An Edge-based Distributed Ledger Architecture for Supporting Decentralized Incentives in Mobile Crowdsensing

(Article begins on next page)

27 April 2024

An Edge-based Distributed Ledger Architecture for Supporting Decentralized Incentives in Mobile Crowdsensing

Paolo Bellavista, Marco Cilloni, Giuseppe Di Modica, Rebecca Montanari,
Pasquale Carlo Maiorano Picone, Michele Solimando
University of Bologna, Bologna, Italy
email:{paolo.bellavista, marco.cilloni2, giuseppe.dimodica, rebecca.montanari,
pasquale.maiorano4, michele.solimando2}@unibo.it

Abstract—**Nowadays, the exploitation of distributed ledger technology (DLT) is increasing among different domains and use cases. Not only within the context of cryptocurrencies, DLT could help the cooperation among untrusted parties in a wide variety of application scenarios. In particular, crowdsensing platforms can benefit from DLT because they need to federate systems belonging to different organizations to share end-user profiles, finally free to move within different domains, maintaining their identity. In this paper, we propose an edge-based distributed ledger architecture for supporting decentralised incentives in a specific mobile crowdsensing paltform called ParticipAct. To motivate the choice we describe two different deployments of ParticipAct, one based on a classical client-server architecture and the other one based on an edge-based model, and we highlight their pro and cons. In particular, our more notable findings rely on an approach based on edge computing and highlight how the three-tier solution improves the scalability, the performance, the security and the fault tolerance of the infrastructure responsible for the management of the federation among untrusted crowdsensing platforms.**

Index Terms—**Distributed Ledger, Blockchain, Mobile Crowd Sensing, Multi-access Edge Computing, Rewarding, Gamification**

I. INTRODUCTION

The last few years have been characterized by a prominent rise of distributed ledger technologies (DLT), driven by a sudden surge of interest towards cryptocurrencies, such as Bitcoint and Ethereum. In particular, relying on strong cryptographic protocols, blockchain technologies, one possible DLT, have been boldly promoted as potentially capable to revolutionize the financial and accounting sectors, challenging the traditionally centralized trust models these sectors rely on.

The concept of a decentralised shared registry promoted by blockchain technologies has been further expanded to comprise application domains outside of the financial sphere by a second wave of distributed ledgers, based on the concept of smart contracts. These small pieces of code allow to enshrine into the ledger the behaviour of a contract between single or multiple parts, without relying on a trusted third party authority. This is achieved by using the peer-to-peer network as a global and distributed virtual machine, where the instructions contained in these contracts can be verified through the cooperation of the entirety of the participants. The long-term viability of these technologies is, however, still up to debate and needs several in-the-field-deployments to evaluate the real effectiveness of DLT adoption in non financial application domains.

Along this direction, we decided to investigate the adoption of blockchain technologies to achieve a better federated reward system for our Mobile CrowdSensing (MCS) infrastructure, i.e.

ParticipAct, and to obtain a scalable and decentralised incentive system platform. ParticiAct mainly consists in a set of tools and utilities that allow city administrators to gather extensive insights on how citizens use resources and commute throughout the day, using devices enrolled by volunteers as a data source.

In ParticipAct, as well as in any MCS, it is crucial and of paramount importance to promote and enroll as many people as possible into MCS campaigns, to improve at the same time both the quantity and quality of the gathered data. One successful measure to increase participation numbers is through incentives and reward programs built around the end goal of securing the loyalty and involvement of the user base, boosting participation and active user involvement. These activity can be made more attractive through the use of gamification, a marketing device which consists in the application of elements of game playing to another area of activity, as a way to encourage user engagement and stimulate involvement. When the user reaches an objective or accomplishes something that is meaningful or of any interest to the platform (i.e., filling up a form or answering to questions) he is rewarded with a price or small gift, which can either have a symbolic value or be useful to the end users.

Furthermore, given the open-source nature of a lot of MCS services, the autonomous and spontaneous aggregation of user-data gathering nodes in urban Smart City scenarios is highly desirable. However, we deem that these nodes should not necessarily need to be certified as trusted, given that they might just wish to contribute to the data gathering and validation processes, and participate to the gamification system itself while remaining independent from the overall infrastructure.

