A Game Theoretical Model addressing Misbehavior in Crowdsourcing IoT

Runbo Su, Arbia Riahi Sfar, Enrico Natalizio, Pascal Moyal, Ye-Qiong Song

A Game Theoretical Model addressing Misbehavior in Crowdsourcing IoT

Runbo Su*, Arbia Riahi Sfar‡, Enrico Natalizio†*, Pascal Moyal§, and Ye-Qiong Song*

*LORIA, CNRS, Université de Lorraine, France

‡STD lab, Military Academy of Tunisia

†Technology Innovation Institute, UAE

§IECL, INRIA, Université de Lorraine, France

***Abstract*—Crowdsourcing technology enables complex tasks to be solved with the aid of a group of workers in the Internet of Things (IoT). On the one hand, crucial sensing data can be collected and processed to enhance smart IoT applications. On the other hand, crowdsourcing IoT (Crowd-IoT) is still facing threats due to the diverse quality of crowdsourced data, and especially the misbehavior of malicious workers. In this paper, we propose a Stochastic Bayesian Game (SBG) to address the Byzantine Altruistic Rational (BAR) based misbehavior, where workers' behavioral types can be deduced reasonably and the requestor can perform optimal actions accordingly by taking the long-term gain into consideration. To validate and evaluate the performance of the proposed model, we simulate various scenarios and conduct a comparison with other solutions. The numerical results show the effectiveness and feasibility of our proposed solution.**

***Index Terms*—Game Theory, Trust, BAR threat model, Malicious behavior, IoT Security, Crowdsourcing.**

I. Introduction

The concept of crowdsourcing was introduced in 2006 as a novel web business solution [1]. To date, it is regarded as a promising technology for IoT since it enables IoT applications to be improved via the contribution of a group of task workers, more precisely, in Crowd-IoT, a group of mobile users (workers) will collect and process data to aid the end user (requestor) in solving a relatively complicated problem [2]. An example of crowdsensing service in IoV (Internet of Vehicles), may be illustrated by traffic-related information (such as transport volume, GPS data, weather conditions, etc.), that can be sensed and gathered by other vehicles or roadside units to optimize the requestor vehicle's routing [3].

While rich and useful information is provided, and massive amounts of sensitive data and private information are being collected and processed, Crowd-IoT systems and participants (task requestors/workers) are very likely to become attack targets. Moreover, identifying misbehavior from malicious task workers remains challenging due to the fact that they can misbehave in a complex strategic manner. For example, in 2011, the UCSD (University of California San Diego) team was a victim suffering from malicious workers in crowdsourcing [4].

In literature, some studies proposed cryptographic schemes to secure IoT/Crowd-IoT. However, such solutions are not efficient as expected in the absence of dedicated infrastructures, and the issue of when detecting insider attackers remains challenging [5]. Furthermore, advances in IoT software and hardware make malicious attackers become more intelligent and capable, which makes existing solutions incapable to deal with the attacker's complex behaviors. For this, some researchers considered the use of trust management to measure the security level in IoT/Crowd-IoT. To date, there are numerous trust management solutions applied in IoT/Crowd-IoT using different approaches, such as Fuzzy inference and Subjective logic [6]. Among these approaches, Game Theory is advantageous in addressing strategic attackers in IoT/Crowd-IoT. However, the majority of existing game-based solutions design a symmetric set of actions for Crowd-IoT. More importantly, only a few works consider the complex behavioral model of attackers, and the distinction between rational selfishness and misbehavior is missing.

In this paper, we design a Stochastic Bayesian Game (SBG) modeling the interactions between Crowd-IoT participants. This model enables the behavioral types of task workers to be deduced by analyzing their behavior, and the requestor can accordingly and optimally act in terms of maximizing his/her long-term gain. The contributions of this work are fourfold:

- We design a SBG game fitting the distributed Crowd-IoT property to model the interactions between participants appropriately.
- We extend the action sets of Crowd-IoT participants to enable a more reasonable analysis of their behaviors, such that the requestor and the worker are no longer homogeneous action-based.
- We consider an evaluation of the behavioral types based on a BAR (Byzantine Altruistic Rational) threat model to address the complex strategic misbehavior of the attacker.
- We conduct a simulation with various scenarios to validate the performance of the proposed model from the perspective of increasing security.

The rest of this paper is organized as follows. Section II summarizes related work. Section III gives the system overview and the definition of the Stochastic Bayesian Game (SBG). After that, Section IV explains game formulation by detailing players, game states, playoffs, and players' strategies. The simulation results, performance evaluation of the proposed model, and a comparative analysis with other approaches are presented in Section V. Lastly, Section VI draws the conclusion and outlines our future work.

