A Blockchain-Enabled Trustless Crowd-Intelligence Ecosystem on Mobile Edge Computing

Jinliang Xu, Shangguang Wang, *Senior Member, IEEE*, Bharat K. Bhargava, *Fellow, IEEE*, and Fangchun Yang, *Senior Member, IEEE*

***Abstract*—Crowd intelligence tries to gather, process, infer, and ascertain massive useful information by utilizing the intelligence of crowds or distributed computers, which has great potential in Industrial Internet of Things. A crowd-intelligence ecosystem involves three stakeholders, namely the platform, workers (e.g., individuals, sensors, or processors), and task publisher. The stakeholders have no mutual trust but interest conflict, which means bad cooperation of them. Due to lack of trust, transferring raw data (e.g., pictures or video clips) between publisher and workers requires the remote platform center to serve as a relay node, which implies network congestion. First, we use a reward-penalty model to align the incentives of stakeholders. Then the predefined rules are implemented using blockchain smart contract on many edge servers (ES) of the mobile edge computing network, which together function as a trustless hybrid human–machine crowd-intelligence platform. As ES are near to workers and publisher, network congestion can be effectively improved. Further, we proved the existence of the only one strong Nash equilibrium, which can maximize the interests of involved ES and make the ecosystem bigger. Theoretical analysis and experiments validate the proposed method, respectively.**

***Index Terms*—Blockchain smart contract, crowd-intelligence ecosystem, hybrid human–machine, mobile edge computing, reward and penalty, strong Nash equilibrium, trustless.**

I. Introduction

CROWD-INTELLIGENCE is to gather, process, infer, and ascertain massive useful information by utilizing the intelligence of crowds or computers, whereby a publisher broadcasts massive tasks to lots of semi skilled workers to obtain reliable answers [1]. It is widely used in knowledge collecting (e.g., mobile crowdsensing [2]), decision making (e.g., tagging of machine learning training dataset [3]), etc. Industrial Internet of Things (IIoT) has the properties of high distribution, high-frequency activity, and is near to massive mobile users, which makes itself a great potential for crowd intelligence [4], [5]. The number of tasks to be completed is too large to complete on time for limited number of professionals, or the tasks are too difficult to be autoprocessed well just by computer programs [3], [6], [7]. A crowd-intelligence ecosystem involves three stakeholders, namely platform, workers, and publisher that can publish tasks to workers.

These three stakeholders may have no mutual trust and their interests conflict with each other [6], [8], [9]. For example, workers try to get more payment from the publisher with low-quality works. It is not easy to align incentives of stakeholders so that they may collaborate smoothly. This harms the long-term development of the whole crowd-intelligence ecosystem, e.g., answers of low quality, unnecessary high cost paid by publisher to workers, delayed completion of the tasks (i.e., exceeding the time constraint set by the publisher), and weakening platform (i.e., loss of professional workers and poorly prespared workers flooding into it). Due to the lack of mutual trust among stakeholders, the raw data for a task (e.g., pictures or video clips that are published by the publisher or submitted by workers [2], [3], [10]) cannot be transferred directly between publisher and workers. The remote cloud center (RC) of the centralized crowd-intelligence platform must serve as the third-party guarantee and the intermediary node of the data transferring network path. As raw data transferring plays a great role in a normal crowd-intelligence ecosystem, trust problem may result in excess bandwidth and delayed response [11]–[13]. This needs to be avoided in time critical use cases, e.g., automobile navigation [10], [14], [15] or disaster recovery [16]. Finally, a crowd-intelligence ecosystem is short of the available workers to complete growing number and variety of tasks and this leads to low quality of final answers and delayed completion of the complete tasks.

We solve the above-mentioned problems from the following three aspects, and build them into an integrated method named blockchain-enabled trustless crowd-intelligence ecosystem operated on mobile edge computing network (Fig. 1, note that RC is not obligatory in general. But without RC, the proposed decentralized crowd-intelligence network still works.).

Manuscript received August 30, 2018; revised December 28, 2018; accepted January 29, 2019. Date of publication February 1, 2019; date of current version June 12, 2019. This work was supported in part by the National Key R&D Program of China under Grant 2018YFB1004801, and in part by the National Science Foundation of China under Grant 61571066. Paper no. TII-18-2261. *(Corresponding author: Shangguang Wang.)*

J. Xu, S. Wang, and F. Yang are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: jlxu@bupt.edu.cn; sgwang@bupt.edu.cn; fcyang@bupt.edu.cn).

B. K. Bhargava is with the Department of Computer Science, Purdue University, West Lafayette, IN 47906 USA (e-mail: bbshail@purdue.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2019.2896965

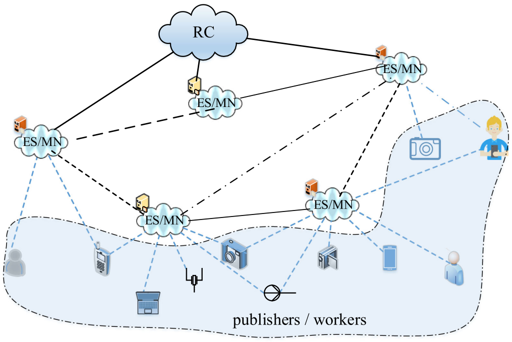


Fig. 1. Proposed blockchain-enabled trustless crowd-intelligence ecosystem. It is decentralized and implements a blockchain smart contract and operates on edge servers/blockchain masternodes (ES/MNs) of a mobile edge computing network. Both human beings and machines can serve as workers or publishers, and and it can be called a hybrid human–machine crowd-intelligence ecosystem.

1) *Incentive problem:* An incentive-compatible and efficient incentive model is particularly important for a decentralized system. We build a reward-penalty model to align incentives of the three stakeholders. We reward or punish workers according to the quality of their committed answers instead of only positive payment [10], [17]. We develop a family of appropriate reward-penalty function couples, by which the amounts of reward and penalty can be computed based on the workers' committing beliefs without destroying the incentive compatibility property of the ecosystem. Different from a single reward-penalty function couple or rewarding only, this model provides a more flexible solution to manage the interests of three stakeholders, and help to reduce latency, improve quality, and do good to platform evolution of the crowd-intelligence ecosystem [10], [17].

