Fig. 2 example of satisfaction function

1) The number of specific classes of agents can be introduced representing specifics of the problem domain with the required level of granularity.
2) Satisfaction function and function of bonuses / penalties are defined by linear combination of multi-criteria objectives, preferences and constraints of each agent.
3) Protocols of agent interaction are defined, which specify how to identify conflicts and find trade-offs with the open slots, shifts and swaps of tasks (operations).
4) The resulting schedule will be formed in the process of self-organization of DRN agents.
5) Selected protocols are triggered when new events occur (for example, arrival of a new order):
   a. An agent is allocated to an order as it arrives into the system.
   b. The Order Agent finds business / technology process from Knowledge Base and creates Agents of Tasks which will be linked by relations and initiate them to start solving the problem.
   c. The Task Agent reads the list of requirements from Knowledge Base and sends a message to all agents assigned to available resources stating that it requires a resource with particular features and it can pay for this resource a certain amount of virtual money.
   d. All agents representing resources with all or some specified features and with the cost smaller or equal to the specified amount of money offer them to the Task Agent.
   e. The Task Agent selects the most appropriate free resource from those on offer. If no suitable resource is free, the Task Agent attempts to obtain a resource, which has already been linked to another order, by offering compensation.
   f. The Task Agent who has been offered some compensation considers the offer. It accepts the offer only if the compensation enables it to obtain a different satisfactory resource and at the same time increase the overall value of the system.
   g. If the Task Agent accepts the offer, it reorganizes the previously established relationship between that order, task and resource and searches for a new relationship with resource increasing the overall value of the system.
   h. The same process is running for Resource agents, which are able to generate proposals to Task agents with specific context-based requirements.
6) The above process is repeated until all resources are linked to orders and there is no way for agents to improve their current state or until the available time is exhausted.
7) At this moment, Agent of System will revise system KPIs and find agents that have minimum satisfaction for most important criteria.
8) Agent of System proposes to these agents to make an attempt and determine the sum of virtual money and value, which can be achievable during negotiations.
9) Receiving proposals from these agents, Agent of System can select the best “Value / Cost” proposal to improve the most important KPIs.

If necessary, the user can interactively intervene the plan at any time and manually rework the schedule by drag-and-dropping the operations - as a result, the plan will be automatically revised and rescheduled.

The general behavior of complex systems is evolving from agent interactions, which in turn constrain their behavior. This behavior is called emergent, and it is non-deterministic and unpredictable, but not chaotic.

Emergency leads to unstable equilibriums, associated with the reasonable consensus of many conflicting agents, which represent well-balanced schedules providing adaptability to the external influences.

In a more detailed way the agent decision making logic and negotiation protocols are presented in [6].

IV. Microeconomics of the Virtual Market

The use of the VM presumes that orders buy the services of the resources that, in their turn, have static or dynamic cost. The dynamic cost of the resource depends on how the resource can be shared.

Agents buy and sell their services / resources and get slots in the schedule. For this purpose, they have virtual accounts and use virtual money (which can be converted to real money).

For example, equipment has a certain cost, but it is distributed between its tasks with some planned profitability given in advance. As it is stated above, the agents can offer each other compensations for shifts and reallocations, the sum of which is defined during negotiations between demand and supply agents. If the cost of functioning is not covered by the income, the resource can decide to switch off.

The decision-making rules for agents are determined by the microeconomics of VM that defines the virtual cost of services, penalties and bonuses, rules for sharing the profits, what taxes should be paid under various conditions, etc. It gives agents an opportunity to accumulate virtual money and use it for getting the best possible options.

To achieve the best results agents use the virtual money that regulates their behavior. The amount of virtual money can be increased by getting bonuses or decreased by penalties, depending on their individual cost functions. The key rule of the designed VM is that any agent that is searching for a new better position in the schedule must compensate losses to other agents that change their allocations to resources, and propagation of such wave of changes is limited by virtual money [16].

The main features of the suggested VM microeconomics are:

- Agents have ideal and current values of multi-objective satisfaction functions, which are used to compute each agent's "satisfaction" by the current plan;
- Order agents enter the system having virtual money to achieve their objectives, including service level, costs, delivery time and some other criteria;
- Resource agents are looking for their maximum utilization but also have their own ideal preferences, constraints and costs which they share between orders;
- Product agents are interested in maximizing quality and minimizing time spent in storage;
- There are dynamic values of weight coefficients of the objective function, which are linked to a virtual money bonus or penalty for Orders, Products and Resources, each criteria has its own coefficient of conversion to virtual money;
- Current virtual budget is used by agents to improve their local allocation in the schedule;
- Agents iteratively improve their satisfaction function to reach better KPIs compensating the losses of other agents from their virtual budget – so that the total value and profit of the company is growing.

Such an approach gives the opportunity to introduce virtual taxes related to agents' job planning and execution, cost of slots and messaging between agents, etc. These tax mechanisms can be used to control the process of self-organization of the schedule to provide a high-quality schedule within limited time.

Currently we consider two types of VM Microeconomics:

- cost of physical resources is taken into consideration;
- cost of scheduling is taken into consideration, including agents routing, messaging, etc.

The aim of Microeconomics is to guide self-organization of scheduling on the level of physical resources (equipment and workers) and agent resources (cost of agents work, etc.). Virtual money plays the role of energy and agents use it to create new schedules or to adapt fragments of the existing ones [16].

V. Application of Ontology for Customizing Multi-Agent Scheduling Engine

A. *State of the Art*

Semantic Web approaches are currently being developed to build a new generation of information processing systems. One of new approaches is based on domain ontologies, which can be represented by the so-called semantic networks which include classes of concepts and relations. Ontologies make it possible to formalize and systematize the domain knowledge and separate it from the source code of systems.

Such developments are actively carried out all over the world in the fields of engineering, pharmaceuticals, robotics, military applications, etc.

However, in the field of enterprise resource management, Semantic Web has not yet achieved sufficient results. Development of approaches in this area is relevant and significant for improving the efficiency of resource management of any enterprises entering the digital economy. Transformation to the digital economy requires development of models and methods to formalize knowledge, which should enable formation of digital models of all aspects of enterprise activities to provide opportunities for managing enterprise resources, support interaction within industry digital platforms, end-to-end traceability of goods and services, and so on.

Such models should specify objects of enterprise activity, requirements for product and process quality, decomposition of departments to the level of each employee with their inherent competencies, the used equipment and tools, technological or business processes, materials, etc.

In this paper, it will be shown that such knowledge can be effectively formalized in the form of ontologies and applied in enterprise resource management. However, for practical application of this approach, a number of fundamental questions must be answered. What should the structure of enterprise ontology be? What kind of tools should be used to create industrial knowledge bases? How is knowledge base related to domain ontologies? What mechanisms should be embedded in the Knowledge Base to ensure automation of knowledge processing and integration with existing databases? What is the life cycle of Knowledge base functioning?