II. Related work

In literature, many solutions have been proposed to address security issues in IoT/Crowd-IoT by employing cryptographic solutions. Authors in [7] studied data-level security through an authentication scheme using encryption against jamming and cloning attacks. While this work shows robustness against these two attacks, analyzing the IoT individuals' complex behaviors remains difficult. As stated before, tracking insider attackers is problematic without evaluating the trustworthiness of Crowd-IoT participants. In this regard, a trust-based model is presented by Saied et al. [8], where a centralized IoT node collects feedback to update the quality of recommendations for monitoring the trustworthiness of the current system. One other work concerning trustworthiness evaluation is proposed in [9], where the decision-making is aided by a classification scheme to determine the evaluated participants' types based on a four-phase trust process scheme. With the purpose of formulating participants' behaviors dynamically, Game Theory has been taken into consideration in trust-based security solutions in Crowd-IoT. Studies adopting prisoner's dilemma (PD) game to analyze malicious behaviors in Crowd-IoT are proposed in [10]. A recent work considering an iterated version of previous prisoner's dilemma (IPD) games to ensure the cooperativeness between Crowd-IoT participants was introduced in [11]. However, these two works [10, 11] are both based on a symmetric payoff matrix treating the requestor and the worker in a homogeneous manner. In [12], authors designed an incentive model using the repeated game for Crowd-IoT, but the defense scheme addressing insider attackers is insufficiently discussed.

From the above review, there are still several limitations unsolved. First, the majority of existing game theoretical trust management solutions focus on a simple set of actions (e.g., cooperate/defect) such that the actions of the requestor and workers are homogeneous, which does not match the Crowd-IoT reality. Second, the complex strategic behavioral model of malicious attackers is not taken into consideration, which means that the attacker is able to switch its actions to mislead the evaluation system. Third, the distinction between self-interested behavior and misbehavior is missing, where the former comes from non-malicious Crowd-IoT participants and causes less damage. Lastly, the cooperativeness between Crowd-IoT participants, i.e., the requestor and workers, is insufficiently discussed. In this context, we propose a game theoretical model using Stochastic Bayesian Game to address the above-mentioned limitations.

III. System model

In this section, we first present the Crowd-IoT architecture considered in our work, and then we give the definition of the Stochastic Bayesian Game (SBG) to establish the base of the game formulation, which will be detailed in the next section.

A. System overview

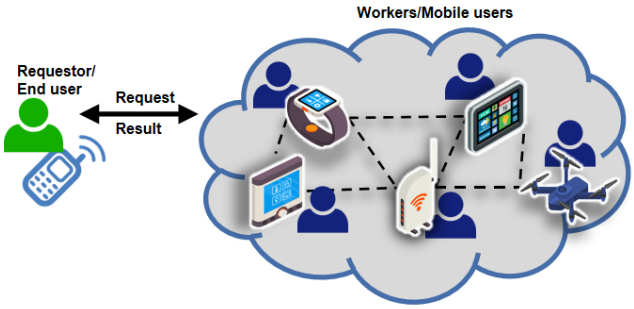

Fig. 1: Crowd-IoT architecture considered in the proposed game theoretical model

Unlike the centralized Crowd-IoT, which is inherently disadvantageous due to a single point of failure, the distributed Crowd-IoT architecture enables the ad-hod network organization and data in-situ processing. Consequently, participants can communicate with each other individually, and request the crowdsourcing service (task) as requestors, or contribute to the task as workers. In this work, we focus on evaluating the interactions between the requestor and workers in distributed Crowd-IoT. Fig. 1 illustrates the task requestor and workers in distributed Crowd-IoT to highlight their roles in crowdsourcing services.

As described in [13], the crowdsourcing service process consists of four main steps: 1) After the communication is established between the requestor and the worker, the former launches the task proposal; 2) The worker will be recruited, and then assigned the task; 3) When the task is completed, the requestor sends the incentive; 4) Once the worker is informed of the reception of the incentive, the crowdsourced data will be released. Given this, we design a SBG game to model the interactions between the requestor and workers appropriately, where workers perform actions independently, i.e., there is no inter-affection between workers.

B. Stochastic Bayesian Game (SBG)

Uncertainties of mixed behaviors may arise when the requestor evaluates the interactions with the workers due to the complex attack strategies of malicious workers. We formulate the problem as Stochastic Bayesian Game (SBG), which was introduced in [14], where opponent players' behaviors can be modeled through a behavior type space and distribution. The methodology can be adapted to our work: where a behavior type refers to one of three categories defined in Section IV-C, and the type distribution can be used for calculating the occurrence frequencies of each type. A general SBG consists of:

