Human-Machine Collective Intelligence Environment for Adaptive Decision Support

Alexander Smirnov, Andrew Ponomarev
St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences
St. Petersburg, Russian Federation
{smir, ponomarev}@iias.spb.su

***Abstract*—Crowd computing has become an important and widely used way to solve various problems by joint effort of humans and machines. In most of the systems, leveraging elements of crowd computing, workflow or algorithm (how to split the original problem into parts, how to distribute them between human participants and how to merge the results received from different human participants) is defined as a part of system design. While it pays back in a wide range of tasks (usually simple ones), it is widely recognized that rigid workflows turn out to be very limiting when applied to complex problems (e.g., decision support). The paper proposes an approach to create flexible and adaptive decision support systems on the basis of an environment, supporting human-machine collective intelligence. The distinctive features of the proposed environment are: a) fusion of elements of collective intelligence and artificial intelligence, b) support for natural self-organization processes in the community of participants (supported by self-organization protocols, convenient for different kinds of participants of a heterogeneous human-machine system), c) interoperability of participants (both human and machine), achieved via multi-aspect ontologies, d) soft guidance in the process of self-organization.**

I. Introduction

Crowdsourcing (also often referred to as "crowd computing") has become an important and widely used way to organize solving problems by joint effort of humans and machines. The role of humans in these systems is typically to execute some tasks that are still hard for machines (semantic interpretation of images and audio, dealing with incomplete information, common sense etc.), while machines do data preprocessing and coordination (as well as quality assurance [1]). The spectrum of problems, where crowd computing has been successfully applied is rather wide, ranging from relatively simple content processing applications on Amazon Mechanical Turk [2], to citizen science projects on Zooniverse [3]. In most of the systems, based on crowd computing, run-time workflow or algorithm (how to split the original problem into parts, how to distribute them between human participants and how to merge the results received from different human participants) is fixed (defined as a part of system design). However, for many complex problems especially in highly dynamic domains it is not viable to arrange a well-defined workflow in advance, because the situation changes very fast [4]. A natural technological answer to that is the creation of a new generation of human-machine systems, characterized by adaptiveness.

It is widely recognized, that organizational decision making is a dynamic and complex process, because decision making requirements are constantly changing over time and vary from person to person [5]. Decision support therefore is one of those kinds of activities that require adaptiveness of the system and flexible workflow, because decision-making very often is based on interactive and iterative exploration of the problem. Besides, modern decision-making leverages a wide spectrum of data processing and reasoning tools (often, with elements of AI). Therefore, it is important to create a palette of methods and technologies that would allow a collective of people and software services with elements of AI to provide decision support defining the required activities in a flexible way.

This paper makes a step in this direction and proposes an environment allowing to utilize human-machine collective intelligence for decision support.

First steps in the direction of creating computational systems including humans and giving them more freedom to define possible actions and a course of solving a problem have already been done ("flash organizations" ([4] and [6]), collective adaptive systems (CAS) [7], hybrid CAS [8] etc.). However, in these concepts, the adaptation processes can be driven only by humans. In the proposed concept, machines (software services) can also play a role in the adaptation processes following various self-organization protocols.

The rest of the paper is structured as follows. Section II describes several relevant areas of research and enumerate most promising results that can be used in environments that support human-machine collective intelligence. Section III discusses the context and positioning of the proposed human-machine collective intelligence environment for decision support. Finally, Section IV enumerates main features of the proposed environment and outlines the proposed ways to implement them.

II. Related Work

In the core of the research is the convergence of three areas: 1) methods to program (and, in general, organize) human-computer effort, especially in the context of complex problems;

2) adaptiveness in decision support systems; 3) self-organization research. In this section, we briefly describe relevant results in each of the areas and underline the differences of the proposed approach.

