LEDGER
ledgerjournal.org

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172

PROCEEDINGS ARTICLE

Blockchain Design for an Embedded System

Sara Falcone,[*†] John Zhang,[‡] Agnes Cameron,[§] Amira Abdel-Rahman[††]

Abstract. This paper proposes a blockchain-based mapping protocol for distributed robotic systems running on embedded hardware. This protocol was developed for a robotic system designed to locomote on lattice structures for space applications. A consensus mechanism, Proof of Validity, is introduced to allow the effort of mining blocks to correlate with the desired tasks the robotic system was designed for. These robots communicate using peer-to-peer LoRa radio. Options, trade-offs and considerations for implementing blockchain technology on an embedded system with wireless radio communication are explored and discussed.

1. Introduction

Recent advances in cryptocurrency have fueled a more general interest in the blockchain as a form of decentralized and Byzantine fault-tolerant consensus mechanism. Concurrently, the field of swarm robotics is growing rapidly, with applications ranging from farming to search and rescue, to space exploration.[1, 2] In both cases, these developments mark a shift from reliance on centralized control mechanisms to trust in distributed and fault tolerant networks.

1.1 Distributed Consensus Mechanisms—Satoshi Nakamoto's paper Bitcoin: a Peer-to-Peer electronic cash system created the first practically-implemented blockchain.[3] The Blockchain forms a distributed ledger: each node in the Bitcoin network maintains a record of every transaction taken place between peers. Consensus is achieved through a trust-less and distributed protocol known as Proof-of-Work (PoW), not via a central arbiter.

The PoW consensus mechanism is a pseudorandom process by which each node in the network competes to add new transaction information to the Blockchain—'mining a block.' Since the release of Bitcoin, there have been several other blockchain-based systems proposed and implemented, most notably Ethereum,[4] which introduces the idea of the 'smart contract,' an agreement to perform some action dependent on events either internal or external to the blockchain. Although mainstream Ethereum also employs PoW as its consensus mechanism, a proposed change to the Ethereum blockchain would implement the Proof-of-Stake protocol (PoStake) in a bid to conserve energy and bring down transaction times. PoStake uses the investment each node has in the network to 'weight' the vote they can contribute, essentially protecting against bad actors using expense, rather than PoW's computational power.

[*] 3FAmdrjGe4MCZCh7LGMQjPMXfZPa2SAoQ9

[†] S. Falcone (sfalcone@mit.edu) is a researcher in the Center for Bits and Atoms at the MIT Media Lab.

[‡] J. Zhang (johnz@mit.edu) is a student at the Massachusetts Institute of Technology. S. Falcone and J. Zhang contributed equally to the present work.

[§] A. F. Cameron (agnescam@mit.edu) is a researcher at the Viral Communications Group at the MIT Media Lab.

[††] A. Abdel-Rahman (amira.abdel-rahman@cba.mit.edu) is a researcher in the Center for Bits and Atoms at the MIT Media Lab.

Other proposed consensus mechanisms are many and varied. Popular in private blockchains is Proof-of-Authority (PoA), in which only 'trusted' entities may maintain the blockchain, with the ability to vote potentially malicious nodes out. While this does not maintain the degree of decentralization achieved by full PoW, it can be useful in cases where trust should be spread amongst a network of entities.

The FOAM whitepaper introduces the concept of a Proof-of-Space (PoSpace) which uses a network of radio beacons to create a secure localization protocol.[5] This addresses a key issue in blockchain-based systems: taking in 'real-world' data, reliably, and appending it to a blockchain.

1.2 Decentralized Robot Systems—There are several examples of decentralized robotic systems that incorporate peer-to-peer communication between the agents using limited local communication, such as line of sight schemes,[6,7] as well as simulations where the communication is virtually restricted to nearest neighbors.[8,9] Many examples, and simulations of distributed robotic systems also exist where communication is received by all nodes and they form a mesh network for routing information,[10] while other mesh applications are focused on covering distances too large for a single point of communication, such as wireless sensor networks (WSN),[11] forming point-to-point and multi-hop networks.