- A state space S with initial state s^0 and terminal state $\overline{s}$;
- A set of players N of cardinality n, and for each player $i \in N$,
  - An action set A_i for player's interaction. Throughout, we set $A = A_1 \times ... \times A_n$;
  - A behavior-type space Θ_i modeling player's type. Throughout, we set $\Theta = \Theta_1 \times ... \times \Theta_n$;
  - A payoff function $u_i : S \times A \times \Theta_i \to \mathbb{R}$ defining the gain/loss of players after executing actions $a \in A$;
  - A strategy function π_i: $\mathbb{H} \times A_i \times \Theta_i \to [0, 1]$, where $\mathbb{H}$ denotes the set of all histories $(H^t : t \geq 0)$ of the form H^t=$\langle s^0, a^0, s^1, a^1, ..., s^t \rangle$, where $s^0, ..., s^t \in S$ and $a^0, ..., a^t \in A$, for all $t \geq 0$.
- A state transition function T: $S \times A \times S \to [0, 1]$;
- A type distribution $\triangle$: $\Theta^+ \to [0, 1]$, where Θ^+ is a finite subset of Θ.

The type θ_i for i is sampled from Θ_i before each round of the game. On the basis of the history H^t up to time t, player i selects an action depending on its strategy $\pi_i(H^t, a, \theta_i)$ until the state $\overline{s}$ is reached.

IV. Game formulation

In this section, we describe the set of actions of the requestor and the worker, and then we present the game states. Next, we give the payoffs with an explication of related constraints. Lastly, we detail the strategies of the worker and the requestor, respectively.

A. Players, actions, and game states

In the proposed game theoretical model, the game is played by the requestor and one typical worker, i.e., we set $N = \{r, w\}$. Their set of actions is given in Table I: A_r=**{S, T, D}**, A_w=**{S, C, I, M}**. The action **S** (Standby) is identical for the requestor and the worker as they both perform waiting as being standby for the new crowdsourcing service. In crowdsourcing task completion, the worker performs either **C** (Cooperate) or **M** (Misbehave), otherwise, it performs **I** (Interruption) in case it does not contribute to the task. The difference between selfish and malicious behaviors through actions **I** and **M** should be noted as the worker does not produce any false information in the crowdsourced data by performing the former action, whereas the latter does, which also leads to more negative consequences caused by the latter action. After receiving the crowdsourced data from the worker, the requestor will perform either **T** (Trust) or **D** (Distrust) depending on its own strategy, which will be discussed in Section IV-D.

Employing the set of actions illustrated in Table I to fit the crowdsourcing service process described in Section III, we consider 7 game states in SBG game, which are given in Table II with a description per each.

We define **I**, **TC**, **DC**, **TM** and **DM** states as $s^t = a^{t-1}$, i.e., the joint action at the previous time slot, as the same as the first experimentation conducted in [15]. This also means that game states are not homogeneous, and we name the states $s^t = a^{t-1}$ 'action states'. In **PE** state, both requestor and

TABLE I: Set of actions of the requestor and the worker

Player	Action	Description
Requestor	S (Standby)	Wait to begin a new crowdsourcing service process.
	T (Trust)	Trust the data crowdsourced by the worker and release the incentives.
	D (Distrust)	Distrust the data crowdsourced by the worker and lower the incentives.
Worker	S (Standby)	Wait to begin a new crowdsourcing service process.
	C (Cooperate)	Task is assigned and complete it with efforts.
	I (Idle)	Task is assigned but not engage in the task.
	M (Misbehave)	Task is assigned but perform misbehavior for crowdsourcing service.

TABLE II: Game states

State	Description
PE (Process End)	The crowdsourcing service process ends or the communication between the requestor and the worker fails.
S (Standby)	Both requestor and work stay at Standby waiting to begin the new crowdsourcing service process.
TC (Trust, Cooperate)	The requestor acts Trust, and the worker acts Cooperate.
DC (Distrust, Cooperate)	The requestor acts Distrust, and the worker acts Cooperate.
TM (Trust, Misbehave)	The requestor acts Trust, and the worker acts Misbehave.
DM (Distrust, Misbehave)	The requestor acts Distrust, and the worker acts Misbehave.
I (Interruption)	The worker is assigned the task but does not engage in the task, and thus the crowdsourcing service process is interrupted.

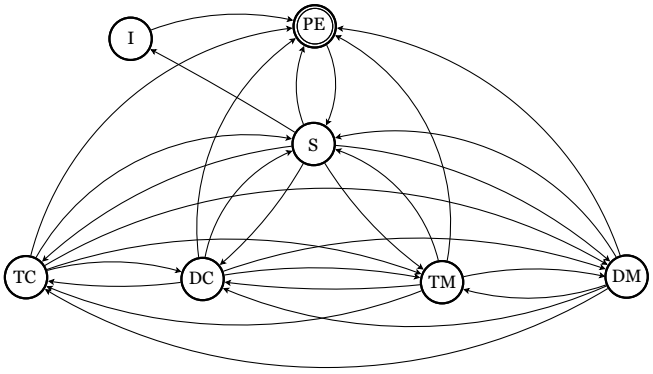

Fig. 2: Diagram of possible transitions between game states of the proposed model