Well-known ontology designers (Protégé, OntoEdit, OntoLingva and others) pursue research goals and are not designed to build full-scale enterprise models, especially within the factories. Moreover, they do not allow building knowledge bases with thousands of objects, and are complex and time-consuming in use.

In this regard, it becomes relevant to develop methods and tools for formal representation of knowledge that will make it possible to unite data of various formats into a single semantic network, provide convenient user access to this information, and make this knowledge suitable for computer processing. It is important to provide users with the possibility for information semantization (classification and parameterization of knowledge quanta, their binding, etc.), including the possibility of constructing semantic descriptors (meta-data) that are convenient for computer processing. However, the very nature of knowledge systems is not static. They are constantly evolving. In this regard, it is necessary to make knowledge active through the use of multi-agent technologies, in order to be able to support the processes of comparing quanta of knowledge, their categorization, classification, binding, etc. As a result, enterprise knowledge can be organized into a certain self-explanatory Wikipedia-type system, in which self-organization processes are maintained to be as complete and coherent as possible, non-contradictory, clear, transparent, and open to change.

Let us consider the current trends in development of smart production, associated with the use of such methods as ontological knowledge representation, knowledge base, machine learning, etc.

A wide range of problems and future prospects of interrelation between machine learning (ML), logical inference and ontologies is presented in [17] in the context of artificial intelligence (AI). Processes organizing applications of learning, inference and ontologies, as well as relations between them, are investigated. Possibilities for solving important problems related to generic AI applications are proposed. The problem of extracting knowledge graphs from the source material for use in machine learning and in logical inference is discussed, and possibilities for achieving AI using methods of iterative knowledge extraction and learning for generic AI applications are analyzed. The paper shows ways of development for ML and Big Data by strengthening the links between learning, logical inference and ontologies.

Application of web services, semantic network, semantic services and ontology in various production sectors are discussed in [18]. It describes problems of developing a knowledge base for storing information resources in machine building. Heterogeneous data is collected from the available separate and segmentary systems of the enterprise based on ontologies. The ontological model of knowledge base of information resources is developed with the help of the Protege editor.

Research development of production ontology is presented in [19], which is considered the key component of the future compatible production systems. The set of basic concepts of production ontology is defined, and their semantics in formal logic is described, based on the use and expansion of the existing standard definitions, taking into account specific features of production. Results of a successful pilot study for testing the application of ontology based on interaction between concepts in design and production of aerospace products are also discussed.

The most important aspects of representation and reuse of knowledge in industrial production are analyzed in [20]. A new approach, called Semantic Manufacturing, is proposed, which combines technologies of production ontologies and Semantic Web.

A smart multi-agent production management system based on ontologies is presented in [21]. With the help of semantic technologies, it ensures high adaptability and reactivity of small-scale production of high-tech products, such as aircraft production at the ramp-up stage.

The method for describing production resources on the basis of a multi-level domain ontology is presented in [22]. Using ontology ensures accuracy and completeness of the search for production resources.

Application of ontology in the distributed semantic network of the enterprise production departments is described in [23]. It is used for the set-up of an efficient and scalable system of interaction and retrieval of data on information requests.

Let us also briefly consider very recent application of ontologies in production, presented at the modern MAS conferences.

The possibility of using ontologies for dividing complex problems into interconnected composite autonomous partsis discussed in [24].

Ontologies can be developed to facilitate proper understanding of the problem domain, and, subsequently, knowledge from external sources can be shared through linked open data or directly integrated (mapped) using an ontology matching approach. This is discussed in [25], which also demonstrates how ontological data description may facilitate interoperability between a company data model and new data sources as well as an update of stored data via ontology matching. This shows the possibility of using ontologies for data integration, but does not solve the problem of their ontological description.

The possibility for configuration of services built on ontologies is demonstrated in [26]. It shows the prototype for a flexible manufacturing system case study to verify the feasibility of greedy local service reconfiguration for competitive and collaborative industrial automation situations.

The possibility of coordination and control of such complex systems is demonstrated in [27]. It can be done through actors or components, methods and techniques used to design, verify and deploy the services, possibly on the fly, making service engineering unavoidable in cyber-manufacturing systems. As far as services can encapsulate or abstract not only software but also physical behavior (as digital twins do), orchestration refers to smart adaptable assembly systems. The paper provides deep semantic information that will complete the abstract layer of service interoperability ontologies.

An example of ontology application is given in [28] for the purpose of task allocation and management, crucial for human-robot collaboration in Urban Search and Rescue response efforts, where the authors design and evaluate an ontology which provides a common vocabulary for team members, both humans and robots. The ontology is used for facilitating data sharing and mission execution, and providing the required automated task management support. Results provide support that the proposed ontology (1) facilitates information sharing during missions; (2) assists the team leader in task allocation and management; and (3) provides automated support for managing an Urban Search and Rescue mission.

From these examples it is already clear that ontologies are becoming a useful tool for managing applications. However, the possibility of a detailed description of domain knowledge for planning has not been investigated. The discussed models and methods also do not provide options for customization of individual features of tasks and production resources in enterprise resource management.

B. *Ontology-Driven Knowledge Base for Adaptive Resource Management*

One of the new steps in multi-agent developments is to design and prototype new ontology-driven knowledge base for adaptive scheduling.

The key idea of this challenge was to collect, formalize and apply domain-specific knowledge on the level of end-users (workers, drivers, etc.) which can help significantly improve the quality of resulting schedules.

The problem is that this usually hidden knowledge is very difficult to formalize at the beginning of software development and it becomes visible only when first computer-formed schedules are deployed to users.

For example, for solving the problem of shop-floor scheduling for airplane factory it could be the knowledge about the possibility of parallelizing part of technological operations or reinforcing them by adding more workers in order to accelerate the work, or knowledge about the competencies of workers, their compatibility in shifts, individual features of tasks, etc. [29].

As the first step of this development, "Basic Ontology of Scheduling" was designed, which describes such key concepts:

- Order – describes what product or service is required;
- Business / Technology Process – describes a set of tasks (operations) to be executed;
- Task (Operation) – specifies the work to be done, including all required resources;
- Resource – humans, machines or materials which are required for tasks / operations;
- Product – input of output products used in operations;
- Department – consists of resources of different types.

A number of classes of relations was introduced for linking the above-mentioned concepts, including the following:

- Order "Creates" Objects – describes what objects will be created for order;
- Object 1 "Is-Part-of" Object 2 – describes assembly of objects;
- Object "Has" Business Process – describes business / technology processes available for producing objects;
- Task 2 "Is-Next or Previous to" Task 1 – describes interdependency of tasks in business process;
- Task "Requires" Resource – describes matching rules for selecting resources;
- Task "Produces" Product – describes output objects for tasks;
- Task "Needs" Materials - describes input objects for tasks, etc.

