

Basic Trends of Decentralized Artificial Intelligence

V. I. Gorodetsky^{a,*}

^a Joint Stock Company EUREKA St. Petersburg, 196006 Russian Federation

*e-mail: vladim.gorodetsky@gmail.com

Abstract—Analysis of modern trends, advanced concepts, and projects in the field of information technologies shows the growing role of artificial intelligence, primarily decentralized artificial intelligence. The paper analyzes the advanced concepts of building new generation applications in the field of information technologies and relevant applied developments and provides a brief analysis of modern achievements in the field of decentralized artificial intelligence and self-organization, which are able to support the practical implementation of such applications. A simplified analog of the roadmap for the development of decentralized and self-organizing artificial intelligence for the near future has also been formulated.

Keywords: IT applications, decentralized artificial intelligence, networked systems, self-organization, peer-to-peer interactions, p2p-communication environment, decentralized machine learning, group control, active data and knowledge

DOI: 10.1134/S105466182303015X

INTRODUCTION

A new wave of interest from the industrial community in the use of artificial intelligence (AI) technologies has a strong impact on the priorities and trends of its development. The modern development of the theoretical and technological basis of AI is mostly limited to centralized architectures and stand-alone applications in the field of machine learning for computer vision, speech processing and voice assistants, natural language, classification, and prediction tasks within business problems of various purposes. AI applications of other classes are much less common and usually at the prototype level. There are many applications in the field of information technology (IT) that claim to use AI without having convincing arguments for doing so.

However, the needs for IT in the use of the latest AI technologies are constantly growing, and this is one of the notable trends in the IT industry, and the leader of these trends is decentralized AI [5]. This is due to the features of a noticeable part of modern applications, which usually consist of a large number of autonomous mobile entities that intensively interact in the process of collaborative problem solving. These include a large number of new applications of the Internet of things and group control of mobile unmanned objects. The globalization of industry and business is also leading to the growing role of distributed and decentralized architectures. There are and most likely will be new classes of relevant applications that will need to solve the problems of

decentralized group control in real time. Military applications, space, harmful environments, natural disasters, and other areas are potential sources of new IT applications and new requirements for AI capabilities.

What all these systems have in common, from a formal point of view is that they operate in a communication environment with a dynamic topology, and therefore each such application is a dynamic network.

In the last few years, new projects and new paradigms of computing in dynamic networks have appeared in IT, which, on the one hand, rely on the achievements of AI and, on the other hand, set new benchmarks and prospects for its development. As many authors believe, these landmarks are most concentrated in *the paradigm and architecture of peripheral computing*, in the concept of *decentralized business* without intermediaries with peer-to-peer (p2p) and blockchain technologies at the core, in a new version of the World Wide Web called *Web3*, and also in a project called *the metaverse* [5].

These projects have already received some development [3, 5, 15, 24, 25, 34], and IT professionals associate them with the future of IT technologies and IT applications of the next-generation, and they also believe that the development of AI, at least in the forthcoming years, will be determined by the requirements on their part. The value of these projects lies in the fact that, on the one hand, they set in a concentrated form the basic requirements for the level of AI technologies of the near future, and, on the other hand, they will become sources of new ideas and tech-

nologies and will stimulate the creation of new tools to support the development of a new generation of IT applications such as those which were listed above and even more difficult ones.

The paper [5] based on the analysis of a new challenges that are now being addressed by the IT paradigms focuses only on the importance of research and development in the field of decentralized AI technologies. The objective of this paper is to take the next step, namely, (1) to identify in these paradigms a basic set of specific difficult problems, the solution of which requires the involvement of ideas and methods of decentralized AI; (2) to conduct a brief analysis of the results, which decentralized AI has already at present; and (3) to focus on some of the most important tasks, which it would makes sense to bring to the attention of researchers and developers in the near future. In the rest of the paper, Section 1 summarizes the objectives and key features of the new projects and it paradigms mentioned above; Section 2 provides minimum information about multi-agent systems (MASs) and self-organizing control principles that form the context needed to understand the subsequent material. Section 3 lists the achievements in the field of decentralized AI and self-organization, which are now ready for use. Section 4 analyzes promising research directions in the field of decentralized AI and self-organization, which will significantly expand the areas of their practical use in IT applications, at least in the near future, ensuring their computational efficiency and fault tolerance. In Conclusion, the main results of the paper are summarized.

1. NEW IT PARADIGMS AND REQUIREMENTS FOR AI CAPABILITIES

Below is a summary of new projects, concepts, and developments in the field of IT, which, according to experts' opinions, determine the requirements for the level of AI in the near future, at least [5].

Peripheral computing. This is the architecture of decentralized networked computing. In this architecture, data processing is carried out as close as possible to their sources and to the users of the results. This architecture is of particular importance for mobile networks, since the transfer of computing to the periphery increases the speed of data processing and reduces delays caused by the limited resources of mobile devices, including power consumption [34]. Systems built on this architecture can generally have different combinations of centralization and decentralization. Individual local subsystems can use a decentralized architecture, but if there is a hierarchy in the network, some functions can be performed in a centralized version. Therefore, peripheral computing

systems can have a hybrid architecture. A detailed description of this architecture and its properties can be found in [5, 34].