A. Programming of Human-Machine Effort

In most publications dedicated to programming of human-machine systems (i.e., crowdsourcing, crowd-computing) human participants are understood as a kind of "computing device" or a service, that can process requests of a certain type (for example, process images by identifying and labeling objects on them). Service-based description of a human participant of a human machine system is so nicely aligned with foundations of a service science and service-oriented architecture (SOA), that it led to the creation of several SOA adaptations for representing services implemented by people (for example, WS-HumanTask, the works of D. Schall [9], [10], or human-computer cloud [11], [12]). With a help of this adapted languages and frameworks, one can develop a service-based system (defining workflow, service compositions and orchestration) abstracting from the nature of an entity executing service call (whether it is human participant, or a computer). In other systems such unification is not done, and the system designer just describes the necessary sequence of steps required for information processing, identifies steps that cannot be effectively performed using software and hardware only, and forms a set of solutions for performing these steps with a help of humans (paying attention to motivation, quality assurance and other issues, caused by the specifics of the actual inclusion of human into a computing system). Anyway, the function of a human participant is reduced to performing a specific task, proposed by the system designer, interacting with the system in a strictly limited manner (usually just reading the details of the task and entering the result into the specific form).

Based on the way the workflow is designed and encoded the existing approaches can be categorized [13] into three groups: a) programming-level approaches (e.g., TurKit [14], CrowdDB [15] and AutoMan [16]); b) parallel-computing approaches (e.g., Turkomatic [17], Jabberwocky [18]); and c) process modeling approaches, deriving from various workflow languages (e.g., CrowdLang [19], CrowdSearcher [20] and CrowdComputer [13]).

Although such rigid division of roles (designer vs. participant of the system) and strict limitation of the participant's capabilities pays back in a wide range of tasks (usually simple ones like annotation, markup, etc.), the creative and organizational capabilities of a human in such systems are discarded. Besides, it is known from the organizational science and specialized research on crowd systems, that rigid workflows turn out to be very limiting when applied to complex problems (see, e.g., [4]). The first experimental crowdsourcing systems where human participants were able to refine the proposed workflow appeared in 2012 [21], but the problem is getting the closest attention of the research community only recently. In particular, in the works of M. Bernstein, who studies the limitations of solutions based on the fixed flow of work for dealing with complex problems and the ways to overcome these limitations with a help of dynamic organizations from members of the crowd (the so-called "flash organizations" [6]). While the concept of "flash organization" represents an important step in understanding how crowd computing can be applied to complex problems, it deals only with human participants. In this research, however, we are building an environment where heterogeneous agents (human and software) would be able to collectively decide on the details of the workflow.

B. Adaptiveness in Decision Support Systems

Decision support systems (DSS) have been formally defined as interactive computer based systems that support decision making processes for decision makers to solve semi-structured and/or unstructured problems [22]. In the current organizational context, characterized with the rapid changes, such systems need to be adaptive in three ways [5]: 1) to obtain various resources needed in the decision making process, 2) to cope with changing (in time and for different people) decision making requirements.

A recent approach to enable adaptiveness in DSSs is connected with service-oriented architecture and SOA-based DSSs correspondingly. The core of the approach is that small functional components of a DSS are represented as services, allowing to easily assemble specialized DSSs (see, e.g., [23], [24]). An elaboration of this approach is the concept of agent-enabled service-oriented DSS [5], where the process of building compositions of services is mitigated with a help of intelligent agents.

While our research is closely related with agent-enabled service-oriented DSS, the authors of [5] don't consider the situation when some of the software services may be represented by humans (with a free will), the role of human participants in the process of adaptation, as well as protocols of adaptation that can be convenient for both human and software participants.

C. Self-Organization

For the purpose of a brief overview, we can divide self-organization research into three streams: descriptive studies, formal studies and constructive studies.

Descriptive studies are aimed mostly on the analysis of how self-organization occurs in natural systems, what mechanisms are used. Due to the inter-disciplinary character of self-organization phenomenon descriptive studies may come from various scientific domains. The most important here is the research on human systems dynamics and self-organization in human collectives (e.g. [25]). And of that, the most relevant is research on how self-organization occurs by means of modern social media technologies, as it directly shows us how human collectives self-organize with a help of information and communication technologies. A prominent impulse for this kind of self-organization is an emergency situation. E.g., [26] discusses the phenomenon of creation of new social ties in the process of self-organization and problem solving by people affected by natural disasters. In particular, she focuses on methods that require shared information space (shared site of work and visible record of activity).