During the second step, the Ontology Editor and Knowledge Base designed as a Semantic Wiki were developed (Fig. 3).

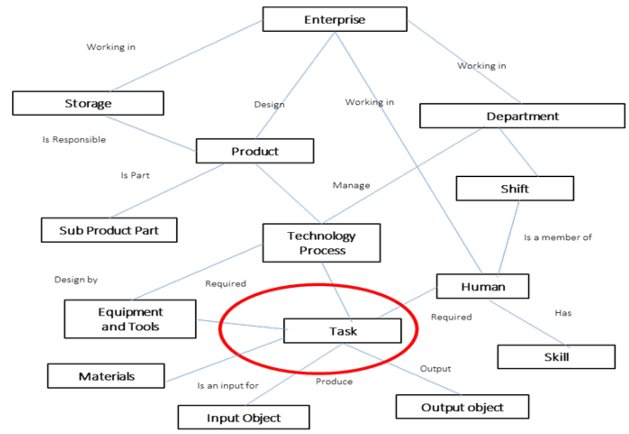


Fig. 3 fragment of basic Ontology of Scheduling

With the use of Ontology Editor one can design Domain Ontology of Scheduling and then with the use of Domain Ontology – specify Ontological Model (Onto-Model) of Enterprise.

Onto-Model of Enterprise contains ontological description of all instances of concepts and relations required for adaptive scheduling of orders to resources:

- Organization Chart of Enterprise, which specifies what resources are available in departments;
- List of Products or Services;
- List of Employees in Departments with their Competencies;
- List of Machines, Equipment and Tools in Departments;
- Initial Schedule of Tasks in Departments;
- State of Products and Materials in Storage, etc.

The core part of the Knowledge Base is the concept "Task" and its instances for performing specific business operations or technological tasks, for example, for factories (Fig. 4) [30].

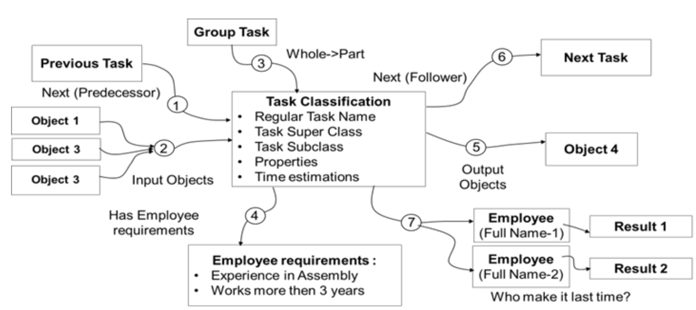


Fig. 4 key Types of Relations in the Concept of "Task"

The key idea here was to develop Task Agent that can read and interpret the relation “Required” as a matching rule – to find the specified resources dynamically.

These relations link the concept “Task” with other entities for scheduling which can be used by Task Agents in the following way:

- “Previous” (1) and “Next” (6) – allow the task agent to find the previous task with the request to move earlier or reschedule, or to find the next task, and to send its agent a delay message;
- “Input objects” (2) and “Output objects” (5) – show which agents should be in place to start the task execution, and also what will be the output as the result of task execution;
- “Is Part (Part-Whole)” (3) – shows that the task is a part of assembly, the agent of which receives parameters of the plan and the fact of the task execution;
- “Requirements for human” (4) – defines the requirements for the employee of the task;
- “Who did it last time?” (7) – finds employees who have already performed such tasks;
- “Required resource” (8) – specifies a resource, predetermined by the technological process.

The developed approach makes it possible to add new rules of matching “on the fly”, taking into account new complexity factors revealed during planning by end-users.

As the third step, the Generic Knowledge-Based Multi-Agent Engine for Scheduling was developed, which can be customized by Knowledge Base for any type of enterprise (within some constraints).

The key option to add new relations in the Knowledge Base as new matching rules gives the opportunity to expand Domain Ontology and Onto-Model of Enterprise by end-users “on the fly" and improve the quality of schedules.

C. The functionality of the solution

The main task of the Multi-Agent Scheduling Engine is formation and adaptive restructuring of the order execution plan, taking into account resource constraints. The Scheduler creates and configures instances of system agents based on the ontological description of the scheduling object, provides a multithreaded environment for their execution, and determines the order of their operation. The events that can trigger adaptation of plans could be the following:

- new order appears but there are not enough resources in the company;
- accepted and partially implemented order is cancelled by customer;
- new and more efficient resource became available for order execution;
- committed resources became unavailable in the process of order execution;
- preferences of customers and suppliers are changed dynamically;
- technology processes are changing, for example, in agriculture, because of weather conditions, attack of insects, etc.

In order to provide event-driven adaptability, such solutions must support the autonomous cycle of resource management, presented in Fig. 5.

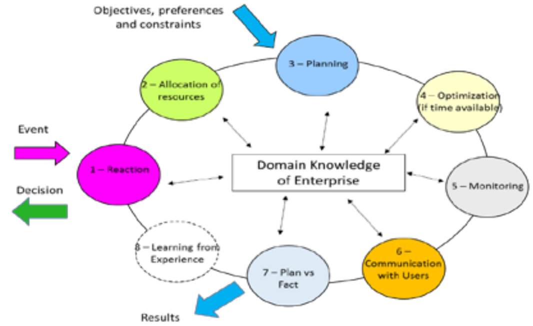


Fig. 5 autonomous cycle of resource management

Autonomous cycle of resource management means that the system is designed as continuously working on server. It has objectives, preferences and constraints, reacts to events and makes adaptive resource re-allocation, re-scheduling, re-optimization of plans using knowledge on problem domain; communicates new computer-formed plan with decision-makers and users and then takes it under control, monitors plan and fact and triggers re-scheduling in case of growing gap between plan and reality.

In the future, such a solution may include real-time learning from experience, real-time forecasting, what-if games in real time, etc.

D. Example 1: Knowledge-Based Adaptive Scheduler for Insurance Repair

Let us consider the example of developing Multi-Agent Scheduler for Supply Chain in Insurance Business. The case is that the Insurance Company is facing a lot of unpredicted claims related to climate changes - bad weather condition, tornado, flood, etc.

Let us assume that Insurance Company has the following resources, which are used to repair insured properties:

- Four Assessors, who can visit Clients using their Own Cars or Rented Cars;
- After visiting the Claim Site, assessors write Claim Reports and send them to the Claim Manager, who decides how to deal with claims and creates Claim Resolution;
- Property repair is carried out by Plumbers, Electricians and/or Builders;
- Repair material is purchased by Buyers and transported from Stores to claim sites by Trucks;
- Own cars, rented cars and trucks reach claim sites by Routes.