In this article, we exploit distributed ledgers to actually try to devise architectures capable to provide a consensus mechanism to store globally the profile of the participants to an MCS platform and the rewards obtained by them by the means of gamification campaigns. A DLT-based platform needs to be capable of efficiently scale from small-sized deployments with only a handful of nodes to large, international data-center infrastructures with hundreds of thousands of members. Given the scalability requirement of such a platform, in our work, we used the support of a third layer provided at the edge of the network that would perform all the ledger-related operations. One of our solutions relies on the new and well-known concept of edge computing, in which a great part of the computation is performed and a full copy of the ledger can be stored to improve the fault tolerance of the system. In fact, moving the transactions' history on the

edge node avoids that a failure of the server node causes loss of users' reward data. As we will see in section II-C, the ETSI European specifications based on the new 5G networks forecast computational possibilities at the edge of the network, which hosts the execution of heavy tasks supported by higher computing power, provisioned on a need basis, than using a dedicated server. Employing edge computing not only gave us an improvement in term of performance, fault tolerance and scalability of the whole platform, but it also improves the security of the system in respect to attacks from both inside and outside the MCS infrastructure, because these nodes are under the control of third-party entities, often telco companies, which should not have conflicts of interest with respect to services running in the smart city.

II. BACKGROUND

This section explores the background related to Distributed Ledgers between non-trusted nodes, and briefly introduces gamification as review mechanism in Mobile Crowd Sensing platforms. The last subsection also gives the reader an abridged introduction to ETSI MEC, a popular telecommunications standard that is widely used in the edge computing field.

A. Mobile Crowd-sensing and gamification

Mobile Crowd Sensing is a paradigm that refers to the distributed gathering of heterogeneous data coming from devices used by crowds. Usually, data collection is performed on portable and power-constrained gadgets, such as simple wearable objects or more sophisticated smartphones. The recent popularity of the MCS platforms comes from the spreading of high-performance versions of the latter, supplied with an ever-increasing number of sensors [1]. The crucial aspects of mobile crowdsensing is the data collection on which to execute machine learning algorithms to obtain usable information [2]. To ensure high participation and a good quality of the crowd-sensed big data, many works propose the gamification approach as a way to stimulate it through incentives [3].

Now we would introduce the ParticipAct [4] crowdsensing platform, our best playground to think to and to test the new architectures we will show in the next sections. The ParticipAct project of the University of Bologna is a complete crowdsensing platform that consists of a sensing client on users' smartphones and a centralized web server receiving and collecting the gathered data. The platform perfectly follows all the guidelines of a good crowdsensing system:

- The client application dealing directly with users and with data gathering operations has a very low footprint on its user's device, in terms of resource consumption and of user's actions needed to collect data, employing a high-performance sensing module called *MoST* [5];
- The server component follows the openness paradigm and it thus results easily extensible and transparent. Furthermore, the collected data can be openly shared with other trusted players, such as other crowdsensing platforms and entities of the smart city;
- The system assures also security and privacy of its users. Integrity and confidentiality are guaranteed through the usage of mechanisms for authentication and secure storage of collected data. For the privacy of the users, the authors leave the data freely accessible from the user that collected them, and they provide notifications to warn about external sharing of their personal data.

The ParticipAct server is only accessible by authorized entities, such as administrators and researchers, that can define the actions users have to complete in order to carry out a so-called campaign. A campaign is a collection of actions, known as *tasks* in the ParticipAct world, that a user has to accomplish in order to collect data and send its contribution to the server. Researchers can customize campaigns choosing tasks to be completed by a selected group of users, a geofence zone of activation and/or completion, and a time frame in which the actions must be completed.

Although in ParticipAct there is already some elements of gamification logic, implemented through scoreboards and badges assigned to users based on their contributions, for now, the platform does not have the possibility to federate different spontaneous systems. The idea would be to allow users of the MCS platform to have their scoreboards and their contributions available while roaming across different ParcticipAct federated servers. For example, if a user is in a city having a ParticipAct server other than her usual one, she can continue to contribute to campaigns without a new subscription, outside of losing the previous contributions. In this work, we address this point, proposing architectural solutions relying on blockchain and edge computing technologies.