worker cannot communicate with each other as the crowdsourcing process ends or their communication fails in this state. If a new crowdsourcing process is launched or the communication recovers, the game turns to a Standby state, where both requestor and worker perform action Standby for crowdsourcing service. After the task proposal is released, the worker will be recruited and assigned the task accordingly. Differing from other action states, the state **I** will be reached if the worker has done no contribution and it must return to **PE** as the crowdsourcing process will be viewed as ended. Or, one of **TC**, **DC**, **TM**, and **DM** states will be reached as the result of the current crowdsourcing service process. To avoid the game being played infinitely, we determine a goal number of interactions as total game rounds. This also means that is not necessary to return to **S** state from action states for every crowdsourcing service process, which can be observed through transitions between action states **TC**, **DC**, **TM**, and **DM**. Based on the above description, Fig. 2 presents the diagram of possible transitions between states, i.e., state space S={**PE**, **S**, **TC**, **DC**, **TM**, **DM**, **I**}, the initial state s^0 and terminal state $\bar{s}$ are both the state **PE**.

Besides, we denote all transitions to **PE** (P(·,PE)) by a value $\mathrm{P_E}$ representing the probability that communication between players fails. On the other hand, as the initial state, **PE** must move to the state **S** in which both players perform Standby, and thus P(PE,S)=1 for beginning a new crowdsourcing service process. To specify the transition P(S, I) such that the worker performs action **I**, we denote P(S, I) by a probability value $\mathrm{P_I}$. As for other transitions depending on the strategies for both requestor and worker, they will be explained in the following subsections.

B. Payoffs

Based on the game states defined in the previous section, the payoffs of the requestor and the worker (u_r, u_w) are given in Table III:

TABLE III: Payoff matrix of the requestor and the worker

r \ w	S	C	I	M
S	-	-	$-\mathrm{Cr_S}$,0	-
T	-	$\mathrm{Gr_{TC}}$ - $\mathrm{Cr_T}$, $\mathrm{Gw_{TC}}$ - $\mathrm{Cw_C}$	-	$-\mathrm{Cr_T}$ - $\mathrm{Lr_{TM}}$, $\mathrm{Gw_{TM}}$ - $\mathrm{Cw_M}$
D	-	- $\mathrm{Cr_D}$, $\mathrm{Gw_{DC}}$ - $\mathrm{Cw_C}$	-	$-\mathrm{Cr_D}$, $\mathrm{Gw_{DM}}$ - $\mathrm{Cw_M}$

Gr=Gain of the requestor; Lr=Loss of the requestor; Cr=Cost of the requestor; Gw=Gain of the worker; Cw=Cost of the worker.

Following the crowdsourcing service process defined in Section III, there are some constraints in payoffs:

- As the gain of the worker represents the incentives offered by the requestor, we impose $\mathrm{Cr_C}$=$\mathrm{Gw_{TC}}$, and likewise for $\mathrm{Gw_{DC}}$=$\mathrm{Gw_{DM}}$=$\mathrm{Cr_D}$.
- We impose $\mathrm{Gr_{TC}}-\mathrm{Cr_T}>\mathrm{Cr_T}$, this is because the overall payoff of the requestor, after a normal crowdsourcing service, should be greater than its cost of performing **T** action. Otherwise, it becomes discouraged to request, due to a non-reasonable payoff obtained. Similarly, as the malicious worker aims to cause damage such that it gains a higher overall payoff than the cost of misbehaving, thus $\mathrm{Gw_{TM}}-\mathrm{Cw_M}>\mathrm{Cw_M}$.
- $\mathrm{Cr_T}>\mathrm{Cr_D}>\mathrm{Cr_S}$ as performing **T** signifies greater incentives are required than the action **D**, and no incentives are offered when the worker performs action **I**.
- Attacking behaviors cost more than cooperating for the malicious worker as it has to create false information based on original crowdsourced data, for this, we impose $\mathrm{Cw_M}>\mathrm{Cw_C}$.
- By convention, the loss and the cost of the requestor should be equal to the gain of the malicious worker. Thus $\mathrm{Lr_{TM}}$+$\mathrm{Cr_T}$=$\mathrm{Gw_{TM}}$.
- As the malicious worker should obtain a higher overall payoff when its misbehavior successfully misleads the requestor, we consider $\mathrm{Gw_{TM}}-\mathrm{Cw_M}>\mathrm{Gw_{TC}}-\mathrm{Cw_C}$.
- $\mathrm{Gw_{DM}}-\mathrm{Cw_M}<0$, otherwise the attacker receives a positive payoff while the requestor performs distrust.