Decentralized business. In this concept, business operates without intermediaries, i.e., on the basis of direct peer-to-peer interactions of its participants. Its most developed version is the decentralized financial business (DeFi). It is an alternative to the traditional financial business and its instruments. In it, all transactions with financial assets are carried out without intermediaries [5, 15].

DeFi manipulates *digital assets* (DeFi tokens), which are understood as entities, perhaps abstract, that have value in the financial market. One example of digital assets is cryptocurrencies. Another example is nonfungible tokens (NFT¹). They are similar to securities, but they are digital. Not everyone can copy them, because they exist in their ecosystem.

The next concept of DeFi is *a digital wallet*, and in fact, it is a user interface for accessing and managing his assets. Another concept of DeFi is called *a smart contract*, which means the procedure for performing certain actions in the DeFi ecosystem, for example, the action to transfer a digital asset from one market participant to another. Manipulations with these DeFi components are subject to certain rules, which must be automatically supported by the DeFi ecosystem. This support is done through the use of peer-to-peer interaction protocols. Examples are rules for performing actions such as “transaction generation,” “data verification and decision making,” and “save the data.” This DeFi chain of operations is performed without any intermediaries [15].

The implementation of DeFi-business is based on blockchain technology [3]. Blockchain is a chain of digital blocks following each other (transactions) connected by a common context. Note that the use of blockchain technology is not limited to the storage of transactions. It is widely used in other tasks, which is due to a number of useful properties that this technology has. These include decentralized storage and execution of operations with digital assets, transparency (access to details and to the trajectory of transactions), immutability of data during storage, the absence of intermediaries (their role is played by consensus protocols), anonymity (authentication of the user without revealing his identity), the ability to audit, built-in protection against attacks, and fault tolerance.

¹ An example of NFT is digital art. Beeple's digital work, 5000 days, was sold at auction for 69 million dollars; see <https://www.heverge.com/2021/3/11/22325054/beeple-christies-nft-sale-cost-everydays-69-million>.

Summarizing the above description of the features of decentralized business, we can conclude that the support of decentralization based on peer-to-peer interactions of objects in a p2p-communication environment determines the essence of the requirements put forward by them in relation to AI.

The concept of Web3 [24]. This concept aims to develop the World Wide Web toward decentralizing content storage. Due to this concept, IT giants such as Google, Amazon, and Apple will lose their monopoly on the storage of content and control over it. The owner of the content in Web3 stores it on his own side, and therefore this content is protected from external blocking. This increases the level of trust in content and its security, makes less vulnerable the transfer of digital businesses such as DeFi to the Internet environment, and expands the opportunities of the digital economy based on tokens as digital assets. It is commonly said that *Web3* is a transition from human texts to texts for computers that can analyze them, make conclusions, use them in the interests of machine learning, etc. Obviously, in the concept of *Web3*, the term *decentralization* is also key in terms of the required properties of IT and AI technologies.

Metaverse. This project expands the concept of cyberspace, adding to it digital twins of the components and aspects of the physical world and the interaction between them. This leads to the creation of a virtual world parallel to the physical world [25]. Judging by the stated goals, the project is preparing a digital explosion of cyberspace. Its predecessors are virtual environments such as social networks, video conferences, augmented reality systems, etc.

In the metaverse, the people of the physical world interact with each other and with this environment through their intermediaries—digital avatars [25]. The creation and development of the application of the metaverse takes place in three stages:

- creating digital twins in a virtual environment;
- the settlement of its “digital aborigines” and the development of digital twins with their participation;
- parallel coexistence, interaction, and mutual enrichment of both worlds and their practical use [25].

According to this scheme, some set of digital twins are first created for a number of aspects of the real world. At this stage, they are called “shadows” of aspects of reality in the virtual world. Further, real-world humans, through their avatar digital aborigines, create connected ecosystems, encompassing such aspects of the physical world as science, culture, economics, laws, and social norms. They are created by analogy with the same ecosystems of the physical world. At the beginning of the development of the metaverse, the digital twins are loosely coupled with similar physical worlds, but over time, digital aborigines establish connections within the ecosystems of the

virtual world and their connections with similar components of the physical world, thus gradually forming a virtual world and a single physical-virtual world with the metaverse as its part.

The worlds interact with data. Data of the physical environment from sensors, from social networks, media, etc., are fed to the input of digital twins, changing their states and initiating dynamic processes. This data can be used for a variety of purposes, including machine learning. Digital twins can generate model data about the real world, which can replenish information about the real world, creating a more complete picture of it in this way. This allows us to more accurately solve the problems of predicting events, facts, and processes of the real world.

Analyzing the requirements for AI technologies necessary to create and use the metaverse, we can conclude that they combine the requirements of other concepts from the list above, and even these concepts themselves.

2. MULTI-AGENT TECHNOLOGIES AND SELF-ORGANIZATION: GENERAL INFORMATION

The analysis of the requirements for the directions of development of AI from the side of next generation of IT applications, the composition of which can be imagined according to the content of the previous section, is advisable to carry out, supported by at least minimal assumptions about the properties of these applications, for example, about architecture. As noted in the introduction, the new most popular and important classes of IT applications are usually network structures of dynamic topology with a large number of nodes and links of various semantics. These networks are characterized by the fact that their different nodes possess different information and are able to solve different problems; they intensively interact with each other and the external environment in solving various problems, many of which are solved jointly by groups of nodes. In the architectures of their software implementation, network nodes are usually put in line with the objects of the application, for example, unmanned aerial vehicles, robots in collective robotics, satellites in the space-based surveillance system, etc.