B. Distributed ledgers

One of the main purposes of Distributed Ledgers technology takes into account, from a Blockchain point of view, the possibility of enforcing trustiness in the environment where all the participants do not trust each other. The enforced trustiness enables the possibility of defining a federation between non-trusted parties, enhances the verifiability of data from participants and allows the implementation of a tamper-proof access control system.

A Distributed Ledger (DL) or, more generally a "Blockchain System", take into account a strict trust model where actors do not trust each other. Using smart contracts they can enforce rules and transaction validation.

Distributed Ledger implementations could be analyzed taking into account several characteristics, among them we want to evaluate which are the best for a Blockchain-based federated system. The main features that we want to evaluate are: permissioned vs permissionless, tokenized vs tokenless, decentralized vs centralized. Permissionless DL allows the join of who ever want to be part of the ledger, and start to submit transactions and smart contract; permissioned, instead, will not allow anyone to join the network and requires certain credentials (like certificates or keys). The second comparison take into account the possibility of a currency for payments/rewards for each operation made on the ledger. A tokenized DL will limit spam in transaction because each one has a fee, but requires a mechanism to "create" the currency like mining which requires a lot of computing power or you can exchange the currency for fiat currencies. The tokenless DL is prone to spam but has not any intrusion in terms of computing power given by mining. The last category taken into account is Decentralized vs Centralized, where centralized means that there is a central authority which have "extra" powers among ledger nodes; the central authority could be distributed between ledger nodes. A decentralized DL does not provide any form of authority, and relies on consensus of the network. Therefore a tokenless distributed ledger model could be coupled with the permissioned and centralized one, and the tokenized model could be also permissionless and decentralized. Regarding our use case we selected two different use cases: permissioned, tokenless
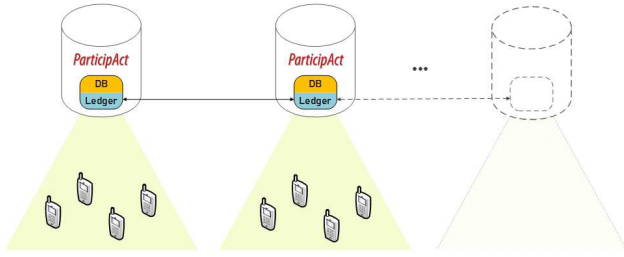
Fig. 1. Hyperledger Fabric model

centralized DL, and permissionless, tokenized decentralized DL with the possibility of deploy a "private" network.

The main platforms considered for our use case are: Hyperledger Fabric and Ethereum. Hyperledger Fabric is a "blockchain framework" implementation developed under the umbrella of the Hyperledger project hosted by the Linux Foundation. Fabric is a permissioned, tokenless and centralized distributed ledger, it defines three types of peers [6]: ordered peers, for intra-peer communication, maintain a consistent state of the ledger, endorsed peers receive and validate the transactions (is the only node which executes smart contracts), and anchor peers, pure ledgers which receive and broadcast transactions and blocks 1.

Ethereum is a tokenized DL which provides a fully-featured platform for distributed applications ("apps"), comprising a full ledger, support for either public and permissionless, or private and permissioned deployments. The fully distributed and decentralized *Ethereum Virtual Machine* (EVM) enables the deployment of user-defined, self-enforcing code entities known as "smart contracts" in order to implement completely decentralized and fair infrastructures based on an easily verifiable set of rules.

C. Multi-access Edge Computing

Before presenting the architectures we deem most suitable to the goal of federating MCS systems to create a common knowledge base consisting of users contributions and gamification scores, we wish to introduce in this section a very popular concept that will be the basis of the next 5G networks, i.e. the Multi-access Edge Computing paradigm [7]. MEC schema gives to developers and service providers an IT service environment at the edge of the network. MEC mitigates the drawbacks arisen in the last years regarding the use of cloud computing resources: by putting these closer to the edge of the network (ideally one-hop far from the devices collecting data), applications could potentially achieve near-real-time communications thanks to ultra-low latency and high bandwidth, obtained through the execution of business logic on edge (cloud) resources [8].

