Usage of Smart Contracts with FCG for Dynamic Robot Coalition Formation in Precision Farming

Alexander Smirnov[1], Leonid Sheremetov[2], and Nikolay Teslya[1](✉)

[1] SPIIRAS, 14th Line 39, St. Petersburg, Russia
{smir,teslya}@iias.spb.su
[2] Mexican Petroleum Institute, Eje Central Lázaro Cárdenas Norte, 152 Mexico City, Mexico
sher@imp.mx

Abstract. In solving the problems of precision farming, an important place has the organization of joint work of robots for processing the field. The paper presents an approach to the dynamic formation of a coalition for solving the problem of precision farming, based on the use of fuzzy cooperative games in determining the structure of the coalition. To collect the initial data and save the result of the calculation of the game, a cyberphysical space is used, built based on the "blackboard" and blockchain technologies. Their use allows to combine the advantages of the concept of the Internet of Things for collecting information from sensors of agricultural robots and the immutability of data blocks to save the results of the calculation of the game in a competitive environment. To ensure the dynamic change of the coalition, smart contracts are used over the blockchain technology. Contracts contain the rules for calculating a fuzzy cooperative game and the rules for changing the composition of the coalition. As a result, the proposed approach provides the dynamic formation of a coalition with the trust of all participants and the ability to collect and disseminate information from robot sensors in a common trusted information space. To implement the blockchain and smart contract, the approach proposes to use the Hyperledger Fabric platform.

Keywords: Fuzzy logic · Coalition · Coalition game · Smart contract · Robot dynamic · Precision farming

1 Introduction

The growth of the Earth's population in the conditions of limited areas of fertile land for food production requires the search for a more efficient organization of agriculture. One of the main directions in this area is the concept of precision farming, the essence of which is the precise control of the parts of a field for growing a crop that most closely matches the part. Given the development of robotics, sensors, and the use of remote sensing of the earth, the solution to this problem can be automated to a very high degree. At the same time, to solve this problem, it is necessary to organize the joint work of many robotic devices that perform the tasks of periodically examining the geological, chemical,

J. Filipe et al. (Eds.): ICEIS 2019, LNBIP 378, pp. 115–133, 2020.
https://doi.org/10.1007/978-3-030-40783-4_7

physical and biological properties of field sites, performing operations to change some of these properties, by irrigation, fertilizing, processing, and other actions.

Among the existing models for the interaction of robots, the coalition collaboration model is most suitable for solving the problem of precision farming. This is explained by the fact that swarm and flock models are based on fairly simple rules that do not take into account or weakly take into account the difference in the functionality of robots, the dynamics of the external environment, and the long-term planning of their own actions. The coalition model, in turn, provides a complex interaction of robots based on the assessment of the gain in achieving the final goal. The implementation of the distribution model of the gain and the description of the actions required to obtain it allows the robots to form coalitions that best meet the requirements of the task, plan the order of their own actions and make decisions in case of unforeseen impacts, for example, in the case of precision farming, when com changing weather conditions or plant disease. In detail, coalition formation models are considered in the work [1].

The previous work by the authors proposes to use the model of fuzzy coalition games to evaluate the winnings of an individual robot and the entire coalition [2]. This model provides the possibility of calculating a coalition based on the estimates of each robot separately about the expected reward and a general assessment of the coalition's effectiveness.

To store the rules of the game, competencies and requirements of robots, as well as information about the current state of coalition and task distribution between robots, it is proposed to use blockchain technology and its extension with smart contracts. Smart contracts as a computerized protocol for storing and carrying out contractual clauses via blockchain become a useful tool used in many industries [3, 4]. In the previous work [2] it was proposed to use two types of smart contracts: first one for storing the rules of a coalition formation, and second one for adjusting the composition of the coalition in order to reflect environment changes. Both types of rules are defined using the theory of fuzzy sets. The contract source code, as well as the current state of the problem solution, is stored in a blockchain-based distributed ledger. This allows providing a trusted information source for robots to store and searching information about stage of task solving and current coalition state. Since the data in any block is linked with other existing blocks by calculating Merkle Tree hash, none of them can be changed without recalculating hashes of other blocks. It makes possible to provide unchangeable process logs by which one can trace the history of operations and, if necessary, find a weak point, to enhance the effectiveness of future coalitions.

Compared to the earlier work by the authors [2], this work presents a detailed description of the problem of precision farming. The process of robots' interaction during the dynamic formation of a coalition is analyzed in detail. The principle of choosing a platform for organizing the blockchain is presented, it is justified, from the point of view of the solution architecture, the use of the Hyperledger platform.

The rest of the paper is organized as follows. Related work about robot coalitions types, concept of fuzzy coalition games and using blockchain for robot coalition organization is revised in the following section. The precision agriculture problem for robot coalition is presented in Sect. 3. A fuzzy cooperative game (FCG) model with core is described in detail in Sect. 4. In Sect. 5, criteria of dynamic robot coalition formation for precision agriculture are analyzed. Section 6 provide information about implementation of smart contracts and frameworks for robots' negotiation during coalition formation.

2 Related Work

2.1 Robot Coalition Formation

There are a lot of existing models of robots' joint work such as swarms, flocks, and coalitions that differ by the freedom of single participant. The swarm model is based on the biological model of ant colony where all members have uniform rules for making decisions about their own actions in the current situation. The rules are quite simple and consistent, which guarantees the coherence of the swarm in solving the common problem. The exchange of information between the swarm participants is minimal [5].

A flock model is similar to a swarm model and differs from it by the presence of a basic hierarchy in which the main participant and his subordinates can be distinguished. The main flow of information is distributed from the higher-level participant to the lower-level, while simple rules of interaction between the lower-level participants allow them to effectively organize joint work on the tasks of the main participant.

In contrast to robots in swarms or flocks where they are limited in actions by strong rules and actions of nearest neighbors, robots in coalitions calculate their next steps based on the common goal reaching according to the current coalition state and set of alternatives provided by norms of coalition [6]. Existing models of task solving in coalition claim that a robot can receive a reward for the successful problem solving according to its contribution. The independency of robots makes it urgent to develop an approach to coalition formation and interaction organization between robots that allows making joint decision during joint solution of the problem the coalition is faced to.