For such systems, the most suitable architecture and development technology is the architecture and technology of the autonomous agent and MAS. This is due to the fact that the concept of an agent on default assumes autonomy and the ability for proactive behavior as its basic properties. Autonomy and proactivity together make it possible to formalize the complex behavior of many autonomous, possibly mobile network objects, which depends not only on the current inputs of each object but also on the prehistory of the state of the external environment and on the behavior

of other objects of the system and their neighbors on the network. For example, a proactive agent can generate messages even if there are no input events, e.g. due to a timeout exceeding. Another important property of the agent, essential for the implementation of networked systems comprising many objects, is its interactivity. Interactivity is defined as the ability of agents to exert particular influence on each other, and it is in this sense that the network of MAS agents is called “weakly-coupled.” As a side effect of agents’ autonomy and interactivity, their ability to solve complex problems through cooperative coordinated behavior emerges [12].

From this brief listing of agent properties, it follows that the agent model for network node individual behavior specification and the MAS model for specification of networked object behavior as a whole are generally well suited for IT applications of the class in question. In addition, another convincing argument in favor of such a choice is that the behavior of the proactive agent is usually specified by the finite-state machine, and the network of such interacting agents is conveniently modeled by a network of such interacting finite-state machines.

In the described architecture, a software agent representing a particular stand-alone application object, on the one hand, controls *the internal behavior* of the software and/or hardware components of “its” own network node in various use cases. On the other hand, this software agent is a representative of “its node” in a network of software agents, where its function is to interact with agents of other nodes of the network by exchanging messages to coordinate the behavior of “its own node” in the joint solution of some common task.

Large-scale network structure systems with a decentralized storage and computing architecture typically allow only the interaction of “neighboring” nodes of the network (agents representing these nodes in the network software environment), for example, in resource planning and group control, regardless of whether the “neighborhood” is determined by the structure of physical communication channels between them or, for example, overlay network in the case of a software-defined network. The individual node of the network (the agent representing it) may not know at all about nodes (agents) that are not its neighbors. Therefore, all calculations in such networks are performed on the basis of local interactions and self-organization principles.

Self-organization is defined as a dynamic process of a system, implemented in it without external interference on the basis of local interactions of its objects, which leads to the emergence and maintenance of a structure on a set of its objects [17, 33]. Self-organizing systems also have a number of specific properties, which include autonomy, global order arising owing to local interactions, emergent behavior, possible instability that occurs without external influences, sensi-

tivity to initial states and small variations of parameters with a jump change of state, multiplicity of stable states (attractors), adaptability as the ability to change behavior and structure when changing local inputs, and the potential complexity of the behavior of simple network objects as a consequence of their large number.

Most of the developed prototypes of self-organizing systems are implemented in the MAS architecture, and this is not accidental. Indeed, the basic requirements for the software implementation of such a system are that its components must be autonomous (be able to control their own behavior aimed at achieving their local goals without external intervention). They should be able to perceive the external world and locally effect it, be able to interact with their neighbors (on the network, in space, etc.), maintain the emergent structure of the system, and have the means to control their behavior. Until now, the MAS has been and remains the alternativeless architecture for the software implementation of self-organizing systems. Moreover, at present, the development of principles and models of self-organization takes place within the framework of research and development in the field of MAS [11, 17].

3. DECENTRALIZATION AND SELF-ORGANIZATION OF AI SYSTEMS. LEVEL OF MODERN ACHIEVEMENTS

In studies on the designated topic, two waves of activity can be distinguished. The first of these covers the period from about the mid-1980-ies to the end of the 1990s. During this period, the main themes of these research and active development, which were funded by DARPA (USA), were the problems of group control of teams of autonomous agents performing a common mission. The second wave dates back to 2002–2010, and its main focus was on two key problems of decentralized AI, namely, decentralized machine learning and the problems of creating p2p-infrastructures to support the p2p-interaction of autonomous network agents in a p2p-communication environment. Around 2000, active research and development began in the field of self-organization as the basic principle of adaptive control in large-scale applications of the network structure. Active research in this area continued until about 2015. It can be argued that four of these problems (group control, decentralized machine learning, p2p-communication networks, and self-organization) at different times were the focus of research related to decentralized AI. Other relevant problems are more specific and the results obtained in them are not analyzed here. Let us take a look at the main results so far on the four issues mentioned.

In the field of *group control* (teamwork), by the mid-1990s, several theories were proposed, but only two of them laid the *theoretical* basis for subsequent and modern models and software tools in this field. Pioneering was the work [9], in which the *Joint Intentions Theory* was proposed. It formulates the basic concepts and general framework that define the group behavior of agents and the characteristics of their interaction in this behavior, as well as the principles of information exchange that can support their situational awareness necessary for decentralized coordination of their individual behavior in order to achieve the group goal. Apparently, the most important and practically very useful result of this theory is the protocol of interaction of members of the team of agents, called the *joint intentions protocol* [22]. It is used by team agents to agree on their commitments and group conventions and to solve common issues of distributed coordination of their behavior in an autonomous mission without outside intervention.

