Cilt 11, Sayı 2, Sayfa 27-35, 2022 Volume 11, Issue 2, Page 27-35, 2022

Araştırma Makalesi https://doi.org/10.46810/tdfd.1017499 Research Article

Türk Doğa ve Fen Dergisi
Turkish Journal of Nature and Science
www.dergipark.gov.tr/tdfd

Application of Blockchain Powered Mobile Robots in Healthcare: Use Cases, Research Challenges and Future Trends

Mehmed Oğuz ŞEN[1*], Fatih OKUMUŞ[2], Adnan Fatih KOCAMAZ[1]

[1] İnönü University, Faculty of Engineering, Department of Computer Engineering, Malatya, Türkiye
[2] İnönü University, Faculty of Engineering, Department of Software Engineering, Malatya, Türkiye
Mehmed Oğuz ŞEN ORCID No: 0000-0002-0533-1859
Fatih OKUMUŞ ORCID No: 0000-0003-3046-9558
Adnan Fatih KOCAMAZ ORCID No: 0000-0002-7729-8322

Corresponding author: oguz.sen@inonu.edu.tr

(Received: 10.03.2022, Accepted: 08.04.2022, Online Publication: 29.06.2022)

Keywords
Blockchain, Mobile robots, Multi robot systems, Hyperledger Fabric, Blockchain in healthcare, Medical robots

Abstract: Using service robots in healthcare is gaining importance in case of emergent situations like pandemics where human labour is considered risky. Multi robot systems of mobile robots have the potential to perform simple but vital tasks in healthcare. However, centralized control with a server computer of these systems carry the risks of single point of failure and ineffective operation of robots, thus decentralized control with blockchain integration offers a better solution. We mention research challenges regarding blockchain powered multi robot systems of mobile robots from use case, blockchain technology and its integration into current computing systems used in medical centers aspects. Then we propose a method for decentralized management and task distribution in a multi robot system by using Hyperledger Fabric as a permissioned blockchain platform and give common use case scenarios. In this system, tasks are assigned to robots depending on the selection of nearest available robots to the task target. Each robot runs the smart contract containing the task assignment method, so that data traffic for the task assignment process is distributed among the network, instead of stacking up on a single line as in a centralized system. Future research issues and directions for future works are also stated as a conclusion.

Blokzincir Destekli Mobil Robotların Sağlık Hizmetinde Uygulanması: Kullanım Örnekleri, Araştırma Zorlukları ve Gelecek Trendler

Anahtar Kelimeler
Blokzincir, Hareketli robotlar, Çoklu robot sistemleri, Hyperledger Fabric, Sağlık hizmetlerinde blokzincir, Tıbbi robotlar

Öz: İnsan emeğinin riskli görüldüğü pandemi gibi acil durumlarda hizmet robotlarının sağlık hizmetlerinde kullanılması önem kazanmaktadır. Mobil robotlardan oluşan çoklu robot sistemleri, sağlık hizmetlerinde basit ama hayati görevleri yerine getirme potansiyeline sahiptir. Ancak bu sistemlerin bir sunucu bilgisayar ile merkezi kontrolü, tek arıza noktası ve robotların etkisiz çalışması risklerini taşır, bu nedenle blok zincir entegrasyonu ile merkezi olmayan kontrol daha iyi bir çözüm sunar. Bu çalışmada blokzincir destekli çoklu mobil robot sistemleri ile ilgili araştırma zorlukları; örnek kullanım durumları, blokzincir teknolojisi ve tıp merkezlerinde kullanılan mevcut bilgi işlem sistemlerine entegrasyonu açılarından ele alınmıştır. Ardından, izne bağlı erişimli bir blok zinciri platformu olan Hyperledger Fabric kullanarak çoklu robot sistemlerinde merkezi olmayan yönetim ve görev dağıtımı için bir yöntem önerilmekte ve ortak kullanım senaryoları verilmektedir. Söz konusu sistemde, görev hedefine en yakın faal robotların seçimine bağlı olarak robotlara görevler atanmaktadır. Her robot, görev atama fonksiyonunu içeren akıllı sözleşmeyi çalıştırmakta; böylece görev atama işlemi için veri trafiği, merkezi bir sistemde olduğu gibi tek bir hatta yığılmak yerine ağ üzerine dağıtılmaktadır. Gelecekteki araştırma konuları ve gelecekteki çalışmalar için yönelimler de sonuç kısmında belirtilmiştir.

1. INTRODUCTION

Pandemics like COVID-19 reminded the importance of utilization of service robots in healthcare not only to protect medical personnel from being infected but also to lower the contagion of viruses. There already exist some applications of service robots being used in medical centers as distribution of patients' bare necessities (food, medicine etc.), diagnosis requiring close contact and sanitization of critical places (surgery rooms, laboratories etc.). In most of these applications, management and task distribution of service robots is performed using centralized systems; the robots are remotely connected to server computers that are integrated into the medical center's automation system. In this paper we present an application framework for multiple mobile medical robots in healthcare, introducing the decentralized approach using blockchain systems.

