Solving the Crowdsourcing Dilemma Using the Zero-Determinant Strategies

Qin Hu, Shengling Wang, *Member, IEEE*, Xiuzhen Cheng, *Fellow, IEEE*, Liran Ma, *Member, IEEE*, and Rongfang Bie, *Member, IEEE*

***Abstract*— Crowdsourcing is a promising technology to accomplish a complex task via eliciting services from a large group of contributors. Recent observations indicate that the success of crowdsourcing has been threatened by the malicious behaviors of the contributors. In this paper, we analyze the attack problem using an iterated prisoner's dilemma (IPD) game and propose a reward-penalty expected payoff algorithm based on zero-determinant (ZD) strategies to reward a worker's cooperation or penalize its defection in order to entice the final cooperation. Both theoretical analysis and simulation studies are performed, and the results indicate that the proposed algorithm has the following two attractive characteristics: 1) the requestor can incentivize the worker to become cooperative without any long-term extra cost; and 2) the proposed algorithm is fair so that the requestor cannot arbitrarily penalize an innocent worker to increase its payoff even though it can dominate the game. To the best of our knowledge, we are the first to adopt the ZD strategies to stimulate both players to cooperate in an IPD game. Moreover, our proposed algorithm is not restricted to solve only the problem of crowdsourcing dilemma - it can be employed to tackle any problem that can be formulated into an IPD game.**

***Index Terms*— Crowdsourcing, malicious attack, game theory, zero-determinant strategies.**

I. Introduction

CROWDSOURCING is a promising technology that takes advantage of contributions from a large group of participants to complete a relatively complicated task [1]–[3]. It has attracted much attention recently, and its success mainly comes from its openness to the crowds. Nevertheless, the prevailing application of crowdsourcing is severely hindered by its susceptibility to malicious actions of the workers. More specifically, greedy workers may inaugurate attacks to the peering non-greedy ones or even the requestor to get more profit by performing malicious actions, such as fabrication and plagiarism, leading to severe damage for task completion. For example, it is well-known that the UCSD team failed to keep the first rank to the end in the DARPA Shredder Challenge 2011 [4] due to the sabotages of the malicious workers during the crowdsourcing process.

Considering that malicious attacks hamper the wide application of crowdsourcing, researchers have made a considerable amount of effort to address this challenge in recent years. For examples, the practicability of launching individual and group attacks in crowdsourcing was analyzed in [5]; Naroditskiy *et al.* in [6], [7] first adopted the prisoner's dilemma (PD) game to model the malicious behaviors between any two workers, and then employed the iterated prisoner's dilemma (IPD) game to formulate and analyze an iterated version of the problem. Nevertheless, none of the works mentioned above proposed any effective countermeasures to address the attack problem. Though the authors in [8]–[11] put forward mechanisms to detect the malicious behaviors in various crowdsourcing applications, those schemes are not systemic or efficient because they function as additional components of the counterattacks.

In this paper, we employ an IPD game to analyze and eliminate the malicious attacks in crowdsourcing. Different from the existing work [6], [7], we consider the interactions between the requestor (i.e., the crowdsourcer) and any worker (i.e., the contributor). This distinct perspective of our analysis is inspired by the facts that the requestor in a crowdsourcing application acts as an employer, and is certainly a victim of any sort of malicious attack in the meanwhile. As a victim, the requestor may receive low-quality results or may need to spend more time and money to get the final submission, thus it has sufficient motivation to get rid of the malicious behaviors; while as an employer who pays remuneration, the requestor is capable of removing the malicious behaviors with the help of market power.

However, it is nontrivial to utilize an IPD game to address the malicious attack problem in crowdsourcing. We summarize two crucial problems as follows, which need to be gradually solved in order to analyze and eliminate the malicious behaviors via an IPD game.

Manuscript received November 20, 2018; revised May 6, 2019; accepted October 16, 2019. Date of publication October 24, 2019; date of current version January 27, 2020. This work was supported in part by the U.S. National Science Foundation under Grant IIS-1741279 and Grant CNS-1704397 and in part by the National Natural Science Foundation of China under Grant 61772080 and Grant 61571049. The work of L. Ma was supported in part by the U.S. National Science Foundation under Grant OAC-1829553 and Grant CNS-1912755. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Shantanu D. Rane. *(Corresponding author: Shengling Wang.)*

Q. Hu is with the College of Information Science and Technology, Beijing Normal University, Beijing 100875, China, and also with the Department of Computer Science, The George Washington University, Washington, DC 20052 USA (e-mail: qinhu@gwu.edu).

S. Wang and R. Bie are with the College of Information Science and Technology, Beijing Normal University, Beijing 100875, China (e-mail: wangshengling@bnu.edu.cn; rfbie@bnu.edu.cn).

X. Cheng is with the Department of Computer Science, The George Washington University, Washington, DC 20052 USA (e-mail: cheng@gwu.edu).

L. Ma is with the Department of Computer Science, Texas Christian University, Fort Worth, TX 76129 USA (e-mail: l.ma@tcu.edu).

Digital Object Identifier 10.1109/TIFS.2019.2949440

1) Can we design an algorithm that can encourage the worker to be finally cooperative via offering the worker more short-term payoff without sacrificing the long-term interest of the requestor?
2) When designing the algorithm mentioned above, can we guarantee its fairness to both the requestor and the worker such that the requestor cannot arbitrarily punish the worker if it dominates the game?

Through answering the above two questions one by one, we make the following contributions (a preliminary version of this paper is presented in [12]):

- The malicious attack problem in crowdsourcing is formulated by an IPD game between two heterogeneous players, i.e., the requestor and the worker.
- Four classic strategies are inspected theoretically and experimentally so as to explore the most reasonable one that could be employed by the worker, which turns out to be the evolutionary strategy.
- For the first problem, we propose a reward-penalty expected payoff algorithm based on zero-determinant (ZD) strategies for the requestor. Both the experimental results and the theoretical analysis on this algorithm demonstrate that i) it can drive the worker to be finally cooperative through a higher short-term payoff but with no long-term extra cost from the requestor; and ii) it is lively in the long run since it is impartial for both the requestor and the worker. As a result, the second problem is also resolved.
- Our proposed algorithm takes advantage of the ZD strategies to drive both players to be finally cooperative in an IPD game. We claim that this algorithm is not restricted to solve only the crowdsourcing dilemma problem; it is also applicable to other iterated two-player-game situations.

The rest of the paper is organized as follows. Section II formulates the attack problem in crowdsourcing with the iterated prisoner's dilemma and identifies the conditions for the game to be valid. Trial studies on the worker's possible strategies are presented in Section III with both theoretical analysis and experimental studies. The reward-penalty expected payoff algorithm based on the ZD strategies is proposed in Section IV and analyzed in Section V. In Section VI, we summarize the most related work; and the paper is concluded in Section VII.

II. Game Model

In this paper, we consider a crowdsourcing scenario where a requestor launches a crowdsourcing request, provides several payment offers, and then recruits workers from a large crowd. This step may be carried out on a crowdsourcing platform such as Amazon Mechanical Turk [13]. Next all recruited workers try their best to complete their jobs and obtain the corresponding payments. We define such a process involving one interaction of the requestor and the workers as a *round* of crowdsourcing. In reality, a worker may attack others such as its peers and the requestor to be more profitable at each round, which is assumed to be hard for the requestor to timely detect due to the lack of evidence or limited computing resource. The requestor may behave maliciously too, by paying the workers with lower payments than they deserve at a round, without knowing whether or not the workers are honest. By this way, the requestor and the workers take actions simultaneously without knowing the exact action of the opponent at each round of the crowdsourcing. We assume that workers in this scenario make decisions independently to play against the requestor, which will never be affected by other peering workers.

Requestor \ Worker	Cooperation	Defection
Cooperation	R_r, R_w	$R_r - m$, $R_w + b$
Defection	$R_r + n$, $R_w - a$	$R_r - m + n$, $R_w + b - a$

Fig. 1. Payoff matrix of the requestor and the worker.

Obviously, an interaction between the requestor and any recruited worker can be modeled as a simultaneous game; when the same worker is employed repeatedly, the game becomes iterated. Here we define that the action of a worker is *cooperation* when he[1] completes the job without any attack, or *defection* when he attacks; and *cooperation* of the requestor implies that she pays the worker a normal payoff without any economic punishment while *defection* means giving a lower payment. At each round, the exact action that one player adopts cannot be known by its opponent. Hence, four combinations of the actions can be derived as $rw = (cc, cd, dc, dd)$, where r and w represent the actions of the requestor and the worker, respectively, while c and d denote cooperation and defection, respectively. Note that since the actions of both players at each round uniquely denote the game at that round, we regard the union of the actions as the state of the game at that round.

Then the payoffs of both the requestor and the worker in each state can be calculated as follows: 1) when both players choose cooperation, they can get their normal payoffs of R_r and R_w; 2) when the requestor adopts cooperation while the worker chooses defection, the defective worker can get more profit b than his normal payoff R_w, but the requestor suffers a loss of m on her normal payoff R_r; 3) when the requestor turns to defection while the worker adopts cooperation, the innocent worker is paid with an economic penalty and receives a payoff of $R_w - a$, where a indicates a direct economic loss as well as other indirect loss such as reputation damage; while the requestor gains more profit n; 4) when both players select defection, the requestor obtains a payoff of $R_r - m + n$ and the worker gets R_w+b-a. Thus, as shown in Fig. 1, the payoff vector of the requestor is $\mathbf{S}_r = (R_r, R_r-m, R_r+n, R_r-m+n)$, while that of the worker is $\mathbf{S}_w = (R_w, R_w + b, R_w - a, R_w + b - a)$, corresponding to the game states $rw = (cc, cd, dc, dd)$. Note that all parameters here are positive.