2) *Trust problem:* A worker's performance history determines the amounts of her reward/penalty in completing tasks. And the predefined management rules/cooperation standards define how the platform works. It is of important to avoid malicious activity or tampering from the bad minorities, and win the trust of stakeholders. We propose to treat it with blockchain smart contract. The smart contract is copied to multiple edge servers (ES) of mobile edge computing and executed. The proposed crowd-intelligence platform is decentralized using blockchain technology and is not hosted on a centralized RC. The ESs are owned by many individuals and behave as blockchain masternodes (MNs) [18]. Without the agreement of more than a half MNs, the smart contract cannot be changed arbitrarily [19]. The stakeholders can collaborate without trusting each other. Specifically, the raw data transferring between publisher and workers can proceed with the nearest ES/MNs serving as intermediary nodes, which can avoid excess bandwidth and delayed response. The ES/MNs charge a small fees from the publisher and workers to keep the security state of the blockchain, and earn money from it [18], [19]. With the help of blockchain smart contract, the involved ESs can maximize profits with strong Nash equilibrium [20] and the proposed ecosystem can attract more ESs to serve as blockchain MNs.

3) *Worker shortage problem:* When the trust problem is solved, a trustless hybrid human–machine crowd-intelligence platform can be built, where both of human beings and machines (including sensors or processors) are able to serve as workers or publishers without mutual trust. Specifically, human workers perform well in highly complex tasks (i.e., crowdsourcing [21]), while machine workers can do better in realtime high-frequency tasks (i.e., crowdsensing [10]). This platform can also help to solve the worker shortage problem.

The rest of this paper is organized as follows: We introduce how to align the interests of three involved stakeholders using the reward and penalty model in Section II. Section III shows how to build a trustless crowd-intelligence ecosystem using blockchain smart contract on a mobile edge computing network. In Section IV, we introduce the hybrid human–machine concept to relieve the worker shortage problem. In Section V, experiments on two synthetic datasets are conducted to validate the proposed crowd-intelligence ecosystem, and the results are supplements to the proof and analysis in the previous sections. Section VI presents the existing related works. And Section VII draws conclusions and discusses our future work.

II. Reward-Penalty Model

An incentive-compatible and efficient incentive model is particularly important for a decentralized system. Not like the traditional centralized crowd-intelligence system, the proposed decentralized version has no an arbitral authority to settle the disputes among different stakeholders. If the system is incentive-compatible, every stakeholder can achieve the best outcome to themselves just by behaving according to predefined rules, which helps to avoid disputes. If the proposed decentralized crowd-intelligence system is efficient, it will have more advantages over the existing centralized competitors. As a result, a good incentive model that can achieving the above targets is needed.

The proposed reward-penalty model does not estimate a worker's proficiency or her reliability on each commit. Instead, it assumes that each worker knows her reliability on a specific task, and its target is to incentive workers to commit the truth. If a worker is assigned a task, she is asked to commit both of task's *type* information (i.e., the answer she selected from the candidates) and *belief value* (i.e., her confidence that measures the probability that her selection will be judged right). Only binary-type task is taken into account here, such that the number of the task's candidate answers is uniformly two, like *Yes* or *No* [8], [17], [22]. *Belief value* is reasonable because that a worker always intentionally or unintentionally estimates in her own mind the probability of giving the right answer when she is encountered with a decision-making task due to tasks' inherent difficulty, different individual experience, and professional knowledge [23], [24].

Some symbols and denotations are as follows. The number of tasks is denoted by T, and there are a total of N workers to complete these tasks. The index of a task is denoted by an integer $1 \leq t \leq T$, and a worker is indexed using an integer $1 \leq n \leq N$. The binary-type space of tasks is denoted as $\{-1, +1\}$. We use s_t to denote the worker set that completed task t. When worker n completes task t, we use $a_{n,t}$ to represent her committing type

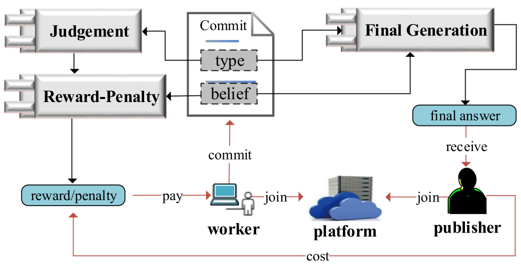


Fig. 2. Workflow of the reward-penalty model.

for this task. The *belief* value is in range [0.5, 1], which means that the committed type is more suitable than the other one. x denotes the committed belief value x, and the real probability that her committed type is judged right is denoted by c. When a worker's committing type is finally judged correct, her reward amount is denoted by the reward function of variable x, i.e., $\text{reward}(x) > 0$, and $\text{penalty}(x) > 0$ is the penalty function if her committing type is judged as wrong.

The workflow of reward-penalty model is shown in Fig. 2, and it contains three modules, namely *judgment*, *final generation*, and *reward-penalty*.

A. Judgment Module

Judgment module uses *benchmark answer* to judge whether a work for a task deserves reward or penalty for a task completion. The *benchmark answer* for task t is defined by the following:

$$\widehat{a_t} = \text{sgn}\left(\sum_{n \in s_t} a_{n,t}\right) \tag{1}$$

where $\text{sgn}(x) = -1$ if $x < 0$, and 1 otherwise.

Note that benchmark answer $\widehat{a_t}$ will not be sent to the publisher as the final answer of a task. It is just used to judge whether or not worker n is due for a reward or penalty for her $a_{n,t}$. Specifically, if $a_{n,t}$ equals to $\widehat{a_t}$, the worker will receive a reward from the publisher, otherwise she will need to pay to the publisher as a penalty.

B. Reward-Penalty Module

This module computes the how much reward $(\text{reward}(x))$ or penalty $(\text{penalty}(x))$ should be given. In statistical terms, the type committed by a worker will be judged correct with probability of c, and occurrence probability of a wrong judgment is the rest $1 - c$. Therefore, the *expected gain payment* is defined as

$$\text{expected}(x) = c \cdot \text{reward}(x) - (1 - c) \cdot \text{penalty}(x). \tag{2}$$

Due to different individual experiences and professional knowledge, value c given by different workers may be different. Moreover, a specific worker's c for different tasks changes according to their respective inherent difficulty.