Formal studies are aimed mostly on the analysis of self-organization from the formal (mostly mathematical) point of view, developing new formal models of it: usually, based on differential equations, automata, or multi-agent paradigm. A prominent approach to formalize self-organization in human system has roots in game theory and describes humans as rational agents, which leads to self-organizing market models.

Constructive studies are aimed mostly on transferring principles of self-organization found in nature into artificial systems. This kind of research is mostly relevant in the paradigm of multi-agent systems, where various approaches to self-organization become a foundation of agent communication protocols. Self-organization mechanisms used in construction of computational (multi-agent) systems [27]: a) mechanisms based on reinforcement learning; b) cooperation-based mechanisms; c) mechanisms based on the use of gradient fields; d) market self-organization mechanisms; e) mechanisms using the holonic system model.

III. Positioning of the Decision Support Environment

The aim of the proposed environment is to support the process of making complex decisions and/or making decisions in complex problem domains. The complexity of making such decisions is usually caused by the number of factors influencing the decision, uncertainty, associated with various parts of the situation description, and/or the incompleteness of the data about the problem situation. All these “complications” have to be addressed in some way, which is determined by the particular situation. Hence, while the methodology of decision-making is quite definite in the upper level (identification of the alternatives, identification of the criteria, evaluation of the alternatives etc.), the exact steps required to collect all the needed data, analyze it and present to the decision maker may be unclear. That is why decision support requires operative (on-the-fly) planning of the low-level activities and may benefit from leveraging self-organizing capabilities of the participants of the decision support process. Besides, currently many decisions are based not only on expert opinions or on intuition, but also rely heavily on problem-relevant private or public data/information sources (big data, linked data and so on). In other words, decision support is in fact human-machine activity, and the environment has to offer a set mechanisms and tools to mitigate this activity.

The environment interacts with three main types of actors (Fig. 1): decision-makers, experts, and data/service providers. Decision-makers are responsible for the analysis of a situation and making a decision. In certain cases, when the decision problem is too complex, the decision-maker may require some additional expertise that can be provided by participants of a human-machine collective intelligence environment. Most likely, decision-maker is a middle-to-top level manager in terms of a typical business hierarchy (because using collective intelligence is usually rather expensive and may be justified mostly for important decisions). However, the environment may be used in different problem domains and with different incentive schemes, so in some applications decision-maker may be just a smart city citizen. After the decision-maker posts the problem to the collective intelligence, he/she may oversee the process of solution and guide it in some way.

Experts have problem-specific knowledge and may contribute into decision support process in two major ways. First, they can come up with procedures of obtaining relevant judgements, participating in the *ad hoc* construction of the decision support workflow, both directly (proposing a sketch of a whole workflow or a part of it), or indirectly (by setting various incentives for other participants). Second, they can use their expertise by providing data as well as processing it to come to some problem-related judgements. In general, an expert can be anyone – within or without the organization boundary, the difference is mostly in the incentives important for the particular expert.

Service providers design and maintain various software tools, services and datasets that can be used for decision support. Their motivation is to receive rewards for providing these tools and data to the other participants of the environment. This is a direct evolution of the on demand service provisioning.

The environment provides a set of protocols, methods and tools, allowing participants of different nature (human and machine) to be able to communicate and decide on the particular steps of decision support process, perform these steps and exchange results, motivated by some external or internal mechanisms, making the whole environment profitable for all parties.