Next, we introduce two theorems to justify why the proposed game model is a PD game in a single round, and an IPD game when multiple rounds are considered.

Theorem 1: When the parameters in the payoff matrix satisfy $n < m$, $b < a$, $n < a$, and $b < m$, the interaction between

[1] We denote the worker as "he" and the requestor as "she" for differentiation.

the requestor and the worker in a round can be modeled by a PD game.

Proof: Generally speaking, two conditions need to be satisfied in a PD game: 1) defection is the dominant strategy for each player while mutual cooperation brings a higher payoff than mutual defection; and 2) mutual cooperation should be superior to all other situations with respect to the overall payoff.

The first condition indicates that 1) whether c or d is chosen by its opponent, a player adopting d can earn a higher payoff than adopting c; and 2) the payoff of each player in state cc is always larger than that in dd. Hence, the requestor's payoff in state $rw = dc$ is higher than that in state $rw = cc$, which is larger than her payoff in state $rw = dd$, and all of them are greater than that in state $rw = cd$. Thus, for the requestor's payoffs, we have

$$R_r + n > R_r > R_r - m + n > R_r - m,$$

which obviously hold when $n < m$.

Similarly, when $b < a$, the payoffs of the worker satisfy:

$$R_w + b > R_w > R_w + b - a > R_w - a.$$

The second condition implies that the following three relationships hold for the payoffs of the requestor and the worker:

$$\begin{aligned} R_r + R_w &> (R_r - m) + (R_w + b), \\ R_r + R_w &> (R_r + n) + (R_w - a), \\ R_r + R_w &> (R_r - m + n) + (R_w + b - a). \end{aligned}$$

It is obvious that they can be satisfied when the constraints $n < a$ and $b < m$ hold.

From an individual perspective, defection is the most rational choice since it can bring more benefit than cooperation for any player, leading to the Nash equilibrium of the crowdsourcing game as mutual defection; while from a global perspective, mutual cooperation brings the highest sum of the payoffs, i.e., social welfare, and thus is defined as the global optimal state. Therefore, the game between the requestor and the worker becomes a PD when $n < m$, $b < a$, $n < a$, and $b < m$ hold. □

As mentioned above, the same worker can be recruited repeatedly by the same requestor to complete continuous jobs such as monitoring the traffic condition in a fixed location. In this case, the game becomes iterated when the constraints in the following theorem are met.

Theorem 2: When the parameters in the payoff matrix conform to $n < m$, $b < a$, $n < a$, and $b < m$, the interactions between the requestor and the worker in multiple rounds can be formulated into an IPD game.

Proof: Besides the two necessary conditions of the PD game, an IPD game needs to avoid the case that a player alternating between cooperation and defection results in more payoff than persisting in cooperation. Thus, the payoff vector of the requestor should satisfy the following inequality:

$$2R_r > (R_r + n) + (R_r - m).$$

It is clear that given $n < m$, the above relationship holds.

Similarly, for the worker, his payoff vector should meet:

$$2R_w > (R_w + b) + (R_w - a),$$

which holds when $b < a$.

Hence, interactions between the requestor and the worker constitute an IPD when $n < m$, $b < a$, $n < a$, $b < m$. □

Actually, all parameter constraints mentioned above have realistic implications. First, $n < m$ and $b < a$ imply that both players earn lower payoffs in mutual defection than those in mutual cooperation. It is totally consistent with reality since the defection of the worker can cause a higher cost of the requestor to complete the whole crowdsourcing job, while the defection of the requestor can bring a lower profit for the worker. Besides, the constraint $n < a$ indicates that the increased amount of payoff received by the requestor, which is resulted from the economic punishment on the worker, is less than the payoff decrement of the worker. This holds true in reality since the requestor's punishment on the worker can bring to him not only the direct economic loss but also indirect cost such as reputation damage. While $b < m$ reveals that the worker's defection behavior produces a payoff increment which is less than the requestor's payoff decrement. This inequality also holds true in practice because a greedy worker can easily obtain more profit with a fake submission without consuming too much energy, but it costs the requestor lots of resources to filter out or fix those intentional errors.

However, it is worth to note that the following study in this paper is not restricted to the scenarios with $n < a$ and $b < m$, but applicable to other cases as long as the general conditions hold, i.e., $n < m$ and $b < a$. If neither of the general conditions is met, the Nash equilibrium state will be coincident with the global optimal state as mutual defection, and then the problem of eliciting mutual cooperation vanishes because there is no motivation for any player to choose cooperation.

III. Strategies for the Worker

A. Strategies for an Evolutionary Player

Intuitively, a worker tends to employ a strategy that can maximize his payoff throughout the game. While in practice, the worker is at a disadvantageous position because he has limited knowledge about the strategies of the requestor and his peers. On the contrary, the requestor stays in an advantageous situation thanks to her global view about the strategies of all workers, which can help her to better estimate a worker's behavior. To handle this situation, a rational worker would like to adaptively search for the optimal strategy, which refers to the cooperation probability in our study.

This sort of optimization seeking method relies on the self-adaptiveness that is similar to the main idea of biological evolution, i.e., "natural selection" and "survival of the fittest". By this means, the global optimal solution is explored in a stochastic or metaheuristic way. Motivated by this observation, we give a loose definition of the evolutionary strategy.

Definition 1 (Evolutionary Strategy): A game player adopts an evolutionary strategy when s/he adjusts it to maximize the payoff without considering the strategy or payoff of the opponent.

In light of the above definition, we present two specific examples of the evolutionary strategy.

1) Example 1: Inspired by [14], we present the first type of evolutionary strategy, denoted as E strategy. Let q_w^t be the cooperation probability of the worker at round t. If the worker adopts the E strategy, the cooperation probability q_w^t should evolve according to

$$q_w^{t+1} = q_w^t \frac{W_c^t}{E_w^t}, \tag{1}$$

where W_c^t is the worker's expected cooperation payoff and E_w^t is his expected total payoff at round t. W_c^t can be calculated as:

$$W_c^t = p_r^t E(cc) + (1 - p_r^t) E(cd), \tag{2}$$

where p_r^t is the cooperation probability of the requestor at round t, $E(cc)$ is the payoff of the worker in state $wr = cc$ and $E(cd)$ is that in state $wr = cd$. Similarly, $E(dc)$ and $E(dd)$ are the payoffs of the worker in states $wr = dc$ and $wr = dd$, respectively. Thus, the expected defection payoff of the worker at round t, denoted as W_d^t, can be calculated likewise:

$$W_d^t = p_r^t E(dc) + (1 - p_r^t) E(dd). \tag{3}$$

Now, for the expected total payoff at round t, E_w^t in (1) can be derived by:

$$E_w^t = q_w^t W_c^t + (1 - q_w^t) W_d^t. \tag{4}$$

According to the above equation, one can see that the numerator in (1), $q_w^t W_c^t$, is always a part of the denominator, i.e., E_w^t. Therefore, $q_w^t \in [0, 1]$ holds for all t. Besides, comparing (2)(3)(4), one can notice that when $W_c^t > E_w^t$, the cooperation probability of the worker increases in the next round; otherwise it keeps unchanging or decreases. This kind of evolution path can help the worker to find the optimal response strategy (i.e., cooperation probability q_w^t) according to the fitness W_c^t / E_w^t, so as to play against the requestor.

2) Example 2: According to [15], one can utilize the following equation to denote another type of evolutionary strategy, denoted as E′ strategy. With the same notation q_w^t mentioned above, when a worker adopts the E′ strategy, his cooperation probability at round t is determined by:

$$q_w^t = \frac{e^{A_c^t - A_d^t}}{1 + e^{A_c^t - A_d^t}}. \tag{5}$$

In (5), A_c^t is the worker's cumulative expected cooperation payoff until the t-th round and can be calculated by $A_c^t = \sum_{\tau=0}^{t} W_c^\tau$, where W_c^τ is his expected cooperation payoff at round τ and can be derived according to (2); while A_d^t is the worker's cumulative expected defection payoff until round t, calculated by $A_d^t = \sum_{\tau=0}^{t} W_d^\tau$, where W_d^τ is the expected defection payoff at round τ and calculated according to (3).

Thus, when adopting the E′ strategy, the worker determines his cooperation probability at round t according to the difference of his cumulative expected payoffs for cooperation and defection in all previous rounds of the game. If his cumulative expected payoff of cooperation is much greater than that of defection, $e^{A_c^t - A_d^t}$ is much greater than 1 and thus q_w^t is close to 1 from the left, which means that the worker cooperates at round t with a high probability. On the contrary, if A_c^t is much less than A_d^t, $e^{A_c^t - A_d^t}$ is close to 0 and q_w^t also approaches 0, which indicates that the worker defects at round t very likely.

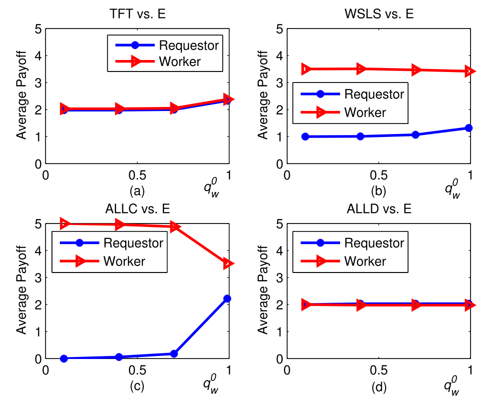

Fig. 2. Average payoffs of the requestor and the worker when the requestor adopts classic strategies and the worker adopts the E strategy.