Many contributions in current literature highlight the benefits of deploying MEC architectures for various scenarios and applications [9] [10] [11] [12].

The European Telecommunications Standards Institute (ETSI) proposes a reference architecture that defines the components of the virtualization infrastructure necessary to run MEC applications within operator networks [13]. A MEC host (referred also as MEC node) consists in an entity capable of supplying all the facilities required to run applications on the edge, with a particular focus toward providing compute, storage, and network resources.

This brief introduction will be useful to understand how the usage of MEC nodes for executing part of the business logic can result useful for our current intent of achieving a fair federation of independent MCS systems, based on gamification profiles shared through distributed ledger technologies.

III. POSSIBLE ARCHITECTURES

This paper tries to identify and define two different architectural approaches for a distributed, federated MCS infrastructure capable of automatically authenticate two or more untrusted entities in a federated scenario. We argue the feasibility of distributed ledger technologies as a way to achieve this goal, discussing the advantages and drawbacks of each deployment; we deem these technologies as being worthy of consideration when designing spontaneous federated systems, without a central third-party authentication authority, thanks to their intrinsically decentralized architecture which radically simplifies the aggregation of untrusted entities. These, however, intrinsically entail a considerable expenditure of networking and computational resources, which, together with their associated battery life costs, may result being prohibitively expensive; it is thus necessary to closely analyze how much performance and scalability issues plague the proposed architectures and ultimately hamper their feasibility.

In this work, we address the introduction of a shared ledger for the recording of the rewards assignment among untrusted and unknown participants in a generic gamification system. We tested different solutions, and in particular, we compared the architectural model with and without the encompassing of ETSI MEC nodes. While discarding some deployments solutions such as the unrealistic case of having a full instance of the ledger on user-owned devices for the sake of saving resources, we tried to investigate the feasibility of relying on edge computing to increase the distribution level of the ledger among multiple close-to-edge deployments. We considered the employment of ETSI MEC nodes in a variant of the architectural model to expand the scalability of the whole system, through the utilization of edge computational and storage resources useful for the execution of the DLT-related functions, assisting the servers in this additional task and making federation transparent to the system. Furthermore, including the edge facilities in our deployment could improve the fault tolerance of the MCS platform thanks to the moving of blockchain knowledge base on a network segment more trustworthy as it is managed by third parties and it will be a crucial tier for future telecommunication networks.

A. Client/Server architecture

Given that directly deploying a full distributed ledger on the sensing nodes of a MCS infrastructure would be unacceptably inexpensive due to network and power consumption constraints; this, however, does not generally apply to the server side of the infrastructure. From this last observation, we devised the first architectural model with the intent of introducing blockchain-like concepts to an MCS system while respecting its non-intrusion principle.

In the schema in Figure 2, the distributed ledger is exclusively localized to server level, and it is used to distribute reward data among federated untrusted nodes. Each one of the organizations participating to the MCS campaign (such as companies and universities) will retain and constantly update a full ledger copy for accounting and cross-validation purposes, in a way completely transparent to the end user.

Fig. 2. A brief, high-level graphical representation of how *Architecture I* could be designed

Fig. 4. An hypothetical ParticipAct Node (Server) where a private Ethereum endpoint has been added
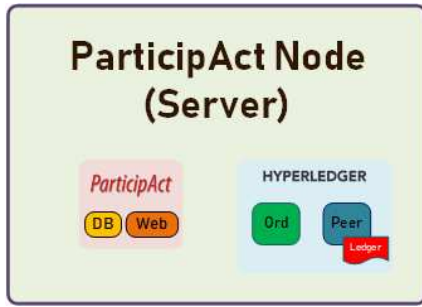

Fig. 3. An hypothetical ParticipAct Node (Server) where a Hyperledger Fabric ledger endpoint has been added. The local deployment hosts both an ordering peer and a full peer; additional components are not shown

Fig. 5. Server-mediated interaction between an end user and the Ethereum network

Each server operates its ledger, aside of its own ParticipAct database, in a completely independent fashion. Each reward is reported and broadcasted to every other node, which, in this case, is associated or allocated to one of the federated servers. This allows the creation of global, cross-institution leader boards and price campaigns; in this way, a client owns a shared MCS-rewarding identity across all of the servers belonging to the federation, which can allow her to roam freely while keeping her score across providers together with a full, irrefutable account of how it has been generated.