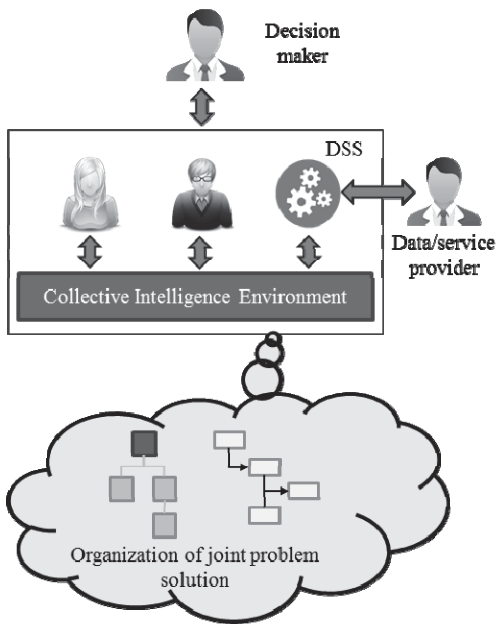

Fig. 1. Main roles of the DSS, based on the environment

IV. Distinguishing Features of the Environment

This section describes the features of the environment, that distinguish it from the related work (Section II), and discusses how these features and capabilities can be achieved.

A. Amalgamation of Collective Intelligence and Artificial Intelligence

Collective intelligence (construed as methods for making people to work together to solve problems) and artificial intelligence are usually considered as alternative methods of decision support (in some tasks former is more convenient, in others – the latter). However, there are a number of possibilities for their joint usage, paving the way for efficient human-machine technologies (see [28], [29], and [30]). A detailed analysis the possibilities to converge the collective intelligence-based and artificial intelligence-based approaches to problem-solving has revealed the following options:

1) The use of artificial intelligence in collective intelligence systems:

a. The use of artificial intelligence methods for efficient and rational organization of people groups for collective problem solving. Existing solutions in this area include, for example, measuring the performance of participants on crowdsourcing sites on different types of problems for efficient task assignment and recommendation [31], [32], and [33]. In a certain sense, this way of convergence can be understood as an application of artificial intelligence techniques at a meta level to the problem of planning and organizing the process of problem solving.

b. Application of problem-oriented methods of artificial intelligence to complement actions of information processing performed by humans. This category, for example, includes optimization of human efforts in systems primarily focused on operations performed by human participants — e.g., bots on Wikipedia, implementing some routine operations on article editing. This also includes the use by a human (who is a part of a collective intelligence system) of some kind of artificial intelligence (at his/her discretion) and the interpretation of the result.

c. The application of artificial intelligence to the processing of specific human characteristics (e.g., recognition of the emotional state), in order to take them into account during the processing and integration of the results obtained by the community.

2) The use of collective intelligence in artificial intelligence systems

a. Ensuring interaction between a person (end user) and an artificial intelligence system (for example, to interpret user requests and translate them into a form "understandable" by a system of formal reasoning), the use of common sense, intuition, etc. for a more accurate specification of a task to be solved by an artificial intelligence system.

b. Learning artificial intelligence models while monitoring human activities [34], addressing a person through active learning protocols [35]. Learning the experience of a group of people, but not only problem domain experience (solving problems that arise in a particular subject area), but also meta-level experience – general social techniques (http://moralmachine.mit.edu).

c. Verification of the results of the work of artificial intelligence models for compliance with ethical principles and social norms, which may be very difficult to formalize.

The identified options (possibilities of collective intelligence and artificial intelligence convergence) can be divided into two levels: foundational and problem-oriented. The foundational level is associated with those possibilities (1 (a), 2 (a)), that can be implemented regardless of the application area. The possibilities of the problem-oriented level (remaining options), on the contrary, require application domain analysis and the development of specific methods for different application domains.

The proposed environment provides the infrastructure for the communication of four types of intelligent software services (Fig. 2):

- Solver. Software that can transform a task description in some way, enriching it with some derived knowledge.
- Data/Knowledge provider. Interface-wise almost similar to the previous type, however, only provides some problem-specific information.
- Tool handler. A utility agent that manages human expert access to some software tool (with GUI). In many cases, certain data processing routines required

