Ontology-Driven Multi-Agent Engine for Real Time Adaptive Scheduling

George Rzevski
Digital Eco-Systems
Imperial Colledge Incubator
London, UK
rzevski@gmail.com

Petr Skobelev
Knowledge Genesis Group
Institute for the Control of Complex Systems of Russian Academy of Sciences
Samara, Russia
petr.skobelev@gmail.com

Alexey Zhilyaev
Samara National Research University
Samara, Russia
zhilyaev@smartsolutions-123.ru

Oleg Lakhin
Knowledge Genesis Group
Samara, Russia
lakhin@smartsolutions-123.ru

Igor Mayorov
Institute for the Control of Complex Systems of Russian Academy of Sciences
Samara, Russia
imayorov@smartsolutions-123.ru

Elena Simonova
Samara National Research University
Samara, Russia
Simonova.Elena.V@gmail.com

***Abstract*—The growing demand for improving business efficiency requests the development of generic resource management systems applicable for solving a wide range of complex problems with minimum cost and time. However, the classical combinatorial or heuristic methods and tools do not provide adequate solutions for solving complex problems of resource management in real time. That is why we consider multi-agent technology as the core part of such solutions – which helps find the balance of many interests and adapt it in a flexible way to unpredictable events, such as a new order, an unavailable resource, etc. In this paper we introduce the use of ontology for scheduling, which provides the opportunity to create ontological model of the enterprise, develop generic multi-agent scheduler and customize matching requirements for each operation in business or technological processes, for example, for applications in manufacturing, project management, supply chains, etc. Semantic Wikipedia on the top of ontology editor will be discussed to support knowledge base of enterprise for resource management. The example of applications for supply chain of insurance company is presented.**

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

I. Introduction

At the present stage of development of production management systems and resource allocation and planning systems, there are significant difficulties typical for enterprises operating in conditions of uncertainty. Uncertainties are associated with the problems of taking into account a variety of heterogeneous factors, such as the need for an individual approach to each order, dynamics of prices for components, variability of supplies from subcontractors, failure to meet deadlines, continuous change in technology, occurrence of various events directly affecting production processes of the enterprise as a whole, etc.

The existing MES systems have wide functionality and integration with executive modules. They support production processes from release to delivery. However, they lack tools supporting decision-making and reacting to dynamic 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 the time and nature of changes, becomes a vitally important task. Modern integrated production systems should respond flexibly and quickly to events, take into account the distributed nature and lack of centralization in most control subsystems, independently make decisions on correction and parrying of unforeseen situations and issue updated agreed plans [5]. The main principles of constructing and implementing such systems are considered in [6]. Solution of such a poorly formalized task 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 recalculate schedules and indicators for hundreds of interrelated production resources and thousands of manufactured products [7-11].

Therefore, in order to take into account unforeseen events, reduce and parry uncertainties in price dynamics and resource availability, multi-agent systems are widely used, in which the current schedule is built by means of negotiating agents in demand-resource networks on the basis of self-organization [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 in 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 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 work was supported by the Russian Foundation for Basic Research under grant #16-01-00759.

The agent of the new order, having entered the scene, refers to the enterprise agent with his requirements. The enterprise agent activates agents of units that create agents of technological processes and agents of operations and subtasks, which send requests for resources - machines, equipment, materials. Agents perform matching according to types, performance and personnel requirements. Resources (machines, specialists) analyze local schedules and check for the possible locations. 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.

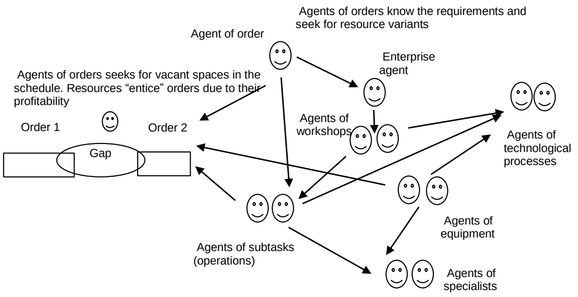

Fig. 1. The DR-world scene.

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

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections A-D below for more information on proofreading, spelling and grammar.