We hypothesize a viable implementation of this design using a distributed ledger solution capable of supporting the smart contracts actually needed to the system's basic functions, which mainly consist in the maintenance of a structured ledger potentially containing data from low to medium complexity (i.e., small records and structures) and the operations associated with its distributed, replicated processing. One very interesting and, in our opinion, viable solution might be designed around a Hyperledger Fabric network, with every server instance running the shared chaincode (written in a supported language such as Go), as shown in Figure 3. For this particular deployment instance, we deemed useful to require the presence of at least one ordering peer, on the side of one or more full peer nodes, on every federation member; this guarantees the presence and availability of at least one instance of this fundamental infrastructural component.

Another potential variation of this solution might involve the deployment of a rewarding infrastructure based on a cryptocurrency, instead of just storing leaderboards. Amounts of this token unit can therefore be handed out to the users as a prize, following the successful completion of crowd sensing tasks. The wallets, each one matching 1:1 a ParticipAct account on one of the federated servers, keep the history of transactions performed by every actor belonging to the chain. Every end user can use its "coins" in any way allowed by the common rule set adopted by the federation,
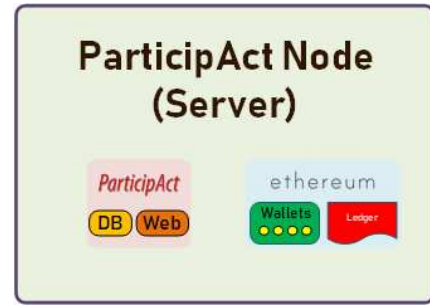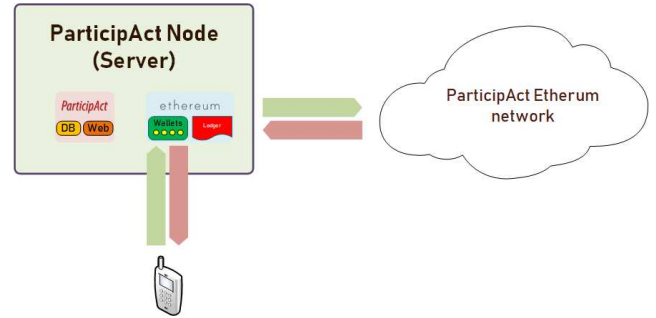
using an 'ad-hoc' application provided by the project; this can either be based on an existing, modified lightweight client, or an extension of the ParticipAct API. While Hyperledger is still a viable solution for this usecase, through the creation and deployment of an appropriate set of chaincode applications implementing a wallet, an already existing cryptocurrency infrastructure with support for private networks and smart contracts is preferable.

One such implementation is Ethereum, one of the major cryptocurrencies and distributed apps (Ðapps) platforms. The contracts, written in a custom language such as Solidity, implement the code and logic behind user wallets, and are deploying in a private, separate subnetwork to which every federation member contributes one or more full nodes, complete with a whole copy of the ledger and an API to allow Wallet access from authenticated members.

The possibility of creating a token based on the public Ethereum network, instead of relying on a private, permissioned Ethereum network separated from the main network has been deemed impractical and unacceptably expensive. While being able to rely on an order of magnitude vaster, and thus much harder to tamper, network would seem like highly desirable from a security standpoint, the unreasonable value of Ether, combined with its high volatility and instability, makes the cost of such a solution unacceptable both in cost and purpose. The rewarding system we aspire to design for ParticipAct is not intended towards handing out monetary rewards, and we strive, therefore, to avoid as much as possible giving the project any sort of financial overtone.