C. Strategies for the worker

In studying security by applying Game Theory, it is essential to define the threat model, specifically our assumptions about the behavioral model of malicious attackers. As stated in Section I, one of the limitations of existing IoT security solutions is the lack of effective distinction between selfish and malicious behaviors. For example, a Crowd-IoT worker performing inactive or selfish cannot be certainly determined as malicious type, it may perform action **I** with the purpose of maximizing its benefit by reducing energy consumption. Given this, the Byzantine Altruistic Rational (BAR) model [16, 17] can be employed for the threat model, where the worker is classified into three categories. Given this, as the type space Θ_w of the worker is unknown, we assume instead that the requestor hypothesizes a user-defined type space $\Theta_w^* = \{\theta_w^A, \theta_w^R, \theta_w^B\}$:

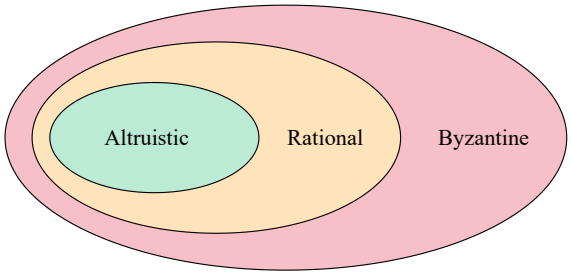

Fig. 3: Euler diagram of behaviors in BAR-based threat model

- **Altruistic** (θ_w^A): performs actively and correctly by carrying out its dedicated task. (Terms equivalent to 'altruistic': 'honest', 'unselfish', and 'self-denying'.)
- **Rational** (θ_w^R): follows the specified crowdsourcing protocol in case the resource needed is sufficient. Otherwise, it may deviate from the suggested protocol. (This type

of worker can also be referred to as 'greedy' or 'self-interested'.)

- **Byzantine** (θ_w^B): performs intentionally disturbing and misleading the requestor, which causes the current Crowd-IoT system to be harmed. (Also called 'compromised', 'malicious', or 'adversary'.)

TABLE IV: Worker types with the definition of strategies

Type	Beh.	Definition
θ_w^A	AC	$\pi_w(H^t, \mathrm{C}, \theta_w^A) = 1$
θ_w^R	RC	$\pi_w(H^t, \mathrm{I}, \theta_w^R) = \mathrm{P_I}$, $\pi_w(H^t, \mathrm{C}, \theta_w^R) = 1 - \mathrm{P_I}$
θ_w^B	RS	$\pi_w(H^t, \mathrm{C}, \theta_w^B) = 1/3, \pi_w(H^t, \mathrm{I}, \theta_w^B) = 1/3, \pi_w(H^t, \mathrm{M}, \theta_w^B) = 1/3$
	AM	$\pi_w(H^t, \mathrm{M}, \theta_w^A) = 1$

Beh.=Behavior; AC=Always Cooperate; RC=Rational Cooperate;
RS= Random Shift; AM=Always Misbehave;

Fig. 3 gives the Euler diagrams of the possible behaviors according to the BAR-based threat model. It can be observed that the altruistic ones can only perform positive actions, such type of worker is regarded as reliable and well-resourced. The rational can act not only altruistically, but also selfishly. The reasons for rational behavior are varied, such as resource constraints that force the worker to interact with others in a selective manner; or the worker, based on its own judgment of utility, desires to refuse cooperation rather than to engage in it. However, rational worker cannot behave in a way that threatens the current IoT system, while the byzantine kind possesses the highest number of possible behaviors, which include all behaviors of the previous two types. To rephrase, a byzantine worker can behave maliciously, altruistically, or rationally, depending on its purpose (i.e., being harmful, hiding its true motivation, and so on).

Table IV defines the behaviors per type of BAR-based threat model: AC worker cooperates in any case. RC worker will perform **I**, if it is not willing to contribute to the task, i.e., with the probability $\mathrm{P_I}$, otherwise it cooperates. In our work, we consider two malicious strategic misbehavior, namely RS and AM. RS worker randomly shifts its behaviors in every game round (i.e., the probability of performing action **C**, **I**, or **M** is identical), and the AM worker misbehaves for all time.

D. Strategies for the requestor

To track the mixed behavioral worker and output the action decision-making of the requestor, we adopt the algorithm *Harsanyi-Bellman Ad Hoc Coordination* (HBA) as the strategies of the requestor. HBA utilizes the concept of Bayesian Nash equilibrium in a planning procedure to find optimal actions in the sense of Bellman optimal control [14]. Here, we still fix r to represent the requestor, and thus the HBA is defined as $a_r^t \sim \arg\max_{a_r \in A_r} \mathbb{E}_{s^t}^{a_r}(H^t)$, where for any state $s \in S$, any action $a_r \in A_r$, and any history $\hat{H}$,