Using static systems for the communication between mobile robots in a multi robot system becomes a hampering factor when new robots are added to the system. Protocols, procedures and hardware used in the communication needs to be different when there are 2, 3, … or n robots operating in the multi robot system. With its decentralized network structure, blockchain makes dynamic communication between robots possible by providing a smooth communication infrastructure which is able to operate fluently, independent from the number of robots.

This study aims to show how a network of mobile service robots for healthcare can communicate and perform tasks assigned to them via a blockchain platform by stating its advantages over centralized systems. Contrary to centralized systems where a robot is expected to communicate only with a server for receiving orders or sending data, a medical service robot can propagate its current state by sharing its position information, status and data acquired from its sensor suite over the blockchain network to the rest of the robots. Therefore, it can inform other robots of its presence as to be distinguished from static obstacles and can be accessed at any time via other robots instead of setting multiple Wi-Fi routers for a seamless connection with a server. In a centralized system, task assignment and robot control are performed on the central server-side, and robots are considered dumb machines. However dynamic task distribution regarding efficient task assignment and autonomous behaviour of robots without the need of a server can be accomplished in a decentralized system. Users can connect to a decentralized network via a client, see the progress of robots, create tasks for robots and request task assignments. This should not be confused with in a server manner; robots are not directly being intervened at client side, only monitoring and task definitions are done here. By increasing dynamism in the robot system, downstream and upstream load decreases significantly thus easing the computation at client side. In addition to this, a blockchain system inherently provides a decent medium for execution of distributed intelligence. We also indicate that Hyperledger Fabric is a more feasible solution than Ethereum as a blockchain platform by stating advantages and disadvantages.

Remainder of this paper is structured as follows: In Section 2 we give background knowledge about blockchain technology and systems of mobile robots in healthcare with related studies. Research challenges regarding blockchain and integration of blockchain powered mobile robots into current medical systems are discussed in Section 3. Our proposed method with example use cases is explained in Section 4. A projection about blockchain applications of medical robot swarms and future directions are presented in Section 5. Finally, we make a conclusion and state our future work in Section 6.

2. BACKGROUND

2.1. Blockchain Technology

Blockchain can be defined as an immutable list of records (i.e. blocks of data) where each record is mapped from the previous one by cryptographic hash functions. Before the production of the first record, a genesis block containing the metadata and block structure is generated by the blockchain network according to the blockchain platform's algorithm. A blockchain network is generally defined as a distributed system which can be accessed publicly or with a permission that is responsible for running the blockchain platform and execution of constantly running and ready-to-be-triggered decentralized programs (defined as smart contracts). Public blockchain networks mostly work in a trustless medium (i.e. malignant nodes trying to capture the network's control or spam with redundant actions to render it unusable) and thus must achieve consensus regarding this situation.

Most known for financial applications like cryptocurrencies, blockchain technology in fact presents a much broader concept; decentralization of network systems, data persistence and immutability in distributed databases, consensus in a trustless distributed system. Introduced in 2008 with the declaration of Bitcoin [1], blockchain technology is a hot research area promising decentralized solutions for various problems.

Successful management of collaboration of mobile robots in healthcare is essential for safe and secure medical treatments to patients. Server computers dedicated for controlling mobile robots are used in current applications. As an alternative to this approach, decentralization with blockchain technology eliminates the necessity of deploying and maintaining a server. In this paper we deal with the benefits of applying blockchain as a decentralized solution for mobile robots in healthcare.

2.2. Mobile Robots in Healthcare

Current studies on mobile robots centric applications include several research domains such as indoor logistics [2], search and rescue [3,4], molecular robotics [5,6] and health [7-9]. Among these research domains, healthcare related studies gained much attention and popularity since the beginning of ongoing COVID-19 pandemic. Common applications of mobile robots in healthcare include medical examination, patient surveillance [7,8] and medical transportation [9] but these applications mostly deal with single robots. Multi robot systems can provide an efficient way to perform healthcare services fluently.

In the centralized management of multiple mobile robots, a server computer's computational workload is going to increase when new service robots are connected to the server. However, if a decentralized approach is preferred then the workload created by the new robots will be distributed among the decentralized network instead of taxing a single point in the multiple mobile robot system.

Specific adaptations of blockchain systems to multi robot systems for multi robot collaboration in fighting COVID-19 pandemic are stated in [10] and [11], but an application for general healthcare services hasn't been proposed yet. Our paper focuses on using a permissioned blockchain platform as a decentralized management for mobile service robots in healthcare.

2.3. Hyperledger Fabric as a Decentralized Solution

Several applications of mobile robots with blockchain for managing both a single robot and multi robot systems including swarm robots [12] have been proposed, with the majority of studies conducted in the literature dated in 2020 and 2021. They can be classified as consensus proposals [13] [14], path planning [15], cooperative working [16], information sharing [17], collective decision making [18], robot partitioning [19], task allocation [20-21]. Ethereum is the preferred blockchain network in most these applications with the mention of tradeoffs in smart contract development.