A concern in decentralized robot systems is robustness to malicious or faulty actors, here referred to as 'Byzantine' agents. Byzantine fault tolerance requires that, even with malicious or faulty actors making up one third of the total agents, consensus may still be reached between valid agents, without requiring a centralized mediator.[12] A key advantage of using blockchains in a swarm robotics context is the robustness of the data structure to faulty data and byzantine agents.[13] However, few implementations of blockchain technologies, if any, have been proposed and implemented on robot-specific hardware using embedded systems.

Organizations such as NASA are interested in leveraging swarm technology to increase robustness and autonomy of their systems.[14] If hardware fails on a single agent in a swarm system the other agents will fill in, replacing the role of the failed agent, continuing the mission. As any agent may fail randomly it may be advantageous for each agent to have a record of what tasks the others in the system have performed, which is an advantage of a distributed data structure.

In addition, storing all data on every agent makes the retrieval of the data easier, particularly of interest in space applications. A convoy can be sent to a single agent, acquiring it and returning it to earth while leaving the rest of the system running on site. Additional agents can be added to the system after the mission begins, and if the robots control scheme is also decentralized the swarm will adopt them and change its behavior to incorporate the additional resources. Alternatively, a "sniffer agent" could be employed purely to retrieve an up-to-date blockchain.

We investigate the use of blockchain technology for a swarm of robots specifically designed to map lattice space structures. Blockchains are not the only way to decentralize, rather this paper aims to explore what can be done with blockchain technology in an embedded system and discuss the necessary trade-offs.

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172

2. Implementation of Blockchain BILL-E

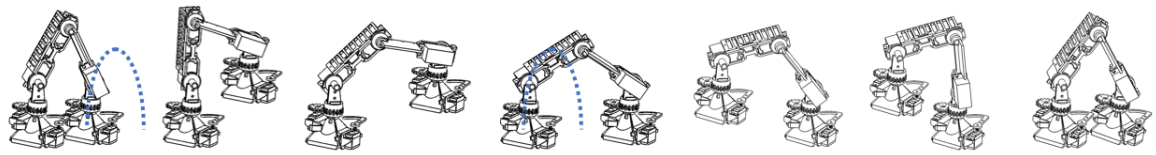

Fig. 1. To take a step forward the robot moves its front foot two voxel forward, then follows with its back foot by moving it one voxel forward. The dashed arcs show the path of the traveling foot.

The robot's body is designed such that it can reach the three voxels immediately in front of it, and step forward by stepping over the central voxel.[15] The robot can also turn 90 degrees in either direction. These constraints vastly reduce the complexity and variability of the robot's motion. We chose to implement a local state machine approach to path planning, though there are many other methods. Figure 2 shows the robot locomoting on a structure and detecting void voxels. As the robots locomote and explore the structure, they expand and fill in a map describing what has been explored.