Disruptive events are the following:

- An assessor cancels a visit to a claim site;
- Delivery of a door is delayed;
- A plumber falls ill;
- A client is not on the site when assessor arrives;

- A new urgent claim arrives.

The objective is to schedule resources to achieve claim resolution in real time and re-schedule affected parts of the Schedule whenever a Disruptive Event occurs.

For the discussed case, as the first step "Ontology for Insurance Supply Chain" was designed as an extension of previously developed "Ontology of Scheduling" (Fig. 6). This Insurance Ontology contains generic business processes and types of claims and resources which are characteristic of Insurance business but not of the specific enterprise. For example, at this level there are descriptions of such concepts as Insurance case, Claim, Employee, Skills and Occupations. In this example, the domain "Insurance Ontology" contains about 50 classes of concepts and relations, which can be used now for specifying concrete insurance company.

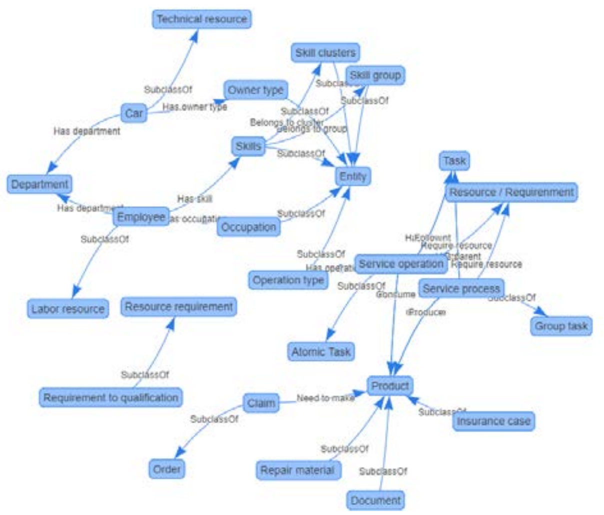


Fig. 6 fragment of Ontology for Insurance Supply Chain

On this basis, the Onto-Model of our discussed specific enterprise "Insurance Company" was developed (Fig. 7), which includes a number of concrete employees with their skills, etc.

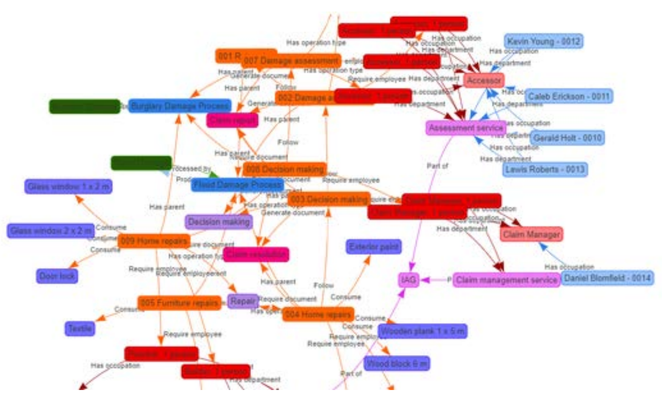


Fig. 7 fragment of Onto-Model of Specified Enterprise "Insurance Company"

The developed Onto-Model of Specified Enterprise "Insurance Company" contains about 70 additional entities, which represent the instances of previously defined concepts and relations. Now Onto-Model of "Insurance Company" can be loaded into Knowledge-Based Multi-Agent Scheduler which will be ready to work at the same time.

Let us assume now that 5 Claims arrive at unpredictable moments:

- 3 claims are related to flooding - water entered the house and the ground floor is flooded;
- 2 claims are related to robbery – damaged doors and windows.

The initial and resulting schedules are presented in Fig. 8–Fig. 10, which show how key operations were shifted and reallocated to different available resources.

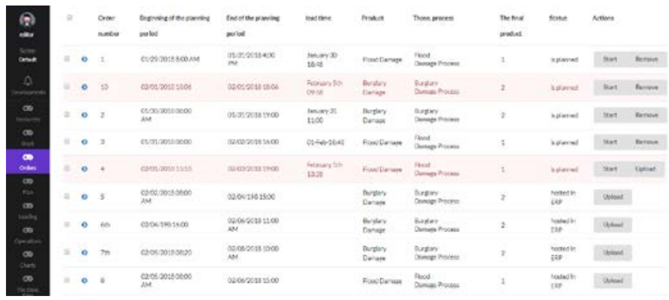


Fig. 8 order list

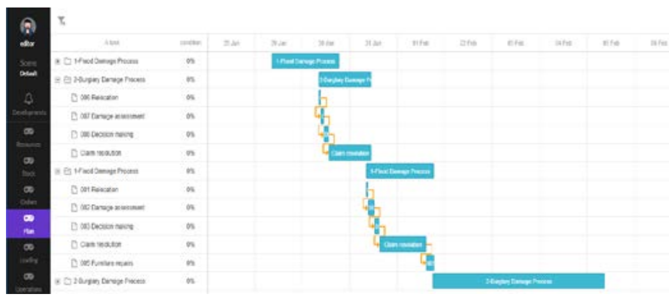


Fig. 9 schedule of "Insurance Company"

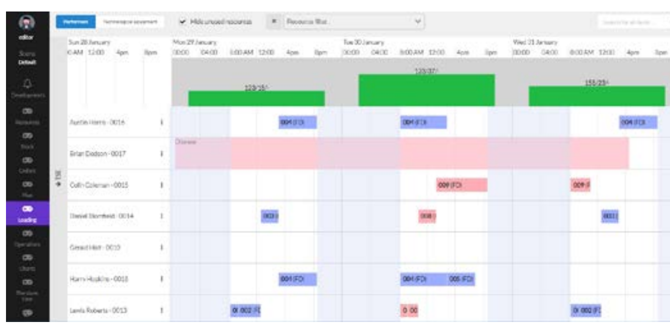


Fig. 10 resulting Schedule of "Insurance Company"

Fig. 11 reflects the schedule value demonstrating impact of unpredictable events and agents decisions taken during negotiations and solving conflicts.

With the use of the basic Ontology of Scheduling and Knowledge-Driven Multi-Agent Scheduler it took the development team only about 3 days to make customization of generic scheduler for new insurance business.

In this process, the one-page specification of client problem was translated into about 300 ontological concepts and relations, which help start scheduling from zero staring point.