Another theory, known as the *Shared Plans Theory*, is constructed a little differently [21]. Its basic concepts are the group plan and the individual mental concepts of autonomous agents. This group plan, in addition to the set of actions of individual agents of the group, which agreed with the set of conditions (time, place, resources, etc.), contains infrastructure components of the model, which convert the set of distributed agents into a single team. Both theories have a strict mathematical justification, but so far their software implementations are limited to software development at the level of simple prototypes.

In subsequent developments, different authors used different combinations of individual ideas of both theories for their models. The two most well-known are STEM [31] and RETSINA [29]. To support the software implementation of the STEM model, the Teamcore environment was developed [30, 31]. In it, the architecture of the software agent that manages the behavior of the agent is divided into two parts. One of them is subject domain-dependent, and the other is subject domain-independent. The domain-independent part of the agent is called a *teamcore agent*. It plays the role of “wrapper” for its domain-dependent part. The wrapper is responsible for the external behavior of the agent and provides it with the capability to work in teams. Note that this productive idea was later developed in different works.

The RETSINA model [29] has an architecture that exploits mainly the Shared Plan Theory. In its software architecture, an agent is allocated, which is called a *cooperative interface agent*. It actually implements the centralization of group control. In an autonomous mission, this solution can be critical, for example, when an object on which an agent with the interface agent role is installed fails. Both models and their supporting toolkits have long been regarded as world leaders in group control theory and models. But despite

their significant multiyear financial support from DARPA and unambiguous focus on military applications, both developments were closed by the early 2000s because the models proposed in them proved unsuitable for practice due to enormous computational complexity, since both were specified based on predicate calculus, extended by modal and temporal operators.

For the subsequent period until about 2010, there was a certain stagnation in the development of the theory and models of group control and its applications, which is clearly noted in [14]. During this period, researchers tried to adapt the theoretical models described above to practical needs. Among the fortunate models of this period is the BITE model proposed in [23]. In it, the individual behavior of team agents and group control are modeled by three structures. The first is the hierarchical structure of the tasks of the team of agents (a structured plan of their actions). The second describes the structure of the agents and their subgroups, which are assigned to the individual mission tasks described in the first structure. The third structure explicitly describes the communications and interactions of agents in the process of distributed coordination of agent behavior. This model significantly simplifies the classical models [9, 21], in which the scenario model is not specified explicitly before the start of the execution of the team’s mission, but must be output dynamically. But, nevertheless, the BITE model also has a big drawback. The authors note that the essence of the group control automation process is to provide automatic control of distributed *scenario execution* according to some standard protocol. They note that this goal in the BITE model has not been achieved, although even more is required to ensure self-organization of group control. This issue has been resolved in a later work [14].

The period from 2002 to 2020 is characterized by active research in the field of decentralized AI in cluster analysis tasks and in decentralized machine learning tasks, although in fairness it is worth noting that in the United States decentralized machine learning based on a set of distributed data was started in the early 1990s. For example, interesting theoretical results brought to practical use were obtained in the works [8, 28]. The authors of these works already then solved the problem of decentralized learning to detect false transactions in a group of US banks, which did not agree to provide their data to machine learning specialists, but agreed that training would be conducted by distributed team of agents using local data of each bank separately (in a decentralized version) with the subsequent use of a common set of rules for detecting false transactions based on local data and on metadata. The method that was developed by the authors of these works was successful and was used already then in practice.

In the period (2002–2006), an innovative project of the European FP6-IST program was carried out under the name “*KD-Ubiq: A Blueprint for Ubiquitous Knowledge Discovery Systems*.” This project explored the difficult problem of decentralized data mining and machine learning algorithms with an emphasis on preserving the confidentiality and privacy of data. The results of this project were published quite fully in the book [26]. Some results, which still retain their great relevance and practical significance, are published in [10, 32]. In particular, these works proposed protocols for decentralized calculation of mathematical expectation, as well as an approximate algorithm for decentralized clustering, built on the basis of the *K-means clustering* method. In both protocols, each agent receives meta-information only from its network neighbors.

It is also worth paying attention to the active development of decentralized machine learning methods and models at that time, which were focused on the use of agents and MASs technology of decentralized machine learning. This direction was called *Agent Mining* [2, 7]. In [6, 27], there is a review that describes the problems of this direction, as well as the most interesting decentralized machine learning algorithms developed by that time.

Work on creating a *p2p-platform* to maintain the interaction and communication of autonomous distributed entities (agents) without centralized yellow pages was initiated in 2004 by the FIPA working group² [13], and already in 2007, the first software implementation of such a platform was published in accordance with the reference model of FIPA [20]. On its basis, during this period, several fairly indicative fully decentralized AI applications were developed ([17, 19], etc.).

As for *self-organization*, the fourth important direction of research in the field of decentralized AI, the current state of research in this field is covered in detail in the works [17, 18, 33]. This direction has been studied quite deeply and there are many practical developments in this area in the period until about 2010. Currently, this topic is not so popular in theoretical studies. However, there is still a great field and a great potential for its development in this direction, and some aspects of this type are discussed in the next section.