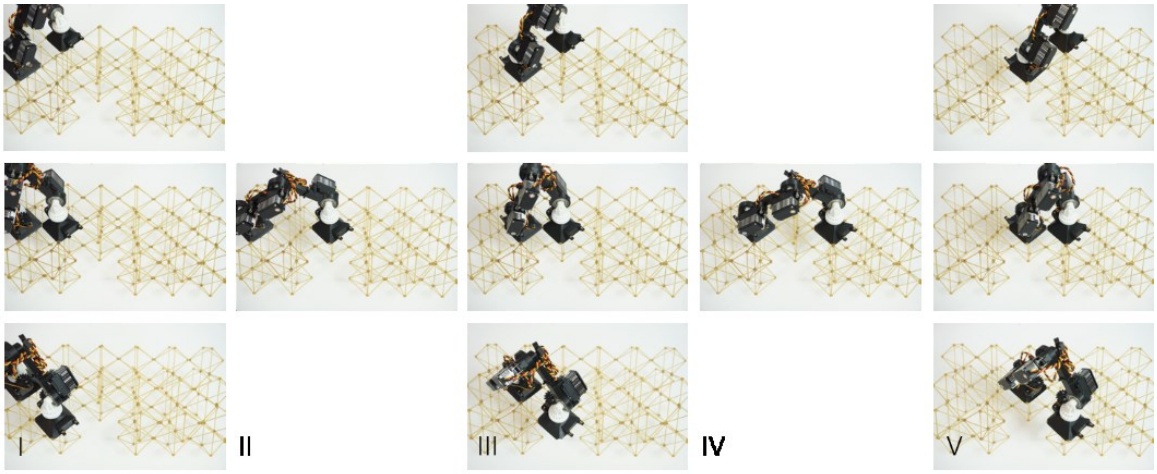

Fig. 2. As the robot explores the lattice structure from I to V it detects if voxels are present or not. In stage I, II and V the robot places its front foot on the 3 voxels immediately in front of itself to determine if they are there or it is void. Notice the robot sensing the empty voxel in the bottom of stage III. In stage II and IV the robot senses the voxel two spaces in front of itself and steps forward if there is a voxel there.

As each robot explores the structure every time a foot steps onto a voxel, or space where a voxel could be, it creates a transaction containing the elements shown in Table 1. These include the coordinates of the voxel and a Boolean: 1 showing there is a voxel present, or 0 indicating a void, and the ID of the robot that mapped the voxel. All robots are listening and record all heard transactions locally in an array, called the mempool. Each block contains a header as well as the transaction data. The data stored in each is listed in Table 1. A low-level hash function is used to maintain traceability between the added blocks.

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172

Table 1. Contents of each block's header and transaction data.

Header Field	**Data**	**Size**
Previous Hash	SHA1 or djb2	20 bytes
Time stamp	UNIX	32 bits
Transaction data		
Voxel coordinates	X, Y	4 bytes
Voxel state	1 for TRUE, 0 for FALSE	1 bit
Diagnostic results	2 x Battery voltage, Float	64 bits
Robot IDs	2 Integers	4 bytes
Transaction timestamps	2 x UNIX, 32-bit	64 bits

As robots explore the structure, they fill in an array of where they have traveled. By traveling along the edge of the structure the robot defines the boundary of the structure and fills in the map towards the center. Every time a robot adds a transaction to the mempool it checks to see if there already exists a transaction with the same coordinates. When a second robot maps the same voxel a second, redundant transaction will be made with the same coordinates. Redundant transactions by separate agents, who both report clean system diagnostics is considered Proof of Validity, allowing the two transactions to be combined into a mined block, and added to the blockchain. This means that each block contains data for a single voxel. If there is conflicting data between two overlapping transactions both robot IDs who reported the inconsistency are flagged, and the transactions are not mined. Future work is needed to develop efficient and secure methods to dealing with these faulty schemes.

Figure 3 shows simulations of two robots following a naive algorithm for mapping and exploration. Each robot is seeded from the same location, which provides a global coordinate system. This assumes the robots arrived at the structure via the same convoy. Each robot first tries to travel to the right, and records if there is a voxel there or not. If it can turn right it does, otherwise it explores the voxel in front of it. If it is present it steps forward, if not it steps to the left. If there are no unexplored options, the robot picks a random direction to travel repeating these steps until the entire map is filled in.

Blockchain implementations stand apart from other decentralized data structures because they require each agent to keep a record of every transaction made by every agent in the system. Many critics of blockchains target the redundancy of this data storage as an unnecessary inefficiency. However, for our application it is advantageous as all robots can use a complete map of the structure they are exploring to decide where to move and explore. In addition, this redundancy allows the retrieval of a single agent, or sniffer, to gather the history of the entire swarm and thus a complete map.

2.1 Blockchain Implementation: Proof of Validity—We propose an alternative method, inspired by as Proof of Stake, which is meaningful to the robotic system adopting this data structure. We incorporate results from a hardware diagnostic test, as well as multiple measurements, into our Proof of Validity (PoV) method for trusting a transaction.