One of the projects within the Hyperledger consortium, Hyperledger Fabric was proposed as a private and permissioned blockchain platform for enterprise level decentralized application development [22]. There are applications proposed for secure and efficient record tracking in the pharmaceutical industry [23], healthcare [24], energy trading [25] and food industry [26] where Fabric is used as a blockchain platform.

Hyperledger Fabric provides a modular system for developers, that means ledger data can be stored in multiple formats and other deterministic consensus algorithms can be set to be used by peers. Channels are mandatorily created for members connected to a Fabric network to join for initiating a ledger to store transactions between members of it.

Blockchain technology and decentralization is the underlying logic of the Bitcoin network operating worldwide, but Bitcoin network uses smart contracts for simple transactions of cryptocurrency, which can't be programmed as demanded, and thus can't be adjusted to provide a decentralized solution for robots in healthcare. Ethereum allows smart contract customization and provides tools to set up a test network [27]. With these characteristics Ethereum is the preferred blockchain platform by most of the developers working in decentralized applications. However, transaction cost in ETH cryptocurrency stands as a tradeoff for decentralized applications running in the main Ethereum network, so developers need to consider code optimizations in their smart contracts [28]. There is another catch that public blockchain networks like the main Ethereum network are not suitable for operating systems of multiple mobile robots, since the consensus algorithm used may demand higher computational power and energy than the robot's limited hardware and power units. Setting a private Ethereum network can solve this issue but Ethereum wallets of the robots connected to the blockchain network must be checked to deal with running out of cryptocurrency and failing to operate. Hence a blockchain platform which can be modified with respect to requirements of a mobile robot system would provide an efficient decentralized solution.

Considering these advantages, Fabric provides a more feasible and customizable blockchain platform than Ethereum. Our proposed method is based on a blockchain network of mobile robots operating Fabric.

3. RESEARCH CHALLENGES

Despite recent advancements in blockchain technology and medical service robots there are still problems regarding the use case scenarios, seamless execution of blockchain systems and integration into current medical systems.

- Robots must be accessorized with specific equipment for treatment.
- In case of an emergency situation, a contingency plan for the robots must be well defined and put into action when need arises. (failure of a specific robot, power outage etc.)
- Each task and its sub tasks for the robots must be defined with utmost clarity and consist of solid and concrete steps.
- Entire system must be secure from external tampering and Byzantine robot behaviours.
- Sanitization of the robots must be included as a post mission step for next tasks.

After stating the challenges at the use case level, we discuss the challenges stemming from the technical aspects of blockchain and robots' integration to current medical infrastructure.

3.1. Challenges in Blockchain

Blockchain systems have been improved to satisfy the varying demands of miscellaneous decentralized applications, but much work is needed to make a full substitution of current centralized systems. First of all, most blockchain platforms provide APIs that are too generic to develop complicated (in terms of complex data types) decentralized applications. For instance, an instantiated object of a custom defined class can't be directly stored to the ledger database in Fabric, it needs to be marshalled to JSON string before storing. Considering robotics (swarm or multi robot systems), there doesn't exist a blockchain platform dedicated to it or tool to integrate blockchain with a middleware (ROS etc.) for various robots. Hence developers must make software designs for decentralized applications simplified enough to match the API of the blockchain platform they choose. Blockchain platforms are also designed for storing data with smaller size; if larger files are needed to be stored in a blockchain, an external distributed storage protocol like IPFS [29] must be integrated with the blockchain system used. Another challenge current blockchain platforms are expected to meet in the future is the ongoing development of quantum computing. SHA-256 is the preferred cryptographic hashing method in popular blockchain platforms and is expected to be rendered unusable with cryptanalytic attacks by quantum computers. Therefore, development of quantum resistant cryptographic hash algorithms and their implementations in blockchain platforms stands as research topics for researchers.

3.2. Challenges in Mobile Robots Integration to Medical Systems

Contemporary medical automation systems used in medical centers are centralized systems which can be developed by different companies but need to include protocols and services for reporting to government officials when requested and interoperability with auxiliary facilities (pharmacy, medical logistics etc.). For instance, the Ministry of Health is responsible for defining the standards of medical treatments and diagnosis algorithms, monitoring the clinical activity in all medical centers including private clinics and public hospitals in Turkey. Integration of the blockchain network as a decentralized system into centralized systems used in medical infrastructure must meet the standards and procedural requirements.

Another challenge arises when data modelling of the chosen blockchain platform needs to comply with data model used in the content management system (CMS) of the medical center. While synchronizing data between robots with blockchain records in their storage and CMS, conversion of complex data types must be committed elaborately to avoid accidental data loss or overwrites. Relational database management systems (RDBMS) are preferred in most CMS applications and thus type standards must be set out to match or at least to be easily convertible while exchanging data between RDBMS of the medical center and ledger databases of the robots connected to the blockchain platform.