Thus, it can be argued that to date, the results obtained in the field of decentralized AI relatively fully cover its main scientific areas and are quite mature. However, these results are mainly from the period up to 2010 and they meet the requirements of the applications of that time. By now, classes of IT applications in which AI technologies could lead to fundamentally

new properties of next generation applications have been significantly tightened, in particular, with regard to their scalability, fault tolerance, and computational complexity, as well as trust in the results obtained.

4. TOPICAL PROBLEMS OF DECENTRALIZED AI

It should be noted that the main results in the field of decentralized AI, despite the sufficient theoretical maturity, currently cannot be directly used in applications of the new generation, since the latter impose different requirements on the development than those that were characteristic of applications ten years ago. First of all, these requirements relate to the scale of applications—modern and promising applications are much larger in scale than those that were focused on the development period (1990–2020). This applies to all four issues discussed in the previous section. As a result of the large scale of modern applications, problems associated with ensuring application computational efficiency, robustness, security, and a number of others mentioned below are exacerbated.

A noticeable increase in the scale of applications leads, first of all, to a worsening the robustness of calculations performed in a decentralized architecture. This problem is not yet given its due, although it is a critical issue in conventional distributed computing, in the computing of statistics, and in decentralized machine learning algorithms. In a new way, the scale of systems affects the use of supercomputers. For example, the well-known parallel computing technologies implemented in the Hadoop ecosystem are not suitable for use in the supercomputers, which are able to realize their capabilities only if all the data is in RAM, when there is almost no need to exchange data with external memory. For these reasons, many decentralized computing algorithms in next-generation applications require either modification or complete renovation.

Decentralization also brings its own problems, among which the security of distributed components and data transmission channels is a priority. New technologies always bring new problems and the problem of security of decentralized systems is one of such new problems.

New challenges for IT applications include the development of new infrastructure and tools. For now, in the current implementation of IT applications, the share of decentralized AI is very small. And the reasons for this lie, first of all, in the fact that decentralized AI technologies are not yet ready for use at the industrial level. The list of priority developments in this area is as follows.

1. Development of new robust algorithms, mature technologies, and scalable software tools for the implementation of the concept of p2p-interactions of autonomous objects in networks with dynamic con-

² FIPA—Foundation for Intelligent Physical Agents, <http://www.fipa.org/>.

nectivity and intensive message passing with large volumes of transmitted information. These tools must provide reliable communications with the temporary loss of availability of individual addressees without loss of addressable information. The creation of new industrial-level platforms to support p2p-communications and dynamic routing in large-scale networks and variable topology is one of the important tasks of decentralized AI in the near future.

2. Development of known and creation of new robust algorithms, technologies, and software tools to support decentralized processes of data mining and machine learning operating, in particular, in real time. In mobile networks with limited computing and communication resources, this will significantly reduce the burden on communication channels and accelerate learning processes. To solve the described problems, there is already a necessary theoretical basis, experimental software developments have been created, but computationally efficient robust industrial-level tools for this purpose have yet to be created.

3. Creation of scalable algorithms and technologies to support applied p2p-services, as well as ecosystems of such services, including, for example, decentralized planning services, p2p-services of distributed coordination of group behavior of objects solving a common problem, and other services of an applied nature.

4. A complicated and important complex of problems of decentralized AI will have to be solved in relation to the tasks of group control. This is a new field of research and development, in which the theoretical basis is based on the protocols, i.e., decentralized (p2p-) algorithms of external behavior of groups of agents in various use cases. A decentralized algorithm is a protocol of interaction of distributed objects, which allows them to coordinate their individual behavior within the framework of the scenario of group behavior of agents intended to solve a common problem. Essentially, these are consensus protocols, leader selection protocols, contract network protocol, auction protocols, a protocol of joint intention, protocols of information exchange in order to maintain situational awareness of teammates in group behavior, and other protocols of decentralized computing, such as decentralized algorithms for data mining and machine learning—all of them can and should become components of libraries of standard algorithms for decentralized computing and group control of networks of autonomous objects represented in the software environment by their agents.

5. Algorithms and technologies of self-organization that work in a broader context, up to the global one. Decentralized AI algorithms form the basis of self-organization. However, the modern concept of self-organizing algorithms involves the use of only local information, i.e., information that an agent can obtain from its neighbors, and its use in local optimization processes. However, there are a number of the-

oretical proposals, as well as specific developments, that allow for the construction of self-organizing systems of decentralized architecture with the involvement of a wider and even global context. Examples are self-organization systems using digital fields [4], as well as self-organization models using the concept of amorphous computing [1].

These and other approaches for the formation of a global context based on local information are united by the concept of *active* data and knowledge bases. By definition, a data and knowledge base is called an active database if it can perform not only the actions that the user *explicitly* specifies. They can perform other actions of a proactive nature in accordance with the rules (knowledge) embedded in the data model. Typically, the activity of the data and knowledge model is used to control and maintain its consistency and integrity. In some cases, the active knowledge model includes the calculation of some attributes. If the values of these attributes fall within a predetermined range, this fact triggers proactive behavior of the system. A typical example of such proactivity in a distributed system is a timeout, which controls the correctness of the system processes in time and generates certain control actions in case of violation of thresholds specified by timeouts.