There are many subject areas that require the use of a coalition of robots to solve a complex problem, including industrial cyberphysical systems, precision farming, and remote or local explore of space objects. Complex tasks in each area can be decomposed to small simple tasks (for instance in precision farming it is needed to scan the relief, check the soil composition, select and put plant or seed in the soil, water it) that are solved by single robots [7]. To form a coalition, robots provide their competences and select tasks that they can perform.

Robots are equipped with different hardware and software as well as expect different levels of reward. Therefore, it is important to consider the heterogeneity and provide common model to consensus reaching during task decomposition and resolution. Each robot is an independent agent with own competencies and goals, which he aims to achieve after the problem solving. In this case, the coalition can be considered as a union of agents with their own interests, which through the negotiation make a decision on a joint solution of the problem and the distribution of the reward.

Most of the approaches to coalition formation are characterized by the exponential nature of the computations and communications complexity. To transition from hyper-exponential and exponential complexity to polynomial, the following parameters are usually limited: the number of agents in one coalition, the number of coalitions, and the rationality of agents [8]. In this case, the additional complexity is caused by the inability to accurately estimate the size of the gain, which introduces fuzziness into the formulation of the problem.

2.2 Fuzzy Cooperative Games

The cooperative nature of modern robotic complexes causes necessity of considering them within the context of cooperative game theory in order to model and understand their cooperative behavior. The main questions of coalition formation are as follows: what coalitions will be formed, how the common winning will be distributed among them and if the obtained coalition structure is stable. Once coalitions are formed and they have a feasible set of payoffs available to its members, the question is the identification of final payoffs awarded to each player. That is, given a collection of feasible sets of payoffs, one for each coalition, can one predict or recommend a payoff (or set of payoffs) to be awarded to each player?

The payoff distribution should guarantee the stability of the coalition structure when no one player has an intention to leave a coalition because of the expectation to increase its payoff. The benefit distribution among the coalition members has proved to be fuzzy, uncertain, and ambiguous [9]. Using the theory of fuzzy cooperative games (FCGs), the uncertainty is processed by means of the introduction of a fuzzy benefit concept through the bargaining process to the conclusion about the corresponding fuzzy distribution of individual benefits among the coalition members [10].

The predictions or recommendations of payment distribution are embodied in different solution concepts. According to [11], cooperative games are divided into two classes based on the way a solution of the game is obtained: games with a solution set and games with a single solution. Games with core considered in this paper, belong to the former class and represent a mechanism for analyzing the possible set of stable outcomes of cooperative games with transferable utilities [12]. The concept of a core is attractive since it tends to maximize the sum of coalition utilities in the particular coalition structure. Such imputations are called C-stable. The core of a game with respect to a given coalition structure is defined as a set of such imputations that prevent the players from forming small coalitions by paying off all the subsets an amount, which is at least as much they would get if they form a coalition (we proceed with a formal definition of a core in the following section). Thus, the core of a game is a set of imputations which are stable.

The drawbacks of the core is that, on the one hand, the computational complexity of finding the optimal structure is high since for the game with n players at least $2^n - 1$ of the total $n^{\frac{n}{2}}$ coalition structures should be tested. On the other hand, for particular classes of the game a core can be empty. Because of these problems, using the C-stable coalition structures was quite unpopular in practical applications [6] and only recently has attracted more attention of the researchers, when the concept of fuzzy cooperative games with core was introduced [13, 14]. For realistic applications like collaborative work of groups of robots, additive environments and the absence of the restrictions on the type of membership functions should be considered [15].

For practical applications of FCGs, one of the key problems is the management of the coalition formation and payoff distribution tasks. In our previous work, a negotiation algorithm has been developed [18]. In this paper, we propose a novel approach using blockchain technology.

2.3 Blockchain in Robot Coalition Organization

Since robot coalitions are characterized by dynamic nature it is required to implement a changes in coalition composition by adding new of removing existing robots, according to changes of the problem being solved [16]. New robots should be quickly familiarized with the current state of the problem solution and provide description of own competences to help to solve the problem. At the same time, the rest part of coalition should operate without any changes as it was defined in their schedule. This problem is usually solved by using external knowledge repositories for storing the history of interaction between coalition members. Such knowledge can be stored in centralized or decentralized knowledge bases. Centralized knowledge base usually provides single access point for connecting robots to the data network. Decentralized knowledge base allows to organize a distributed network without any single access point in which the knowledge base is distributed among all participants with a share of the backup, which makes the general information space more resistant to the disconnection of one or more nodes.

Regarding the organization of robots' interaction, the blockchain is mostly used as immutable storage for information exchange and platform for smart contracts. Information stored in the blockchain could contain records about task and consumables distribution [17, 18], smart contracts and reward transactions [19], as well as global knowledge about coalition previous actions [20]. In combination with cooperative games blockchain technology can provide more trust for communication between robots, due to the storing information about transactions in immutable log that are verified by every coalition participant. In contrary to existing approaches, blockchain does not require central authority that provide trust for all nodes. All nodes negotiate with each other coming to consensus with one of possible mechanisms: Proof of Work, Proof of Stake, or practical byzantine fault tolerance [21]. The blockchain is used to provide safe and trustiness logging of robots' task distribution and rewarding for task solving.

It is also noted that the combination of the peer-to-peer network and the cryptographic algorithms used in blockchain technology allow for a negotiation process and consensus building without the presence of any controlling authorities. The distributed nature of the blockchain is proposed to be used in swarm robotics to store global knowledge about swarm actions [20]. At the same time, due to blockchain, the security of the transmitted data is ensured (garbage data can affect the achievement of a common goal), distributed decision making (creating a distributed voting system for the solution), separation of robots behavior (switching between behavior patterns depending on the role in the swarm), the emergence of new business models using the swarm. In addition, the availability of a distributed transaction ledger allows new robots to join the swarm and gain all the knowledge they have gained prior to the moment of inclusion by downloading and analyzing the transaction history.

3 Precision Farming Case with Robot Coalition

In this section, an example of solving the problems of precision farming by coalition of robots is considered. The problem is stated as follows (see Fig. 1). There is a field with various geological and ecological characteristics of soils, suitable for growing several types of crops that require different growth conditions. The field is processed by several robots equipped with devices for plowing, loosening, planting, watering, fertilizing and harvesting crops.