4. PROPOSED METHOD

Due to its decentralized nature, blockchain systems can provide a decent solution for management of multi robot systems. The advantages of autonomous multi-robot collaboration include easily scalable, better fault tolerance against any possible failure of robots, better adaptability to the environment, less memory, less processing power, better options for exception controlling, and less capability for individual robots. A use case scenario for a multi robot system using Fabric is demonstrated in Figure 1. As it is seen in the figure there are five mobile service robots operating in a blockchain powered multi robot system at a healthcare facility. Robot 1 is appointed to deliver medicine to Patient A and proceeds to the medical supplies room to pick it up. As a post task objective, Robot 2 goes to the nearest sanitization unit to be sanitized for the next task. Robot 3 is appointed to deliver food to Patient C and is serving food to patient. After gathering samples from Patient B, Robot 4 brings them to doctors in the examination room. Robot 5 is appointed to a physical examination of Patient D and is carrying it out. While the robots are at work, they all have the same copies of ledgers containing robot and task information (namely RobotLedger and TaskLedger) at the same time and update the ledgers by committing new information records after achieving consensus. Medical personnel can connect to the Fabric network via an application at client side to create tasks and send a request for assignment of them to available robots. Since all of the peers in a Fabric network share the same ledger database, each robot in the system has the position and task information of other robots and updates its position and status regularly.

Since locations of facility areas (cafeteria, sanitization etc.) and patient beds remain static during robot operations; their position information can be stored in both client side and robots, there is no need to store this data in blockchain.

4.1. Definition of a Robot for Robot Ledger

A robot is recorded on blockchain with following attributes as shown in Table 1.

Table 1. Attributes of a robot definition in blockchain

Attribute Type	Attribute Name
Integer	RobotID
Integer	TaskCategoryID
Integer	TaskCounter
Integer	AssignedTaskID
Floating-point	Position_x
Floating-point	Position_y
String	Status

RobotID: Each robot has a unique identifier as a positive integer value.
TaskCategoryID**:** Stores which type of task the robot is assigned to and currently carries out. Its value is limited to the values (as shown in Table 2) enumerated for task categories.

Table 2. Enumeration of task categories

TaskCategoryID	Task Category
1	Medical supply delivery
2	Food delivery
3	Medical Textile Delivery
4	Physical examination
5	Sanitization
6	Charging

TaskCounter: Stores the total number of tasks successfully completed by the robot since initial deployment. This variable can be reset to zero at the end of the day after archiving its value to track the robot's performance.

AssignedTaskID: Stores the unique identifier of the task the robot is carrying on. This variable is used for querying task details on the task ledger on demand. If unassigned its value is 0 or null.

Position_x: Stores the horizontal (x coordinate) position of the robot.

Position_y: Stores the vertical (y coordinate) position of the robot.

Status: Shows the current status of the robot. If a task is assigned to the robot or the robot is getting sanitized, it is set to BUSY. If the robot is being charged at a charging station, it is set to CHARGING. Otherwise it is set to IDLE.

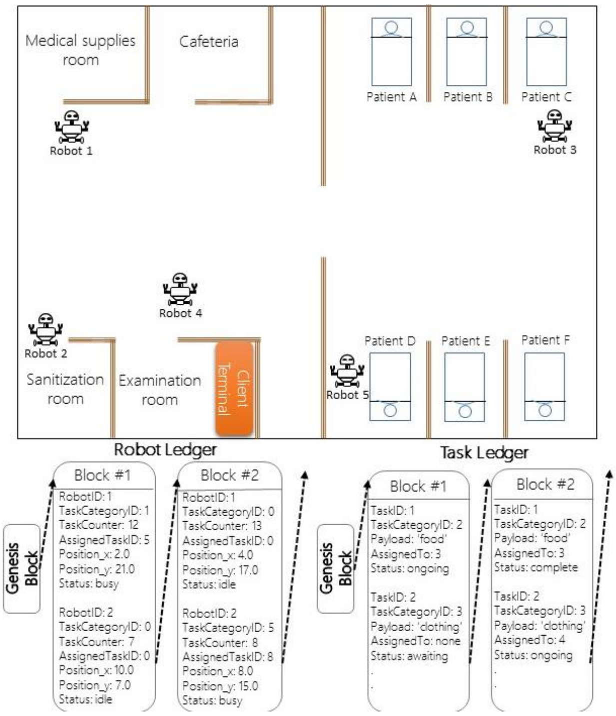

Figure 1. An example of a blockchain powered multi robot system in action. Robot 1 moves to gather medicine for the task of delivering it to Patient A. After completing a task, Robot 2 arrives at the nearest sanitization unit to be sanitized for the next task. Robot 3 delivers food to Patient C. Robot 4 brings the samples gathered from Patient B to doctors in the examination room while Robot 5 is conducting a routine physical examination of Patient D. Each robot has the same copies of RobotLedger and TaskLedger records at any time. Medical personnel in the examination room can create tasks and send a request for assignment of the task at the client terminal connected to the Fabric network.