Then the model is formulated by the following:

$$\begin{cases} \text{argmax}_x \ \text{expected}(x) = c \\ \langle \text{reward}, \text{penalty} \rangle \geq 0, \langle \frac{\partial \text{reward}}{\partial x}, \frac{\partial \text{penalty}}{\partial x} \rangle > 0 \\ \text{reward}(0.5) = \text{penalty}(0.5) = 0, \text{reward}(1) = 1 \\ 0.5 \leq x \leq 1 \end{cases} \tag{3}$$

of which the first means that a worker must commit c to maximize her expected gain, i.e., a worker can get the largest expected gain payment just only if she chooses to commit the truth [25], [26]; the second means that the reward and penalty have a positive correlation with x, a larger belief value of its corresponding committing type has a larger influence on the benchmark and final answer of the task, and it may be beneficial or harmful [23], [24]; the third means the boundary values; and the fourth means the definitional domain. A fixed boundary value is to ensure *easy solvability* of reward-penalty functions [27]. As a commit with $x = 0.5$ cannot offer any useful information, no stakeholder will gain or lose for it.

By solving this model, we successfully find a family of reward-penalty functions by the following:

$$\begin{cases} \text{reward}_k(x) = \frac{-(k-1)2^k x^k + 2^k k x^{k-1} - (k+1)}{2^k - k - 1} \\ \text{penalty}_k(x) = \frac{(k-1)2^k x^k - (k-1)}{2^k - k - 1} \end{cases} \tag{4}$$

where $k \geq 2$ can be considered as the order of a reward-penalty function couple and *personal order value* of the corresponding worker. Then, we can obtain the expected gain function with kth order by plugging (4) into (2):

$$\text{expected}_k(c) = \frac{2^k c^k - 2ck + k - 1}{2^k - k - 1}. \tag{5}$$

C. Final Generation Module

This module generates the *final answer* as the final result of the corresponding task, which is what the publisher wants. The *final answer* a_t^* for task t can be generated by the following:

$$a_t^* = \text{sgn}\left(\sum_{n \in s_t} \text{expected}_{n,t} a_{n,t}\right). \tag{6}$$

Compared to *benchmark answer* in (1), *final answer* in (6) is more like the weighted majority rule or weighting aggregation rule that are widely used in crowdsourcing [28]. It can generate the true answer even if majority workers' committing types are not right. As a larger belief value means its corresponding type is more reliable than a smaller one, it is reasonable to consider workers' belief value into computing the final answer.

The expected payment $\text{expected}_{n,t}$ is suitable for the weight coefficient of committing type as follows:

1) It has monotonicity property in $0.5 \leq c \leq 1$ for any $k \geq 2$, which guarantees a larger influence of a committing type if it has a higher belief value.
2) It is a better fit than either one of them in measuring the reliability of a committing type.
3) It has more distinguishing ability in comparison to belief value c.

4) The more the publisher pays, the better quality she gains, which helps to make the most of publisher's money.

The reward-penalty model is incentive compatible and efficient, which helps to align stakeholders' interests, latency, and quality control as follows:

1) The publisher needs to pay only for good committing answers, and can get some compensation from the penalty for bad commits.
2) It is easy to design *personal order value* into a negative function of a worker's performance history. Then, a professional worker can gain more than a badly behaved worker if they commit the same for a task. This way model can attract more good workers to the platform.
3) When a worker feels that the assigned task is very hard, she can commit with a tiny belief value (i.e., approximating to 0.5) for a tiny gain, by which difficult tasks will not be left behind and undue latency would not occur.

The reward-penalty model needs blockchain technology as follows.

1) Workers and publishers are the comparatively weak side against the centralized platform. The proposed reward-penalty model entitles workers and publishers to exert influence on all of the three involved stakeholders. This is consistent with the decentralization idea of blockchain and mobile edge computing.
2) As the reward-penalty model requires workers to deposit in advance a certain amount of fund to the platform as the possible the source of penalty, the platform may misuse workers' money without proper supervision, which is not workers do not want.

   Blockchain technology can help to avoid diversion of fund by decentralizing fund management.

In the following, we will introduce how a crowd-intelligence ecosystem can benefit from blockchain and mobile edge computing.

III. Trustless Crowd-Intelligence Platform on Mobile Edge Computing

A. Trustless Platform Enabled by Blockchain Smart Contract and Mobile Edge Computing

The trust problem of a crowd-intelligence ecosystem results from the following two ways:

1) *High centralization:* The reward-penalty model in Section II relies too heavily on the crowd-intelligence platform in aligning incentives of stakeholders. A worker's reward/penalty amounts for task completion is closely related to her performance history (i.e., order k in (4)). In addition, the predefined management rules of the platform and the collaboration standards of stakeholders define how the ecosystem works. All of these information and rules are controlled by and stored in a highly centralized crowd-intelligence platform (a RC) [2], [10], which poses the following risks: the platform itself has the right to tamper them, and natural disaster may damage them easily [16]. Without solving the risks brought by the centralized platform, the crowd-intelligence ecosystem cannot win the trust of other stakeholders.

2) *Competing interest:* Interest conflict of stakeholders leads to lack of mutual trust [6], [8], [9]. So data for a crowd-intelligence task cannot be transferred directly between publisher and workers. A less-than-ideal alternative is to invite RC of the centralized crowd-intelligence platform to serve as the third-party guarantee and the intermediary node of the data transferring network path. This increases the bandwidth overburden and response time [11]–[13], as raw data transferring plays a great role in a normal IIoT or crowd-intelligence ecosystem, where pictures or video clips are published by the publisher or submitted by workers [2], [3], [10], [16].

We propose to use blockchain smart contract to implement the reward-penalty model, workers' performance history data, and the predefined management rules of the platform and the collaboration standards, which can reduce the risks due to a centralized platform; we propose to use many ES of mobile edge computing as blockchain MNs to host and run the smart contract, which can relieve network congestion. We present details in two steps as follows:

1) New activities among stakeholders will be recorded and packed into a newly created data block at set intervals. A new block has a pointer to the unique hash code of its previous block, and all blocks form a chain, which is called blockchain. It is scarcely possible to tamper or damage a blockchain. As a result, if some information has been stored on blockchain, everyone can trust it [29]. Blockchain smart contract is a piece of computer program that is owned by all blockchain nodes, and it cannot be changed arbitrarily without the consensus of majority of nodes [19]. What we should implement using blockchain smart contract includes the reward-penalty model, workers' performance history data, the predefined management rules of the platform, and the collaboration standards of stakeholders. When the presupposed condition is met, the agreed steps will proceed automatically. In this way, a centralized crowd-intelligence ecosystem is transformed into a decentralized one that is operated on a blockchain network, and it can have the trust of all stakeholders.
2) The blockchain's consensus algorithm is designed to rely on a certain range of memory and bandwidth, which limits the corresponding mining hardware on ES or some other computing devices at edge network like that [19], [30]. This design can make the ecosystem distribute wider, and fit into mobile edge computing. These hardwares store an exact replica of the blockchain, and we can call them MNs. To relieve bandwidth overburden and response delay, the nearest ES/MN is able to replace a RC to serve as an intermediary node and third-party guarantee during a raw data transfer between a worker and publisher. As most of workers and publishers have limited hardware resources (e.g., mobile phones and sensors), they need only be a node of simplified payment verification (SPV node),[1] and do not have to host a full replica of the blockchain like ES/MNs. In addition, as an ES/MN can

[1] https://en.bitcoinwiki.org/wiki/Simplified_Payment_Verification

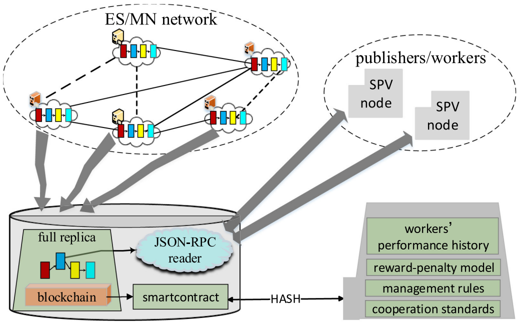


Fig. 3. ES/MN nodes and SPV nodes of the crowd-intelligence ecosystem on mobile edge computing network.

be in close proximity to workers/publishers than the RC, it can help them preprocess the task data at edge network (e.g., videos clips or images). This can further relieve network congestion, and increase the system's ability to respond to massive tasks [16], [31]. The relationship of ES/MN node and SPV node is shown in Fig. 3.

The reasons that the proposed blockchain is deployed on ES are as follows:

1) An ES/MN should have enough storage resource to host the block data of the blockchain. In addition, it should have enough computation resource (e.g., CPU, GPU) to provide services to workers and publishers. What is more, as ES/MNs need to communicate with each other, network bandwidth is also needed. The demand of storage/computation/network resources decides that ES/MN should be at least a small datacenter.
2) For an ES/MN, to save cost and enhance its competitiveness against other ES/MNs, it should be near to workers or potential worker. That is to say, ES/MNs should be deployed near to people. As a result, the ideal place for the proposed system is ES.

B. Incentives for Collaboration Between ES/MNs

The set of ES/MNs play a key role in the proposed trustless ecosystem. Many ES/MNs together function as a decentralized crowd-intelligence platform, and regulate and guide the behaviors of involved stakeholders. Beyond that, with the help of blockchain smart contract, the other stakeholders can trust the platform and collaborate with each other, and the network congestion resulted by transferring data among stakeholders can be relieved. What is more, if the number of ES/MNs increases, all stakeholders in the ecosystem can benefit more. Specifically the following confditions hold:

1) As ES/MN network covers more and more geographical areas, more workers/publishers will be able to access the crowd-intelligence services through the ES/MNs around them, especially the people in remote rural areas with poor and expensive network service.
2) As the deployment density of ES/MNs in one region grows, workers/publishers will have more choices for data preprocessing and intermediary nodes for transferring data.

TABLE I
PAYOFF MATRIX FOR TWO ESS IN A TRADITIONAL CROWD-INTELLIGENCE ECOSYSTEM WITHOUT BLOCKCHAIN SMART CONTRACT

		ES #B	
		join	leave
ES#A	join	$b+d$ $a+c$	$b+d+\alpha$ $a+c-\alpha$
	leave	$b+d-\beta$ $a+c+\beta$	$\underline{\boldsymbol{b}}$ $\underline{\boldsymbol{a}}$

* $0 < \alpha \leq c \wedge 0 < \beta \leq d$.

3) A large ES/MN network can generate a fully competitive market environment, which can reduce the maintenance cost of the crowd-intelligence ecosystem.

We now show why the proposed trustless crowd-intelligence ecosystem can attract more ES to serve as MN in it, and how the involved ES/MNs can maximize their profits with the help of blockchain smart contract [20]. As is shown in Fig. 1, an ES at edge network can be considered as a very small cloud with limited resources (e.g., CPU, memory, bandwidth), while the RC at the core network will not encounter resource shortage problem [31]. If too many mobile workers gather around an ES/MN at some point, the resources of the ES will not meet the demands of these workers and the corresponding ES must buy resources from nearby ESs or the RC [32], [33]. In this process, two kinds of trust problems arise: 1) no mutual trust between two ESs can ensure a resource trading, unless they find a RC as the third-party guarantee [31], [34]; 2) ESs are not willing to trust the RC as the RC can provide its own resources to ESs. If the RC has dominated the resource market, it will leverage its status of the intermediary to squeeze ESs out.

The following section expresses our views: First, we introduce the prisoner's dilemma without blockchain smart contract on a traditional crowd-intelligence platform, which is why it can not attract more ESs and make the ecosystem grow bigger [31], [34]. We suppose all ESs are rational economic man, and the game is played as follows: if both of two ESs #A and #B choose to leave the ecosystem and trade resources with the RC, they will gain a and b, respectively; if they choose to join and trade with each other, the transferring distance of the traded resources between two neighboring ESs is much less than that between ESs and the RC. As a result, both of ESs #A and #B can gain more than before; if one choose to join while the other choose to leave, the former must give more profit to the latter to keep it inside the ecosystem. The payoff matrix of this game is shown in Table I. The dominant strategy for both involved ESs is to leave the ecosystem, which is the only strong Nash equilibrium. However, both of them gain the least (i.e., a and b). So a traditional crowd-intelligence platform is absolutely a prisoner's dilemma for ESs. ESs of mobile edge computing have no incentives to join the ecosystem.