$$\mathbb{E}_s^{a_r}(\hat{H}) = \sum_{\theta_w^* \in \Theta_w^*} \Pr(\theta_w^* | \hat{H}) \sum_{a_w \in A_w} Q_s^{(a_r, a_w)}(\hat{H}) \pi_w(\hat{H}, a_w, \theta_w^*) \quad (1)$$

is the long-term payoff of the requestor taking the action a_r in the state s after history $\hat{H}$, and for all $a \in A$,

$$Q_s^a(\hat{H}) = \sum_{s' \in S} T(s, a, s') \left[u_r(s, a) + \gamma \max_{a_r \in A_r} \mathbb{E}_{s'}^{a_r}(\langle \hat{H}, a, s' \rangle) \right] \quad (2)$$

determines the long-term payoff for the requestor r when joint action a is executed in state s after history $\hat{H}$, $\gamma \in [0, 1]$ is the discount factor, and $\langle \hat{H}, a, s' \rangle$ in (2) denotes the concatenation of $\hat{H}$ and (a, s').

We consider that the behavior of the requestor r is completely specified by HBA, that is, in our model r has a single fixed type: $\Theta_r^+ = \{\theta_r^{HBA}\}$, where θ_r^{HBA} is outputted by (1) and (2). The Posterior Belief Probability (PBP) in (1) is defined as follows: for any history $\hat{H}$ and any $\theta_w^* \in \Theta_w^*$,

$$\Pr_w(\theta_w^* | \hat{H}) = \frac{L(\hat{H} | \theta_w^*) P_w(\theta_w^*)}{\sum_{\hat{\theta_w^*} \in \Theta_w^*} L(\hat{H} | \hat{\theta_w^*}) P_w(\theta_w^*)}, \quad (3)$$

where $P_w(\theta_w^*)$ is the prior belief (probability) that the worker w is of type θ_w^* before any action is observed, and $L(\hat{H} | \theta_w^*)$ is the likelihood of history $\hat{H}$ under the assumption that the worker w is of type θ_w^*. To specify the likelihood L in (3), we consider the sum posterior given in [14], which allows HBA to learn mixed type distribution. Thus, $\hat{\theta_w^*}$ in (3) refers to all possible hypothesized types of the worker w in $\hat{H}$.

V. Simulation

In this section, we have simulated various scenarios of the proposed game theoretical model by using MATLAB platform to validate its effectiveness. Besides, we have conducted a comparative analysis with other approaches modeling opponent players.

A. Scenario descriptions

In the simulation, the type-based behaviors of the worker defined in Table IV are all taken into consideration, namely AC, RC, RS, and AM. We designed two kinds of history, as illustrated in Table V.

TABLE V: Scenario description

π_r	controlled by HBA				
π_w	AC	RC	RS		AM
H	Ø	Ø	Ø	F	Ø

Ø= Empty history; F=Favorable history.

- An empty history where the requestor and the worker have yet to interact.

- A favorable history in which only **TC** is reached among action states, and this signifies a possible situation where the attacker hides its true behavioral type by performing only cooperate in the past and it starts misbehaving at a moment given, this also corresponds to the intelligent attack types of the insider attacker analyzed in [18].

The favorable history and RS misbehavior will be utilized for the comparative

analysis with other approaches since they, somehow, represent a more complex context in simulation.

B. Parameter settings

TABLE VI: Simulation parameter values

Parameter	Value	Parameter	Value
P_E	0.1	Lr_{TM}	0.55
P_I	0.2	Gw_{TC}	0.45
γ	0.9	Gw_{TM}	1
Gr_{TC}	1	Gw_{DC}	0.1
Cr_S	0.05	Gw_{DM}	0.1
Cr_T	0.45	Cw_C	0.2
Cr_D	0.1	Cw_M	0.45

As illustrated in Table VI, we set P_E 0.1. For the same IoT protocol, its failure rate calculated in [19] is 12%, and its communication stability given in [20] is 0.92, we take the average of these values to represent the probability that the communication fails, i.e., P_E=0.1 (obviously [0.12+(1-0.92)]/2=0.1).

r \ w	S	C	I	M
S	-	-	-0.05 ,0	-
T	-	0.45 , 0.25	-	-1 , 0.55
D	-	- 0.1 , -0.1	-	-0.1 , -0.35

TABLE VII: Payoff matrix with parameter values

In our proposed model, we consider the rational worker may perform action **I** in case of resource-constrained, e.g., in trouble of insufficient battery. In the above table, P_I is fixed at 0.2 as we employ the value of a parameter in work [21] describing the capability of solving battery issues, which is set to 0.8. Thus we consider P_I=1-0.8=0.2. The discount factor γ is fixed at 0.9 as in [14]. Besides, the initial prior considered is uniform prior for the empty history, where three types have identical prior values. For the favorable history, the initial prior is calculated by (3), as the posterior of the previous time slot. We fix the maximal gain to 1 for both the requestor and the worker. By respecting the constraints of payoffs mentioned in Section IV-B, the rest of the parameters are accordingly assigned as given in Table VII: $Gr_{TC}(1)$-$Cr_T(0.45)$>$Cr_T(0.45)$; $Gw_{TM}(1)$-$Cw_M(0.45)$>$Cw_M(0.45)$; $Cr_T(0.45)$>$Cr_D(0.1)$>$Cr_S(0.05)$; Gw_{DC}=Gw_{DM}=Cr_D=0.2; Gw_{TC}=Cr_C=0.4; $Gw_{TM}(1)$-($Cw_M(0.45)$>Gw_{TC})(0.45)-$Cw_C(0.2)$. Fixing the target number of interactions between the requestor and the worker, we run 50 game rounds for simulation.