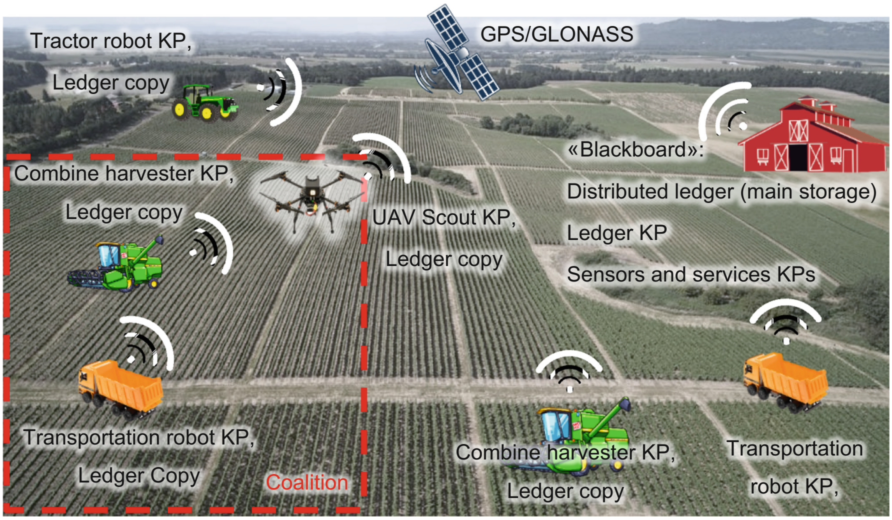

Fig. 1. Coalition formation for precision farming task (based on [2]).

In the set of robots, the following types can be distinguished, according to the main tasks to be solved: combine harvesters, the list of tasks of which includes planting and harvesting, studying the composition of the soil; a transport robot that carries out the movement of seed and finished crop, as well as fertilizers between the combine harvester and the warehouse; a robot scout (wheeled or UAV) that perform periodic field reconnaissance to measure light characteristics, wind direction and strength, soil moisture, and plant status. Each robot is equipped with a set of sensors and actuators that allow exploring the soil structure, light and humidity conditions ant take a picture in each sector of the field. Based on the explored data a map of the field is built, where the current conditions are bound with the coordinates from GPS/GLONASS satellites. Crops are selected for each sector based on the sector conditions that are the most favorable in terms of yield, as well as technologies are selected for their care of planted agriculture. The technology of caring for each type of crop requires the use of robots that are capable of carrying out specific operations for the culture chosen, while some robots are capable of performing operations on several technologies, or the technologies can have common steps being solved by the same type of robots. Storing of the history of fieldwork and crops can help both in subsequent decision-making and in drawing up special reporting on the production cycle, which is increasingly required by the laws of developed countries. In addition, the history of growing process can be shared with customers to provide insight information about a process and help customers to get food based on the quality of farming process. This, as well as the requirement of storing the history of fieldwork requires the presence of a repository, in which the history of actions and the results of field processing will be recorded.

Robots are interacting through the cyberphysical framework presented on Fig. 2. The framework is based on the smart cyberphysical space created by extending of smart space concept (based on the "blackboard") with distributed ledger based on blockchain [22, 23]. This combination provides the ability to organize basic interaction of robots in the physical and cyber (virtual) spaces with storing history of each robot actions sequence in immutable blockchain structure. The interaction includes solo and joint manipulations with physical objects, information exchange about the current state of robots and objects for planning further joint actions during the coalition formation.

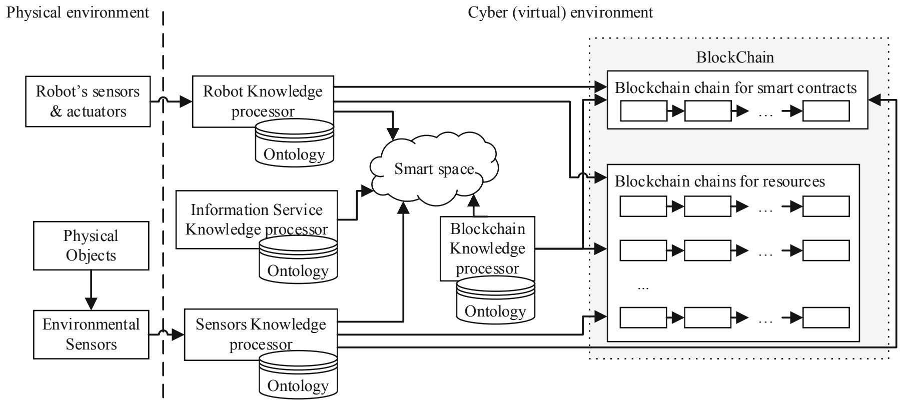

Fig. 2. Cyberphysical framework with blockchain support [2].

One of the typical coalitional tasks for precision agriculture is the field exploration where different types of robots are engaged. The overall task of the study is divided between them into subtasks, according to the available competences of the robots. In this case, task division is performed using the cooperative game model for the dynamic coalition formation. Within the framework of this model, individual robots interact with each other, putting forward their competencies and requirements based on which the selection of coalition participants is being carried out and their effectiveness in solving the assigned task is estimated. The process of coalition formation is presented with the sequence diagram on Fig. 3.

4 Fuzzy Cooperative Game Model with Core

A generalized model of a fuzzy cooperative game (FCG) with core was proposed in [15, 24, 25]. As shown in [15], the concept of a core is attractive since it tends to maximize

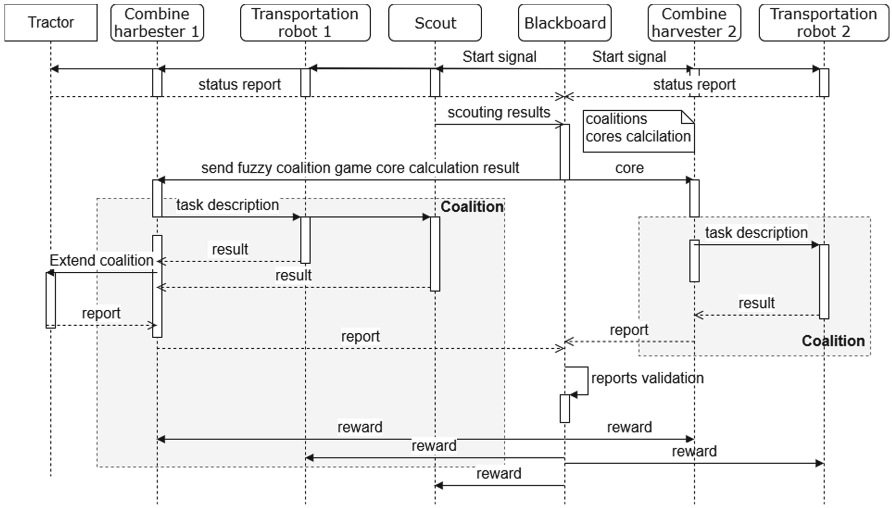

Fig. 3. Sequence diagram of coalition formation in precision agriculture task.