The concept of active knowledge is very useful in self-organizing systems. Expanding the locally available context about the states of network objects to make decision based on active knowledge is one of the options for increasing the autonomy and situational awareness of distributed decentralized systems. We demonstrate the ability of active knowledge to expand the local context of decision-making mechanisms of self-organization by example.

Example: Dynamic routing in a network with dynamic topology. We consider the concept of amorphous computing proposed at MIT in 2000 [1]. It is based on a model of self-organization borrowed from morphogenesis. This model considers a vector model of the morphogen, in which each coordinate can be used to control a particular process, property, etc.

The amorphous computing model considers a large number of equally programmed simple devices that are distributed on the surface or in some volume randomly. Each device can perceive the external environment and effect on it. It is believed that the devices have very limited resources and perceive a very limited amount of local information, and they can also fail. It is also assumed that each device has its own execution thread and is capable of generating random numbers. Each device has an internal state depending on its previous actions. Devices can exchange messages over a communication channel with a short range of reach. Network devices initially know nothing about the topology of the communication network; the network does not have a centralized source of information,

global time, and beacons for binding to coordinates. The communication environment supports the propagation of the digital field of each device, which is defined by a spatially dependent data structure called a digital field [4].

It turns out that such a fairly simple computing model can be used to build very effective mechanisms of self-organization by spreading part of the global context across the network. For example, let some source device can send its name and “morphogen” to neighbors—a number equal to zero. After receiving such a message, each neighbor sends it to its neighbors, adding 1 to the value of the morphogen with the name of its source. This process continues until all nodes in the communication network are reached.

Each device can store the minimum value of this morphogen. If necessary, the device can use it as the value of the shortest path to the source node, if the path length is measured in terms of hops on it. Further, each node can determine the “local orientation” of the node, i.e., the direction to or from the source, requesting the value of the morphogen of this node from neighbors. The direction to the source is determined by the node (or nodes) in which the target node morphogen value is minimal. Similarly, the direction from the source corresponds to the direction of nodes with a value of the morphogen one more than its own value.

If network agents are used as digital field sources, the agent of any network node can obtain information about the “local orientation” of the network node to the source by polling neighboring nodes. If the source is the addressee of the message that a node must send, then as a result of this survey it can dynamically determine the first hop of the route of transmission of the message to its addressee. Next, subsequent nodes on the path from source to destination can do this. As a result, the desired route of transmission of the message will be formed automatically in a decentralized style using an amorphous computing model.

This example shows the important role of active knowledge in solving the problem of addressing messages in systems with mobile objects and limited communication range. Note that, in this case, the local context of the network node is expanded owing to a digital field that delivers distributed information about the global connectivity of the communication network to each node.

Thus, the use of active knowledge in distributed systems with self-organization allows one to get more effective self-organizing systems by “delivering” information about the global context to local decision-making nodes. Therefore, the use of active knowledge should be considered as one of the important future principles of building decentralized AI systems.

New challenges for IT applications, as well as new requirements and new opportunities for decentralized AI, are a natural development process. At this stage, the issues and tasks summarized in this section are considered as the key ones.

CONCLUSIONS

The large scale, high level of complexity, and distributed nature of the new generation of IT applications and the tasks that they must solve, as a rule, in real time and in uncertainty conditions is a modern reality. Complex next-generation IT applications are networks of mobile objects operating in a wireless communications environment. These objects, owing to the combination of their resources and intensive interaction, are able to solve the most complex problems even with limited computing resources. For modern AI, this situation is new, and it presents new requirements that cannot be addressed without revising previously accepted control paradigms and approaches and without tightening various indicators of the quality of the system.

Analysis of modern trends in the field of advanced concepts and projects under development shows the growing role of AI, primarily decentralized AI.

The paper analyzes the advanced concepts of building new generation IT applications and specific developments and provides a brief analysis of modern achievements in the field of decentralized AI and self-organization, which are theoretically able to support the practical implementation of such applications. However, these developments are not yet ready for use at the industrial level, taking into account new realities and requirements for them from modern applications. The basic guidelines for the development of AI in this context can be characterized by such key areas of development as the following:

- decentralized computing in dynamic networks based on algorithms of p2p-interactions;
- dynamic p2p-communication networks and environments;
- group behavior modeling and protocol-based group control;
- libraries of application protocols for interaction of agents in different tasks and in cases of use and ecosystems that support the operation of objects in accordance with these protocols;
- large-scale robust computationally efficient algorithms for decentralized computing in p2p-networks;
- new decentralized algorithms, technologies, and software tools to support data mining and machine learning processes;

- scalable algorithms and technologies to support applied p2p-services, as well as ecosystems of such services;
- self-organization in a global context.

CONFLICT OF INTEREST

The author declares that he has no conflicts of interest.