C. Performance evaluation

To evaluate the performance of the proposed model, we focus on the changes in PBP (Posterior Belief Probability) given by (3) and the occurrence rate of game states obtained per scenario, and the average payoff of each scenario will also be assessed.

a) AC scenario: As we can see in Fig. 4, since the action **C** can also be performed by a Rational worker, this PBP value of the Rational type increases a little at the beginning. However, this value goes down rapidly due to no action **I** performed at all by the worker, and finally converges to 0. On the other hand, the PBP value of the Altruistic type converges to 1, which corresponds to the occurrence rate diagram in the same figure, showing that the requestor only performed the action **T** based on HBA to optimize the requestor's long-term payoff.

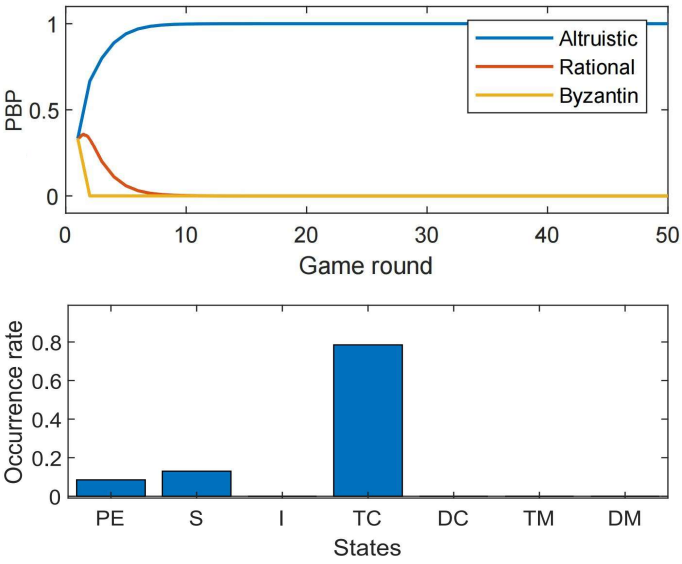

Fig. 4: Changes in PBP and the occurrence rate of game states in AC worker scenario

b) RC scenario: Differently, the changes in PBP values in Fig. 5 show that the Rational worker performed action **I** at the very early game rounds, and thus the PBP value of the Altruistic type decreases. After that, the worker cooperated with the requestor so that we can observe a rise in the PBP value of the Altruistic type around the tenth game round. With more and more action **I** being performed by the worker, the PBP of the Rational type increases steadily while the worker performed cooperate in some cases. Since **I** state must return to **PE** state by definition, it can be noticed that the occurrences of **PE** and **S** state are relatively higher than in other scenarios. Furthermore, it can be noticed that the occurrence of **I** state is much lower than that of **TC** state, which matches the value of P_I=0.2.

c) RS scenario: As the most complex malicious behavior defined in Table IV, the RS worker will randomly shift its actions between **I**, **C**, and **M**, it can be observed in Fig. 6 that the occurrences of state **I**, **DC**, and **DM** are close. The changes

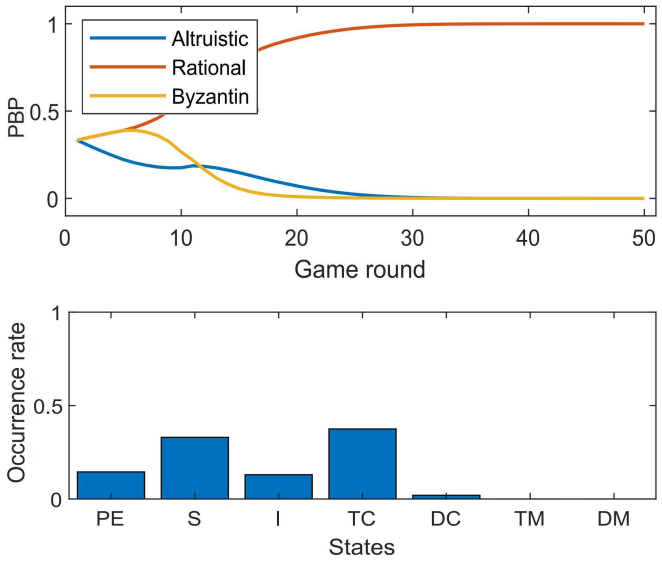

Fig. 5: Changes in PBP and the occurrence rate of game states in RC worker scenario