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172

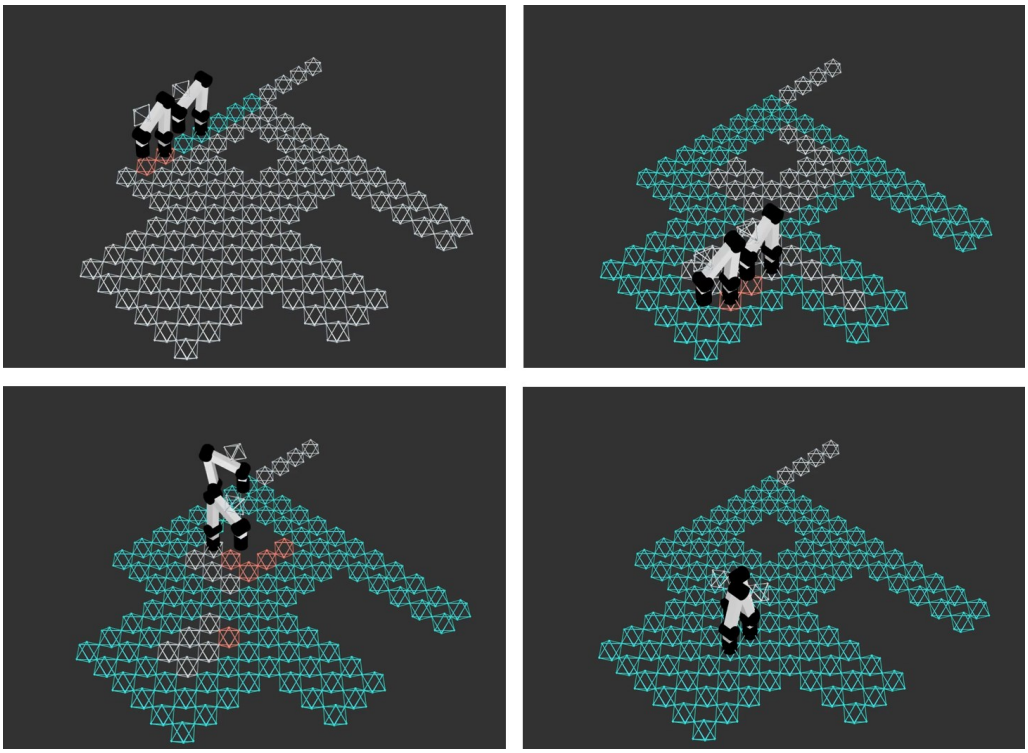

Fig. 3. These snapshots are from a simulation showing two robots explore and map an arbitrary lattice surface. In the simulations white voxels are unexplored, red voxels have been explored by a single robot, and blue voxels are mined.

The hardware diagnostic test can include readings such as temperature, power usage, odometry readings in comparison of robots' location via an external system, etc. Hardware diagnostics such as these are commonly incorporated into products that boast reliability. Before data can be added to the blockchain using PoV, the robot associated with it must pass a system diagnostic with a clean "bill of health" to ensure the efficacy of the sensor readings.
Incorporating this hardware diagnostic into the Proof of Validity flags hardware failures and keeps a constant record of the hardware's performance enabling research and development for future systems.

Sample size is an important consideration for experimentation. For our proof of concept exploration, we were mainly interested in implementing a useful form of blockchain that would work with BILL-E, but the developed protocol can be utilized towards other systems. If the robot were not constringed to the lattice structure one could imagine them traversing the surface of a planet. In that case a map is much more extensive than an array of Boolean values describing if a voxel is present of not. It would be nearly impossible to record redundant topological readings with meaningful resolution given noise and actual changes that can occur in exposed environments, though average readings with a threshold deviation could be considered PoV.

With PoV difficulty is not determined by incrementing a nonce, instead the amount of time robots need to map determines the cadence between block addition. Incorporating statistical metrics on the sensor readings and adjusting the threshold could also be used to control the mining difficulty.

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172

3. Limitations of Embedded Systems for Blockchain Implementation

Usage of the blockchain paradigm on embedded systems for distributed or multi-agent robotics is still uncommon. The primary factor behind the lack of embedded blockchain implementations is the limitations of embedded hardware. These limitations include processor speed, SRAM, storage, and bandwidth. Traditionally, blockchain implementations for cryptocurrencies require significant amounts of dedicated hardware to run. For instance, the current Bitcoin blockchain is over 145 GB (at the time of writing) and grows at a rate of nearly 10 MB per block added every ten minutes.[16] Additionally, the Proof-of-Work algorithm used for Bitcoin, SHA256, is computationally intensive and requires substantial processing capability for a miner to be successful. [17] The peer-to-peer nature of blockchain networks imposes the additional requirement of fast wireless communication between nodes, a field that is only recently penetrating embedded systems in the form of nascent Internet-of-Things technology.[18]