the sum of coalition utilities in the particular coalition structure. The core of a game is a set of imputations, which are stable. The proposed model helped solving the problems of the computational complexity of finding the optimal structure and of the empty core, which enabled its use in practical applications of selecting robots in coalitions.

A FCG is defined as a pair $(Robot, w)$, where $Robot$ is nonempty and finite set of players, subsets of $Robot$ joining together to fulfil some task T_i are called coalitions K, and w is called a characteristic function of the game, being $w : 2^n \to \Re^+$ a mapping connecting every coalition $K \subset Robot$ with a fuzzy quantity $w(K) \in \Re^+$, with a membership function $\mu_K : R \to [0, 1]$. A modal value of $w(K)$ corresponds to the characteristic function of the crisp game $v(K)$: $\max\mu_K(w(K)) = \mu_K(\nu(K))$. For an empty coalition $w(\emptyset) = 0$. A fuzzy core for the game $(Robot, w)$ with the imputation $X = \left(x_{ij}\right)_{i\in I, j\in Robot} \in \Re^+$ is a fuzzy subset C_F of $\Re^+$:

$$C_F = \left\{ x_{ij} \in \Re^+ : \nu f = \left(w(Robot), \sum_{\substack{i\in I,\\ j\in Robot}} x_{ij}\varphi_{ij} \right), \min_{\substack{K_i\in k\\ j\in Robot}} \left(\nu f = \left(\sum_{j\in K_i} x_{ij}\varphi_{ij}, w(K_i) \right) \right) \right\} \tag{1}$$

where x_{ij} is the fuzzy payment of a robot j participating in a coalition i, $i = 1, 2, \ldots, I$, $j = 1, 2, \ldots, N$, $\bar{k} = [K_1, K_2, \ldots, K_l]$ is the ordered structure of effective coalitions; φ is a fuzzy partial order relation with a membership function $\nu f =: R \times R \to [0, 1]$, and φ_{ij} is a binary variable such that:

$$\varphi_{ij} = \begin{cases} 1, if\ robot\ j\ participates\ in\ a\ coalition\ i; \\ 0, otherwise. \end{cases} \tag{2}$$

This variable can be considered as a result of some robot's strategy on joining a coalition.

A fuzzy partial order relation is defined as follows (for more details see [26]). Let a, b be fuzzy numbers with membership functions μ_a and μ_b respectively, then the possibility of partial order $a\phi = b$ is defined as $\nu\phi = (a, b) \in [0, 1]$ as follows:

$$\nu\phi = (a, b) = \sup_{\substack{x, y \in R \\ x \geq y}} (\min(\mu_a(x), \mu_b(y))) \tag{3}$$

The core C_F is the set of possible distributions of the total payment achievable by the coalitions, and none of coalitions can offer to its members more than they can obtain accepting some imputation from the core. The first argument of the core C_F indicates that the payments for the grand coalition are less than the characteristic function of the game. The second argument reflects the property of group rationality of the players, that there is no other payoff vector, which yields more to each player. The membership function $\mu_{C_F} : R \to [0, 1]$, is defined as:

$$\mu_{C_F}(x) = \min\left\{ \nu f = \left(w(Robot), \sum_{\substack{i \in I \\ j \in Robot}} x_{ij}\varphi_{ij} \right), \min_{\substack{K_i \in k \\ j \in Robot}} \left(\nu f = \left(\sum_{j \in K_i} x_{ij}\varphi_{ij}, w(K_i) \right) \right) \right\} \tag{4}$$

With the possibility that a non-empty core C_F of the game $(Robot, w)$ exists:

$$\gamma_{C_F}(Robot, w) = \sup\left(\mu_{C_F}(x) : x \in \Re^n\right) \tag{5}$$

The solution of a cooperative game is a coalition configuration (S, x) which consists of (i) a partition S of $Robot$, the so-called coalition structure, and (ii) an efficient payoff distribution x which assigns each robot in Robot its payoff out of the utility of the coalition it is member of in a given coalition structure S. A coalition configuration (S, x) is called stable if no robot has an incentive to leave its coalition in S due to its assigned payoff xi.

It was proved that the fuzzy set of coalition structures forming the game core represents a subset of the fuzzy set formed by the structure of effective coalitions. In turn, this inference allows us to specify the upper possibility bound for the core, which is a very important condition for the process of solution searching, because in this case, the presence of a solution that meets the efficiency condition may serve as the signal to terminate the search algorithm [25].

The game purpose is to generate an effective structure of robot coalitions for executing some task. In turn, the generated structure of robot coalitions represents the optimal configuration of the grand coalition.

Individual robots use the technique of nonlinear fuzzy regression to estimate the parameters of utility functions for their payments [27]. A "coalition robot" is enabled for constructing membership functions (MF) of coalitions and generating the game core (fuzzy-number generator). The algorithm of fuzzy number summation for obtaining coalition membership functions represents an important element of the model. The sum operation is based on Zadeh extension principle [26] for fuzzy numbers a and b (which are convex sets normalized in R):

$$\mu_{a(*)b}(Z) = \sup_{z=x*y} \min(\mu_a(x), \mu_b(y)) \tag{6}$$

where * can designate the sum ⊕ or the product • of fuzzy numbers. Each fuzzy set is decomposed into two segments, a non-decreasing and non-increasing one. The operation * is performed for every group of n segments (one segment for each fuzzy set) that belong to the same class (non-decreasing or non-increasing one). Thus, a fuzzy set is generated for every group of n segments. The summation result is derived as superposition of these sets, which gives the membership function as the sum of n fuzzy numbers.