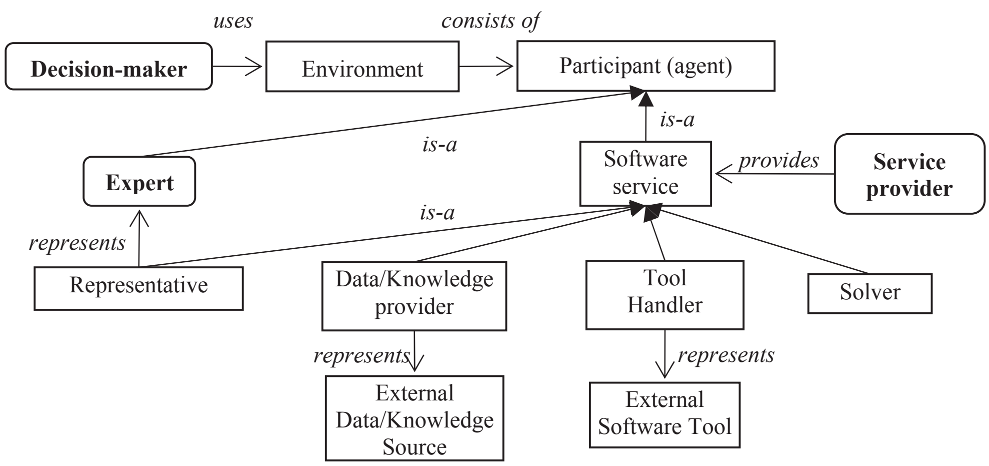

Fig 2. Core entities of the environment

for decision-making can be implemented with some software (or, SaaS). It is not practical to re-implement it in a new way, however, granting an access to such tools might be useful for all the involved parties.
- Representative. Allowing expert to communicate with other services.

General structure of all types of software services contains several elements: communicative structure of the service, responsible for making agreements with other participants of the environment based on goals (given by the provider of the service) and competencies.

B. Protocols for Self-Organization

An important distinguishing feature of the proposed approach is to allow the participants (both human and software services) to dynamically decide on the details of the workflow. Therefore, agents should be able to coordinate and decide on roles, task distribution etc., in other words a group of agents should be able to self-organize. Any self organization in the collective is based on the communication of the parties. The rules of such communication as well as semantic entities required to describe possible implementations of the decision support process and responsibilities of participants in the scope of a possible implementation make up a protocol for self-organization.

The protocols of self-organization in such environment have to respect both machine and human requirements. The latter means that widely used models of bio-inspired self-organization turn out to have less potential to be applied, as they mostly are taken from the analysis of primitive behaviors (e.g., of insects). On the other hand, market (or, economics) based models have much more potential to be applied in this environment, because they account for the economic behavior of humans and match the incentive structure of on-demand service provisioning. Another possible source are socio-inspired mechanisms and protocols. They are by design natural for people, and there are already some attempts to adapt them for artificial systems (e.g., see [39]).

Particularly, the necessary activities are: a) research on and formalization of the mechanisms of self-organization of human groups, b) development of technological solutions based on the information technology to support the self-organization mechanisms, c) transformation of the self-organization mechanisms into the form of protocols that can be used by software services (components) (socio-inspired self-organization)

The research efforts will further develop the models of the socio-inspired self-organization [39] earlier proposed by one of the authors and consistently apply these models for the organization of different kinds of interactions in the course of decision support in the human-machine collective intelligence environment.

C. Interoperability of Agents

To sustain various coordination processes, as well as information flow during decision-making there multilevel interoperability has to be provided inside the collaborative environment. This is especially acute in the case of mixed collectives, consisting of human and machine agents.

To implement any self-organization protocols, the participants of the system have to exchange several types of knowledge:

- Domain knowledge. What object and what relationships between objects are in the problem area.
- Task knowledge. Both goal description, and possible conceptualization of the active decision support task, e.g., mapping some concepts to alternatives, functions to criteria.
- Protocol knowledge. Terms of interaction, incentives, roles etc.