We show how we break through the prisoner's dilemma using blockchain smart contract, and incentive ESs to join the ecosystem as MNs. The emphasis is to help ESs in this ecosystem to get rid of the RC, and form an autonomous community. In the proposed trustless crowd-intelligence ecosystem, first all trading-related information and rules are stored with blockchain

TABLE II
PAYOFF MATRIX FOR TWO ESS OF THE PROPOSED TRUSTLESS CROWD-INTELLIGENCE ECOSYSTEM

		ES #B	
		join	leave
ES#A	join	$\underline{\boldsymbol{b+d}}$ $\underline{\boldsymbol{a+c}}$	$b+d-\epsilon_2$ $a+c-\epsilon_2$
	leave	$b+d-\epsilon_1$ $a+c-\epsilon_1$	b a

* $\epsilon_1, \epsilon_2 < c, d \wedge 0 < \epsilon_1, \epsilon_2 \ll c + d \wedge \epsilon_1, \epsilon_2 < \alpha, \beta$.

smart contract. During a resource trading, both of the corresponding two ESs must follow these already recognized rules. So ESs can buy or sell resources with others without mutual trust or third-party guarantee. Table II shows the payoff matrix for two ESs of the proposed trustless crowd-intelligence ecosystem. The difference of payoff matrix in Table II from that in Table I are the payoff values for two ESs' different choices. Specifically, the two ESs can trust the blockchain smart contract as a third-party guarantee, and trade resources under the guidance of the blockchain smart contract without trusting each other. In this case, the blockchain smart contract helps to match the most suitable seller/buyer. In return, the blockchain smart contract platform will charge small fees from two ES/MNs to keep the blockchain running (i.e., ϵ_1, ϵ_2 in the payoff matrix). We can see that joining is the dominant strategy for both of the two involved ESs, which is the only strong Nash equilibrium of the game. That means the ESs can maximize their profits and have enough incentives to join the proposed trustless crowd-intelligence ecosystem to serve as MNs.

The root cause of trading resources between ES/MNs is to reduce cost. When two MN/ES cannot match each other, it means that they cannot reach consensus over the price of resources. When two ES/MN cannot reach consensus over the price, it means that the buyer cannot reduce cost by buy resources from others. Then, it will process the work by itself, and cannot provide more services before this ES/MN can free more resources.

C. Brokerage From the Publisher to ES/MNs

ES/MNs will not offer services (including collaboration between ES/MNs) to workers and publishers for free. Someone should pay for it. In this paper, the publisher should pay brokerage to its directly connected ES/MNs. The brokerage model is as follows:

1) The publisher give the brokerage ratio η when she publish the tasks. If a ES/MN accepts the ratio, it will join in to provide crowd-intelligence services.
2) When the publisher pays reward r to a worker for completing as task, it also pays ηr brokerage to all of ES/MNs that have provide services for completing this task, and the brokerage is equally distributed between these ES/MNs.
3) If the worker should pay penalty p back to the publisher for this task, the ES/MNs associated with this task have to share ηp penalty.

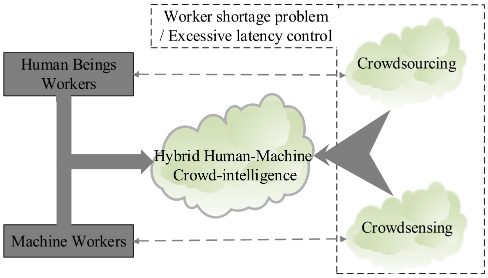


Fig. 4. Crowd-intelligence platform with hybrid human–machine workers.

For an ES/MN, the money it has received minus the money it has paid out is its net remuneration. Every ES/MN has full responsibility for its own profit and loss. The brokerage model has no impact on the incentive compatibility of the reward-penalty model. So as long as the workers have positive overall expected gain, the ES/MN will have positive net remuneration.

There is need to build a completely new blockchain for this paper. A complete new blockchain that can fit crowd intelligence is the best. With the tailor-made transaction structure and smartcontract, the system would be simple and easy to use. However, a complete new blockchain is not needed. The core use of Blockchain in this paper is to provide trust by store some information into blockchain. As a result, we can build an application layer on one of the existing blockchain platform (e.g., Bitcoin or Ethernum), which stores information into the third-party blockchain, and uses this information to provide crowd-intelligence service.

IV. HYBRID HUMAN–MACHINE WORKERS

A crowd-intelligence platform needs to attract machines like sensors and processors in addition to human beings as its workers, and develop itself into a hybrid human–machine ecosystem as follows:

1) Integrating advantages both of human beings and machines. Specifically, the crowd-intelligence platform can assign human workers more highly complex crowdsourcing tasks, e.g., labeling of training dataset in machine learning [3], [21]; and assign machine workers more realtime frequent crowdsensing tasks, e.g., mobile crowdsensing in IIoT [4] and automobile navigation [10]). The proposed platform is able to cover both crowdsourcing and crowdsensing.
2) Relieving the worker shortage problem and reducing excessive latency of the whole tasks. As more workers can complete more tasks within the same time period, excessive latency of the whole tasks is hard to occur if only human workers are used.

The hybrid human–machine crowd intelligence is shown in Fig. 4.

The enabling technology of the proposed hybrid human–machine crowd-intelligence platform is blockchain smart contract. The blockchain smart contract is responsible to guide and monitor the behaviors of workers, and prevent frauds among them. So from a management perspective of the decentralized crowd-intelligence platform, human beings and machines are just indiscriminate workers represented by SPV nodes. All kinds of computing devices, especially AI devices like inhome routers, monitoring camera, and mobile phones at edge network can work as machine workers. With the blockchain smart contract, these machine workers can use the earnings to pay for their continued existence, e.g., hardware resources, access to more useful information, and software improvements in their whole life cycle without human intervention.

V. Experiments

A. Experimental Settings

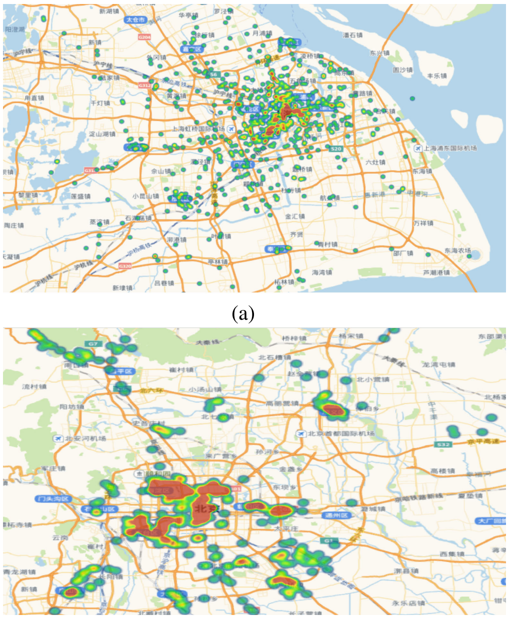
We use **BeTrustMEC** to represent the proposed blockchain-enabled trustless crowd-intelligence ecosystem. There are two baselines, namely **Major-CI** and **ReNalty-CI** as follows:

1) *Major-CI:* It is widely used in current crowdsourcing platforms like MTurk[2] and FigureEight,[3] where the type selected by the largest number of workers is determined as the final answer. Major-CI is totally centralized in that data transferring between a publisher and workers must go through the remote platform center on an RC.
2) *ReNalty-CI:* The only difference of ReNalty-CI from Major-CI is that it uses the reward-penalty model to align incentives of stakeholders and obtain the final answer. ReNalty-CI gives publishers and workers more rights to influence the ecosystem. Like Major-CI, publishers and workers of ReNalty-CI have no mutual trust.