B. Experimental Evaluation on the Evolutionary Strategies

In this section, we carry out a few experimental studies to testify the potential of the evolutionary strategies. More specifically, we simulate the game process when the worker adopts an evolutionary strategy, i.e., either the E strategy or the E′ strategy, while the requestor utilizes one of the following classic strategies: tit-for-tat (TFT), win-stay-lose-shift (WSLS), all-cooperation (ALLC), and all-defection (ALLD). Basically, when the requestor adopts the TFT strategy, she cooperates in the first round and then chooses the previous action of the opponent (worker) in each of the following rounds; when the requestor is a WSLS player, she does not change her action if and only if this action brought her a high payoff in the previous round; if the requestor chooses ALLC, it remains cooperative throughout the game regardless of the worker's action; while adopting ALLD strategy indicates that the requestor chooses defection unconditionally no matter what action the worker adopts.

We set the parameters in the payoff matrix as $R_w = R_r = 3$, $a = m = 3$, $b = n = 2$ in these experiments. Accordingly, the payoff vectors of the requestor and the worker are $\mathbf{S}_r = (3, 0, 5, 2)$ and $\mathbf{S}_w = (3, 5, 0, 2)$, respectively. Each experiment runs 100 times to get the average value for eliminating the statistical uncertainties. Note that we also conduct simulations with other parameter settings and obtain very similar results.

The average payoffs of the requestor and that of the worker *vs.* the worker's initial cooperation probability q_w^0 when the worker adopts the E strategy while the requestor adopts different classic strategies are plotted in Fig. 2. It is easy to see that the average payoff of the worker is always larger than or equal to that of the requestor under the four game scenarios. Particularly, the worker yields a higher payoff when the requestor adopts WSLS and ALLC, while gets similar payoffs when the requestor adopts TFT and ALLD.

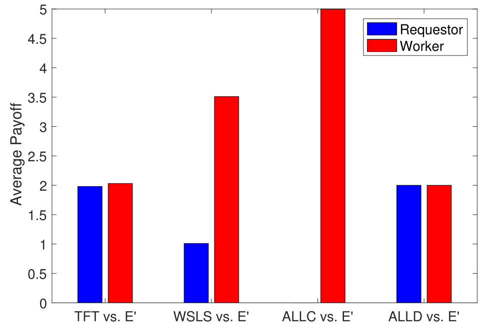

Fig. 3. Average payoffs of the requestor and the worker when the requestor adopts classic strategies and the worker adopts the E′ strategy.

When the worker adopts the E′ strategy, the average payoffs of the requestor and that of the worker are presented in Fig. 3. It is clear that no matter what kind of classic strategy the requestor employs, the worker with the E′ strategy can get an average payoff that is similar to or higher than the requestor. More specifically, the worker's payoff is much higher when the requestor takes WSLS or ALLC.

C. Analysis on the Evolutionary Strategies

In this section, we present our theoretical analysis on the above experimental results. We start by giving definitions on the mixed strategies of the requestor and the worker based on conditional probabilities.

Definition 2 (The Requestor's Mixed Strategy $\mathbf{p}^t$*): The mixed strategy of the requestor at round t is $\mathbf{p}^t = (p_1^t, p_2^t, p_3^t, p_4^t)$, where each element is the requestor's conditional probability to cooperate at round t given the outcome of previous round $rw \in \{cc, cd, dc, dd\}$.*

Definition 3 (The Worker's Mixed Strategy $\mathbf{q}^t$*): The mixed strategy of the worker at round t is $\mathbf{q}^t = (q_1^t, q_2^t, q_3^t, q_4^t)$, where each element is the worker's conditional probability to cooperate at round t given the outcome of previous round $wr \in \{cc, cd, dc, dd\}$.*

According to [16], we can calculate the requestor's expected payoff E_r^t and the worker's expected payoff E_w^t at round t using the mixed strategies defined above:

$$E_r^t = \frac{D(\mathbf{p}^t, \mathbf{q}^t, \mathbf{S}_r)}{D(\mathbf{p}^t, \mathbf{q}^t, \mathbf{1})}, \quad E_w^t = \frac{D(\mathbf{p}^t, \mathbf{q}^t, \mathbf{S}_w)}{D(\mathbf{p}^t, \mathbf{q}^t, \mathbf{1})}. \tag{6}$$

Given any vector $\mathbf{x} = (x_1, x_2, x_3, x_4)$, $D(\mathbf{p}^t, \mathbf{q}^t, \mathbf{x})$ can be computed by:

$$D(\mathbf{p}^t, \mathbf{q}^t, \mathbf{x}) = det \begin{bmatrix} -1 + p_1^t q_1^t & -1 + p_1^t & -1 + q_1^t & x_1 \\ p_2^t q_3^t & -1 + p_2^t & q_3^t & x_2 \\ p_3^t q_2^t & p_3^t & -1 + q_2^t & x_3 \\ p_4^t q_4^t & p_4^t & q_4^t & x_4 \end{bmatrix}.$$

Since it is very typical for the players in a PD game to have symmetric payoffs, we assume that $R_r = R_w = R$, $a = m$, and $b = n$. Based on these assumptions, we derive the following theorems to analyze the above experimental results.

Theorem 3: If the requestor employs the TFT strategy and the worker adopts an evolutionary strategy, the expected payoffs of both players are similar.

Proof: When the requestor employs the TFT strategy, her mixed strategy at round t can be expressed as $\mathbf{p}^t = (1, 0, 1, 0)$. Thus, the requestor's expected payoff E_r^t and the worker's expected payoff E_w^t at round t can be calculated according to (6). Thus, we have

$$\begin{aligned} E_r^t = E_w^t = &\frac{R - a + b - Rq_1^t - Rq_3^t + 2Rq_4^t + aq_1^t + aq_3^t}{2q_4^t - q_3^t - q_1^t + q_1^t q_3^t - 2q_1^t q_4^t + q_2^t q_4^t + 1} \\ &+ \frac{-aq_4^t - bq_1^t - bq_3^t + bq_4^t + Rq_1^t q_3^t - 2Rq_1^t q_4^t}{2q_4^t - q_3^t - q_1^t + q_1^t q_3^t - 2q_1^t q_4^t + q_2^t q_4^t + 1} \\ &+ \frac{Rq_2^t q_4^t - aq_1^t q_3^t + aq_1^t q_4^t + bq_1^t q_3^t - bq_1^t q_4^t}{2q_4^t - q_3^t - q_1^t + q_1^t q_3^t - 2q_1^t q_4^t + q_2^t q_4^t + 1}. \end{aligned}$$

□

Theorem 4: If the requestor employs the WSLS strategy and the worker adopts an evolutionary strategy, we can find a constant T, such that $\forall t \geq T$, the worker's expected payoff E_w^t is larger than the requestor's expected payoff E_r^t.

Proof: If the requestor employs the WSLS strategy, her mixed strategy at any round τ is $\mathbf{p}^\tau = (1, 0, 0, 1)$. Thus, according to (6), we have

$$\begin{aligned} E_r^\tau = &-\frac{2R - 2a + b - 2Rq_1^\tau - 2Rq_2^\tau + Rq_3^\tau + Rq_4^\tau + 2aq_1^\tau}{2q_1^\tau + 2q_2^\tau - q_3^\tau - q_4^\tau - 2q_1^\tau q_2^\tau + q_1^\tau q_3^\tau + q_2^\tau q_4^\tau - 2} \\ &-\frac{2aq_2^\tau - bq_1^\tau - bq_2^\tau + bq_3^\tau + 2Rq_1^\tau q_2^\tau - Rq_1^\tau q_3^\tau}{2q_1^\tau + 2q_2^\tau - q_3^\tau - q_4^\tau - 2q_1^\tau q_2^\tau + q_1^\tau q_3^\tau + q_2^\tau q_4^\tau - 2} \\ &-\frac{(-Rq_2^\tau q_4^\tau - 2aq_1^\tau q_2^\tau + bq_1^\tau q_2^\tau - bq_1^\tau q_3^\tau)}{2q_1^\tau + 2q_2^\tau - q_3^\tau - q_4^\tau - 2q_1^\tau q_2^\tau + q_1^\tau q_3^\tau + q_2^\tau q_4^\tau - 2}, \\ E_w^\tau = &-\frac{2R - a + 2b - 2Rq_1^\tau - 2Rq_2^\tau + Rq_3^\tau + Rq_4^\tau + aq_1^\tau}{2q_1^\tau + 2q_2^\tau - q_3^\tau - q_4^\tau - 2q_1^\tau q_2^\tau + q_1^\tau q_3^\tau + q_2^\tau q_4^\tau - 2} \\ &-\frac{aq_2^\tau - aq_3^\tau - 2bq_1^\tau - 2bq_2^\tau + 2Rq_1^\tau q_2^\tau - Rq_1^\tau q_3^\tau}{2q_1^\tau + 2q_2^\tau - q_3^\tau - q_4^\tau - 2q_1^\tau q_2^\tau + q_1^\tau q_3^\tau + q_2^\tau q_4^\tau - 2} \\ &-\frac{(-Rq_2^\tau q_4^\tau - aq_1^\tau q_2^\tau + aq_1^\tau q_3^\tau + 2bq_1^\tau q_2^\tau)}{2q_1^\tau + 2q_2^\tau - q_3^\tau - q_4^\tau - 2q_1^\tau q_2^\tau + q_1^\tau q_3^\tau + q_2^\tau q_4^\tau - 2}. \end{aligned}$$

Next, we can calculate their difference as

$$E_w^\tau - E_r^\tau = \frac{(a + b)(1 - q_1^\tau)(q_2^\tau + q_3^\tau - 1)}{(1 - q_2^\tau)(2(q_1^\tau - 1) - q_4^\tau) + q_3^\tau(q_1^\tau - 1)}. \tag{7}$$

Since the cooperation probability certainly resides in [0, 1], we have $q_1^\tau, q_2^\tau, q_3^\tau, q_4^\tau \in [0, 1]$, and thus it is clear that the denominator of the above equation is non-positive. Besides, the denominator can be transformed from the denominator of E_r^τ and E_w^τ; thus it should not be zero. In this case, the denominator of (7) is negative.