5 Criteria for Dynamic Robot Coalition Formation

Joint problem solving requires a well-coordinated interaction of the participants' actions during the coalition formation. Regardless of the coalition model used, coalition formation process can be considered as three types of interrelated actions:

- Generation of a coalition structure. The structure includes a subset of robots that will jointly interact to coordinate their activities on problem solving;
- Solving the problem of optimization of each coalition; union of agents' competencies for effective problem solving. On this stage, the task is dividing to the subtasks and robots are assigned to each subtask based on the benefit it can bring to coalition. The benefit is estimated based on the functions, winning expectations and efficiency of robot;
- Profit sharing between agents.

If the actions presented above are performed before problem solving, a static coalition formation is considered. The structure of static coalitions does not change over the time. Such a situation is typical for environments with quite low dynamic. At the time of optimization of the coalition, in parallel to coalition structure also a plan for solving the problem is calculated as well as all possible deviations from the plan. The deviations can be predicted due to the known patterns or equations of situation development. In case of a deviation, for example, due to the failure of one of the coalition members, the correction of the plan is carried out by the forces of the last coalition members taking into account the changed conditions in order to return to the original plan with minimal losses. This approach is quite rough due to the situation that coalition will fail in case of unpredicted deviation happens or a set of deviations will be accumulated and coalition cannot fix all of them based only on pre-calculated actions.

A more complex, but flexible case of a coalition formation is the dynamic formation. In this case, during the optimization, an initial plan of problem solving is formed same as for the static coalition. However, in case of deviation from the plan, a return is made by changing the structure of the coalition, for example, by adding a new participant or reassigning and rescheduling subtasks. To do this, the rules for the formation of the coalition should describe actions for extraordinary situations, as well as the overall benefit of the coalition, so the plan of action is dynamically recalculated considering the context of the task has changed.

One of the following parameters can be used to evaluate the coalition efficiency:

- Minimizing the energy spent. Since all robots are autonomous it means that they used electrical or fuel power (or both) to move and perform any kind of actions. Therefore, the solution of each task or sub-task can be estimated by the energy (charge of the battery of fuel level) $E_k(T_i)$ of the robot k that is spent to solve it by using own competencies:

$$E_k(T_i) = \sum\nolimits_j f_{T_i}\left(b_j^k\right) \cdot \varphi(T_i, k, j) \tag{7}$$

The exact amount of energy spent on solving the problem is not possible to estimate precisely due to the influence of a large number of external and internal factors. However, based on average data over the similar problems, it is possible to obtain an approximate estimation, which, however, introduces fuzziness into the final decision to form a coalition. In this case, the robots are interested in spending minimum energy with the maximum efficiency. The coalition efficiency can be estimated as relation of the number of solved problems to the total energy expended:

$$v\left(K_{T_i}\right) = Payoff(T_i) - \min_{K_i \in k} \sum\nolimits_{j \in K_i} E_j(T_i) \tag{8}$$

- Robot uptime can serve as an analogue of the estimated energy expended. Robots are consisting of a great amount of parts and units and each of them has the probability of failure, which increases as the operation proceeds. Solution of each task requires a certain time of unit operation. Thus, the estimation of failure probability is the ratio of the time difference between the time of the node work and the average uptime of this type of robot units: $\mathrm{P_{C_i}} = \frac{\mathrm{T^w_{c_i} - T^m_{c_i}}}{\mathrm{T^{avg}_{c_i}}}$, where P_{C_i} – failure probability of unit c_i by robot $\mathrm{r_j}$, $T^w_{c_i}$ – total work duration of the unit c_i, $T^m_{c_i}$ – last maintenance time point. The probability of entire robot failure is evaluated according to the maximum probability of nodes failure $\mathrm{P_r} = \max_i \mathrm{P_{c_i}}$. An estimation of this probability is also approximate and bring fuzziness in solution of coalition game. The efficiency criterion in this case will be the maximum duration of the coalition's overall work to the next maintenance, which requires such a distribution of tasks among the participants, so that the probability of coalition member failure will be minimal.
- Maximizing the coalition benefit. For example, in relation to precision farming, the coalition's benefit is the cumulative crop of all cultures on the field. This requires coordinated and timely interaction of all robots in a dynamic coalition. The value of the solution of the problem decreases with the passage of time: the longer the task is postponed, the less benefit it can provide. For example, untimely watering due to the lack of robots in a coalition with a enough supply of water can cause the death of a crop, which will reduce the potential benefit. Thus, the choice of coalition participants and the distribution of tasks among them should be carried out in such a way as to minimize downtime and, accordingly, to maximize the overall benefit of the coalition.

6 Implementation of a Fuzzy Cooperative Game Over Smart Contracts

In this section, the implementation of the rules of the coalition game is proposed by means of smart contracts that describe the interaction of robots during the coalition formation. This is enabled by the ability of smart contracts within the scope of blockchain technology to describe complex algorithms by using the Turing-complete programming language. Examples include Solidity for the Ethereum platform [28] or GoLang and JavaScript for Hyperledger Fabric [29]. To date, many platforms for blockchain organization have been developed, a comparison of the main ones is presented in Table 1.

Table 1. Overview of existing blockchain platforms.

Platform	Permissions	Consensus protocol	Smart contract (language)	Performance (trans/sec, tps)
Bitcoin [30]	Public	Proof-of-Work	Partly	Up to 7
Ethereum [28]	Public & private	Proof-of-Work	Yes (Solidity)	15–25
Corda [31]	Public	PBFT, PoET	Yes (Java)	170
HyperLedger (Fabric, Burrow) [29]	Public & private	Pluggable (PBFT, PoET)	Yes (Go, Java, Python)	3500
Symbiont [21]	Private	BFT	Yes (Python)	~80000
Kadena	Public & private	PBFT, SmartBF	Partly (Pact lang.)	7000
Quorum (ETH, enterprise)	Private	FBA	Yes (Solidity)	35–130
HydraChain (ETH)	Private	Proof-of-Work	Yes (Python)	15–25
Exonum [32]	Private	FBA, PBFT	Yes, Rust	5000

To use the blockchain for storing the results of a coalition game and organizing interaction between robots in precision farming, several requirements must be observed. The platform used should support the organization of public and private blockchain structures. Since the consistency of the data exchanged by robots is important when solving the problem of precision farming, the consensus algorithm used should correctly handle disconnection or coupling of some nodes (presented by robots). To store the core of the coalition and the particular tasks of robots, the ability to describe any algorithm in a smart contract is required. For this, the language used to describe the contract must be Turing-complete, that is, provide the ability to describe algorithms of any complexity. And the last requirement is the speed of transaction processing for the rapid dissemination of verified data between coalition members, which will ensure the speed of decision-making in a critical situation. Table 1 provides a comparison of the main platforms according to the requirements presented above.