As no specific off-the-shelf dataset exists, we choose to generate two pieces of synthesized datasets to validate the proposed ecosystem. Fig. 5(a) shows the spatial distribution of 3234 cellular base stations in Shanghai, China. The cellular distribution is unbalanced and is consistent with the distribution of crowd-intelligence tasks. In experiments of two baselines, the RC is located at the same location, where the cellular base station distribution has the highest density in Fig. 5(a). Another dataset is as shown in Fig. 5(b), which shows the distribution of 967 cellular base stations in Beijing, China.

Based on the base station distribution data, the three stakeholders (i.e., publisher, worker, and ES/MN) and crowdsourcing tasks for the proposed BeTrustMEC are generated according to the following ways:

1) We normalize the distances between two base stations to range (0, 1], and consider the normalized distances as the corresponding network latency between the two locations.
2) Five base stations are randomly selected as publishers, and each has 2000 tasks to publish.
3) Hundred workers are selected from remaining base stations to complete these tasks.
4) Twenty ES/MNs are randomly selected from the rest base stations.


(a)

(b)

Fig. 5. Spatial distribution heatmap of cellular base stations. (a) Shanghai city of China. (b) Beijing city of China.

In the simulation of BeTrustMEC, when a worker receives data from or submits data to a publisher, they will choose the ES/MN with least sum of distances to them to transfer the data without trust. In this way, a decentralized crowd-intelligence ecosystem that covers the whole city is constructed.

Different from the proposed BeTrustMEC, baselines Major-CI and ReNalty-CI are essentially traditional centralized crowdsourcing systems, which have only one remote crowdsourcing platform on an RC instead of decentralized ES/MNs. The location of the remote platform is determined each time according to the following ways:

1) We compute the sum distance of each base station to the rest base stations, and select the least 20 of them as the set of candidate platform locations. This is quite different from site selection in practical cloud computing industry, where a data center is always located far from downtown due to operating costs [35]. We make this change here to make the baselines to generate the best performance, especially in latency time of completing the whole tasks, which in turn validates the effectiveness of the proposed BeTrustMEC in the subsequent experimental results.
2) We select one from the set of candidate platform locations as the location of the platform each time. This step adds uncertainty to the location selection, which is just like the selection of the set of ES/MNs for BeTrustMEC.

Now we introduce the way to generate crowd-intelligence tasks and commits of workers. A task is generated as follows: 1) the real type value y_0 of a task takes $+1/-1$ at the same probability; 2) the real belief value c_0 of the tasks is uniformly distributed in range (0.5, 1]. A worker is supposed to have a bias value $b \in [-0.5, 0.5]$ toward c_0, and her commit for a task with c_0 is generated as follows:

[2] https://www.mturk.com/

[3] https://www.figure-eight.com/

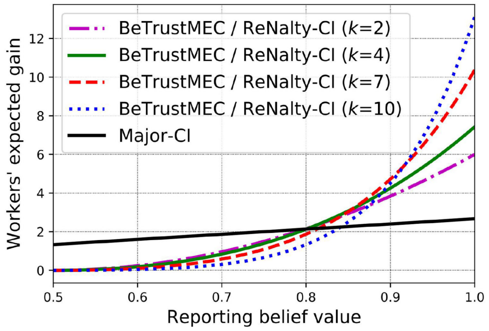
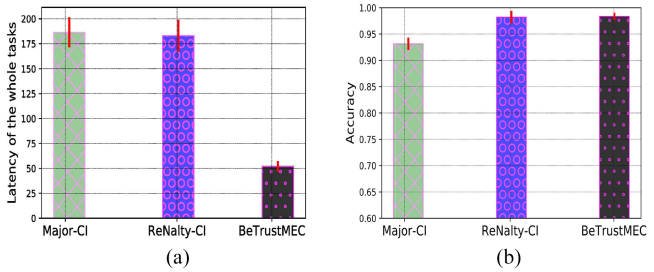
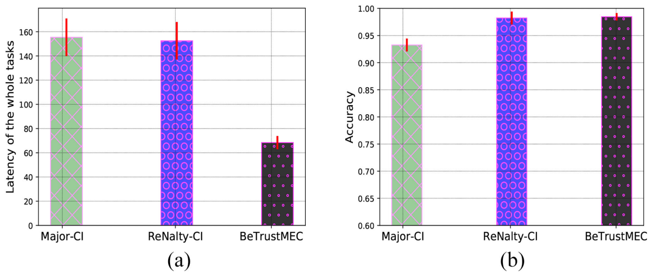


Fig. 6. Workers' expected gain for different belief values.



(a) (b)

Fig. 7. Experimental results on data of Shanghai base stations. (a) Performance on latency time of the whole tasks. (b) Performance on accuracy of the final answers.



(a) (b)

Fig. 8. Experimental results on data of Beijing base stations. (a) Performance on latency time of the whole tasks. (b) Performance on accuracy of the final answers.

1) If $c_0 + b < 0.5$, she commits the wrong type with belief value $1 - c_0 - b$.
2) If $c_0 + b > 1$, she commits the right type with belief value 1.
3) Else, she commits the right type with belief value $c_0 + b$.

In addition, a worker cannot process two tasks at a time and a task cannot be processed more than once by one worker. When a task obtains three commits, its benchmark and final answers are generated and this task is considered complete.