While for the numerator of (7), we know that $(a + b)$ and $(1 - q_1^\tau)$ are non-negative, and thus the comparison between E_w^τ and E_r^τ is determined by the value of $(q_2^\tau + q_3^\tau - 1)$. Note that q_2^τ is the worker's cooperation probability at round τ when the outcome at round $\tau - 1$ is $wr = cd$, where the cooperation probability of the requestor $p_r^{\tau-1}$ is exactly 0. Thus according to (2), we obtain that the expected cooperation payoff of the worker is $W_c^{\tau-1} = p_r^{\tau-1}E(cc) + (1 - p_r^{\tau-1})E(cd) = E(cd) = R - a$, and the expected defection payoff is $W_d^{\tau-1} = p_r^{\tau-1}E(dc) + (1 - p_r^{\tau-1})E(dd) = E(dd) = R + b - a$.

Next, we consider the situation where the worker adopts the E strategy. In this case, his cooperation probability evolves according to (1), which leads to

$$q_2^\tau = q_w^\tau = q_w^{\tau-1}\frac{W_c^{\tau-1}}{E_w^{\tau-1}} = q_w^{\tau-1}\frac{R-a}{R-a+b(1-q_w^{\tau-1})}. \tag{8}$$

Because $\forall\tau$, $q_w^\tau \in [0,1]$, $R-a$ is always less than $R-a+b(1-q_w^{\tau-1})$. Thus, when τ increases , q_2^τ always decreases. q_3^τ has the same trend after similar computation process.

When the worker adopts the E′ strategy, his cooperation probability will evolve according to (5). As mentioned above, the worker's expected cooperation payoff W_c^τ is always less than his expected defection payoff W_d^τ at any round τ; thus the cumulative expected cooperation payoff A_c^τ should be less than the cumulative expected defection payoff A_d^τ as τ increases. Therefore we have

$$q_2^\tau = q_w^\tau = \frac{e^{A_c^\tau - A_d^\tau}}{1+e^{A_c^\tau - A_d^\tau}} = \frac{e^{\sum_{x=0}^{\tau}(R-a)-\sum_{x=0}^{\tau}(R+b-a)}}{1+e^{\sum_{x=0}^{\tau}(R-a)-\sum_{x=0}^{\tau}(R+b-a)}}. \tag{9}$$

It is clear that q_2^τ becomes smaller and smaller as τ increases. Similar results can be obtained for q_3^τ.

Therefore, $\exists T$ such that $\forall t \geq T$, $q_2^t + q_3^t < 1$ holds, which causes the numerator of (7) to be negative. Thus the value of (7) is positive, leading to $E_w^\tau > E_r^\tau$. □

Corollary 1: In the case where the worker gets a payoff of zero when the outcome is $wr = cd$, the requestor's expected payoff when adopting the WSLS strategy is always less than the worker's expected payoff when adopting an evolutionary strategy.

Proof: When the worker adopts the E strategy, if $E(cd) =$ R-a $= 0$, according to (8), for $\forall t$, $q_2^t = 0$; the same argument holds for q_3^t. Thus we have $E_r^t < E_w^t$ based on (7). When the worker adopts the E′ strategy, according to (9), we have $q_2^t < \frac{1}{2}$; similarly, $q_3^t < \frac{1}{2}$. Therefore, we also get $E_r^t < E_w^t$. □

Note that Corollary 1 can help explain the observed phenomena of WSLS vs. E and WSLS vs. E′ in Figs. 2 and 3, respectively.

Theorem 5: If the requestor employs the ALLC strategy and the worker adopts an evolutionary strategy, the expected payoff of the worker is always larger than that of the requestor.

Proof: When the requestor employs the ALLC strategy, her mixed strategy at any round t is $\mathbf{p}^t = (1,1,1,1)$. Thus, the expected payoffs of the players are:

$$E_r^t = \frac{R-a-Rq_1^t+Rq_3^t+aq_1^t}{q_3^t+1-q_1^t},$$
$$E_w^t = \frac{R+b-Rq_1^t+Rq_3^t-bq_1^t}{q_3^t+1-q_1^t}.$$

Then

$$E_w^t - E_r^t = \frac{(a+b)(1-q_1^t)}{q_3^t+1-q_1^t}.$$

It is clear that the above equation is non-negative because $q_1^t, q_3^t \in [0,1]$. Therefore, we have $E_w^t \geq E_r^t$. □

Theorem 6: If the requestor employs the ALLD strategy and the worker adopts an evolutionary strategy, the expected payoff of the requestor is similar to that of the worker when t is very large.

Proof: When the requestor adopts the ALLD strategy, her mixed strategy is $\mathbf{p}^t = (0,0,0,0)$, and the expected payoffs of both players are:

$$E_r^t = \frac{R-a+b-Rq_2^t+Rq_4^t+aq_2^t-bq_2^t+bq_4^t}{q_4^t+1-q_2^t},$$
$$E_w^t = \frac{R-a+b-Rq_2^t+Rq_4^t+aq_2^t-aq_4^t-bq_2^t}{q_4^t+1-q_2^t}.$$

Thus we have

$$E_w^t - E_r^t = \frac{-(a+b)q_4^t}{q_4^t+1-q_2^t}.$$

It is obvious that the value of the above equation is non-positive, and whether it is zero is determined by q_4^t, which is the worker's cooperation probability given that the previous outcome is $wr = dd$. According to (2), $W_c^{t-1} = 0\cdot E(cc) + 1\cdot E(cd) = E(cd) = R-a$, and according to (3), $W_d^{t-1} = 0\cdot E(dc)+1\cdot E(dd) = E(dd) = R+b-a$. Therefore, when the worker adopts the E strategy, $q_4^t = q_w^t = q_w^{t-1}\frac{W_c^{t-1}}{E_w^{t-1}} = q_w^{t-1}\frac{R-a}{R-a+b(1-q_w^{t-1})}$, and thus $\lim_{t\to\infty} q_4^t = 0$, resulting in $E_w^t = E_r^t$; when the worker adopts the E′ strategy, $q_4^t = q_w^t = \frac{\exp(A_c^t - A_d^t)}{1+\exp(A_c^t-A_d^t)} = \frac{\exp(\sum_{x=0}^{t}(R-a)-\sum_{x=0}^{t}(R+b-a))}{1+\exp(\sum_{x=0}^{t}(R-a)-\sum_{x=0}^{t}(R+b-a))}$, and we can also get $\lim_{t\to\infty} q_4^t = 0$, leading to $E_w^t = E_r^t$. □

Corollary 2: In the case where the worker gets a zero payoff when the outcome is $wr = cd$, the requestor employing the ALLD strategy has a similar expected payoff compared to that of the worker adopting an evolutionary strategy.

Proof: We start by considering that the worker adopts the E strategy. If $E(cd) = 0$, we have for $\forall t$, $q_4^t = q_w^t = q_w^{t-1}\frac{W_c^{t-1}}{E_w^{t-1}} = q_w^{t-1}\frac{E(cd)}{E_w^{t-1}} = 0$, which means $E_w^t = E_r^t$ at each round t. When the worker adopts the E′ strategy, given that $E(cd) = 0$, we have $q_4^t = q_w^t = \frac{\exp(\sum_{x=0}^{t}(R-a)-\sum_{x=0}^{t}(R+b-a))}{1+\exp(\sum_{x=0}^{t}(R-a)-\sum_{x=0}^{t}(R+b-a))} = \frac{\exp(-\sum_{x=0}^{t} b)}{1+\exp(-\sum_{x=0}^{t} b)}$, which is close to zero when t is large. Thus we have $E_w^t \approx E_r^t$. □

Note that Corollary 2 can justify the results of ALLD vs. E and ALLD vs. E′ in Figs. 2 and 3, respectively.

In summary, both the experimental results and the theoretical analysis indicate that the payoff of the worker when adopting an evolutionary strategy is never less than that of the requestor. Even though the situation with ALLD strategy is unfavorable for the worker, a rational and interest-driven requestor would never take this stubborn strategy. While from the perspective of the worker, he is employed by the requestor and stays at a disadvantageous position, lacking the global information of the opponent; thus, getting the above expected payoffs seems to be good enough for him. In the following we assume that the worker adopts an evolutionary strategy due to its great potential.