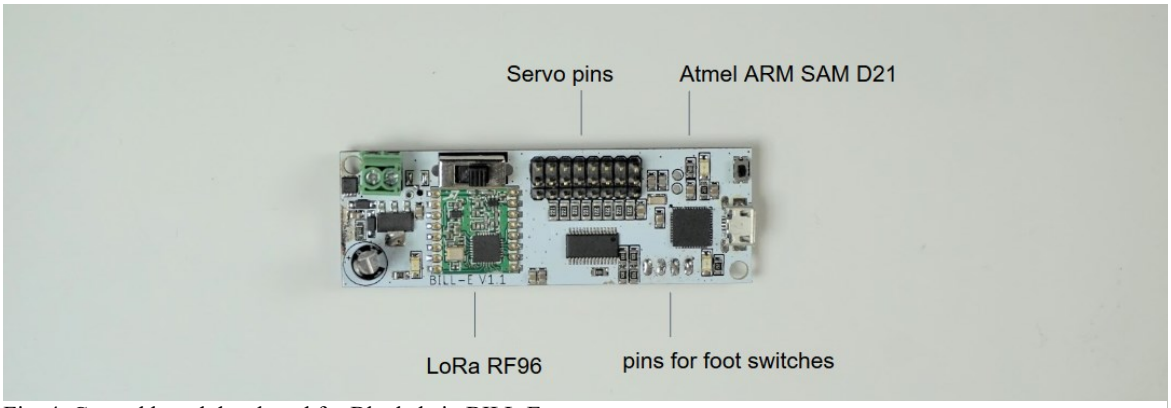

Fig. 4. Control board developed for Blockchain BILL-E

The BILL-E robot controller design, shown in Figure 4, is based off of a Feather M0 featuring a ATSAMD21G18 ARM Cortex M0 processor clocked at 48 MHz. The inherent limitations in embedded systems make transferring data structures from application software non-trivial. For instance, though work-arounds exist, the C programming language does not natively support mutable arrays, which are pivotal for blockchain implementation.

The processor implementation of the BILLE has 256KB of FLASH and 32KB of RAM but no EEPROM. Although the FLASH and RAM storage are sufficient to host a blockchain of the required size for the BILL-E lattice mapping operation, the lack of EEPROM for this system means that the blockchain is lost after power cycle. This is a fundamental limitation of this robotic platform that exposes any blockchain implementation to the risk of permanent erasure.

A further consideration is the implementation of the peer-to-peer network. The BILL-E uses LoRa for wireless communication between nodes. LoRa is advantageous for small distributed robotic platforms because of its low power consumption, low cost, and long range.[19] The tradeoff of this wireless communication standard is the lower bandwidth. Table 2 summarizes the typical maximum bitrate of LoRa and competing wireless communication technologies.

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172

Table 2. Comparison of bit rate between communication standards.

Wireless Communication Standard	Maximum Bitrate
LoRa	50 kbps
Bluetooth	800 kbps
Wifi	600 Mbps

With the M0 processor, the SHA256 hash algorithm, standard to Bitcoin, cannot be computed in a reasonable amount of time. Here, djb2 is used for ease of implementation, though it is not a cryptographic hash function. Given the bandwidth limitations a hashing algorithm that computes a short hash is ideal, allowing more blocks to be transferred within a single packet. For instance, SHA-1 results in a 20 bytes hash, though there is an extreme trade-off in security.

For BILL-E, the maximum packet size and message latency for reliable communication was experimentally found to be 200 bytes and 200 milliseconds. This is for continuous broadcast during which other robotic operations such as movement are necessarily suspended. Given Table 1, which shows each transaction needs 385 bits, we found that we can only communicate five transactions in a single packet. The blockchain would be meaningless if it were restricted to the size of a LoRa packet. To provide a sense of scale, if we constrained the blockchain to fit in a single packet, and the blockchain were the lyrics to "Lucy in the Sky with Diamonds" by the Beatles, it would be truncated at the first stanza:

```
Picture yourself in a boat on a river
With tangerine trees and marmalade skies
Somebody calls you, you answer quite slowly
A girl with kaleidoscope eyes
Cellophane flowers of yellow and green
```

Although a multi-packet communication protocol is possible, it has not yet been implemented in this first demonstration as it is not necessary for standard addition of blocks, where each node keeps up to date with the network. It is however necessary to develop this capacity for adding new robots into a system, or to merge divergent blockchains. This is not a "light node" approach as the hardware stores the entirety of the blockchain, as it has space for several thousand transactions.