B. Edge-based Architecture

Our second architectural proposal, as detailed in in Figure 6, improves upon the first one by trickling down the ledger infrastructure towards the network infrastructure edge. This is accomplished through the use of Multi-access Edge Computing nodes (MEC) as specified by ETSI in its ETSI-MEC specification,
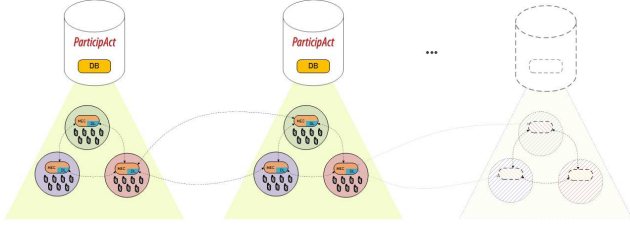
Fig. 6. An hypothetical deployment that relies on ETSI MEC nodes hosting ledger instances
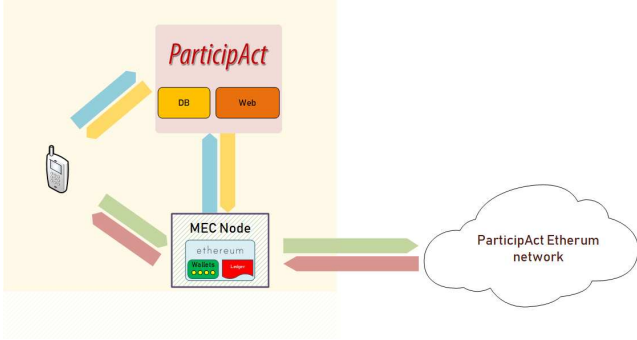

Fig. 7. An MEC-powered version of the infrastructure describe in figure 5. Notice how Ethereum nodes are now hosted inside of edge computing instances

as previously introduced in subsection II-C. Each participant to the crowd-sensing campaign still adheres and is enrolled to a single, specific PartcipAct infrastructure and server, to which one or more of the aforementioned nodes have been added. Among their several services, the nodes contain a complete distributed ledger node, in the form of (ideally) a full replica and a wallet service. The clients, like specified in the previous subsection, still rely on the full nodes contained in their closest MEC node to access, operate and control their rewarding account records.

The infrastructure still remains for all intents and purposes unaltered, with collected data still privately kept by servers independently from each other. Infrastructural elements report to their closest MEC node after becoming aware of a correctly validated task result, including its relative point allotment; this information is then shared by the platform to every other node under the control of federated members, granting them the capability to quickly verify the correctness of the system in awarding rewards.

To reach this end, as previously described for client/server deployment, the system relies on a fine-grained set of smart contracts that implement an advanced system capable of autonomously hand over rewards, which can, also in this case, consist in amounts of token cryptocurrency. Figure 7 shows how an high-level Ethereum-based implementation of how such a system might appear from an architectural standpoint.

IV. COMPARISON

Both the architectural approaches proposed in the previous section are subject to several inherent benefits and drawbacks. First, we can assume that, given a correct technological deployment without any malicious actor tampering with critical security parameters (such as configurations or certificates), the solutions devised in this paper could potentially increase the trustworthiness of the overall gamification scheme through partial decentralization. That is to say, while the overall administration of the initiative is still in the hands of the federated nodes

instead of being put under direct user control, we deem that the proposed systems make single institutions more accountable with respect to malicious activities compared to a centralized solution. It is, indeed, much harder to tamper a ledger where every single, slightest transaction is under the scrutiny of a large number of entities, as long as the majority of these can be assumed as not being actively conspiring against its integrity; this can be considered a significant improvement compared to a system relying on a negligent or corruptible centralized authority.

Comparing the two architectures, we can characterize the first one as potentially flawed and ineffective in preventing tampering of the ledger when the number of participating institutions is low, which tends to be by far the most common case. A relatively low number of nodes enrolled in the federation would make the risk of 50% + 1 attacks not negligible, by creating a situation where malicious actor can potentially hijack ledger consensus by taking over a majority of nodes. This eventuality fundamentally voids the advantages carried by a distributed ledger when the number of nodes or independent institutions is insufficient, given that under the aforementioned circumstances such a deployment cannot be considered trustworthy without the presence of some form of authority capable of monitor the network for improper behaviours. It is therefore necessary to determine a minimum number of participants below which a centralized authority can be considered more secure and thus preferable. The main advantage of this solution compared to the MEC-powered one mainly consists in being potentially easier to deploy, by sharing resources with those already allocated to the server infrastructure.