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 $f_{ij}^{task}(x_i-x_{ij}^{id})$. 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 $f_{kl}^{res}(z_k-z_{kl}^{id})$, 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-Resources network the optimization problem is formulated as solving (1) - (3).

The developed method is based on DRN concept where agents operate in the 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 agent 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.

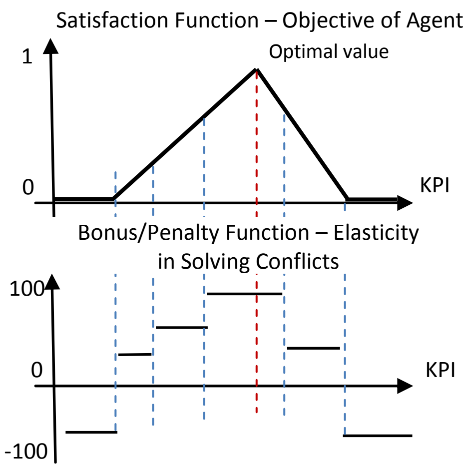

Fig. 2. Example of satisfaction function.

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 be not worsen.

Different protocols can define different methods of conflicts solving.

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

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 agents 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 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 time available is exhausted.
7. At this moment Agent of System will revise system KPIs and find agents which have minimum satisfaction for most important criteria.
8. Agents of System propose 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 drop the operations - as a result, the plan will be automatically revised and rescheduled.

The general behavior of complex systems is evolving from agents 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 the 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 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 schedule and for this aim they have virtual accounts and use virtual money (can be converted to real money).

For example, a truck has a certain cost, but it is distributed between its cargoes with some planned profitability given in advance. As it is stated before, 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 define 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 best possible options.

To achieve the best possible results agents use the virtual money that regulates their behaviour. 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 and delivery time and some other criteria;
- Resource agents are looking for their maximum utilization but also have their own ideal preferences and constraints and costs which they share between orders;
- Product agents are interested to maximize quality and minimize 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 total value and profit of the company is growing.

Such an approach gives the opportunity to introduce virtual taxes related to agents' jobs 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 good 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 (trucks or drivers) 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 [16].

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

A. Ontology-Driven Knowledge Base for Adaptive Resource Management

Transition to the digital economy involves development of methods and means of formalized knowledge representation, which should enable unified construction 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 include description of the object of enterprise activity, requirements for product and process quality, composition of units to the level of each employee with their inherent competencies, the used tools, technological or business processes and other regulatory and technical documentation, regulations of interactions between departments and specialists, results of model and full-scale testing of facilities, etc.

Our work shows that such knowledge can be effectively stored in the form of ontologies and applied in enterprise resource management. For practical application of this approach, a number of fundamental scientific tasks are required. What should the structure of enterprise ontology be? What kind of tools should be used to create industrial knowledge bases based on ontologies? What mechanisms should be embedded in the work of such Knowledge Base to ensure automation of knowledge processing and integration with existing databases? What is the life cycle of their functioning?

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

In this regard, it becomes relevant and important 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, it is 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, noncontradictory, clear, transparent, and open to change.

The current trends in development of smart production are associated with the use of such methods as ontological knowledge representation, knowledge base, machine learning, etc., which are very important and promising for the use in industry.

[17] presents a smart multi-agent production management system based on ontologies. 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.

[18] dwells on the method for describing production resources on the basis of a multilevel ontology of production resources, as well as a multi-level ontology for managing production resources. Using ontology ensures the accuracy and completeness of the search for production resources. However, this method does not provide customization of individual features of tasks and production resources in the ontology of resource allocation for management and planning in the enterprise.

[19] describes application of ontology in the distributed semantic network of the enterprise's production departments for the set-up of an efficient and scalable system of interaction and retrieval of data on information requests.

Let us briefly consider application of ontologies in production, which were presented at the HoloMAS 2017 conference, devoted to application of MAS in production, at the conference SOHOMA 2017, which examined concepts, methods and solutions aimed at development of management technologies with distributed intelligence based on agents, as well as at the ICAART 2018 conference, which, among others, considered the use of ontologies to represent knowledge in multi-agent systems [20-24] .

[20] dwells on the possibility of using ontologies for dividing complex problems into composite, autonomous, ones. However, an ontological description of individual operations is not made.