REFERENCES

1. H. Abelson, D. Allen, D. Coore, C. Hanson, G. Homsy, T. Knight, R. Nagpal, E. Rauch, G. Sussman, and R. Weiss, “Amorphous computing,” *Commun. ACM* **43** (5), 74–82 (2000). <https://doi.org/10.1145/332833.332842>
2. Agent Mining Special Interest Group. <http://www.agentmining.org>.
3. M. N. M. Bhutta, A. A. Khwaja, A. Nadeem, H. F. Ahmad, M. Kh. Khan, M. A. Hanif, H. Song, M. Alshamari, and Yu. Cao, “Survey on blockchain technology: Evolution, architecture and security,” *IEEE Access* **10**, 61048–61073 (2021). <https://doi.org/10.1109/ACCESS.2021.3072849>
4. M. Camurri, M. Mamei, and F. Zambonelli, “Urban traffic control with co-fields,” in *Environments for Multi-agent Systems III*, Ed. by D. Weyns, H. V. D. Parunak, and F. Michel, Lecture Notes in Artificial Intelligence, Vol. 4389 (Springer, Berlin, 2007), pp. 239–253. https://doi.org/10.1007/978-3-540-71103-2_14
5. L. Cao, “Decentralized AI: Edge intelligence and smart blockchain, metaverse, Web3, and DeSci,” *IEEE Intell. Syst.* **37** (3), 6–19 (2022). <https://doi.org/10.1109/mis.2022.3181504>
6. L. Cao, V. Gorodetsky, and P. A. Mitkas, “Agent mining: The synergy of agents and data mining,” *IEEE Intell. Syst.* **24** (3) (2009). <https://doi.org/10.1109/MIS.2009.45>
7. L. Cao, G. Weiss, and P. S. Yu, “A brief introduction to agent mining,” *Auton. Agents Multi-Agent Syst.* **25**, 419–424 (2012). <https://doi.org/10.1007/s10458-011-9191-4>
8. Ph. K. Chan, W. Fan, and S. J. Stolfo, “Distributed data mining in credit card fraud detection,” *IEEE Intell. Syst.* **14** (6), 67–74 (1999). <https://doi.org/10.1109/5254.809570>
9. P. Cohen and H. J. Levesque, “Teamwork,” *Noûs*, No. 25, **487** (1991). <https://doi.org/10.2307/2216075>
10. S. Datta, K. Bhaduri, C. Giannella, R. Wolff, and H. Kargupta, “Distributed data mining in peer-to-peer networks,” *IEEE Internet Comput.* **10** (4), 18–26 (2006). <https://doi.org/10.1109/mic.2006.74>
11. G. Di Marzo Serugendo, M. P. Gleizes, and A. Karageorgos, “Self-organisation in multi-agent systems,” in *Rapport de recherche IRIT/2005_18_R* (Univ. Paul Sabatier, Toulouse, 2005).
12. J. Ferber, “Foreword,” in *Handbook on Agent-Oriented Design Processes*, Ed. by M. Cossentino, V. Hilaire, A. Molesini, and V. Seidita (Springer, Berlin, 2014). <https://doi.org/10.1007/978-3-642-39975-6>
13. FIPA P2P NA WG6. Functional Architecture Specification Draft 0.12. <http://www.fipa.org/subgroups/P2PNA-WG-docs/P2PNA-Spec-Draft0.12.doc>.
14. K. Geihs, “Engineering challenges ahead for robot teamwork in dynamic environments,” *Appl. Sci.* **10**, 1368 (2020). <https://doi.org/10.3390/app10041368>
15. I. Gidasov and A. Akopyan, “What is DeFi: Everything about decentralized finance.” <https://currency.com/ru/chto-takoe-defi>
16. V. Gorodetsky, “Multi-agent autonomous group control in collective robotics-based assembly,” *CEUR Workshop Proc.* **2648**, 1 (2020).
17. V. I. Gorodetskii, “Self-organization and multiagent systems: I. Models of multiagent self-organization,” *J. Comput. Syst. Sci. Int.* **51**, 256–281 (2012). <https://doi.org/10.1134/s106423071201008x>
18. V. I. Gorodetskii, “Self-organization and multiagent systems: II. Applications and the development technology,” *J. Comput. Syst. Sci. Int.* **51**, 391–409 (2012). <https://doi.org/10.1134/s1064230712020062>
19. V. I. Gorodetskii, O. V. Karsaev, V. V. Samoilov, and D. Trotskii, “Development tools for open agent networks,” *J. Comput. Syst. Sci. Int.* **47**, 429–446 (2008). <https://doi.org/10.1134/S1064230708030131>
20. V. Gorodetsky, O. Karsaev, V. Samoylov, and S. Serebryakov, “P2P agent platform: Implementation and testing,” in *Agents and Peer-to-Peer Computing*, Ed. by S. R. H. Joseph, Z. Despotovic, G. Moro, and S. Bergamaschi, Lecture Notes in Computer Science, Vol. 5319 (Springer, Berlin, 2010), pp. 41–54. https://doi.org/10.1007/978-3-642-11368-0_4
21. B. J. Grosz and S. Kraus, “Collaborative plans for complex group action,” *Artif. Intell.* **86**, 269–357 (1996). [https://doi.org/10.1016/0004-3702\(95\)00103-4](https://doi.org/10.1016/0004-3702(95)00103-4)
22. N. R. Jennings, “Controlling cooperative problem solving in industrial multi-agent systems using joint intentions,” *Artif. Intell.* **75**, 195–240 (1995). [https://doi.org/10.1016/0004-3702\(94\)00020-2](https://doi.org/10.1016/0004-3702(94)00020-2)
23. G. A. Kaminka and I. Frenkel, “Towards flexible teamwork in behavior-based robots,” in *Proc. 4th Int. Joint Conf. on Autonomous Agents and Multiagent Systems* (Association for Computing Machinery, New York, 2005), pp. 1355–1356. <https://doi.org/10.1145/1082473.1082770>
24. A. Karelina, “What is Web3: Explanation in simple words.” <https://secretmag.ru/enciklopediya/chto-takoe-web3-obyasnyam-prostyimi-slovami.htm>