Fig. 11 value Chart reflecting Value of Events and Decisions of Agents

E. Example 2: Knowledge-Based Adaptive Scheduler for managing maintenance of the infrastructure of complex technical objects

Maintenance and repair (MR) of the infrastructure of complex technical objects is the process of planning and performing operations to maintain and restore the working capacity of these facilities during their operation. In the proposed intelligent system for managing maintenance of the infrastructure of complex technical objects (IS), a knowledge base is used, built upon ontology with descriptions of composition of instances in ontological classes of objects together with their scheme of division, service regulations, employee competences, spare parts, tools, accessories and other parameters.

This approach involves creating ontological models of enterprises servicing the infrastructure of complex technical objects in the knowledge base, which can be loaded into the generic scheduler of maintenance and repair, thereby setting it for the specific application.

The knowledge base is used by software agents that represent the key elements of the physical world.

Ontology for managing maintenance of the infrastructure of complex technical objects has been designed as an extension of the previously developed "Ontology of Scheduling".

The example of the basic ontology of MR includes description of the infrastructure and the performed targets, MR processes, taking into account the requirements for the used spare parts, tools and accessories, personnel and maintenance intervals (Fig. 12).

This ontology can be expanded in accordance with the specific features of the structure, business processes and regulations of the enterprise.

On this basis, the Onto-Model of a specific enterprise was developed (Fig. 13), which includes a number of concrete regulations and maintenance tasks, spare parts, tools and accessories, materials, employees with their competencies, etc.

Fig. 14 shows the diagram of the sequence of service interaction with the knowledge base in the IS.

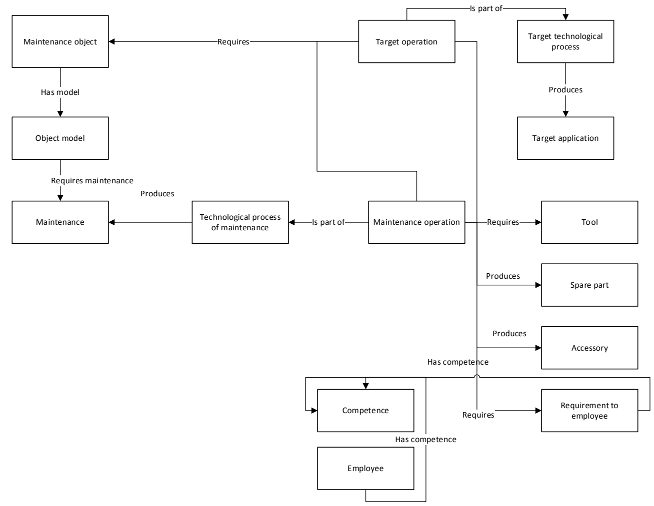


Fig. 12 basic ontology of maintenance and repair

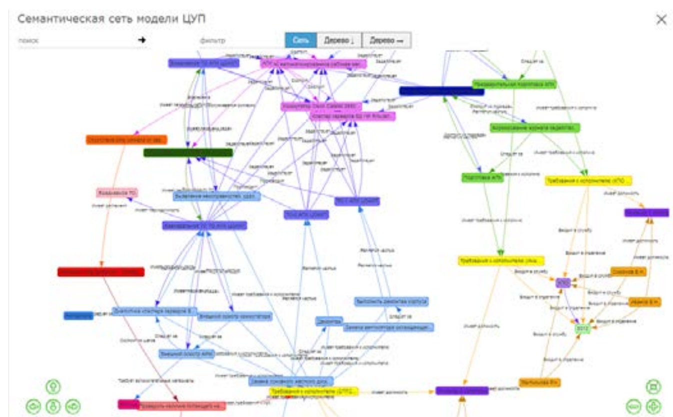


Fig. 13 fragment of Onto-Model of a Specific Enterprise for managing complex technical object

One of the main services in the proposed smart system is the forecasting with the following input data:

• knowledge base on the facilities of a complex technical object and their component parts;

• the target plan for the use of facilities of a complex technical object and the maintenance and repair plan;

• maintenance and repair regulations;

• characteristics of the facilities of a complex technical object and components (telemetry);

• history of breakdowns;

• calendar of availability and involvement of performers.

If this information is available, it becomes possible to construct a predictive maintenance and repair plan, which is aimed at improving the efficiency of executing target tasks by performing maintenance and repair in advance (Fig. 15).

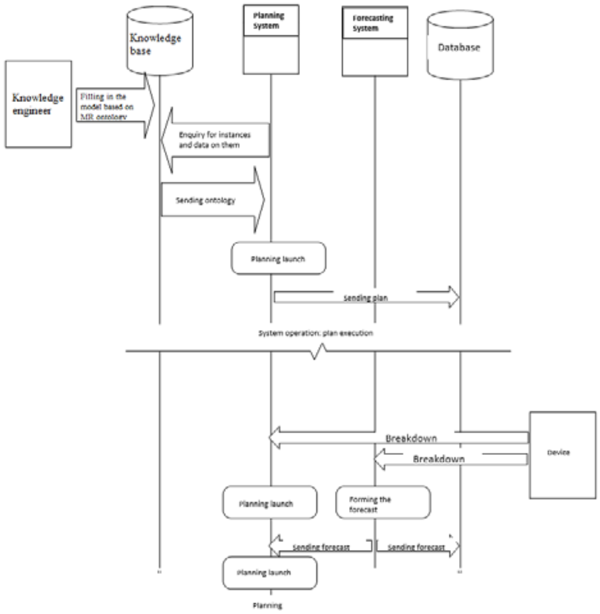


Fig. 14 scheme of interaction of services with the knowledge base

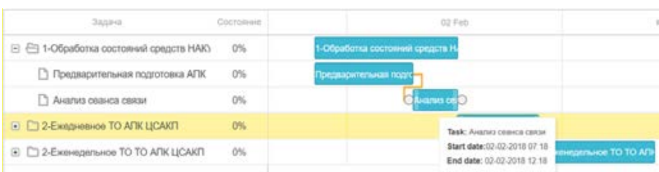


Fig. 15 maintenance and repair plan for a technical object

VI. The Results for Adaptive Resource Management

The objectives of these developments can help achieve the following results:

- manage complexity for manufacturing and transportation, supply chain and other networks;
- “crack” unsolvable problems – provide the ability to bring more factors, criteria, preferences and constraints;
- support collaborative team work with users, forming the balance of interests;
- allow for detailed representation of objects and processes in the real world;
- respond adaptively to real-time events (new order from a VIP client, a road closed, equipment or vehicle late or broken, etc.);
- use domain- and company-specific knowledge to produce feasible schedules, which are in match with conflicting preferences and constraints of decision-makers involved;
- visualize decision-making processes and support user interventions for interactive adjustments of decisions for smart manual recalculation of schedules;
- provide the ability to override constraints and balance cost, profit and service level individually for orders and resources;
- bring visibility and integration of scheduling processes across all the company and business processes;
- provide what-if games and simulations in parallel with the main thread of operational scheduling;
- support activity-based detailed cost calculation for orders and resources;
- address limitations in the existing technology solutions;
- obtain economic effect by optimizing the technical operation strategy for each complex technical object.