D. Cooperativeness of the Worker

With such a powerful evolutionary strategy adopted by the worker, we explore his cooperativeness when playing against the requestor with several intuitive strategies such as TFT, WSLS, ALLC, ALLD, E and E′, which turn out to be all failures in eliciting the final cooperation from the worker (detailed in [12]). In fact, when the worker is an E-strategy player, the cooperation probability evolves according to $q_w^{t+1} = q_w^t \frac{W_c^t}{E_w^t}$, where

$$\begin{aligned} W_c^t &= p_r^t * E(cc) + (1-p_r^t) * E(cd) = R_w - a + ap_r^t, \\ W_d^t &= p_r^t * E(dc) + (1-p_r^t) * E(dd) = R_w - a + ap_r^t + b, \\ E_w^t &= q_w^t * W_c^t + (1-q_w^t) \\ &\quad * W_d^t = R_w - a + ap_r^t + b(1-q_w^t). \end{aligned}$$

It is clear that for $\forall t$, $W_c^t = R_w - a + ap_r^t$, $E_w^t = R_w - a + ap_r^t + b(1-q_w^t)$, so $W_c^t \le E_w^t$ for $p_r^t, q_w^t \in [0,1]$. Hence $\lim_{t\to\infty} q_w^{t+1} = q_w^t \frac{W_c^t}{E_w^t} = 0$. When the worker employs the E$'$ strategy, q_w^t evolves according to $q_w^t = \frac{e^{A_c^t - A_d^t}}{1+e^{A_c^t - A_d^t}}$, where

$$\begin{aligned} A_c^t &= \sum_{\tau=0}^{t} W_c^\tau = \sum_{\tau=0}^{t} (R_w - a + ap_r^\tau), \\ A_d^t &= \sum_{\tau=0}^{t} W_d^\tau = \sum_{\tau=0}^{t} (R_w - a + ap_r^\tau + b). \end{aligned}$$

Then it is easy to see that $A_c^t - A_d^t$ becomes smaller and smaller when t increases, and hence $e^{A_c^t - A_d^t}$ approaches to zero gradually. Thus, $\lim_{t\to\infty} q_w^t = 0$.

From the above analysis, we can find that if W_c^t does not get increased, the evolutionary worker should always defect. This is because when the worker adopts an evolutionary strategy, he adjusts his cooperation probability only based on his own obtained payoff regardless of the strategy or payoff of his opponent. Therefore, only if the external environment faced by the worker changes, such as the payoff matrix getting changed, his cooperation probability can switch to another evolutionary path. Yet, once the requestor increases the worker's payoff when he is cooperative, it costs the requestor more to get the whole job done in the long run. This challenge elicits the following problem.

Problem 1: Does there exist an algorithm that can stimulate the cooperation of the worker via only increasing his short-term payoff without extra long-term cost of the requestor?

IV. ZD Strategies Based Algorithm

To solve the above problem, we propose a novel algorithm to help the requestor drive the worker to become cooperative via increasing the short-term payoff of the worker without extra long-term expense of the requestor. The proposed algorithm is based on ZD strategies, including the pinning strategy and the extortion strategy, which are introduced as follows with respect to our problem scenario.

A. Introduction and Analysis of the ZD Strategies

The ZD strategies [16] proposed by Press and Dyson can help us produce a revolutionary recognition of the IPD problem. By this means, a ZD player can unilaterally set the opponent's expected payoff or form an extortionate linear relationship between the expected payoffs of both players in an IPD game. As a matter of fact, ZD strategies can be applied to any iterated two-player game scenario. Hence, the requestor can have unilateral priority to play against the worker by taking advantage of the ZD strategies. Next, we introduce the theoretic basis for applying the ZD strategies in our algorithm to tackle Problem 1.

A linear relationship of the expected payoffs can be obtained according to (6):

$$\forall t,\ \alpha E_r^t + \beta E_w^t + \gamma = \frac{D(\mathbf{p}^t, \mathbf{q}^t, \alpha \mathbf{S}_r + \beta \mathbf{S}_w + \gamma \mathbf{1})}{D(\mathbf{p}^t, \mathbf{q}^t, \mathbf{1})}. \quad (10)$$

When the strategy of the requestor $\mathbf{p}^t$ satisfies $(-1+p_1^t, -1+p_2^t, p_3^t, p_4^t) = \alpha \mathbf{S}_r + \beta \mathbf{S}_w + \gamma \mathbf{1}$, the numerator in the right side of (10) is zero. Hence, a linear relationship between the two expected payoffs exists,

$$\forall t,\ \alpha E_r^t + \beta E_w^t + \gamma = 0. \quad (11)$$

Based on (11), two special kinds of ZD strategies can be derived, i.e., *pinning strategy* and *extortion strategy*, where the former can help the ZD player set the payoff of the opponent to a fixed value, while the latter can facilitate the ZD player to extortionately control the payoff of the opponent at a low level. The details are presented as follows.

1) Pinning Strategy: In light of (11), when $\alpha = 0$, $E_w^t = -\frac{\gamma}{\beta}$ holds. That is, when the requestor's strategy $\mathbf{p}^t$ meets $(-1+p_1^t, -1+p_2^t, p_3^t, p_4^t) = \beta \mathbf{S}_w + \gamma \mathbf{1}$, the requestor can set the expected payoff of the worker E_w^t to a certain value unilaterally. More specifically, to set the appropriate value of $\mathbf{p}^t$, we need to solve the following equations,

$$\begin{cases} -1 + p_1^t = \beta R_w + \gamma, \\ -1 + p_2^t = \beta(R_w + b) + \gamma, \\ p_3^t = \beta(R_w - a) + \gamma, \\ p_4^t = \beta(R_w + b - a) + \gamma. \end{cases}$$

Here, p_2^t and p_3^t can be represented using p_1^t and p_4^t as:

$$p_2^t = \frac{ap_1^t - b(1+p_4^t)}{a-b}, \quad p_3^t = \frac{b(1-p_1^t) + ap_4^t}{a-b}. \quad (12)$$

Accordingly, the expected payoff of the worker at round t is:

$$E_w^t = \frac{(1-p_1^t)(R_w + b - a) + p_4^t R_w}{1 - p_1^t + p_4^t}. \quad (13)$$

Theoretically, given $p_2^t, p_3^t \in [0,1]$, the equations in (12) have feasible solutions only when p_1^t is left close to 1 and p_4^t is right close to 0. According to (13), E_w^t is a weighted average of $R_w + b - a$ and R_w; therefore $R_w + b - a \le E_w^t \le R_w$. We denote the maximum and the minimum of E_w^t in this pinning strategy as max_p and min_p, respectively.

2) Extortion Strategy: According to (11), when $\alpha = \phi$, $\beta = -\phi\chi$, $\gamma = \phi(\chi(R_w + b - a) - (R_r - m + n))$, the linear relationship between the expected payoff of the requestor and that of the worker $E_r^t - (R_r - m + n) = \chi(E_w^t - (R_w + b - a))$ can be obtained, where $\chi \ge 1$ is called the extortion ratio and ϕ is employed to ensure that the elements of $\mathbf{p}^t$ are in $[0,1]$. In other words, if the requestor chooses the strategy $\mathbf{p}^t$ satisfying $(-1+p_1^t, -1+p_2^t, p_3^t, p_4^t) = \phi[(\mathbf{S}_r - (R_r - m + n)) - \chi(\mathbf{S}_w - (R_w + b - a))]$, she can enforce an extortion relationship between her payoff and that of the worker. Similarly, the specific strategy of the requestor can be solved by the following equations:

$$\begin{cases} -1 + p_1^t = \phi[m - n - \chi(a - b)], \\ -1 + p_2^t = \phi[-n - \chi a], \\ p_3^t = \phi[m + \chi b], \\ p_4^t = 0. \end{cases} \tag{14}$$

It is clear that the existence of a feasible solution is determined by any χ and a sufficiently small ϕ. Specifically, to ensure that the elements of $\mathbf{p}^t \in [0, 1]$, ϕ should satisfy:

$$0 \le \phi \le \overline{\phi}, \tag{15}$$

where $\overline{\phi} = \min\{\frac{1}{m+\chi b}, \frac{1}{n+\chi a}, \frac{1}{\chi(a-b)-(m-n)}\}$.

According to [16], when the worker unconditionally defects, i.e., $\mathbf{q}^t = (0, 0, 0, 0)$, he gets the minimum payoff,

$$\min E_w^t = E_w^t|_{\mathbf{q}^t=(0,0,0,0)} = R_w + b - a; \tag{16}$$

while when he unconditionally cooperates, i.e., $\mathbf{q}^t = (1, 1, 1, 1)$, the maximum payoff can be obtained,

$$\begin{aligned} \max E_w^t &= E_w^t|_{\mathbf{q}^t=(1,1,1,1)} \\ &= \frac{R_w n + am - an + \chi(R_w a - a^2 + ab)}{n + \chi a}. \end{aligned} \tag{17}$$

Note that (17) is monotonically decreasing with respect to the extortion ratio χ. Since it is beneficial to the requestor when $\chi \ge 1$, the maximum of E_w is obtained when $\chi = 1$, i.e.,

$$\begin{aligned} \max E_w^t &= E_w^t|_{\mathbf{q}^t=(1,1,1,1),\chi=1} \\ &= R_w + \frac{a}{n+a}(m - n + b - a). \end{aligned} \tag{18}$$

Remarkably, when $m - n = a - b$, $\max E_w^t = R_w$. We denote the maximum and minimum of E_w^t in the extortion strategy as max_e and min_e, respectively.