With any wireless implementation Equation 1 can be balanced to inform the design of a custom blockchain. The size of the data in the header, H, plus the transaction data, T must balance with the total packet size, P, minus the required meta data to send a message, M, which includes time information and message type, for however many blocks are to be fit into this single packet, N. N will be the length of the blockchain transmitted. With this ratio we calculate the length of a message.

$$\frac{P-M}{N} = H + T \tag{1}$$

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172

4. Discussion

Proof of Validity (PoV) is a means of constructing a distributed, mapping framework. This is an exploratory implementation with very limited hardware, which has greatly influenced the design. This system is still in an experimental phase; however, we have successfully demonstrated the implementation of a blockchain on an embedded system. In addition, we outline a protocol for linking mobile agents with LoRa communication and propose a framework for optimization over limited bandwidth. In further developing the system, it will be necessary to address the sever bandwidth limitation by implementing some degree of multipacket reception or a different communication protocol.

The consensus algorithm should allow the robots to traverse the entirely of the surface without collision, and with as few robot steps as possible explore each voxel twice, such that its map can be incorporated into the blockchain. Scenarios such as the robots diverging and forming independent blockchains should be more carefully evaluated. Traditionally, when two blockchains rejoin the longer is favored for consensus and the shorter blockchain gets truncated to the last common node, orphaning blocks. Given the environment and limited recourses this system is designed for, it is not desirable to discard data. It is also not as likely or advantageous for byzantine agents to corrupt the data and add extended false blocks. However, considering these system trade-offs, we pose the question: at what point does the word 'blockchain' stop being a meaningful descriptor?

This exploration brought up many considerations regarding trade-offs in robotic design for use with blockchain. For instance, the amount of hardware power and time used for robot-specific action versus blockchain processing. Assuming resources such as time, energy and processing power are either dedicated to robot functions, *i.e.* completing the tasks the robot was designed for, or blockchain actions, necessary to validate the robot functions, Equation 2 can be used to characterize the amount of resources that are invested in 'useful work' for cost-benefit analysis of blockchain implementation on specific systems.

$$System\ efficiency\ = \frac{Robot\ Function}{Blockchain\ Actions + \ Robot\ Function} \tag{2}$$

Using multiple processors would allow simultaneous operations for blockchain- and robot-specific tasks, more effectively parallelizing the process. Our hardware also lacked long term memory, which is an obvious necessity for a robust system.

We believe this proposal remains true to the intention of blockchain technologies, keeping a peer-to-peer, distributed data structure which is fault tolerant and does not necessitate a central arbiter to "keep peace." Though this architecture has yet to be fully realized, it is a step towards enabling blockchain technologies in robotic applications.

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172

Acknowledgements

Thank you to L. Cai, J. Seale, M. Feldmeier, H. Robertson and W. Langford for their feedback and suggestions on this project. This work was supported by the Center for Bits and Atoms research consortia; thank you!

Author Contributions

SF and JZ co-developed the framework for this protocol and this paper. AC contributed significant guidance and blockchain specific knowledge. AAR developed the multi-robot simulation.

Notes and References

[1] SAGA. "SAGA – Swarm Robotics for Agricultural Applications." (accessed 9 March 2019) http://echord.eu/saga/.

[2] Bloss, R. "Advanced Swarm Robots Addressing Innovative Tasks Such As Assembly, Search, Rescue, Mapping, Communication, Aerial and Other Original Applications." *Industrial Robot: An International Journal* **41.5** 408–412 (2014) https://dx.doi.org/10.1108/IR-05-2014-0337.

[3] Nakamoto, S. "Bitcoin: A Peer-to-Peer Electronic Cash System." No Publisher (2008) https://bitcoin.org/bitcoin.pdf.