Table 1 also shows that the Hyperledger Fabric platform meets the most presented requirements. Its detailed description and adaptation to the tasks of precision farming will be presented later in Sect. 6.2.

6.1 Smart Contract Theory

The idea of smart contract was proposed in 1994 by Nick Szabo. He had defined smart contract as "a set of promises, specified in digital form, including protocols within which the parties perform on these promises." [33] The example of resource exchange is presented on Fig. 4.

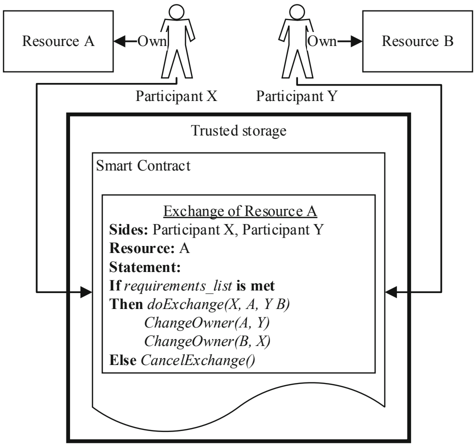

Fig. 4. Smart contract usage example [2].

In scope of the current level of information systems, smart contracts are viewed as decentralized applications that are available to all sides of the contract through the cloud of in decentralized way, for instance, blockchain. Due to the use of Turing-complete language for contract description, it is possible to implement rather complex algorithms. At the same time, it is mandatory to have conditions under which the contract must be executed as well as the list of actions assigned to the submitted conditions. All conditions of a smart contract must be described in a strong mathematical way and provide clear execution logic. In this regard, the first smart contracts in the blockchain are created to formalize the simplest relationships and consist of a small number of conditions.

To be valid and trusted smart contracts have to be signed by all sides with their private key [34] and sent as a transaction to be written to in the cloud or decentralized storage. After signing by all contract sides, the smart contract comes into force. To ensure the automated performance of contract obligations, an environment of existence is required that allows fully automated execution of contracts. This means that smart contracts can only exist within an environment that has unrestricted access to executable code of smart contract objects. Having unimpeded access to the objects of the contract, the smart contract monitors the specified conditions of achievement or violation of the points and makes independent decisions based on the programmed conditions. Thus, the main principle of a smart contract is the complete automation and reliability of the performance of contractual relations between participants.

6.2 Smart Contracts for Robot Coalition Formation

Figure 5 shows the scheme of interaction of robots in the coalition by means of a blockchain. It is proposed to use two kinds of chains in the blockchain network system for robot interaction: (i) for storing resources and (ii) for storing contracts. All system resources including consumables, energy, reward, which are represented by tokens, are stored in the resources chains. In the chain with contracts, the rules of cooperative game are stored, which are used by the robot's coordinators during the coalition forming and the distribution of tasks. The first contracts in the chain of contracts are rules for processing tasks and assigning coalition core. New task is formed with a program interface outside a coalition by problem manager, or by the cores of another coalition in case of obtaining a new context that cannot be processed by the existing coalition. New tasks are stored in the contract chain of the blockchain, from where they become available for all coalition cores. Tasks contain a formalized description of the goal, the initial parameters and the amount of reward for the solution.

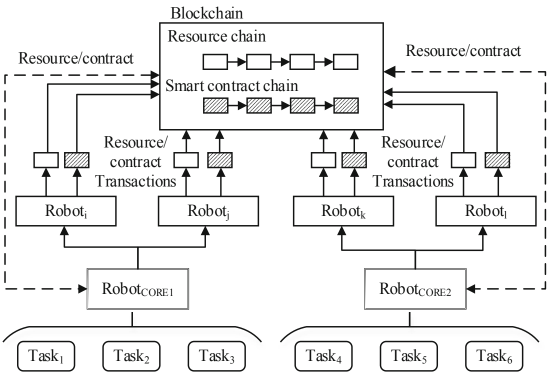

Fig. 5. Robot interaction in coalition through blockchain and smart contracts [2].

The robot coordinator selects robots guided by contracts that describe their competencies and reward expectations, as well as the rules of the cooperative game, defined for the subject area to which the task belongs. If the robot can participate in several coalitions, each robot coordinator calculates the cooperative game core and win for each of the coalitions, as well as the availability of sufficient resources for the robot successful work. If there are enough resources for robot's operations, it can participate in several coalitions. Otherwise, the robot is assigned to a coalition for which it can bring the highest benefit. The reward for the successful solution of the problem is distributed among the coalition members based on the reward rules for the cooperative game, described in the code of the relevant contract.

The blockchain network for the case study has been implemented based on the Hyperledger Fabric platform that is provided by community of software and hardware companies leading by IBM [29]. The platform provides possibilities of wide range configurations: changing of a core database for transactions and block storing, changing of consensus mechanisms, and changing signature algorithms for peers' interaction with blockchain. For the case study presented in the paper, the default configuration has been used that includes Byzantine Fault Tolerate consensus mechanism based on BFT-SMaRT core [35], Apache CouchDB as a database and an internal solution for peer certification. This configuration provides processing of more than 3500 transactions per second with latency of hundred milliseconds.

The choice of the Hyperledger Fabric platform is also justified by the peculiarities of its architecture, which makes it easy to adapt the coalition structure obtained when calculating the coalition game into the platform structure (see Fig. 6). The main elements of the architecture are nodes, divided into three levels: "Client", "Peer", "Orderer". Client level corresponds to robots whose main task is to conduct reconnaissance and send data, or to perform operations and report on their performance. In precision farming, such robots can be individual harvester combine tools, scouts, and transport robots. Above them, in terms of level, are devices that collect information and execute the code of contracts – "Peer" that can be presented by control block of harvester combine. Their main task is to collect information from the lower level, process it using smart contracts and transfer it to the upper level, in which information will be disseminated and stored. The upper level – "Orderer" - corresponds to the "Blackboard" device in the robot interaction scheme on Fig. 2. Its task is to store information in the appropriate block chain, to ensure the coordination and distribution of the new block between other Orderers or between Peers.