VII. Conclusions

The paper considers application of multi-agent approach, knowledge bases and ontologies for resource planning in real time. The key requirements to the system and agent behavior affecting the decision-making process and optimization of the current plans for incoming events are discussed. The paper also shows that multi-agent systems ensure adaptability of the constructed plan, while the use of the domain ontology ensures the flexibility of the system in everyday use. The advantage in using ontologies is matching rules between the concepts of tasks and resources, which makes it possible to take into account peculiarities of new entities, and to carry out planning in view of new tasks and resources. The proposed approach is shown in resource management for supply chain to solve scheduling problems in insurance and managing the maintenance of complex technical objects.

When planning maintenance and repair operations, the use of the knowledge base makes it possible to take into account not only the design features of a complex technical object in operation, but also the current operating conditions and the actual condition, as well as the target plan for the use of the objects under consideration. To prevent object breakdowns, risks are identified in a timely manner and measures are taken ahead of time to minimize these risks by providing technical maintenance in advance. This data is taken from the knowledge base of the history of operation and repair for the specific object. Thus, the system ensures reliability of functioning of complex technical objects.

References

[1] B. Saenz de Ugarte, A. Artiba, et a,. “Manufacturing execution system—a literature review”, *Production Planning & Control*, 20(6), 2009, pp. 525–39.

[2] J. Kletti, “Manufacturing Execution System”, MES, Berlin-Heidelberg: Springer, 2010, 272 p.

[3] B. W. Jeon et al. “An architecture design for smart manufacturing execution system”, *Computer-Aided Design and Applications*, 14(4), 2017, pp. 472–485.

[4] D. Strang, N. Galaske, R. Anderl, “Dynamic, adaptive worker allocation for the integration of human factors in cyber-physical production systems”, *Advances in Ergonomics of Manufacturing: Managing the Enterprise of the Future*, Springer International Publishing, 2016, pp. 517–529.

[5] A. Giret, D. Trentesaux, V. Prabhu, “Sustainability in Manufacturing Operations Scheduling: A State of the Art Review”, *Journal of Manufacturing Systems*, 37(1), 2015, pp. 126–140.

[6] G. Rzevski, P. Skobelev, “Managing complexity”, WIT Press, 2014, 198 p.

[7] J. M. Framinan, R. Leisten, R. R. García, “Manufacturing Scheduling Systems”, London: Springer-Verlag, 2014, p.400.

[8] J. Leung, “Handbook of Scheduling: Algorithms, Models and Performance Analysis”, CRC Computer and Information Science Series, Chapman & Hall, 2004.

[9] S. Vos, “Meta-heuristics: The State of the Art in Local Search for Planning and Scheduling”, A. Nareyek (Ed.), Berlin: Springer-Verlag, 2001, pp. 1–23.

[10] S. Binitha, S. Siva Sathya, “A Survey of Bio inspired Optimization Algorithms” International Journal of Soft Computing and Engineering (IJSCE), vol.2, is., 2012.

[11] F. Xhafa, A. Abraham, “Metaheuristics for Scheduling in Industrial and Manufacturing Applications”, Series Studies in Computational Intelligence, vol. 128, Berlin, Heidelberg: Springer-Verlag, 2008, p. 346.

[12] P. Skobelev, “Multi-Agent Systems for Real Time Adaptive Resource Management”, in *Industrial Agents: Emerging Applications of Software Agents in Industry*, Paulo Leitão, Stamatis Karnouskos (Ed.), Elsevier, 2015, pp. 207–230.

[13] P. Skobelev, “Open multi-agent systems for decision making support”, *Avtometriya, Journal of Siberian Branch of Russian Academy of Science*, n6, 2002, pp.45-61.

[14] P. Skobelev, V. Vittikh, “Models of Self-organization for Designing Demand-Resource Networks”, *Automation and Control, Journal of Russian Academy of Science*, n1, 2003, pp. 177-185.

[15] V. Vittikh, P. Skobelev, “The compensation method of agents interactions for real time resource allocation”, *Avtometriya, Journal of Siberian Branch of Russian Academy of Science*, n2, 2009, pp.78-87.

[16] I. Mayorov, P. Skobelev, “Towards thermodynamics of real timescheduling”, *International Journal of Design & Nature and Ecodynamics*,WIT Press, vol. 10, is. 3, 2015, pp. 213-223.

[17] K. Baclawski, M. Bennett, G. Berg-Cross, D. Fritzsche, T. Schneider, R. Sharma, R. Sriram and A. Westerninen, “Ontology summit 2017 communiqu’e—AI, learning, reasoning and ontologies”. Available: https://s3.amazonaws.com/ontologforum/OntologySummit2017/Communique/OntologySummit2017Communique_v8.pdf.

[18] N. Khilwani, J. Harding, A. Choudhary, “Semantic web in manufacturing”, *Journal of Engineering Manufacture*, 223(7), 2009, pp. 905-924.

[19] Z. Usman, R. Young, N. Chungoora et al., “Towards a formal manufacturing reference ontology”, *International Journal of Production Research*, vol. 51, is. 22, 2013, pp. 6553-6572.

[20] L. Ramos, “Semantic Web for manufacturing, trends and open issues: Toward a state of the art”, *Computers & Industrial Engineering*, vol. 90, 2015, pp. 444-460.

[21] D. T. Meridou, A. P. Kapsalis, M.-E. Ch. Papadopoulou et al, “An Ontology-Based Smart Production Management System”, *IT Professional*, vol. 17, is. 6, 2015, pp. 36 – 46.

[22] Ontology-based manufacturing capacity multi-granularity description framework and method, Patent № CN104615671, 2015.

[23] S. Saeidlou, M. Saadat, E. Amini Sharifi, G. D. Jules, “An ontology-based intelligent data query system in manufacturing networks”, *Production & Manufacturing Research Journal*, vol. 5, is. 1, 2017, pp. 250-267.

[24] M. Klima, J. Gregor, O. Harcuba, V. Marik, “Agent-Based Shop Floor Scheduling”, in *Proc. 8th Int. Conf. on Industrial Applications of Holonic and Multi-Agent Systems (HoloMAS’2017)*, August 28-30, 2017, Lion, France, LNAI 10444, 2017, Switzerland: Springer International Publishing, pp. 24-38.

[25] V. Jirkovsky, P. Kadera, N. Rychtyckyj, “Semi-automatic Ontology Matching Approach for Integration of Various Data Models in Automotive”, in *Proc. 8th Int. Conf. on Industrial Applications of Holonic and Multi-Agent Systems (HoloMAS’2017)*, August 28-30, 2017, Lion, France, LNAI 10444, 2017, Switzerland: Springer International Publishing, pp. 53-65.