[4] Buterin, V. "A Proof of Stake Design Philosophy." *Medium* (30 December 2016). https://medium.com/@VitalikButerin/a-proof-of-stake-design-philosophy-506585978d51.

[5] King, R. J. "Introduction to Proof of Location: The Case for Alternative Location Systems." *FOAM* (16 October 2018) https://blog.foam.space/introduction-to-proof-of-location-6b4c77928022.

[6] Chen, J., Gauci, M., Li, W., Kolling, A., Gros, R. "Occlusion-Based Cooperative Transport with a Swarm of Miniature Mobile Robots." *IEEE Transactions on Robotics* **31.2** 307–321 (2015) https://doi.org/10.1109/TRO.2015.2400731.

[7] Rubenstein, M., Cornejo, A., Nagpal, R. "Programmable Self-Assembly in a Thousand-Robot Swarm." *Science* **345.6198** 795–799 (2014) https://doi.org/10.1126/science.1254295.

[8] Kim, Y.-J., Choi, H.-H., Lee, J.-R. "A Bioinspired Fair Resource-Allocation Algorithm for TDMA-Based Distributed Sensor Networks for IoT." *International Journal of Distributed Sensor Networks* **12.4**. 7296359 (2016) https://doi.org/10.1155%2F2016%2F7296359.

[9] Tanner, H. G., Jadbabaie, A., Pappas, G. J. "Stable Flocking of Mobile Agents. I. Fixed Topology," in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475),* 2010–2015 (2003) https://doi.org/10.1109/CDC.2003.1272910.

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172

[10] Kim, J. Y., Colaco, T., Kashino, Z., Nejat, G., Benhabib, B. "mROBerTO: A Modular Millirobot for Swarm-Behavior Studies," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, South Korea* 2109–2114 (2016) https://doi.org/10.1109/IROS.2016.7759331.

[11] Rodenas-Herraiz, D., Garcia-Sanchez, A.-J., Garcia-Sanchez, F., Garcia-Haro, J. "Current Trends in Wireless Mesh Sensor Networks: A Review of Competing Approaches." *Sensors* **13.5** 5958–5995 (2013) https://dx.doi.org/10.3390%2Fs130505958.

[12] Lamport, L., Shostak, R., Pease, M. "The Byzantine Generals Problem." *SRI International* (1982) https://people.eecs.berkeley.edu/~luca/cs174/byzantine.pdf.

[13] Strobel, V., Castelló Ferrer, E., Dorigo, M. "Managing Byzantine Robots via Blockchain Technology in a Swarm Robotics Collective Decision Making Scenario," in E. Andre, S. Koenig (Eds.) *AAMAS '18 Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems* 541-549 (2018) https://dl.acm.org/citation.cfm?id=3237464.

[14] For a recent example, see the yearly NASA "Swarmathon" competition: http://nasaswarmathon.com/.

[15] B. Jenett and K. Cheung, "BILL-E: Robotic Platform for Locomotion and Ma-nipulation of Lightweight Space Structures," in *25th AIAA/AHS Adaptive Structures Conference AIAA SciTech Forum, (AIAA 2017-1876)* (2017) https://doi.org/10.2514/6.2017-1876.

[16] As of publication, that number has risen to over 200GB "Download Bitcoin Core." *Bitcoin.org.* https://bitcoin.org/en/download.

[17] Kobie, N. "How Much Energy Does Bitcoin Mining Really Use? It's Complicated." *Wired* (2 December 2017) https://www.wired.co.uk/article/how-much-energy-does-bitcoin-mining-really-use.

[18] No Author. "State of the Market: Internet of Things 2017." *Verizon*. (accessed 9 March 2019) https://www.verizon.com/about/sites/default/files/Verizon-2017-State-of-the-Market-IoT-Report.pdf.

[19] Sinha, R. S., Wei, Y., Hwang, S.-H. "A Survey on LPWA Technology: LoRa and NB-IoT." *ICT Express* **3.1** 14–21 (2017) https://doi.org/10.1016/j.icte.2017.03.004.

ISSN 2379-5980 (online)
DOI 10.5915/LEDGER.2019.172