After setting up and deploying the Fabric network with robots as peers, each robot in the system is expected to carry out tasks, which can be generated on demand by medical staff.

4.2. Definition of a Task for Task Ledger

A task is recorded on blockchain with following attributes as shown in Table 3.

Table 3. Attributes of a task definition in blockchain

Attribute Type	Attribute Name
Integer	TaskID
Integer	TaskCategoryID
String	Payload
Integer	AssignedTo
String	Status

TaskID: Each task has a unique identifier as a positive integer value.
TaskCategoryID: Stores which type of task the robot is assigned to and currently carries out. Its value is limited to the values enumerated for task categories.

Payload: Used to define what robot is carrying to target. Can be food, clothing, medicine etc. If the robot is not carrying anything it is set to NONE.

AssignedTo: Used to determine which robot the task is assigned to. If unassigned its value is 0 or null.

Status: Shows the current status of the task, if task is not assigned to a robot, it is set to AWAITING. If the task is assigned, it is set to ONGOING. If it is completed, then it is set to COMPLETE.

Tasks (shown in Figure 2) to be carried out by the robots can be categorized as follows:

1. *Medical supply delivery*: Robot goes to the initial position (pharmacy, nurse room etc.) of the requested item (medicine, scrubs etc.) and picks it up. Then the robot is ordered to move to the patient's location as the target position and delivers the requested item(s). Finally, the robot is ordered with the sanitization task before accepting new tasks.

2. *Food or clothing delivery*: Robot goes to the initial position (cafeteria, laundry room) of the requested item (food, scrubs etc.) and picks it up. Then the robot is ordered to move to the patient's location as the target position and delivers the requested item(s). If anything needs to be carried away from the patient (empty dishes, dirty clothing etc.), the robot picks it up and moves to where it needs to be dropped (cafeteria, laundry room etc.). Finally, the robot is ordered with the sanitization task before accepting new tasks.

3. *Physical examination*: Being equipped with required tools for close examination of the patient (stethoscope etc.), robots are ordered to go to the patient's location (bed, examination chair or couch etc.) as the target position. After the examination the robot moves to medical personnel to bring the samples gathered from the patient if necessary. Finally, the robot is ordered with the sanitization task before accepting new tasks.

4. *Sanitization*: After the completion of any other tasks, the robot is ordered to go to the nearest sanitization unit to be ready for the next tasks. The robot's status needs to be busy in order not to be accidentally ordered for a new task. After the sanitization process is completed, the robot gets ready to await new tasks.

5. *Charging*: When a robot's battery life is below a threshold value (e.g. 10% of total) the robot might not be able to complete a task successfully, so after checking its battery life, a robot is appointed to be charged at the nearest possible charging station automatically. While being charged, a robot's status is assigned to `CHARGING` and it won't be considered among available robots.

Figure 2. Task definitions for robots. Steps to be performed are explained for each task category.

4.3. Task Assignment Procedure

Using the client application running on the terminal depicted in Figure 1, authorized medical personnel can connect to the Fabric blockchain network of the robots and oversee its operation. Functions that a user can perform in the client application are as follows:

- *Create a Task*: The user can define tasks for the robots on demand. Details of a task are determined by the user according to the data model described in Section 4.2. After entering task information, the task is submitted to the blockchain network to be committed to the task ledger.
- *Send Task Assignment Request*: The user can select a task stored in the task ledger and request it to be carried out. Since the blockchain network operates decentralized, assignment of a task to a robot is first submitted to the network as a request, then assignment is done after execution of the task assignment algorithm. First, available robots are determined by checking the status of each robot in the network (step 1). Then for each available robot, its current position is gathered and Euclidean distance to the target position of the task is calculated. (step 5) Finally, the robot with the nearest Euclidean distance is selected for assigning the task. (step 6) If more than one robot has the same distance, the robot which demanded the task earlier by completing execution of smart contract faster is chosen (step 8). Pseudocode of this task assignment algorithm is as follows.
-

ALGORITHM 1: Task assignment to robots

```
1 availableRobots = CheckRobotStatus();
2 nearestRobotID=0;
3 for each availableRobot in availableRobots
4  initialPosition = getPosition();
5  distance = EuclideanDistance(initialPosition,
targetPosition);
6  if (distance < nearestDistance)
7       then nearestRobotID = RobotID
8  if (distance = nearestDistance)
9       then if(TimeUnit.nearestRobotID >
TimeUnit.RobotID)
10          then nearestRobotID = RobotID
```

- *Network Status*: The user can see the current status of the robots (i.e. performing a task, standing idle), number of blocks committed and the block data in both robot and task ledgers.