The second infrastructure, thanks to its reliance on ETSI MEC, is intrinsically more resilient to 50% + 1 attacks, due to the ledger being distributed among a vastly larger number of edge nodes, some of which potentially (and highly desirably so) under the control of independent (and ideally less corruptible) third parties such as telecommunications operators. This factor alone really helps in making small- to medium-scale attempts in tampering the ledger less likely to succeed, due to the larger investment required in altering so many instances under an heterogeneous ownership.

The reliance on a vast array of edge nodes can, however, result in being a drawback for this architecture, due for the need for the ledger of being highly distributed and replicated among the installs, a circumstance that can lead to slowdowns and difficulty in reaching consensus in case of subpar network performance. Given the eventuality that some the participants could be operating in a moderately hostile environment, plagued by unpredictable network latencies, some nodes might find themselves unable to keep their ledger copy up to date, thus creating a potential source of performance degradation and unfairness; this can, in fact, result in these nodes being incapable to participate to the voting process like their better performing peers with a reliable, fast network connection.

To conclude, the drawback of a distributed ledger approach mainly consists in increased complexity. While maintaining a centralized authority can be challenging and potentially costly, it requires a vastly lower amount of network traffic and time to reach an efficient consensus, given that in this circumstance decisions are arbitrarily settled by the aforementioned trusted arbiter in the limits of its powers. We thus recommend any interested user to seriously evaluate the potential benefits and costs must of a distributed ledger solution with respects to their specific use case.

V. Conclusion

This paper proposes an original dissertation about how the capabilities provided by distributed ledger systems can be exploited to address the issues arisen while designing federations of autonomous, different, and untrusted MCS systems. After a brief introduction on how gamification can improve both quality and quantity of data harvested by rewarding useful behaviours, we thought about gamification scores and profiles of the MCS platforms users could be shared among members of a federation.

It will be desirable in the modern smart cities scenario the coexistence of different MCS platforms without the need of central third-party authority for their federation. The users should share their profiles among all the systems of the federation, without logging-in every time and with the possibility to retrieve its own scores at all systems. Employing a distributed ledger can be onerous in term of resources consumption and intrusion in user's daily habits during the crowdsensing campaigns, so in this paper, we made a comparison between two possible architectures for federated systems, focusing on the strengths and the weaknesses of both.

Our first evaluation is based on the proposal of a complete ledger within the sensing devices, unrealistic due to limitations of these nodes characterized by excessive mobility, such as power consumption and constrained resources. Consequently to this consideration, we pondered two considerable alternative architectures. Consequently to this consideration, we pondered two considerable alternative architectures, giving access to the distributed ledger only to servers, not affected by the limited resources problem of the leaf nodes.

In the first schema, each server owned by an organization (such as companies, universities, and public administration) is responsible for the updating of the distributed ledger with the user's rewards information. In this configuration, each server-node is part of the Hyperledger Fabric Network for storing and sharing leaderboard information. We considered also a variation of this schema in which the rewarding infrastructure involves a cryptocurrency infrastructure (such as Ethereum). In this case, the token unit can be handed out to the users as a prize for the successful completion of sensing activities. Each user with an account on ParticipAct has its own wallet with coins expendable in any way allowed by the federation. An existing infrastructure system with support for private networks and smart contracts is preferable to the chaincode installation on peers, like the case with the Hyperledger Fabric Network. However, we do not aim to obtain a monetary system behind the crowdsensing platform, so we believe that using the public Ethereum network, instead of a private/permissioned one, is impractical and excessive, even if much harder to tamper.

The main alternative to the previous architecture is the splitting of the whole system in three layers, adding the MEC layer to the client and server ones. Although the system has the same features as the previous one, the architecture differs for the place where the ledger is located. The MEC nodes now contain a full distributed ledger (and a wallet implementation), and each sensing device refers to the nearest MEC node to record information regarding tasks completion and the consequent updating of the score. Each server relies on a number of MEC nodes, associated with him, to verify the correctness of the system in awarding rewards, instead of handling on its own all the distributed ledger parts. In the edge part of the network, we can put the heavy computation involved in the blockchain-related tasks, with significant savings of resources on the ParticipAct servers. We think about the calculation of rewards and execution of blockchain client that implements wallet and ledger.