B. Algorithm Design

The main idea of our proposed algorithm is to coerce the worker to be cooperative by properly setting his expected payoff using ZD strategies. According to the above analysis, we have $E_w^t \in [min, max]$ at any round t, where $min = min_p$, $max = max_p$ when the pinning strategy is adopted and $min = min_e$, $max = max_e$ when the extortion strategy is employed. Thus, our proposed algorithm can utilize the ZD strategies to set the worker's expected payoff from min to max at each round. Generally speaking, when the worker is likely to be cooperative in a round, his expected payoff in that round can be improved and vice versa. The detailed steps of our algorithm are elaborated in Algorithm 1, which involves an M-round preparatory stage to obtain rational initial values of the parameters. Note that the larger the M, the more precise the parameter values.

In order to estimate the worker's action at the $(t + 1)$-th round based on his action at the t-th round, we employ a state transition probability vector $\mathbf{P_s} = (P_{cc}, P_{cd}, P_{dc}, P_{dd})$, where $P_{ij}, i, j \in \{c, d\}$ is the statistical transition probability from state i to state j calculated from the previous rounds. A simple calculation method is to get the ratio of the number of rounds at which the worker's action changes from state i to j to the total number of rounds. In practice, the worker cannot know the requestor's real strategy at each round, and thus it is not practical to use (2) to compute W_c^t, or further to get E_w^t according to (4)[2]. To make the proposed algorithm practicable, we employ the following equations to compute:

$$w_c^t = f_r^t E(cc) + (1 - f_r^t)E(cd), \tag{19}$$

$$w_d^t = f_r^t E(dc) + (1 - f_r^t)E(dd), \tag{20}$$

$$\overline{w}^t = f_w^t w_c^t + (1 - f_w^t) w_d^t, \tag{21}$$

where f_r^t and f_w^t are the cooperation frequency of the requestor and that of the worker in the previous $M + t$ rounds, respectively. When the worker adopts the E strategy, his cooperation probability evolves according to:

$$q_w^{t+1} = q_w^t \frac{w_c^t}{\overline{w}^t}. \tag{22}$$

While if he employs the E′ strategy, the corresponding cooperation probability changes based on:

$$q_w^t = \frac{e^{\lambda_c^t - \lambda_d^t}}{1 + e^{\lambda_c^t - \lambda_d^t}}, \tag{23}$$

where $\lambda_c^t = \sum_{\tau=0}^t w_c^t$ and $\lambda_d^t = \sum_{\tau=0}^t w_d^t$.

To initialize E_w^0 (Step 1 in Algorithm 1), we compute $\overline{w}^0$ with f_r^0 and f_w^0 obtained from the first M rounds and assign it to E_w^0. After that, the requestor adjusts the worker's expected payoff E_w^t at round t according to the prediction of the worker's cooperativeness. When the worker's previous move is cooperation (Step 3), if $P_{cc} > P_{cd}$, the requestor regards the worker friendly in the current round and sets a higher expected payoff $E_w^t = E_w^{t-1} + (max - E_w^{t-1})/2$ (Steps 4-6); if not, a lower E_w^t is given (Steps 7-9), which equals $E_w^t = E_w^{t-1} - (E_w^{t-1} - min)/2$. Similarly, when the worker's previous move is defection, if $P_{dc} > P_{dd}$, the requestor gives the worker a higher E_w^t (Steps 12-14), and vice versa (Steps 15-18). Actually, the expected payoff at round t is directly reflected in $\overline{w}^t$ for the worker, and he can conclude that his move at round t results in the change of $\overline{w}^t$, where cooperation causes the change of w_c^t and defection is responsible for the change of w_d^t. At the end of each round, the value of the state transition probability updates (Steps 20-22) by including the action change of the worker in the current round. When the total number of rounds is reached, the whole process terminates.

To implement the proposed algorithm, a key problem is how to solve $\mathbf{p}^t$ in order to get an appropriate E_w^t satisfying our algorithm at each round. When the pinning strategy is employed, the requestor can derive the appropriate $\mathbf{p}^t$ according to (12) and (13). While when the extortion strategy is utilized, the requestor needs to first infer the strategy of the worker, i.e., cooperation probability $\mathbf{q}^t$, and then the desirable $\mathbf{p}^t$ can be calculated as follows. According to $E_r^t = \frac{D(\mathbf{p}^t, \mathbf{q}^t, \mathbf{S}_r)}{D(\mathbf{p}^t, \mathbf{q}^t, \mathbf{1})}$ and (14), the expected payoff of the requestor at round t, E_r^t, with respect to χ and ϕ, can be easily calculated. And once an appropriate ϕ is given, χ can be solved through

[2]In the previous section, we employ q_w^t and p_r^t to compute W_c^t and E_w^t in order to precisely analyze the problem in an ideal condition.

Algorithm 1 Reward-Penalty Expected Payoff Algorithm Based on ZD Strategies

Require: $\mathbf{p}^t$: the requestor's strategy at round t and its initial value is the one used in the last round of the preparatory stage; $\mathbf{P_s} = (P_{cc}, P_{cd}, P_{dc}, P_{dd})$: the state transition probability of the worker, and its initial value is statistically calculated by the data collected in the preparatory stage; N: the total number of rounds.

Initialize(E_w^0)
for $t = 1$ to N **do**
  if The worker's previous move is c **then**
    if $P_{cc} > P_{cd}$ **then**
      Calculate $\mathbf{p}^t$ which makes
        $E_w^t \leftarrow E_w^{t-1} + (max - E_w^{t-1})/2$
    else ▹ $P_{cc} \leq P_{cd}$.
      Calculate $\mathbf{p}^t$ which makes
        $E_w^t \leftarrow E_w^{t-1} - (E_w^{t-1} - min)/2$
    end if
  else ▹ The worker's previous move is d.
    if $P_{dc} > P_{dd}$ **then**
      Calculate $\mathbf{p}^t$ which makes
        $E_w^t \leftarrow E_w^{t-1} + (max - E_w^{t-1})/2$
    else ▹ $P_{dc} \leq P_{dd}$.
      Calculate $\mathbf{p}^t$ which makes
        $E_w^t \leftarrow E_w^{t-1} - (E_w^{t-1} - min)/2$
    end if
  end if
  if The current round terminates **then**
    Update $\mathbf{P_s}$
  end if
end for

$E_r^t - (R_r - m + n) = \chi(E_w^t - (R_w + b - a))$, which in turn determines the appropriate $\mathbf{p}^t$ in the extortion strategy.

Notably, the proposed algorithm can not only prevent the malicious attack initiated by a single worker but also resist the attack from multiple workers in a crowdsourcing scenario. Because we counter the malicious attack of the worker with the game model between the requestor and any one of the workers, by which the worker can be stimulated to cooperate eventually, the powerful Algorithm 1 can also lead to the workers' full cooperation when faced with simultaneous attacks launched by multiple workers. Note that since the collusion among several workers may complicate the basic problem model, we leave this as our future work.

V. Evaluation on the Proposed Algorithm

A. Simulation Study

To evaluate the performance of the above algorithm, we simulate the whole game process with the same parameters as in the previous simulations. We test our algorithm based on both the pinning strategy and the extortion strategy and obtain very similar results. Hence, we only present the results of the pinning strategy in this section to avoid redundancy. As before, we first investigate the situation where the worker adopts the E strategy. Considering that both players adopt evolutionary strategies in the preparatory M rounds and their cooperation probabilities in the M-th round are almost the same, we set $p_r^0 = q_w^0$ in this simulation study.

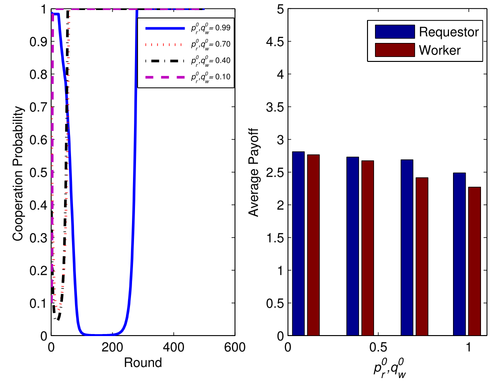

Fig. 4. Cooperation probability of the E worker and average payoffs.

The cooperation probability of the worker and average payoffs of both the requestor and the worker are presented in Fig. 4. The left side demonstrates the evolution of the worker's cooperation probability changing over time with different initial cooperation probabilities of the requestor and the worker, i.e., different p_r^0 and q_w^0, and it evolves into 1 as t increases. Note that there are some differences among the situations with smaller p_r^0, q_w^0 and larger ones. The existence of concave part exactly shows the effect of our proposed algorithm as otherwise all the curves will decrease to zero. The underlying reason is that a smaller q_w^0 results in a huge reward space for the expected payoff when he cooperates; while a larger one leads to a higher initial expected payoff and a smaller reward space. In this case, with the decrease of the large q_w^0, the power of penalty increases gradually and finally drives the worker to be cooperative. In the right side of Fig. 4, the bar graph presents the average values of the actual payoffs of both players with different p_r^0 and q_w^0. It is clear that the actual payoff of the worker is no more than $R_w = 3$, which means that the requestor does not need additional expense to induce the worker to be cooperative finally. In addition, as the initial cooperation probabilities increase, the payoffs decrease slightly. The reason lies in that a larger q_w^0 causes a trough on the evolution of q_w^t, during which the payoffs are lower.