For the blockchain network shown in Figure 3, a medical personnel logins to the client application and requests a swab sample of Patient F. For this, the medical personnel creates a task in the physical examination category and determines the payload and sets Patient F as the target position. The task is stored in the task ledger with the status set to `AWAITING` (demonstrated in Figure 3(c)). Then the medical personnel sends a task assignment request to the blockchain network. For the assignment of the task, status of each robot is checked and Robot 1, Robot 2, Robot 3 and Robot 4 (which already have swab kits) are found as the available robots at that moment. While running the task assignment algorithm described above, Euclidean distances of Robot 1, Robot 2, Robot 3 and Robot 4 are computed as 8, 13, 8 and 17 respectively. Having the least Euclidean distance to Patient F, Robot 1 and Robot 3 are both candidates for appointment but since Robot 1 runs the smart contract faster and broadcasts its availability earlier, it is assigned with the task the medical personnel just created and the confirmation message of successful task assignment is returned to the user.

It should be noted that the client application is only responsible for providing the interface for task generation and request of task assignment to an available robot. The task assignment algorithm described above is implemented as a method (i.e. `AssignTaskToRobot`) in Java source code of smart contract defined for robots. When requesting a task assignment, the client application actually makes a remote procedure call (with `SubmitTransaction` method in Fabric Java API) for invoking the method in the smart contract running on the Fabric network (as shown in Figure 4). Since the position of the robots can be accessed from the robot ledger, Euclidean distances are computed at each robot and the nearest robot is appointed in a distributed way. Since the outcome of this invocation might require a ledger update (a successful assignment needs to be recorded), consensus of the peers (robots in this case) must be achieved first. For this, robots execute the smart contract deployed on the network and send their appointment requests to other robots over the network. Since each robot in the network is involved in achieving consensus, data transition is distributed among the network, instead of overloading the bandwidth of a single node (i.e. a server node). Hence utilizing decentralization with distribution of the whole task assignment process over the network mitigates the burden of a server in a centralized system.

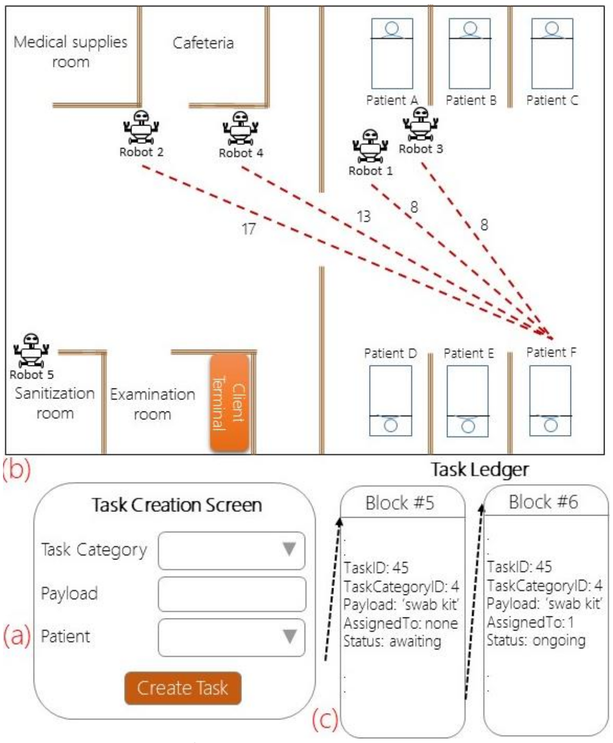

Figure 3. Creation of a task and its assignment to Robot 1. Medical personnel creates the task by entering task details in the client application connected to Fabric network which is demonstrated in (a). Execution of the task assignment yields that Robot 1, Robot 2, Robot 3 and Robot 4 are the available robots in the network at that moment with Euclidean distances shown in (b) to Patient F. Robot 1 and Robot 3 have the least Euclidean distance to Patient F but since Robot 1 acted first, it is appointed to the task and task information is stored in the task ledger as shown in (c). It can be seen that the `AssignedTo` attribute now has the robot ID value of Robot 1 and the `Status` attribute is set to `ONGOING` in the sixth block of the task ledger.