It is proposed to use ontologies as the main means ensuring the interoperability. The key role of the ontology model is in its ability to support semantic interoperability as the information represented by ontology can be interpreted both by humans and machines, therefore, ontology-based information representation can provide the interoperability for all kinds of possible interactions (human-human, human-machine, machine-machine). Ontologies have proven themselves as a means resolving the problem of semantic interoperability, but applying ontologies can still be a problem due to different terminologies and formalisms that the members of the system use. Therefore, to realize the potential of ontologies to serve as a *lingua franca* in the human-computer environment for collective intelligence a number of fundamental tasks has to be solved. First, it is needed to develop an ontology model for representation and processing of data produced by the decision support processes. Second, it is needed to develop methods to support conciliated ontologies that capture different views on the same problem.

Three main groups of approaches to solving the problem of conciliated ontologies support can be distinguished: development of a universal common ontology, development of an ontology ecosystem, and development of a multi-aspect ontology.

The development of a universal common ontology is complicated by the amount of information it is supposed to represent. Besides, the diversity of service providers that make up the ecosystem, as well as its dynamic development, also complicates the problem of defining some common terminology and formalism (it is often reasonable to use different formalisms for solving different problems).

The development of an ontology ecosystem assumes existence of correspondences between the ontologies that make up the ontology ecosystem. With this approach, there are also significant difficulties associated with the ecosystem dynamics (the correspondences between ontologies must be constantly updated). One of the techniques to solving this problem may be ontology matching. However, at present, the methods supporting automatic ontology matching are relatively reliable only for specific narrow domains, and manual ontology matching requires considerable time and efforts. There are studies on enrichment of ontology facilities (e.g., extensions to ontologies in DAML+OIL for representation of the configuration problem [36] are developed; semantic annotations are introduced [37], etc.), but these studies still cannot solve the problem of integrating heterogeneous information and knowledge with different terminologies.

Taking into account the heterogeneity of the participants of the human-machine collective intelligence systems and the multidimensionality of the decision support activities, it is proposed to use multi-aspect ontologies. Such approach will enable to develop a model that can be applied to a broad spectrum of activities arising from the description and interaction of participants of the considered type of systems. Besides, multi-aspect ontologies will avoid the need for standardization of all services of a digital ecosystem through providing one aspect (some viewpoint on the domain) to services of one ecosystem community (services of one producer, services that jointly solve a certain task, etc.) for the service collaboration.

D. Soft Guidance

Though the execution process in the proposed environment is self-orchestrated and driven by negotiation protocols, human participants, however, will need intelligent assistance when communicating with other agents in the environment. The role of this assistance is to offer viable organization structures and incentive mechanisms based on current goals. An important aspect during the soft guidance is mapping actions defined by decision-making methodologies to human-computer collaboration scenarios. It means that the environment (or representative service) uses the existing knowledge on decision making to offer agents viable collaboration structures. In the context of classic prescriptive (recommended) decision making models (for instance, Simon's model), the activities delegated to the human-machine environment could be identification of criteria for decision support in the current situation, ranking and determining the criteria importance, identification and comparison of alternatives. All these activities often require a comprehensive analysis of different dimensions of the problem situation, taking into account the experience and, sometimes, the intuition of experts, which makes it advisable to use human-machine environments to carry out the mentioned activities. In the decision support theory, there have been proposed a large number of approaches (e.g., [38]) to solve the decision-making problems. Such approaches can constitute initial patterns for organization of a decision support process and can either be reproduced exactly or be refined and modified w.r.t. the problem situation and the decision maker's preferences.

V. CONCLUSION

Motivated by recognized in the scientific community limitations of modern crowd-based systems in dealing with complex problems (and in complex domains), the paper discusses a novel class of decision support systems, based on an environment, leveraging human-machine collective intelligence.

The distinctive features of the proposed environment are: a) fusion of elements of collective intelligence and artificial intelligence, b) support for natural self-organization processes in the community of participants (supported by self-organization protocols, convenient for different kinds of participants of a heterogeneous human-machine system), c) interoperability of participants (both human and machine), achieved via multi-aspect ontologies, d) soft guidance in the process of self-organization.

ACKNOWLEDGMENT

The research is funded by the Russian Science Foundation (project # 19-11-00126).