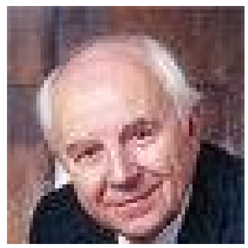
[26] N. Rodrigues, P. Leitão, E. Oliveira, “An Agent-Based Approach for the Dynamic and Decentralized Service Reconfiguration in Collaborative Production Scenarios”, in *Proc. 8th Int. Conf. on Industrial Applications of Holonic and Multi-Agent Systems (HoloMAS’2017)*, August 28-30, 2017, Lion, France, LNAI 10444, 2017, Switzerland: Springer International Publishing, pp. 140-154.

[27] P. André, O. Cardin, “Trusted Services for Cyber Manufacturing Systems”, in *Proc. 7th Workshop on Service Orientation in Holonic and Multi-Agent Manufacturing (SOHOMA17)*, October 19-20, 2017, Nantes, France, in Borangiu T., Trentesaux D., Thomas A., Cardin O. (eds), Service Orientation in Holonic and Multi-Agent Manufacturing, Studies in Computational Intelligence, vol 762, Springer, Cham, 2018, pp. 359-370.

[28] E. Saad, K. V. Hindriks, M. A. Neerincx, “Ontology Design for Task Allocation and Management in Urban Search and Rescue Missions”, in *Proc. 10th Int. Conf. on Agents and Artificial Intelligence (ICAART 2018)*, Funchal, Madeira, Portugal, 16-18 January, vol.2, 2018, SCITEPRESS, pp. 622-629.

[29] P. Skobelev, S. Kozhevnikov, I. Mayorov, D. Poludov, E. Simonova, “Smart Projects: Multi-Agent Solution for Aerospace Applications”, *International Journal of Design & Nature and Ecodynamics*, WIT Press, vol. 12, is. 4, 2017, pp. 492-504.

[30] P. Skobelev, V. Eliseev, I. Mayorov, V. Travin, A. Zhilyaev, E. Simonova, “Designing distributed multi-agent system for aggregate and final assembly of complex technical objects on ramp-up stage”, in *Proc. 10th Int. Conf. on Agents and Artificial Intelligence (ICAART 2018*), Funchal, Madeira, Portugal, 16-18 January, vol.1, 2018, SCITEPRESS, pp. 250-257.



George Rzevski is of Russian origin. His family emigrated from Russia in 1918 and settled in Serbia, where he was born in 1932 and educated at the University of Belgrade. In his late twenties, he was given an opportunity to establish a new design office in Belgrade. He hand-picked his staff employing only talented young engineers and the design bureau grew under his leadership into a major organization capable of undertaking large-scale electrical engineering projects. At the age of 29, George was Chief Designer of all major railways electrification schemes in Yugoslavia. George moved to the UK in the 1960s where he attended a postgraduate refresher course at Imperial College before joining Kingston. Professor Rzevski began his academic career in the UK at Kingston Polytechnic, later Kingston University, where he was Professor and Founder Head of Information Systems. At Kingston, he launched new undergraduate and postgraduate courses aimed at bringing together disciplines of Information Technology (IT) and Business and led a successful research centre in Computer-Integrated Manufacturing. The Centre worked in close cooperation with leading high-technology companies, including ICL, Xerox and IBM.

He is an academic, entrepreneur and consultant based in London, UK. He is Emeritus Professor, Centre for Complexity and Design, The Open University, Milton Keynes, UK and Executive Chairman, Multi-Agent Technology Ltd, London, Helsinki, Moscow and Samara.

The list of his publications includes more than 110 works. His main publications are:

1. G. Rzevski, P. Skobelev, “Managing complexity”, WIT Press, 2014, 198 p.;
2. G. Rzevski, C. Brebbia (eds), “Complex Systems in Business, Administration, Science and Engineering II”, WIT Press, 2016;
3. G. Rzevski, C. Brebbia (eds), “Complex Systems: Fundamentals and Applications”, WIT Press, 2016.

For several years, George has delivered a regular series of lectures on Economic, Social and Cultural Implications of Global Networks to postgraduate students at London School of Economics. George acted as Visiting Professor at London School of Economics; Brunel University, West London; Wuhan University, China; Cologne University of Applied Sciences, Germany; and Moratuwa University, Colombo, Sri Lanka and he serves on the Board of Directors, Wessex Institute of Technology, New Forest, England. He served as editor-in-chief of the Journal of Artificial Intelligence in Engineering, published by Elsevier. Together with his long-term business partner, Professor Petr Skobelev from Russia, Professor Rzevski has founded a network of advanced technology companies, which include Magenta Corporation, London, UK, and Samara, Russia; Rzevski Solutions Ltd, London; Knowledge Genesis Ltd, London, UK, and Hanover, Germany; Smart Solutions Ltd, Samara, Russia; and Multi-Agent Technology Ltd, London, UK, Helsinki, Finland and Moscow, Samara, Russia. The companies are engaged in developing and marketing complex adaptive systems. George and Petr created advanced multi-agent technology, which their companies use for developing complex adaptive systems.

Until his retirement in 1999, George was Professor in Department of Design and Innovation at The Open University, Milton Keynes, where he was Director, Centre for the Design of Intelligent Systems. His Centre was well funded by grants from government and industry and his Department was rated 5 out of 5 in the two UK Research Assessment Exercises. As a tribute to his successful research career, the University established “The George Rzevski Complexity Science Laboratory”.

Skobelev Petr Olegovich, born in 1960. Prof. Skobelev graduated from Samara Aerospace University in 1983. He obtained a PhD degree in Functional Programming Languages and Artificial Intelligence for Aerospace Applications in 1986 and the Professor degree in Multi-agent Systems for Real-time Resource Management in 2003 at the Institute for Control of Complex Systems of the Russian Academy of Sciences. Areas of his scientific interests include multi-agent technology, distributed intelligence, collective management.

Prof. Skobelev is the President of the Group of companies “Knowledge Genesis” in Samara, Russia (with the daughter company - Software Engineering Company “Smart Solutions”). Besides, he is a member of the Russian Association for Artificial Intelligence and a member of the IEEE International Committee for Industrial Agents. Prof. Skobelev is the head of Laboratory of Intelligent Systems and Technologies at the Institute of Control of Complex Systems of the Russian Academy of Sciences, Professor at Samara National Research University and the head of Department of Electronic Systems and Information Security of Samara Technical University.

The list of his publications includes more than 150 works in the fields of the theory of complex adaptive systems, new approaches to enterprise resource management, emergent intelligence (swarm intelligence), multi-agent technology, knowledge bases, systems of systems, intelligent Internet of people and things (Internet of agents), and digital platforms for creating eco-systems of smart services.