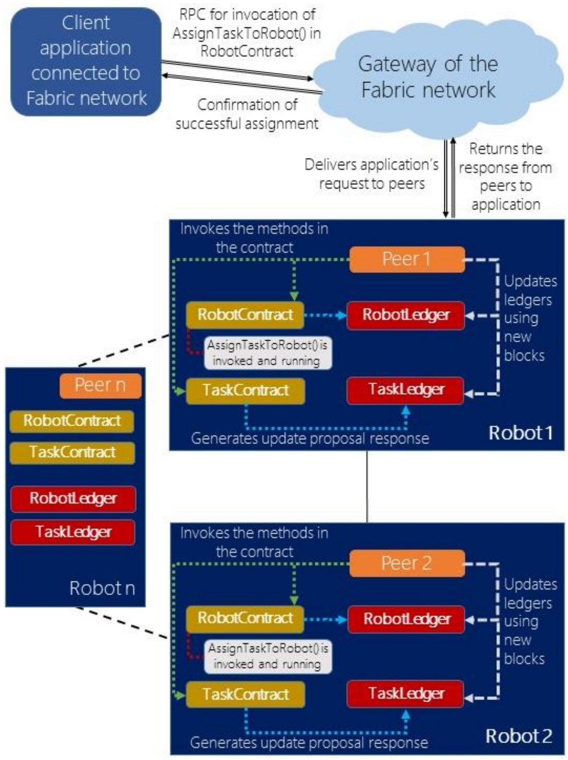

Figure 4. Simplified overview of robots as peers in the Fabric network. Each peer (i.e. robot) runs the smart contracts (RobotContract and TaskContract) deployed on the network. Along with other method definitions for CRUD transactions, RobotContract contains `AssignTaskToRobot` method which is the actual implementation of the task assignment algorithm described above. Users can request the assignment of the task they created by invoking AssignTaskToRobot remotely via client application. Invocation requests from client app and their responses are transmitted to the peers via the gateway of the network. Interactions depicted inside a robot are the same for 2nd, 3rd … nth robots.

5. FUTURE TRENDS

Automatisation and robotics applications in healthcare services provides not only protection of patients against contagion of diseases but also decreases the cost and manpower requirements in medical centers. Service robots being developed come with more powerful hardware thus they can run their sensor suite and still operate in a blockchain network. Blockchain systems overcome trust problems in distributed networks of mobile robots. Since Hyperledger project aims to provide enterprise level decentralized solutions besides cryptocurrencies, a project dedicated for management of systems of mobile robots with its integration into popular robotics middlewares like Robot Operating System could be launched as a specialized form of Fabric for robotics. In addition to the application framework we propose in here, more detailed and complicated task definitions and robot integrations with experiments on simulations and real robots are planned for future work.

6. CONCLUSION

Using autonomous systems of mobile robots in healthcare is a beneficial approach thanks to not only human resource management but also decreasing the contagious risks of human labour in medical centers. Blockchain systems provide distributed data storage while preserving integrity and immutability in the execution and management of multi robot systems. Also communication between robots such as sharing position, orientation and sensor data is simplified through blockchain infrastructure. However, public blockchain networks like Ethereum aren't feasible for operating multi robot systems since limited hardware capabilities of the robots can't compete with the powerful cluster systems while running the Proof of Work consensus algorithm. Setting a private Ethereum network can be suggested but robots need to be checked for not to be drained out of cryptocurrency. We introduce the idea of using Hyperledger Fabric as a permissioned blockchain platform to eliminate these disadvantages and state use case scenarios where task categories with definitions for service robots are explained. Being in nascent stages, blockchain integration into multi robot systems has the potential to be a golden standard in management and operation of distributed intelligence for mobile robots. As the service robots in healthcare continue to improve, various applications of blockchain in medical services are expected to emerge and become widespread.

REFERENCES

[1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system. Decentralized Business Review. 2008;21260.

[2] Okumuş F, Dönmez E, Kocamaz AF. A cloudware architecture for collaboration of multiple agvs in indoor logistics: Case study in fabric manufacturing enterprises. Electronics. 2020;9(12): 2023.

[3] Dadgar M, Couceiro MS, Hamzeh A. RbRDPSO: Repulsion-Based RDPSO for Robotic Target Searching. Iranian Journal of Science and Technology - Transactions of Electrical Engineering. 2020;44(1): 551–563.

[4] Trotta A, Montecchiari L, Felice MD, Bononi L. A GPS-Free Flocking Model for Aerial Mesh Deployments in Disaster-Recovery Scenarios. IEEE Access. 2020;8: 91558–91573.

[5] Kabir AMR, Inoue D, Kakugo A. Molecular swarm robots: recent progress and future challenges. Science and Technology of Advanced Materials. 2020;21(1): 323–332.

[6] Keya JJ, Kabir AMR, Inoue D, Sada K, Hess H, Kuzuya A, Kakugo A. Control of swarming of molecular robots. Scientific Reports. 2018;8(1): 1–10.

[7] Holland J, Kingston L, McCarthy C, Armstrong E, O'Dwyer P, Merz F, et al. Service Robots in the Healthcare Sector. Robotics. 2021;10(1): 47.

[8] Farkh R, Marouani H, Al Jaloud K, Alhuwaimel S, Quasim MT, Fouad Y. Intelligent autonomous-robot control for medical applications. Computers, Materials and Continua. 2021;68(2): 2189-2203.

[9] Fang B, Mei G, Yuan X, Wang L, Wang Z, Wang J. Visual SLAM for robot navigation in healthcare facility. Pattern Recognition. 2021;113: 107822.