Next, we explore the performance of our proposed algorithm when the worker adopts the E′ strategy, which is not related to the initial cooperation probability. Accordingly, we investigate the impact of the difference between the worker's initial expected cooperation payoff and defection payoff. As shown in Fig. 5, the left-side subgraph presents the evolution path of the worker's cooperation probability, when $W_c^0 - W_d^0$ varies, while the right one plots the average payoff of the requestor and that of the worker. It is clear that our proposed algorithm is also effective when the worker adopts the E′ strategy. More specifically, the cooperation probability of the worker increases to 1 and the worker's average actual payoff is no more than $R_w = 3$ under all situations, which means that with the help of our proposed algorithm, the requestor can drive the worker to be fully cooperative without extra cost.

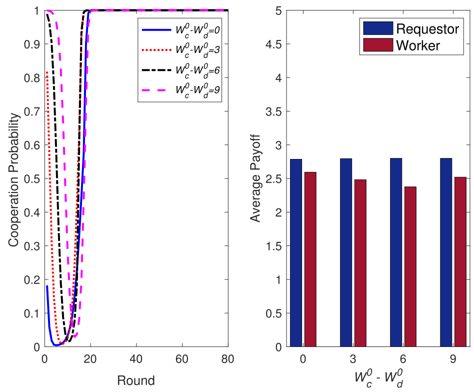

Fig. 5. Cooperation probability of the E′ worker and average payoffs.

B. Theoretical Analysis

Actually, the above experimental results can be verified theoretically, which are summarized into the following theorems.

Theorem 7: In the proposed algorithm, the cooperation probability of the worker can increase to 1 when the number of game rounds increases, which indicates that the worker eventually cooperates.

Proof: When the worker adopts the E strategy, one can see that the increase of q_w^{t+1} happens only when $w_c^t > \overline{w}^t$ at round t according to (22). We can combine with the calculation of $\overline{w}^{t+1}$ in (21) to transfer the condition $w_c^t > \overline{w}^t$ to $w_c^t > w_d^t$.

On the other hand, when the worker chooses the E′ strategy, q_w^t increases only when $\lambda_c^t - \lambda_d^t$ increases at round t compared to that at round $t-1$ according to (23). Considering that $\lambda_c^t = \sum_{\tau=0}^t w_c^t$ and $\lambda_d^t = \sum_{\tau=0}^t w_d^t$, one can conclude that the increase of $\lambda_c^t - \lambda_d^t$ is possible only when $w_c^t > w_d^t$.

Next, we need to prove that the relationship $w_c^t > w_d^t$ holds when our proposed algorithm is applied. We need to consider the following two circumstances:

1) Once the worker's action at round $t+1$ is estimated to be c, he can obtain E_w^{t+1} that is higher than E_w^t as a reward. The increase of E_w^{t+1} can result in the increase of $\overline{w}^{t+1} = \frac{(M+t)\overline{w}^t + E_w^{t+1}}{M+t+1}$ when we have $\overline{w}^t \approx E_w^t$. And the cooperation frequency of the worker from round t to $t+1$ satisfies $f_w^{t+1} = \frac{f_w^t(M+t)+1}{M+t+1} = f_w^t + \frac{1-f_w^t}{M+t+1}$. According to the definition of $\overline{w}^{t+1}$, we have

$$\begin{aligned} w_c^{t+1} &= \frac{\overline{w}^{t+1} - (1 - f_w^{t+1})w_d^{t+1}}{f_w^{t+1}} \\ &= \frac{\overline{w}^{t+1} - (1 - f_w^t - \frac{1-f_w^t}{M+t+1})w_d^{t+1}}{f_w^t + \frac{1-f_w^t}{M+t+1}}. \end{aligned}$$

When $M+t$ is large enough, we have

$$\begin{aligned} w_c^{t+1} &\approx \frac{\overline{w}^{t+1} - (1 - f_w^t)w_d^{t+1}}{f_w^t} \\ &> \frac{\overline{w}^t - (1 - f_w^t)w_d^{t+1}}{f_w^t} = w_c^t, \end{aligned}$$

where the equality holds because $w_d^{t+1} = w_d^t$.

2) Once the worker's action at round $t+1$ is estimated to be d, we have $E_w^{t+1} < E_w^t$, which leads to $\overline{w}^{t+1} < \overline{w}^t$. Then $f_w^{t+1} = \frac{f_w^t(M+t)}{M+t+1} = f_w^t - \frac{f_w^t}{M+t+1}$. Considering that w_c^{t+1} does not change, we have

$$\begin{aligned} w_d^{t+1} &= \frac{\overline{w}^{t+1} - f_w^{t+1} w_c^{t+1}}{1 - f_w^{t+1}} \\ &= \frac{\overline{w}^{t+1} - (f_w^t - \frac{f_w^t}{M+t+1})w_c^{t+1}}{1 - f_w^t + \frac{f_w^t}{M+t+1}}. \end{aligned}$$

When $M+t$ is large enough, we have

$$w_d^{t+1} \approx \frac{\overline{w}^{t+1} - f_w^t w_c^{t+1}}{1 - f_w^t} < \frac{\overline{w}^t - f_w w_c^{t+1}}{1 - f_w^t} = w_d^t.$$

Considering the above two situations, one can see that at any round t either w_c^t increases when w_d^t remains unchanged, or w_d^t decreases when w_c^t keeps constant. Therefore, $\exists T \in \mathbb{N}^+$ such that $\forall\ t > T$, $w_c^t > w_d^t$. □

Theorem 8: In the proposed algorithm, the worker's actual payoff is never more than R_w in the long run.

Proof: As shown in Theorem 7, q_w^t increases to 1 eventually. If $q_w^t \to 1$, it is easy to prove that the worker's expected payoff $E_w^t \to R_w$ according to Algorithm 1 (Steps 6 and 14). Considering that E_w^t is determined by the requestor's strategy $\mathbf{p}^t$, we can get $\mathbf{p}^t$ when $E_w^t \to R_w$ via solving

$$\frac{(1-p_1^t)(R_w + b - a) + p_4^t R_w}{1 - p_1^t + p_4^t} = R_w. \tag{24}$$

It it clear that when $p_1^t = 1,\ p_4^t \neq 0$, the above equation holds. And $p_1^t = 1$ implies that the requestor remains cooperative when the outcome of the last round is mutual cooperation. On the other hand, the cooperation probability of the worker increases to 1, so the outcome of the game stays at mutual cooperation once $rw = cc$ is achieved. Therefore, when $t \to \infty$, $p_r^t = 1$.

Then, as t increases, the worker's actual payoff is $\lim_{t\to\infty} \frac{w_c^t - (1-p_r^t)(R_w+b-a)}{p_r^t} = \lim_{t\to\infty} w_c^t = \lim_{t\to\infty} \frac{\overline{w}^t - (1-q_w^t)w_d^t}{q_w^t} \overset{q_w \to 1}{=} \lim_{t\to\infty} \overline{w}^t = \lim_{t\to\infty} \frac{M\overline{w}^0 + \sum_1^t E_w}{t} = R_w$. □

Problem 2: Since the worker becomes cooperative eventually, what if a greedy requestor intends to earn a higher payoff by maliciously defecting?

The above problem can be solved by the following theorem.

Theorem 9: In the proposed algorithm, the actual payoff of the requestor is never more than R_r in the long run.

Proof: As analyzed in Theorem 8, when $q_w^t \to 1$, $p_1^t \to 1$ holds, which indicates that the requestor keeps cooperation after $rw = cc$ is reached and she never defects in the long run. Therefore, her actual payoff will never larger than R_r. □

Theorem 9 verifies that Algorithm 1 is lively because of its fairness for both players. To be specific, even though the requestor can utilize powerful ZD strategies, she cannot arbitrarily defect for higher payoffs once the worker cooperates. The ultimate outcome of the game then can be $wr = cc$ under the impact of the proposed algorithm, which not only increases the payoffs of both players but also is impartial to them, making the algorithm easily accepted by both players in the long term.

VI. Related Work

The research on thwarting the malicious behaviors in crowdsourcing has been thriving in recent years. In [10], Wang *et al.* proposed a two-phase method, i.e., collecting "ground truth" and comparing/contrasting, to detect and characterize malicious workers. A practical and empirical method was presented in [11] for adversary attack detection in crowdsourcing. In [8], Zhang *et al.* studied the malicious data injection attacks in crowdsourcing-based cooperative spectrum sensing by concurrently employing external *detectors* to test the truthfulness and considering the worker's long-term reputation. Lasecki *et al.* [5] indicated that some workers might curiously obtain the private information of other participants and proposed possible methods to deal with these threats in Amazon's Mechanical Turk platform. Xu *et al.* [9] detected spammers in crowdsourced online Q&A community to distinguish the malicious spammers from the normal users.

Game-based approaches have been receiving more and more attention in recent years. Researchers in [7] and [17] respectively made use of PD and IPD to analyze the attack problem in crowdsourcing, and demonstrated that the existence of malicious behaviors in crowdsourcing is the norm and repeated interactions cannot improve or solve the crowdsourcing dilemma. In [18], Anta *et al.* modeled the actions of the workers, either normally performing the task or submitting fabricated results for saving cost, as a game in crowdsourcing computing, and derived the required conditions for the existence of the Nash Equilibrium where the requestor could obtain the correct computing results. In an Android-based crowdsourcing recommendation system named RecDroid, Rashidi and Fung [19] tackled the malicious attack problem between the system users and the server using a static Bayesian game, where the system server has incomplete information about the users. Focusing on vehicular crowdsensing scenarios, Xiao *et al.* [20] formulated a static game and a dynamic game, with the first one considering both the accumulative sensing tasks and the best-quality ones to achieve tradeoff between the sensing quality of the vehicles and the overall payment of the server, and the second one involving Q-learning-based sensing and payment strategies. The same two games were also considered in [21], which adopted a Stackelberg game to model the interactions between the server and the users in mobile crowdsensing, where the equilibria of the static game and a deep Q-network based payment strategy for the dynamic game were proposed to solve the problem of fake sensing attacks.