B. Experimental Results

Fig. 6 shows curves of the expected gain payment that workers gain with different belief values of commits. As both of ReNalty and the proposed BeTrustMEC use the reward-penalty model, they are represented by the same curves. The parameter k means workers' *personal order value*. Belief value is not needed in Major-CI. However, it still affects the accuracy of a worker's committing answer and her expected gain. A larger committing belief value can gain more in BeTrustMEC or ReNalty-CI than Major-CI, while a less committing belief results in the opposite. As personal order increases, this trend becomes more evident. So the ReNalty and the proposed BeTrustMEC can attract more professional workers.

Experimental results on data of Shanghai base stations is shown in Fig. 7. Fig. 7(a) shows that BeTrustMEC performs much better than Major-CI and ReNalty-CI on latency control, which means that tasks published by BeTrustMEC can be completed in much less time. The reason is that BeTrustMEC is a decentralized ecosystem where workers and publishers can transfer data using the nearby ES/MNs instead of RC center. What is more, BeTrustMEC has lower variance value than baselines, which means its decentralization helps to suppress latency fluctuation. Fig. 7(b) shows that BeTrustMEC performs better than Major-CI in accuracy of final answers. ReNalty-CI has similar accuracy to BeTrustMEC, which is because of the same reward-penalty model is used. But ReNalty-CI has much worse variance, which may result from the location uncertainty of the remote platform.

Experimental results on data of Beijing base stations are shown in Fig. 8. The results are very similar to that in Fig. 7. Fig. 8(a) shows that the proposed BeTrust-MEC performs much better than both of Major-CI and ReNalty-CI on latency control. While in respect to accuracy of final answers, Fig. 8(b) shows that BeTrustMEC performs better than Major-CI, and similar to ReNalty-CI.

As we can see from the experimental results, BeTrustMEC performs better than the two baselines in general, which together with the previous theoretical analysis helps to validate the effectiveness of the proposed BeTrustMEC.

VI. Related Works

Past works cannot fit the decentralization features of mobile edge computing, and meet the differentiated demands of a crowd-intelligence ecosystem [1], [3], [8], [10], [17]. For example, workers' reliability must be inferred from workers' history records [1], [3], and at the same time much money is wasted for bad workers and no penalty for their bad affects as compensation is paid to the task publisher [10], [17]. The reward-penalty model gives a much more flexible way to manage the interests of three stakeholders. The incentive system in crowd-intelligence platform, like the reward-penalty model, is complex enough, and easy to be tampered by bad minorities, which results in trust problem.

In this paper, the trust problem among crowd-intelligence stakeholders is solved by integrating mobile edge computing and blockchain smart contract together. The past works mainly focus on how to identify the trustworthiness of workers or their commits [1], [3], [10], [17], but ignore how to ensure the trustworthness of publishers and the crowd-intelligence platform, which is unfair to the workers. We are the first to discussed the trust problem among the ES/MNs of the decentralized crowd-intelligence platform based on blockchain smart contract. Satyanarayanan *et al.* [31] mentioned the relationship between ES and

RC, but not studied how ES compete with RC in providing edge services, and how they can collaborate with each other without mutual trust.

The past crowdsourcing [21] mainly focuses on dealing with highly complex tasks (e.g., labeling of training dataset in machine learning [3]) using human beings workers, while crowdsensing mainly deals with real-time frequent tasks using sensors or other computing devices. As crowdsourcing and crowdsensing use quite different workers, it is not easy to integrate the two in order to obtain the advantages of both. While by using blockchain smart contract, the proposed hybrid human–machine platform can consider human beings and machines as indiscriminate workers, which can expand the applications of crowd intelligence to crowdsourcing and crowdsensing. As human beings and machine workers can complement each other, the hybrid human–machine platform can help to reduce excessive latency of the whole tasks.

VII. Conclusion

This paper presents a trustless crowd-intelligence ecosystem based on the common decentralization feature of mobile edge computing and blockchain technology. Its reward-penalty model provides a flexible way to align the interests of three stakeholders. It expands the applications of crowd intelligence to crowdsourcing and crowdsensing domains. What is more, it can make use both of human beings and machines as workers to solve the worker shortage problem. It can offer many data-related solutions to IIoT, and benefit from the IIoT's infrastructures at the same time. For future work, we plan to 1) implement the proposed ecosystem using blockchain technology and deploy it in a real mobile edge computing network for more test and promotion; and 2) further make it work within limited total budget.

References

[1] B. C. Ooi *et al.*, "Contextual crowd intelligence," *ACM SIGKDD Explor. Newslett.*, vol. 16, no. 1, pp. 39–46, 2014.

[2] B. Guo, Q. Han, H. Chen, L. Shangguan, Z. Zhou, and Z. Yu, "The emergence of visual crowdsensing: Challenges and opportunities," *IEEE Commun. Surv. Tut.*, vol. 19, no. 4, pp. 2526–2543, Oct.–Dec. 2017.

[3] R. Jurca and B. Faltings, "Error rate analysis of 1abeling by crowdsourcing," in *Proc. 30th Int. Conf. Mach. Learn. Workshop*, 2013, pp. 1–8.

[4] K. Zhang and A. Marchiori, "Crowdsourcing low-power wide-area IoT networks," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2017, pp. 41–49.

[5] J. Fernandes *et al.*, "Iot lab: Towards co-design and IoT solution testing using the crowd," in *Proc. IEEE Int. Conf. Recent Adv. Internet Things*, 2015, pp. 1–6.

[6] Y. Gao, Y. Chen, and K. J. R. Liu, "On cost-effective incentive mechanism in micro-task crowdsourcing," *IEEE Trans. Comput. Intel1. AI Games*, vol. 7, no. 1, pp. 3–15, Mar. 2015.

[7] J. Ren, Y. Zhang, K. Zhang, and X. Shen, "Exploiting mobile crowdsourcing for pervasive cloud services: Challenges and solutions," *IEEE Commun. Mag.*, vol. 53, no. 3, pp. 98–105, Mar. 2015.

[8] J. Xu, S. Wang, N. Zhang, F. Yang, and X. S. Shen, "Reward or penalty: Aligning incentives of stakeholders in crowdsourcing," *IEEE Trans. Mobile Comput.*, pp. 1–13, 2018, doi: 10.1109/TMC.2018.2847350.

[9] J. Jiang, G. Han, L. Shu, S. Chan, and K. Wang, "A trust model based on cloud theory in underwater acoustic sensor networks," *IEEE Trans. Ind. Inform.*, vol. 13, no. 1, pp. 342–350, Feb. 2017.