[10] Alsamhi SH, Lee B. Blockchain-Empowered Multi-Robot Collaboration to Fight COVID-19 and Future Pandemics. IEEE Access. 2021;9: 44173–44197.

[11] Alsamhi SH, Lee B, Guizani M, Kumar N, Qiao Y, Liu X. Blockchain for decentralized multi-drone to combat COVID-19 and future pandemics: Framework and proposed solutions. Transactions on Emerging Telecommunications Technologies. 2021;32: e4255.

[12] Strobel V, Ferrer EC, Dorigo M. Blockchain technology secures robot swarms: a comparison of consensus protocols and their resilience to byzantine robots. Frontiers in Robotics and AI. 2020;7: 54.

[13] Liu J, Xie M, Chen S, Ma C, Gong Q. An improved DPoS consensus mechanism in blockchain based on PLTS for the smart autonomous multi-robot system. Information Sciences. 2021;575: 528-541.

[14] Singh PK, Singh R, Nandi SK, Ghafoor KZ, Rawat DB, Nandi S. An efficient blockchain-based approach for cooperative decision making in swarm robotics. Internet Technology Letters. 2020;3(1): e140.

[15] Mokhtar A, Murphy N, Bruton J. Blockchain-based multi-robot path planning. 5th World Forum on Internet of Things, WF-IoT 2019. Limerick: IEEE; 2019. p. 584-589.

[16] Karthik S, Chandhar NP, Akil M, Chander S, Amogh J, Aditya R. Bee-Bots: A Blockchain Based Decentralised Swarm Robotic System. 6th International Conference on Control, Automation and Robotics, ICCAR 2020. Singapore: IEEE; 2020. p. 145-150.

[17] Nishida Y, Kaneko K, Sharma S, Sakurai K. Suppressing chain size of blockchain-based information sharing for swarm robotic systems. 6th International Symposium on Computing and Networking Workshops, CANDARW 2018. Hida Takayama: IEEE; 2018. p. 524-528.

[18] Nguyen TT, Hatua A, Sung AH. Blockchain approach to solve collective decision making problems for swarm robotics. International Congress on Blockchain and Applications, BLOCKCHAIN'22 2019. Ávila: Springer; 2019. p. 118-125.

[19] Tran JA, Ramachandran GS, Shah PM, Danilov CB, Santiago RA, Krishnamachari B. SwarmDAG: A partition tolerant distributed ledger protocol for swarm robotics. Ledger. 2019;4(1): 25-31.

[20] Basegio TL, Michelin RA, Zorzo AF, Bordini RH. A decentralised approach to task allocation using blockchain. International Workshop on Engineering Multi-Agent Systems, EMAS 2017. Sao Paulo: Springer; 2019. p. 75-91.

[21] Grey J, Godage I, Seneviratne O. Swarm Contracts: Smart Contracts in Robotic Swarms with Varying Agent Behavior. 2020 IEEE International Conference on Blockchain (Blockchain). Rhodes Island: IEEE; 2020 p. 265-272.

[22] Androulaki E, Barger A, Bortnikov V, Cachin C, Christidis K, De Caro A, et al. Hyperledger Fabric: A distributed operating system for permissioned blockchains. Proceedings of the Thirteenth EuroSys Conference, EuroSys'18 2018. Porto: ACM; 2018. p. 1-15.

[23] Uddin M. Blockchain Medledger: Hyperledger fabric enabled drug traceability system for counterfeit drugs in pharmaceutical industry. International Journal of Pharmaceutics. 2021;597: 120235.

[24] Uddin M, Memon MS, Memon I, Ali I, Memon J, Abdelhaq M, et al. Hyperledger fabric blockchain: Secure and efficient solution for electronic health records. Computers, Materials and Continua. 2021;68(2): 2377–2397.

[25] Lohachab A, Garg S, Kang BH, Amin M B. Performance evaluation of Hyperledger Fabric-enabled framework for pervasive peer-to-peer energy trading in smart Cyber–Physical Systems. Future Generation Computer Systems. 2021;118: 392–416.

[26] Surjandari I, Yusuf H, Laoh E, Maulida R. Designing a Permissioned Blockchain Network for the Halal Industry using Hyperledger Fabric with multiple channels and the raft consensus mechanism. Journal of Big Data. 2021;8(1): 10.

[27] A Next-Generation Smart Contract and Decentralized Application Platform. Ethereum

Project White Paper. Technical Report. V Buterin; 2014 [cited 2021 Oct 18]. Available from: https://ethereum.org/en/whitepaper/

[28] Zarir AA, Oliva GA, Jiang ZM, Hassan AE. Developing Cost-Effective Blockchain-Powered Applications: A Case Study of the Gas Usage of Smart Contract Transactions in the Ethereum Blockchain Platform. ACM Transactions on Software Engineering and Methodology (TOSEM). 2021;30(3): 1-38.

[29] InterPlanetary File System [Internet]. 2021 [cited 2021 Oct 18]. Available from: https://github.com/ipfs/ipfs