[21] shows that 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. The paper 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.

[22] demonstrates the possibility for configuration of services built on ontologies. It shows the prototype for a flexible manufacturing system case study to be allowed to verify the feasibility of greedy local service reconfiguration for competitive and collaborative industrial automation situations.

[23] shows the possibility of coordination and control of such complex systems 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 important example of ontology application is given in [24] 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 between 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 tool for managing applications. However, the possibility of a detailed description of problems has not been investigated.

One of the new steps in our multi-agent developments was 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 arrive to users.

For example, for solving the problem of shop floors scheduling for airplane factory it could be the knowledge about 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, etc. [25].

As the first step of this development, we designed "Basic Ontology of Scheduling" 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-A-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).

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 entities required for adaptive scheduling of orders to resources:

- Organization Chart of Enterprise which specifies what resources are available at 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.

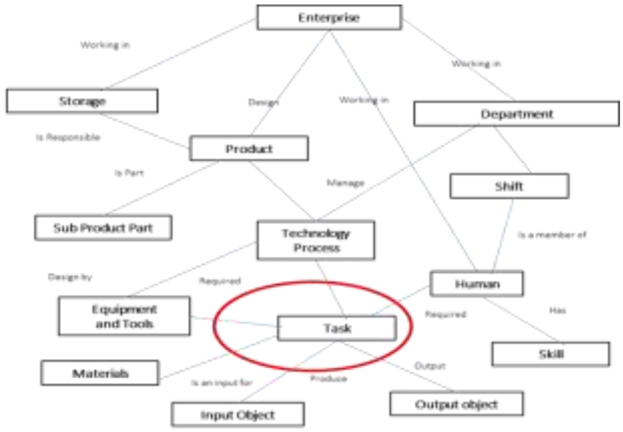

Fig. 3. Fragment of basic Ontology of Scheduling.

The main 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) [26].

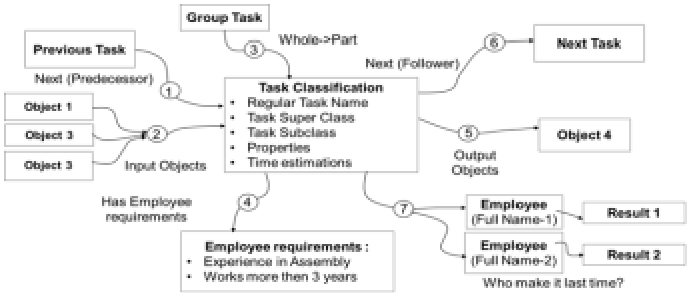

Fig. 4. Key Types of Relations in the Concept of "Task".

The key idea here was to develop Task Agent that can read and interpret relation "Required" as a matching rule – to find 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 Scheduler was developed which can be customized by Knowledge Base for any type of enterprise.

The key option to add new relations in 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.

B. The functionality of the solution

In practice the discussed solutions should be designed as event-driven systems – the list of such events includes but is not limited to such examples:

- 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.

However, the most important idea is that in order to provide event-driven adaptability, such solutions must support autonomous cycle of resource management, presented in Fig. 5.

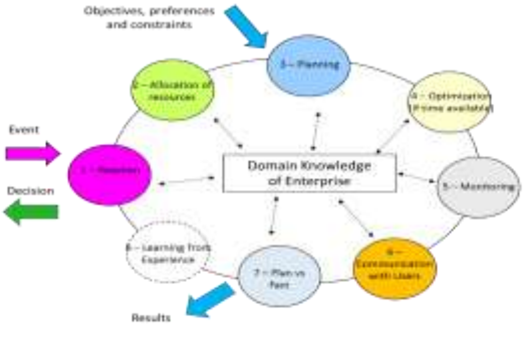

Fig. 5. Autonomous cycle of resource management.

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.

C. Example: 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 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 fell ill;
- A client was not on the site when assessor arrived;
- A new urgent claim arrived.

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 specific for Insurance business but not to the specific enterprise.