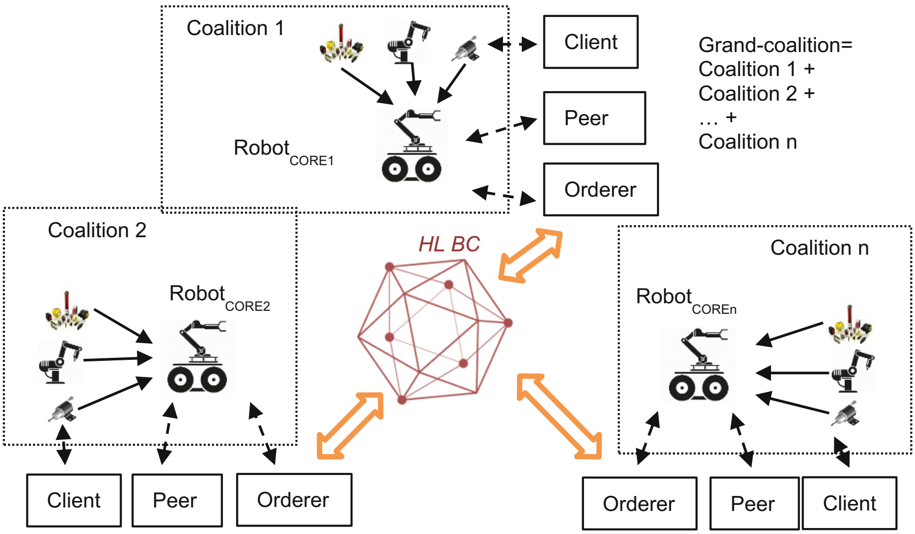

Fig. 6. Model of robot interaction through Hyperledger Fabric blockchain platform.

The Hyperledger Fabric platform also provides possibility to create smart contracts called chaincodes (program code that describes interaction between resources) using Go or Java programming languages. The chaincodes are running in isolated containers of core peers of Hyperledger based on the Docker technology stack. Each chaincode contains rules for cooperative fuzzy game that used for coalition participants negotiation. The example of chaincode for core calculation is presented at listing 1.

Listing 1. Example of a chaincode for coalition core calculation [2].

```
var robots []Robot // Robot list
var tasks []Task     // Tasks to be solved
var core []FCG // Fuzzy coalition core
var coreMaxGain FGC  // Core with max gain
func coreCalc(stub shim.ChaincodeStubInterface, args []string)
(string, error) {
    robots[i], tasks[j] = args[i], args[j]
    core = FCGCalculation(robots, tasks) //according to equa-
    tion(2)
    for c in core {
      if c.gain > coreMaxGain.gain
        coreMaxGain = c
      }
      for rob in robots{
      // bind task for robot according to formula (1)
      stub.PutState(rob, c.getTask(rob)) // Estimate and fix
      processing time
      stub.PutState(c.getTask(rob), CalcProcTime(rob))
    }
  }
}
```

7 Conclusions

Solving the problems of precision farming requires the development of a new approaches, which provide the dynamic formation of a coalition of robots for processing fields taking into account the current situation. This paper presents an approach based on the use of fuzzy coalition games and blockchain technology.

The main difference of the presented approach is the integration of the mathematical apparatus of the fuzzy coalition game and the trusted information space based on the "blackboard" and blockchain technologies. Their combination ensured that the results of the coalition game are preserved in an unchanged form, which is important in the case of the interaction of competing agents, which are robots in the coalition. Competition arises because robots, in addition to interest in achieving common goals, pursue their own interest, which consists in obtaining the maximum individual gain.

Fuzziness in the presented approach serves as the fundamental component of realistic cooperation models when there exist fuzzy expectations of player and coalition benefits. When an effective solution is found, individual benefits for players (the agreement efficiency) increase, as well as the capability of the coalition to find an effective and stable agreement. The blockchain model allows to avoid the synchronization problem, which is critical for distributed negotiation algorithms with large robot populations.

The integration of fuzzy coalition games with smart contracts can make coalition formation more transparent and to smooth out the operations of the tasks. The use of Internet of Things (IoT) concept with the blockchain, provides continuous tracking of food, from warehouses to manufacturers and enables verification at each stage of the precision agriculture task solving. If any stakeholder fails to meet the terms of the contract, for instance if a robot did not perform some operation on time, it would be clear for every party to see and new coalitions can be arranged dynamically.

The future work is aimed in developing smart contracts for participants changing in coalition. The changing process will be based on the negotiation between coalition core and robots outside the coalition that can perform task instead of failed coalition members.

Acknowledgements. The present research was supported by the projects funded through grants # 17-29-07073, of the Russian Foundation for Basic Research. The part of research is supported by the program №7 “New developments in the areas of energy, mechanics and robotics” of the Russian Academy of Sciences.

References

1. Mouradian, C., Sahoo, J., Glitho, R.H., Morrow, M.J., Polakos, P.A.: A coalition formation algorithm for Multi-Robot Task Allocation in large-scale natural disasters. In: 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC). IEEE, pp. 1909–1914 (2017)
2. Smirnov, A., Sheremetov, L., Teslya, N.: Fuzzy cooperative games usage in smart contracts for dynamic robot coalition formation: approach and use case description. In: ICEIS 2019 - Proceedings of the 21st International Conference on Enterprise Information Systems, pp. 349–358 (2019)
3. Cong, L.W., He, Z., Zheng, J.: Blockchain disruption and smart contracts. SSRN Electron. J. **48** (2017). https://doi.org/10.2139/ssrn.2985764
4. Delmolino, K., Arnett, M., Kosba, A., Miller, A., Shi, E.: Step by step towards creating a safe smart contract: lessons and insights from a cryptocurrency lab. In: Clark, J., Meiklejohn, S., Ryan, P.Y.A., Wallach, D., Brenner, M., Rohloff, K. (eds.) FC 2016. LNCS, vol. 9604, pp. 79–94. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53357-4_6
5. Ordaz-Rivas, E., Rodríguez-Liñán, A., Torres-Treviño, L.: Collaboration of robot swarms with a relation of individuals with prey-predator type. Smart Technology, pp. 121–132. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (2018)
6. Klusch, M., Gerber, A.: Dynamic coalition formation among rational agents. IEEE Intell. Syst. **17**, 42–47 (2002). https://doi.org/10.1109/MIS.2002.1005630
7. Kardos, C., Kovács, A., Váncza, J.: Decomposition approach to optimal feature-based assembly planning. CIRP Ann. **66**, 417–420 (2017). https://doi.org/10.1016/j.cirp.2017.04.002
8. Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Wooldridge, M., Sierra, C.: Automated negotiation: prospects, methods and challenges. Group Decis. Negot. **10**, 199–215 (2001). https://doi.org/10.1023/A:1008746126376