[10] X. Fan, J. Liu, Z. Wang, Y. Jiang, and X. Liu, "Crowdsourced road navigation: Concept, design, and implementation," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 126–128, Jun. 2017.

[11] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 17–27, Mar. 2017.

[12] P. Lindgren, J. Eriksson, M. Lindner, A. Lindner, D. Pereira, and L. M. Pinho, "End-to-end response time of IEC 61499 distributed applications over switched ethernet," *IEEE Trans. Ind. Inform.*, vol. 13, no. 1, pp. 287–297, Feb. 2017.

[13] A. U. R. Khan, M. Othman, S. A. Madani, and S. U. Khan, "A survey of mobile cloud computing application models," *IEEE Commun. Surv. Tut.*, vol. 16, no. 1, pp. 393–413, Jan.–Mar. 2014.

[14] J. Kang *et al.*, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet Things J.*, 2018, to be published, doi: 10.1109/JIOT.2018.2875542.

[15] H. Liu, Y. Zhang, and T. Yang, "Blockchain-enabled security in electric vehicles cloud and edge computing," *IEEE Netw.*, vol. 32, no. 3, pp. 78–83, May/Jun. 2018.

[16] M. Satyanarayanan, G. Lewis, E. Morris, S. Simanta, J. Bo1eng, and K. Ha, "The role of cloudlets in hostile environments," *IEEE Pervasive Comput.*, vol. 12, no. 4, pp. 41–49, Oct.–Dec. 2013.

[17] G. Radanovic and B. Fa1tings, "A robust Bayesian truth serum for non-binary signal," in *Proc. 27th AAAI Conf. Artif. Intell.*, 2013, pp. 833–839.

[18] H.-W. Kim and Y.-S. Jeong, "Secure authentication-management human-centric scheme for trusting personal resource information on mobile cloud computing with blockchain," *Human-Centric Comput. Inf. Sci.*, vol. 8, no. 1, 2018, Art. no. 12.

[19] S. Underwood, "Blockchain beyond Bitcoin," *Commun. ACM*, vol. 59, no. 12, pp. 15–17, 2016.

[20] M. Shubik, "Game theory, behavior, and the paradox of the prisoner's dilemma: Three solutions," *J. Conflict Resolution*, vol. 14, no. 2, pp. 181–193, 1970.

[21] A. I. Chittilappilly, L. Chen, and S. Amer-Yahia, "A survey of general-purpose crowdsourcing techniques," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 9, pp. 2245–2266, Sep. 2016.

[22] J. Witkowski and D. C. Parkes, "Peer prediction without a common prior," in *Proc. l3th ACM Conf. E1ectron. Commerce*, 2012, pp. 963–981.

[23] C. Vullioud, F. Clément, T. Scott-Phillips, and H. Mercier, "Confidence as an expression of commitment: Why misplaced expressions of confidence backfire," *Evol. Human Behav.*, vol. 38, no. 1, pp. 1–36, 2016.

[24] G. Radanovic and B. Fa1tings, "Incentives for subjective evaluations with private beliefs," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 1014–1020.

[25] J. Witkowski and S. Seuken, "Incentive-compatib1e escrow mechanisms," in *Proc. 25th AAAI Conf. Artif. Intell.*, 2011, pp. 751–757.

[26] N. B. Shah and D. Zhou, "Doub1e or nothing: Multiplicative incentive mechanisms for crowdsourcing," in *Proc. 28th Adv. Neural Inf. Process. Syst.*, 2015, pp. 1–9.

[27] S. Liang and J. Zhang, "Positive solutions for boundary value problems of nonlinear fractional differential equation," *Nonlinear Anal., Theory, Methods Appl.*, vol. 71, no. 11, pp. 5545–5550, 2009.

[28] D. Berend and A. Kontorovch, "Consistency of weighted majority votes," in *Proc. 24th Adv. Neura1 Inf. Process. Syst.*, 2014, pp. 3446–3454.

[29] S. Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System*, 2008, pp. 1–9. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[30] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Commun. Surv. Tut.*, vol. 18, no. 3, pp. 2084–2123, Jul.–Sep. 2016.

[31] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct–Dec. 2009.

[32] Z. Li, J. Kang, R. Yu, D. Ye, Q. Deng, and Y. Zhang, "Consortium blockchain for secure energy trading in industrial Internet of Things," *IEEE Trans. Ind. Inform.*, vol. 14, no. 8, pp. 3690–3700, Aug. 2018.

[33] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains," *IEEE Trans. Ind. Inform.*, vol. 13, no. 6, pp. 3154–3164, Dec. 2017.

[34] M. Patel *et al.*, "Mobile-edge comput. introductory technical white paper," Mobile-edge Computing Industry Initiative White Paper, vol. 1, no. 1, pp. 1–36, 2014.

[35] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: Research problems in data center networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 1, pp. 68–73, 2008.

Jinliang Xu received the bachelor's degree in electronic information science and technology from Beijing University of Posts and Telecommunications, Beijing, China, in 2014. He is currently working toward the Ph.D. degree in computer science at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications.

His research interests include mobile cloud computing, blockchain, AI, and crowdsourcing.

Shangguang Wang (SM'11) received the Ph.D. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2011.

He is Professor and Vice-Director with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. He has authored and coauthored more than 150 papers recent years, and played a key role at many international conferences, such as general chair and PC chair. His research interests include service computing, cloud computing, and mobile edge computing.

Prof. Wang is a senior member of the IEEE, and Editor-in-Chief of the *International Journal of Web Science*.

Bharat K. Bhargava (F'93) received the Ph.D. degree from Purdue University, West Lafayette, IN, USA, in 1974. He is currently a Professor of computer science with Purdue University, West Lafayette, IN, USA.

Prof. Bhargava is the Editor-in-Chief for four journals and serves on over ten editorial boards of international journals. He is the founder of the IEEE Symposium on Reliable and Distributed Systems, IEEE conference on Digital Library, and the ACM Conference on Information and Knowledge Management. He has authored/coauthored hundreds of research papers and has won five best paper awards in addition to the technical achievement award and golden core award from IEEE.

Fangchun Yang (SM'94) received the Ph.D. degree in communication and electronic system from the Beijing University of Posts and Telecommunication, Beijing, China, in 1990.

He is currently a Professor with the Beijing University of Posts and Telecommunication, China. His research interests include network intelligence and communications software.

Prof. Yang is a Fellow of the IET.