His main publications include the following:

1. G. Rzevski, P. Skobelev, Managing complexity, WIT Press, 2014, 198 p.;

2. P. Skobelev, Multi-Agent Systems for Real-Time Adaptive Resource Management. in Industrial Agents: Emerging Applications of Software Agents in Industry, Paulo Leitão, Stamatis Karnouskos (Ed.), Elsevier, 2015, pp. 207-230;

3. V. Gorodetsky, P. Skobelev, Industrial Applications of Multi-agent Technology: Reality and Perspectives, SPIIRAS Proceedings, is. 6 (55), 2017, pp. 11-45.

Zhilyaev Alexey Alexandrovich, born in 1992. Graduated from Samara State Aerospace University in 2014. Area of scientific interests - multi-agent systems for resource management.

He is a post-graduate student of the Department of Aircraft Design of Samara National Research University, and senior programmer at SEC "Smart Solutions".

His list of publications includes 36 papers. The main publications are:

1. I. Belokonov, P. Skobelev, E. Simonova, V. Travin, A. Zhilyaev, Multiagent planning of the network traffic between nanosatellites and ground stations, Proceda Engineering: Scientific and Technological Experiments on Automatic Space Vehicles and Small Satellites, 104 (2015), Elsevier, pp. 118-130;

2. P. Skobelev, E. Simonova, A. Zhilyaev, Using the multi-agent technology for distributed management of the cluster of remote sensing satellites, International Journal of Design & Nature and Ecodynamics, WIT Press, vol. 11 (2016), is. 2, pp. 127-134;

3. P. Skobelev, V. Eliseev, I. Mayorov, V. Travin, A. Zhilyaev, E. Simonova, Designing distributed multi-agent system for aggregate and final assembly of complex technical objects on ramp-up stage, Proceedings of the 10th International Conference on Agents and Artificial Intelligence (ICAART 2018), Funchal, Madeira, Portugal, 16-18 January, 2018, vol.1, SCITEPRESS, pp. 250-257.

Oleg Ivanovich Lakhin, born in 1976. Graduated from the Volga Institute of Informatics, Radiotechnics and Telecommunications in 1998. In 2017, he obtained his PhD degree in technical sciences on the topic: "Managing the cargo flow of complex remote-based technical objects on the basis of multi-agent technology".

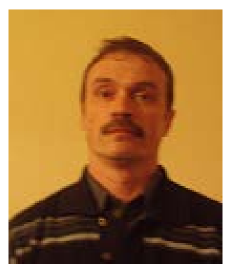
Areas of his scientific interests include multi-agent systems for resource management, development of knowledge support systems for aerospace enterprises, ontology modeling of the domain.

He is CEO at SEC “Smart Solutions” LLC. He is the co-author of more than 60 publications. His main publications are:

1. P. Skobelev, O. Lakhin, I. Mayorov, Method of Adaptive Cargo Flow Scheduling for ISS RS Based on Multi-Agent Technology, V. Marik et al. (Eds.), Proceedings of the 8th International Conference on Industrial Applications of Holonic and Multi-Agent Systems (HoloMAS'2017), August 28-30, 2017, Lion, France, LNAI 10444, Springer International Publishing, Switzerland, 2017, pp. 3-10, 2017;

2. G. Rzevski, J. Knezevic, P. Skobelev, N. Borgest, O. Lakhin, Managing Aircraft Lifecycle Complexity, International Journal of Design & Nature and Ecodynamics, WIT Press, vol. 11 (2016), is. 2, pp. 77-87;

3. I. Iurygina, A. Anisimov, O. Lakhin, et al., Ontology and Multi-Agent Technology Application for Intellectual Property Knowledge Management, Indian Journal of Science and Technology, Vol. 9 (46), 2016, pp. 35-43.



Mayorov Igor Vladimirovich, born in 1963. Graduated from Samara State University in 1989. In 2017, he obtained his PhD degree in technical sciences on the topic: "Multi-agent models and technologies for situational management of enterprise resources in conditions of uncertainty". Areas of his scientific interests include management of complex systems, multi-agent systems, real-time modeling.

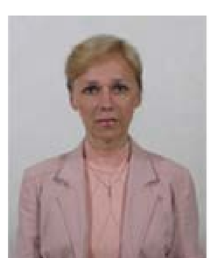
He is the head of the department for development of mathematical models, methods and algorithms at SEC "Smart Solutions" LLC.

He has published more than 40 papers in Russian and foreign journals. His main publications are:

1. V. Gorodetsky, O. Bukhvalov, P. Skobelev, I. Majoryov, Current state and prospects of industrial applications of multi-agent systems, Control of large systems, Moscow: IPP RAS, Issue 66, 2017, pp. 94-157;

2. I. Mayorov, P. Skobelev, Towards thermodynamics of real-time scheduling, International Journal of Design & Nature and Ecodynamics, WIT Press, vol. 10 (2015), is. 3, pp. 213-223;

3. P. Skobelev, O. Granichin, D. Budaev, V. Laryukhin, I. Mayorov. Multi-agent tasks scheduling system in software defined networks, IOP Journal of Physics: Conference Series, IOP Publishing, 510 (2014) 012006.



Simonova Elena Vitalevna, born in 1962. Graduated from Samara State Aerospace University in 1985. In 1994, she obtained her PhD degree in technical sciences in the field of simulation of complex technical systems. Areas of her scientific interests include development of intelligent decision support systems, ontological representation of knowledge, and development of multi-agent technologies.

She is an associate professor of the Department of Information Systems and Technologies at Samara National Research University, and the leading analyst at SEC "Smart Solutions".

The list of her publications includes more than 100 papers. The main publications are:

1. P. Skobelev, S. Kozhevnikov, I. Mayorov, D. Poludov, E. Simonova, Smart Projects: Multi-Agent Solution for Aerospace Applications, International Journal of Design & Nature and Ecodynamics, WIT Press, vol. 12 (2017), is. 4, pp. 492-504;

2. P. Skobelev, E. Simonova, A. Zhilyaev, V. Travin, Application of the Multi-Agent Technology in the Scheduling System of the Swarm of Earth. Remote Sensing Satellites, Procedures Computer Science, Elsevier, 103 (2017), pp. 396-402;

3. P. Skobelev, O. Lakhin, A. Polnikov, E. Simonova, Approach to the solution of aerospace product lifecycle management problem based on network-centric principles, V. Marik, J.I. et al. (Eds.), Proceedings of the 7th International Conference on Industrial Applications of Holonic and Multi-Agent Systems (HoloMAS'2015), September 2-4, 2015, Valencia, Spain, LNAI 9266, Springer International Publishing, Switzerland, 2015, pp. 169-178, 2015.