REFERENCES

[1] A. Ponomarev, "Quality control methods in crowd computing: literature review", *SPIIRAS Proceedings*, 54, 2017, pp. 152–184.
[2] Amazon Mechanical Turk, Web: https://mturk.com
[3] Zooniverse, Web: https://www.zooniverse.org
[4] D. Retelny, M. Bernstein, M. Valentine, "No Workflow Can Ever Be Enough: How Crowdsourcing Workflows Constrain Complex Work," *in Proc. ACM Human-Computer Interact,* vol. 1, no. 2, 2017, article 89.
[5] C.S.J. Dong and A. Srinivasan, "Agent-enabled service-oriented decision support systems," *Decis. Support Syst.*, vol. 55, no. 1, 2013, pp. 364–373.
[6] M. Valentine, "Flash Organizations," *in 2017 CHI Conference on Human Factors in Computing Systems – CHI '17*, pp. 3523–3537.
[7] M. Viroli, G. Audrito, J. Beal, F. Damiani, D. Pianini, "Engineering Resilient Collective Adaptive Systems by Self-Stabilisation", *Journal ACM Transactions on Modeling and Computer Simulation*, vol. 28, issue 2, 2018, article 16.
[8] S. Dustdar, S. Nastic, O. Scekic, *Smart Cities: The Internet of Things, People and Systems*. Springer, 2017.
[9] D. Schall, *Service-Oriented Crowdsourcing: Architecture, Protocols and Algorithms*. Springer, New York, 2012.
[10] D. Schall, "Service Oriented Protocols for Human Computation", *in Handbook of Human Computation*, Springer, New York, 2013, pp. 551–560.
[11] A. Smirnov, A. Ponomarev, T. Levashova, N. Teslya, "Human-computer cloud for decision support in tourism: Approach and architecture", in Proceedings of the 19th Conference of Open Innovations Association (FRUCT), 2016, pp. 226-235.
[12] A. Smirnov, A. Ponomarev, N. Shilov, A. Kashevnik, N. Teslya, "Ontology-based human-computer cloud for decision support: architecture and applications in tourism", *International Journal of Embedded and Real-Time Communication Systems (IJERTCS)*, vol. 9, issue 1, 2018.
[13] S. Tranquillini, F. Daniel, P. Kucherbaev, F. Casati, "Modeling, enacting, and integrating custom crowdsourcing processes", *ACM Trans. Web*, vol. 9, no. 2, 2015, pp. 7:1–7:43.
[14] G. Little, "Exploring iterative and parallel human computation processes," *in Ext. Abstracts on Human Factors in Comp. Sys.*, ser. CHI EA '10, 2010, pp. 4309–4314.
[15] M.J. Franklin, D. Kossmann, T. Kraska, S. Ramesh, R. Xin, "CrowdDB: Answering queries with crowdsourcing", *in Proc. 2011 ACM SIGMOD Intl. Conf. on Management of Data*, 2011, pp. 61–72.
[16] D. Barowy, C. Curtsinger, E. Berger, A. McGregor, "Automan: A platform for integrating human-based and digital computation", *SIGPLAN Not.*, vol. 47, no. 10, 2012, pp. 639–654.
[17] A. Kulkarni, M. Can, B. Hartmann, "Turkomatic: Automatic recursive task and workflow design for mechanical turk", *in CHI '11 Ext. Abs. on Human Factors in Comp. Sys.*, 2011, pp. 2053–2058.
[18] S. Ahmad, A. Battle, Z. Malkani, S. Kamvar, "The jabberwocky programming environment for structured social computing," *in Proc. 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*, pp. 53–64.
[19] P. Minder, A. Bernstein, "Crowdlang: A programming language for the systematic exploration of human computation systems", *in Social Informatics, LNCS, vol.* 7710, 2012, pp. 124–137.