We think both presented solutions increase the trustworthiness of the overall gamification system, partially decentralizing the duties and the verification process. Obviously, a federated MCS system with a large number of participant is more difficult to manipulate by malicious users, so a future work could be to test a good trade-off to give the best security solution to the federated system, based on the number of the participating servers. Mostly for the first solution we presented, a low number of nodes enrolled in the federation would put a non-negligible risk of 50% + 1 attacks, with a potential. We identified the main advantage of the first architecture in having an easy of deployment respect to the MEC-powered solution. Moreover, the MEC nodes are under the control of entities not participating in the federation, such as telco operators, therefore more difficult to corrupt. The MEC layer, adding a lot of participant to the distributed ledger network, is more strong respect to the vulnerability to a 50% + 1 attack. However, involving a lot of MEC nodes, resulting in high distribution, can represent a bottleneck during the consensus process, in the event of a loss in the performance of the global network that connects all nodes.

Next study will aim to establish a trade-off between the difficult deployment of a distributed ledger-like solution and the security benefits it could provide. Surely the actual implementation of the presented alternative solutions will provide us with guidelines to draw up the characteristics that a federated MCS system must have to justify the implementation and use of a distributed ledger.

References

[1] K. Abualsaud, T. El-Fouly, T. Khattab, E. Yaacoub, L. Ismail, M. H. Ahmed, and M. Guizani, "A survey on mobile crowd-sensing and its applications in the iot era," *IEEE Access*, vol. PP, pp. 1–1, 12 2018.

[2] M. Pouryazdan, B. Kantarci, T. Soyata, L. Foschini, and H. Song, "Quantifying user reputation scores, data trustworthiness, and user incentives in mobile crowd-sensing," *IEEE Access*, vol. 5, pp. 1382–1397, 2017.

[3] X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, and X. Mao, "Incentives for mobile crowd sensing: A survey," *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 54–67, Firstquarter 2016.

[4] G. Cardone, A. Corradi, L. Foschini, and R. Ianniello, "Participact: A large-scale crowdsensing platform," *IEEE Transactions on Emerging Topics in Computing*, vol. 4, no. 1, pp. 21–32, Jan 2016.

[5] G. Cardone, A. Cirri, A. Corradi, L. Foschini, and R. Montanari, "Activity recognition for smart city scenarios: Google play services vs. most facilities," in *2014 IEEE Symposium on Computers and Communications (ISCC)*, June 2014, pp. 1–6.

[6] "Hyperledger fabric." [Online]. Available: https://www.hyperledger.org/projects/fabric

[7] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5g network edge cloud architecture and orchestration," *IEEE Communications Surveys Tutorials*, vol. 19, no. 3, pp. 1657–1681, thirdquarter 2017.

[8] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, Feb 2018.

[9] Z. Wu, J. Zhang, W. Xie, and F. Yang, "Cdn convergence based on multi-access edge computing," in *2018 10th International Conference on Wireless Communications and Signal Processing (WCSP)*, Oct 2018, pp. 1–5.

[10] P. Zhang, M. Durresi, and A. Durresi, "Mobile privacy protection enhanced with multi-access edge computing," in *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, May 2018, pp. 724–731.

[11] R. Mogi, T. Nakayama, and T. Asaka, "Load balancing method for iot sensor system using multi-access edge computing," in *2018 Sixth International Symposium on Computing and Networking Workshops (CANDARW)*, Nov 2018, pp. 75–78.

[12] N. Dao, Y. Lee, S. Cho, E. Kim, K. Chung, and C. Keum, "Multi-tier multi-access edge computing: The role for the fourth industrial revolution," in *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, Oct 2017, pp. 1280–1282.

[13] ETSI. (2019) Multi-access edge computing (mec); framework and reference architecture. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/02.01.01_60/gs_MEC003v020101p.pdf