9. Hosam, H., Khaldoun, Z.: Planning coalition formation under uncertainty: auction approach. In: Proceedings - 2006 International Conference on Information and Communication Technologies: From Theory to Applications, ICTTA 2006, pp. 3013–3017. IEEE (2006)
10. Aubin, J.-P.: Cooperative Fuzzy Games. Math. Oper. Res. **6**, 1–13 (1981). https://doi.org/10.1287/moor.6.1.1
11. Kahan, J.P., Rapoport, A.: Theories of Coalition Formation. Lawrence Erlbaum Associates, Inc., Hillsdale (1984)
12. Gillies, D.B.: Some theorems on n-person games. Princeton University (1953)
13. Mareš, M.: Fuzzy Cooperative Games. Physica-Verlag HD, Heidelberg (2001). https://doi.org/10.1007/978-3-7908-1820-8
14. Shen, P., Gao, J.: Coalitional game with fuzzy payoffs and credibilistic core. Soft. Comput. **15**, 781–786 (2010). https://doi.org/10.1007/s00500-010-0632-9
15. Smirnov, A.V., Sheremetov, L.B.: Models of coalition formation among cooperative agents: the current state and prospects of research. Sci. Tech. Inf. Process. **39**, 283–292 (2012). https://doi.org/10.3103/S014768821205005X
16. Bayram, H., Bozma, H.I.: Coalition formation games for dynamic multirobot tasks. Int. J. Rob. Res. **35**, 514–527 (2015). https://doi.org/10.1177/0278364915595707
17. Verma, D., Desai, N., Preece, A., Taylor, I.: A block chain based architecture for asset management in coalition operations. In: Pham, T., Kolodny, M.A. (eds.) Proceedings of the SPIE 10190, Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR VIII, p. 101900Y (2017)
18. Dorri, A., Kanhere, S.S., Jurdak, R.: Towards an optimized BlockChain for IoT. In: Proceedings of the Second International Conference on Internet-of-Things Des Implement - IoTDI 2017, pp. 173–178 (2017). https://doi.org/10.1145/3054977.3055003
19. Zhang, Y., Wen, J.: The IoT electric business model: using blockchain technology for the internet of things. Peer-to-Peer Netw. Appl. **10**, 983–994 (2017). https://doi.org/10.1007/s12083-016-0456-1
20. Ferrer, E.C.: The blockchain: a new framework for robotic swarm systems. Adv. Intell. Syst. Comput. **881**, 1037–1058 (2019). https://doi.org/10.1007/978-3-030-02683-7_77
21. Cachin, C., Vukolić, M.: Blockchain Consensus Protocols in the Wild, 24 (2017). https://doi.org/10.4230/LIPIcs.DISC.2017.1
22. Teslya, N., Ryabchikov, I.: Blockchain-based platform architecture for industrial IoT. In: Conference of Open Innovation Association, FRUCT (2018)
23. Smirnov, A., Kashevnik, A., Ponomarev, A., Shilov, N.: Context-aware decision support in socio-cyberphysical systems: from smart space-based applications to human-computer cloud services. In: Demazeau, Y., Davidsson, P., Bajo, J., Vale, Z. (eds.) PAAMS 2017. LNCS (LNAI), vol. 10349, pp. 3–15. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59930-4_1
24. Sheremetov, L.B., Smirnov, A.V.: A fuzzy cooperative game model for configuration management for open supply networks. Contrib. Game Theory Manag. **4**, 433–446 (2011)
25. Sheremetov, L.B.: A model of fuzzy coalition games in problems of configuring open supply networks. J. Comput. Syst. Sci. Int. **48**, 765–778 (2009). https://doi.org/10.1134/S1064230709050116
26. Zadeh, L.A.: Similarity relations and fuzzy orderings. Inf. Sci. (Ny) **3**, 177–200 (1971). https://doi.org/10.1016/S0020-0255(71)80005-1
27. Haekwan, L., Tanaka, H.: Fuzzy approximations with non-symmetric fuzzy parameters in fuzzy regression analysis. J. Oper. Res. Soc. Jpn. **42**, 98–112 (1999)
28. Buterin, V.: A next-generation smart contract and decentralized application platform. Etherum 1–36 (2014). https://doi.org/10.5663/aps.v1i1.10138
29. Androulaki, E., et al.: Hyperledger fabric. In: Proceedings of the Thirteenth EuroSys Conference on - EuroSys 2018, pp 1–15. ACM Press, New York (2018)

30. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system. WwwBitcoinOrg 9 (2008). https://doi.org/10.1007/s10838-008-9062-0
31. Brown, R.G., Carlyle, J., Grigg, I., Hearn, M.: Corda : an introduction 1–15 (2016). https://doi.org/10.13140/RG.2.2.30487.37284
32. What is Exonum - Exonum Documentation (2018). https://exonum.com/doc/get-started/what-is-exonum/. Accessed 1 Mar 2018
33. Szabo, N.: Smart contracts: building blocks for digital markets copyright. In: alamut.com (1996). http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html. Accessed 16 Sep 2017
34. Goldreich, O.: Foundations of cryptography, 1st edn. Cambridge University Press, Cambridge (2006)
35. Bessani, A., Sousa, J., Vukolić, M.: A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform. In: Proceedings of the 1st Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers - SERIAL 2017, pp. 1–10. ACM Press, New York (2017)