[20] A. Bozzon, M. Brambilla, S. Ceri, A. Mauri, R. Volonterio, “Pattern-based specification of crowdsourcing applications”, *in Proc. 14th Intl. Conf. on Web Engineering (ICWE)*, 2014, pp. 218–235.
[21] A. Kulkarni, M. Can, B. Hartmann, “Collaboratively crowdsourcing workflows with turkomatic,” in Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work. Seattle, Washington, USA (2012).
[22] G.A. Gorry, M.S.C. Morton, “A framework for management information systems,” *Sloan Management Review*, 13 (1), 1971, 55–70.
[23] A. Nada, M. Nasr, and M. Salah, “Service oriented approach for decision support systems,” *2014 IEEE 7th Jt. Int. Inf. Technol. Artif. Intell. Conf. ITAIC 2014*, 2014, pp. 409–413.
[24] H. Demirkan and D. Delen, “Leveraging the capabilities of service-oriented decision support systems: Putting analytics and big data in cloud,” *Decis. Support Syst.*, vol. 55, no. 1, 2013, pp. 412–421.
[25] R. Dale, R. Fusaroli, N. Duran, D. Richardson, “ The Self-Organization of Human Interaction”, *Psychology of Learning And Motivation*, vol. 59, 2013, pp. 43–95.
[26] M. Kogan, “ Digital Traces of Online Self-Organizing and Problem Solving in Disaster”, *in Proceedings of the 19th International Conference on Supporting Group Work - GROUP ’16*, 2016, pp. 479–483.
[27] V. Gorodetskii, “Self-Organization and Multiagent Systems: I. Models of Multiagent Self-Organization”, *Journal of Computer and Systems Sciences International*, vol. 51, no. 2, 2012, pp. 256–281.
[28] E. Kamar, “Directions in Hybrid Intelligence: Complementing AI Systems with Human Intelligence”, *IJCAI Invited Talk: Early Career Spotlight Track*, 2016.
[29] B. Nushi, E. Kamar, E. Horvitz, D. Kossmann, “On human intellect and machine failures: Troubleshooting integrative machine learning systems”, *31st AAAI Conference on Artificial Intelligence*, 2017, pp. 1017–1025.
[30] S. Verhulst, “Where and when AI and CI meet”, *AI & Society*, vol. 33, issue 2, May 2018, pp. 293–297.
[31] P. Dai, Mausam, D. Weld “Decision-Theoretic Control of Crowd-Sourced Workflows”, *in National Conference on Artificial Intelligence – AAAI* (2010).
[32] P. Dai, Mausam, D. Weld “Artificial Intelligence for Artificial Artificial Intelligence”, *in The 25th AAAI Conference on Artificial Intelligence*, 2011, pp. 1153–1159.
[33] M.-Ch. Yuen, I. King, K.-S. Leung, “TaskRec: A Task Recommendation Framework in Crowdsourcing Systems”, *Neural Processing Letters*, vol. 41, issue 2, 2015, pp. 223–238.
[34] P. Abbeel, A. Ng, “Apprenticeship learning via inverse reinforcement learning”, *in 21st International Conference on Machine Learning (ICML),* 2004.
[35] B. Settles, “Active Learning Literature Survey”, *Computer Sciences Technical Report* 1648. University of Wisconsin–Madison, Web: http://pages.cs.wisc.edu/~bsettles/pub/settles.activelearning.pdf.
[36] A. Felfernig, A. et al., “Configuration knowledge representations for Semantic Web applications”, *Artif. Intell. Eng. Des. Anal. Manuf*, vol. 17, 2003, pp. 31–50.
[37] Y. Liao, M. Lezoche, H. Panetto, N. Boudjlida, “Semantic annotations for semantic interoperability in a product lifecycle management context”, *Int. J. Prod. Res*, vol. 54, 2016, pp. 5534–5553.
[38] D. Forsyth, “Decision making”, *in Group Dynamics* (5th Ed.), Cengage Learning, 2006, pp. 317-349.
[39] A. Smirnov, N. Shilov, “Service-Based Socio-Cyberphysical Network Modeling for Guided Self-Organization”, *Procedia Computer Science*, vol. 64, 2015, pp. 290–297.