25. L. Lee, T. Braud, P. Zhou, L. Wang, D. Xu, Z. Lin, A. Kumar, C. Bermejo, and P. Hui arXiv:2110.05352 (2021).
26. M. May and L. Saitta, *Ubiquitous Knowledge Discovery: Challenges, Techniques, Applications*, Lecture Notes in Computer Science, Vol. 6202 (Springer, Berlin, 2010). <https://doi.org/10.1007/978-3-642-16392-0>
27. Ch. Moemeng, V. Gorodetsky, Z. Zuo, Yo. Yang, and Ch. Zhang, "Agent-based distributed data mining: A survey," in *Data Mining and Multi-agent Integration*, Ed. by L. Cao (Springer, Boston, 2009), pp. 47–58. https://doi.org/10.1007/978-1-4419-0522-2_3
28. A. L. Prodromidis, Ph. K. Chan, and S. J. Stolfo, "Meta-learning in distributed data mining systems: Issues and approaches," in *Advances of Distributed Data Mining* (AAAI Press, 2000).
29. K. Sycara, M. Paolucci, M. Van Velsen, and J. Giam-papa, "The RETSINA multi-agent infrastructure," *Auton. Agents Multi-Agent Syst.* **7**, 29–48 (2003). <https://doi.org/10.1023/a:1024172719965>
30. M. Tambe, W.-M. Shen, M. Mataric, D. Pynadath, D. Goldberg, P. Modi, Z. Qiu, and B. Salemi, "Teamwork in cyberspace: Using TEAMCORE to make agents team-ready," in *AAAI Spring Symp. on Agents in Cyberspace* (1999), pp. 136–141.
31. M. Tambe, "Towards flexible teamwork," *J. Artif. Intell. Res.* **7**, 83–124 (1997). <https://doi.org/10.1613/jair.433>
32. R. Wolff and A. Schuster, "Association rule mining in peer-to-peer systems," *IEEE Trans. Syst., Man Cybern., Part B (Cybern.)* **34**, 2426–2438 (2004). <https://doi.org/10.1109/tsmcb.2004.836888>
33. D. Ye, M. Zhang, and A. V. Vasilakos, "A survey of self-organization mechanisms in multiagent systems," *IEEE Trans. Syst., Man, Cybern.: Syst.* **47**, 441–461 (2017). <https://doi.org/10.1109/tsmc.2015.2504350>
34. Ya. Zhang, "Mobile edge computing," in *Mobile Edge Computing*, Simula SpringerBriefs on Computing, Vol. 9 (Springer, Cham, 2021), pp. 9–21. https://doi.org/10.1007/978-3-030-83944-4_2



Vladimir Ivanovich Gorodetsky.

Doctor of Technical Sciences (1973), Professor (1990), Honored Scientist of the Russian Federation. He graduated from the Leningrad Air Force Engineering Academy with the master degree in mechanics (1960) and the Faculty of Mathematics and Mechanics of the Leningrad State University with the master degree in mathematics (1970). He defended his candidate (1967) and doctoral (1973) dissertations on

the problems of optimal control. From 1967 to 1988 he worked at the Mozhaisky Military Space Engineering Academy in research and teaching positions. From 1988 to August 10, 2018, he worked at the St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences as head of the laboratory and chief researcher of the laboratory of intelligent systems. He published more than 250 works (over 100 in foreign cited publications) and 9 monographs and manuals; he was coeditor of 12 issues of the "Lecture Notes in Artificial Intelligence" and "Lecture Notes in Computer Science" series (2001–2015), guest editor of a special issue of the *IEEE Intelligent Systems* journal (Q1) on "Agent and Data Mining Interaction" (2009), etc. Member of the Russian and European Association of Artificial Intelligence, IEEE, IEEE Computer Science, International Foundation for Autonomous Agents and Multi-agent Systems (IFAAMAS), International Society of Information Fusion (ISIF). Associate Editor of the *Data Science and Analytics* international journal (Springer) and a member of the editorial board of the *Design Ontology* Russian journal. Winner of the D.A. Pospelov Award of the Russian Association of Artificial Intelligence (2022).

Science metric data (as of January 1, 2023): RSCI citation 3166, h-index = 24, WoS citation 219, h-index = 7, Scopus-citation 755, h-index = 12.

Areas of scientific interest: theory of optimal control, celestial mechanics, applied statistics, planning and scheduling, artificial intelligence and decision-making, multi-agent systems, group management, self-organizing agent networks, software tools, data mining and machine learning, efficient and robust big data processing algorithms, semantic computing, distributed and p2p-learning, computer security, dealing with uncertainty, recommendation systems, transport and production logistics, steganography, etc.

Web page: https://en.wikipedia.org/wiki/Vladimir_Gorodetski.