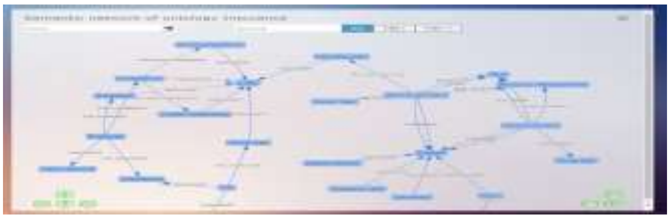

Fig. 6. Fragment of Ontology for Insurance Supply Chain.

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.

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. 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 of time:

- 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. 8-10 where it is shown how key operations were shifted and reallocated to different available resources.

Fig. 8. Initial Schedule of “Insurance Company”.

Fig. 9. Status of Materials in Storage.

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.

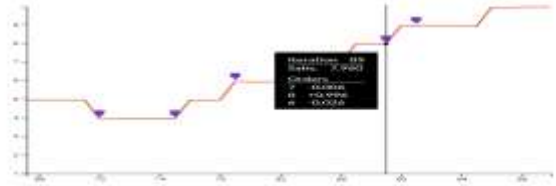

Fig. 11. Value Chart reflecting Value of Events and Decisions of Agents.

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.

VI. The Results for Adaptive Resource Management

The objectives of developments can help achieve the following results:

- Manage complexity for manufacturing and transportation, supply chain and other networks;
- “Crack” unsolvable problems – provide ability to bring more factors, criteria, preferences and constraints;
- Support collaborative team work with users forming balance of interests;
- Allows for detailed representation of objects and processes in the real world;
- Respond adaptively to real-time events (new order from VIP client, 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 users interventions for interactive adjustments of decisions for smart manual re-work of schedules;
- Ability to override constraints and balance cost, profit and service level individually for orders and resources;
- 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;
- Addresses limitations in existing technology solutions.

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 subject area ontology ensures the flexibility of the system with respect to accounting and overcoming of constraints and construction of a consensus solution. The advantage in using ontologies is building and accounting the links between the concepts of tasks and resources, which makes it possible to take into account peculiarities of new tasks, and to carry out planning in view of new descriptions of tasks and resources. The proposed approach is shown in planning of resource management tasks and decision-making in supply chain to solve scheduling problems in insurance.

References

[1] B. Saenz de Ugarte, A. Artiba, et al. "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] 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.

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

[19] 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.

[20] M. Klima, J. Gregor, O. Harcuba, V. Marik. "Agent-Based Shop Floor Scheduling", 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, 2017, Switzerland: Springer International Publishing, pp. 24-38.

[21] V. Jirkovsky, P. Kadera, N. Rychtyckyj. "Semi-automatic Ontology Matching Approach for Integration of Various Data Models in Automotive", 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, 2017, Switzerland: Springer International Publishing, pp. 53-65.

[22] N. Rodrigues, P. Leitão, E. Oliveira. "An Agent-Based Approach for the Dynamic and Decentralized Service Reconfiguration in Collaborative Production Scenarios 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, 2017, Switzerland: Springer International Publishing, pp. 140-154.

[23] P. André, O. Cardin. "Trusted Services for Cyber Manufacturing Systems", Proceedings of the 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.

[24] E. Saad, K. V. Hindriks, M. A. Neerincx. "Ontology Design for Task Allocation and Management in Urban Search and Rescue Missions", Proceedings of the 10th International Conference on Agents and Artificial Intelligence (ICAART 2018), Funchal, Madeira, Portugal, 16-18 January, vol.2, 2018, SCITEPRESS, pp. 622-629.

[25] 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.

[26] 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, vol.1, 2018, SCITEPRESS, pp. 250-257.

Published: George Rzevski, Petr Skobelev, Alexey Zhilyaev, Oleg Lakhin, Igor Mayorov, Elena Simonova. Ontology-Driven Multi-Agent Engine for Real Time Adaptive Scheduling // Proceedings of the International Conference on Control, Artificial Intelligence, Robotics and Optimization (ICCAIRO 2018), Prague, Czech Republic, May 19-21, 2018. – https://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=8684744 –

IEEE Xplore, IEEE. – P. 14-22. https://ieeexplore.ieee.org/document/8698403. DOI: 10.1109/ICCAIRO.2018.00011.