In this paper, we adopt the ZD strategies to set the expected payoff of the worker so as to elicit the final cooperation. A ZD player can unilaterally set the opponent's expected payoff to a fixed value or enforce a linear relationship between the expected payoffs of both players. Although some studies [22] challenged the evolutionary stability of the ZD strategies, they are still broadly investigated. In [23], an extension of the ZD strategies from the two-player game to the multi-player game was proposed; while in [24], Rong *et al.* studied the influence of the strategy selection timescale, which was also mentioned in [16].

VII. Conclusion

In this paper, we propose a novel algorithm to resist malicious behaviors of the workers in crowdsourcing. We start by modeling the interactions between the requestor and a worker as an iterated two-player game. Next, we consider that the worker adopts evolutionary strategies and demonstrate its power by simulation studies and theoretical analysis. To play against the powerful opponent, we propose a novel algorithm for the requestor to entice the worker's cooperation by changing the expected payoff of the worker based on ZD strategies. Both theoretical analysis and simulation studies demonstrate that our proposed algorithm can force the worker to cooperate in the long run by only increasing his short-term payment without any increase on the overall payoff, which is proved to be lively since it is fair for both players. In our future research, we will consider the impact of colluding workers on the cooperation probabilities of both the initiator and the workers in crowdsourcing applications; we will also investigate ZD based incentive strategies to tackle other crowdsourcing challenges as well as other problems that can be modeled as two-player or multiple-player games.

References

[1] C. L. Tucci, A. Afuah, and G. Viscusi, *Creating and Capturing Value Through Crowdsourcing*. New York, NY, USA: Oxford Univ. Press, 2018.

[2] J. Lu, Y. Xin, Z. Zhang, X. Liu, and K. Li, "Game-theoretic design of optimal two-sided rating protocols for service exchange dilemma in crowdsourcing," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 11, pp. 2801–2815, Nov. 2018.

[3] Y. Wang, Z. Cai, X. Tong, Y. Gao, and G. Yin, "Truthful incentive mechanism with location privacy-preserving for mobile crowdsourcing systems," *Comput. Netw.*, vol. 135, pp. 32–43, Apr. 2018.

[4] C. Palmer. (2011). *UC San Diego Team's Effort in DARPA's Shredder Challenge Derailed by Sabotage*. [Online]. Available: https://jacobsschool.ucsd.edu/news/news_releases/release.sfe?id=1150

[5] W. S. Lasecki, J. Teevan, and E. Kamar, "Information extraction and manipulation threats in crowd-powered systems," in *Proc. 17th ACM Conf. Comput. Supported Cooperat. Work Social Comput.*, Feb. 2014, pp. 248–256.

[6] V. Naroditskiy, N. R. Jennings, P. Van Hentenryck, and M. Cebrian, "Crowdsourcing dilemma," Apr. 2013, *arXiv:1304.3548*. [Online]. Available: https://arxiv.org/abs/1304.3548

[7] V. Naroditskiy, N. R. Jennings, P. Van Hentenryck, and M. Cebrian, "Crowdsourcing contest dilemma," *J. Roy. Soc., Interface*, vol. 11, no. 5, pp. 1–8, 2014.

[8] R. Zhang, J. Zhang, Y. Zhang, and C. Zhang, "Secure crowdsourcing-based cooperative pectrum sensing," in *Proc. INFOCOM*, Apr. 2013, pp. 2526–2534.

[9] A. Xu, X. Feng, and Y. Tian, "Revealing, characterizing, and detecting crowdsourcing spammers: A case study in community Q&A," in *Proc. IEEE INFOCOM*, Apr./May 2015, pp. 2533–2541.

[10] T. Wang, G. Wang, X. Li, H. Zheng, and B. Y. Zhao, "Characterizing and detecting malicious crowdsourcing," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 4, pp. 537–538, 2013.

[11] G. Wang, T. Wang, H. Zhang, and B. Y. Zhao, "Man vs. machine: Practical adversarial detection of malicious crowdsourcing workers," in *Proc. 23rd USENIX Conf. Secur. Symp.*, Apr. 2014, pp. 239–254.

[12] Q. Hu, S. Wang, L. Ma, R. Bie, and X. Cheng, "Anti-malicious crowdsourcing using the zero-determinant strategy," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2017, pp. 1137–1146.

[13] (Aug. 17, 2012). *Amazon Mechanical Turk*. [Online]. Available: https://www.mturk.com/

[14] J. M. Smith, *Evolution and the Theory of Games*. Cambridge, U.K.: Cambridge Univ. Press, 1982.

[15] R. M. Dawes, A. J. Van De Kragt, and J. M. Orbell, "Not me or thee but we: The importance of group identity in eliciting cooperation in dilemma situations: Experimental manipulations," *Acta Psychologica*, vol. 68, nos. 1–3, pp. 83–97, 1988.

[16] W. H. Press and F. J. Dyson, "Iterated prisoner's dilemma contains strategies that dominate any evolutionary opponent," *Proc. Nat. Acad. Sci. USA*, vol. 109, no. 26, pp. 10409–10413, 2012.
[17] K. Oishi, M. Cebrian, A. Abeliuk, and N. Masuda, "Iterated crowdsourcing dilemma game," *Sci. Rep.*, vol. 4, Feb. 2014, Art. no. 4100.
[18] A. F. Anta, C. Georgiou, M. A. Mosteiro, and D. Pareja, "Algorithmic mechanisms for reliable crowdsourcing computation under collusion," *PloS ONE*, vol. 10, no. 3, 2015, Art. no. e0116520.
[19] B. Rashidi and C. Fung, "A game-theoretic model for defending against malicious users in RecDroid," in *Proc. IFIP/IEEE Int. Symp. Integr. Netw. Manage. (IM)*, May 2015, pp. 1339–1344.
[20] L. Xiao, T. Chen, C. Xie, H. Dai, and V. Poor, "Mobile crowdsensing games in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1535–1545, Feb. 2017.
[21] L. Xiao, Y. Li, G. Han, H. Dai, and H. V. Poor, "A secure mobile crowdsensing game with deep reinforcement learning," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 1, pp. 35–47, Jan. 2018.
[22] J. Chen and A. Zinger, "The robustness of zero-determinant strategies in iterated prisoner's dilemma games," *J. Theor. Biol.*, vol. 357, pp. 46–54, Sep. 2014.
[23] L. Pan, D. Hao, Z. Rong, and T. Zhou, "Zero-determinant strategies in iterated public goods game," *Sci. Rep.*, vol. 5, Aug. 2015, Art. no. 13096.
[24] Z. Rong, Z.-X. Wu, D. Hao, M. Z. Q. Chen, and T. Zhou, "Diversity of timescale promotes the maintenance of extortioners in a spatial prisoner's dilemma game," *New J. Phys.*, vol. 17, no. 3, Mar. 2015, Art. no. 033032.

Qin Hu received the Ph.D. degree in computer science from The George Washington University in 2019. She is currently an Assistant Professor with the Department of Computer and Information Science, Indiana University - Purdue University Indianapolis. Her research interests include wireless and mobile security, crowdsourcing/crowdsensing, and blockchain.

Shengling Wang (M'15) received the Ph.D. degree from Xi'an Jiaotong University, in 2008. She did her Post-Doctoral Research in the Department of Computer Science and Technology, Tsinghua University. From 2010 to 2013, she was an Assistant and an Associate Professor with the Institute of Computing Technology of the Chinese Academy of Sciences. She is currently an Associate Professor with the College of Information Science and Technology, Beijing Normal University. Her research interests include mobile/wireless networks, game theory, and crowdsourcing.

Xiuzhen Cheng (F'15) received the M.S. and Ph.D. degrees in computer science from the University of Minnesota Twin Cities in 2000 and 2002, respectively. She is currently a Professor with the Department of Computer Science, The George Washington University, Washington, DC, USA. Her current research interests focus on privacy-aware computing, wireless and mobile security, dynamic spectrum access, mobile handset networking systems (mobile health and safety), cognitive radio networks, and algorithm design and analysis. She has served on the Editorial Boards of several technical publications and the Technical Program Committees of various professional conferences/workshops. She has also chaired several international conferences. She worked as a Program Director for the U.S. National Science Foundation (NSF) from April to October 2006 (full-time), and from April 2008 to May 2010 (part-time).

Liran Ma (M'08) received the Ph.D. degree in computer science from The George Washington University. He is currently an Associate Professor with the Department of Computer Science, Texas Christian University. His current research interests include wireless, mobile, and embedded systems, including security and privacy, smartphones, smart health, mobile computing, the Internet of Things, and cloud computing.

Rongfang Bie (M'15) received the M.S. and Ph.D. degrees from Beijing Normal University in 1993 and 1996, respectively. She has been a Visiting Scholar of University of Cambridge in 2003. She is currently a Professor with the College of Information Science and Technology, Beijing Normal University. Her current research interests include the Internet of Things, data mining, and knowledge engineering.