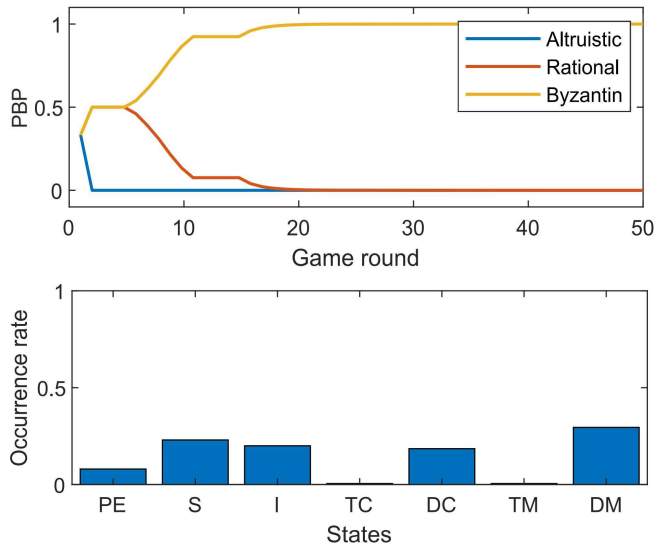

Fig. 6: Changes in PBP and the occurrence rate of game states in RS worker scenario

in PBP value demonstrate that the worker performed action **I** at the beginning. And then the value of the Rational type goes down immediately due to the misbehavior of the worker, we can also observe that the worker repeated action **I**, which leads to the PBP value of the Rational type remaining unchanged. On the other hand, the requestor performed very rarely action **T**, this is because performing **D** enables maximization of the long-term payoff. In other words, to cope with a RS malicious worker, performing action **D** is optimal based on the calculation of HBA. Finally, with more and more action **M** being performed, the PBP of the Byzantine type converges to 1.

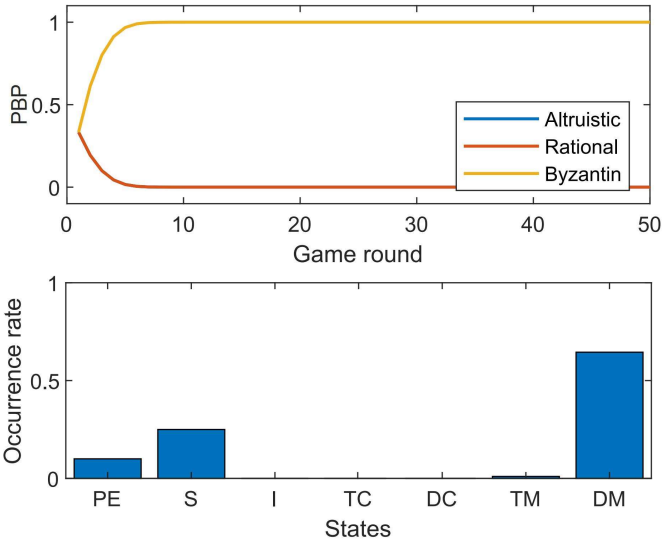

Fig. 7: Changes in PBP and the occurrence rate of game states in AM worker scenario

d) AM scenario: The AM worker will misbehave immediately from the beginning, and the requestor will perform action **T** at the first game round since the HBA maximizes its long-term payoff for the first interaction with the worker, which outputs that it will perform action **T**. As the malicious worker continuously misbehaves during the game, the PBP of the Byzantine type in Fig 7 increases till it converges to 1, and PBP values of Altruistic and Rational types are overlapping and both decrease to 0. Except one **TM** state is reached as the requestor performed action **T**, only **DM** is reached among all action states as the type of the worker is reasoned as Byzantine, the requestor will keep distrusting the worker. In our simulation, we run 50 game rounds even though the type of malicious worker has been identified. Indeed, such malicious worker will be removed from the group of workers once it is remarked as the attacker.

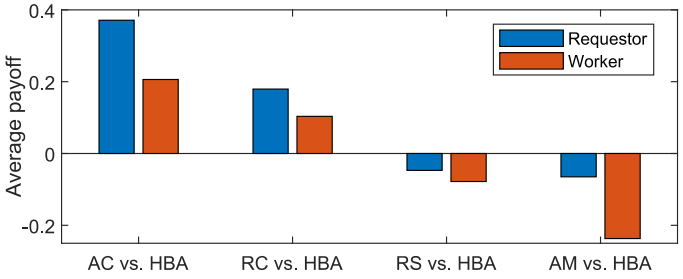

Fig. 8: Average payoff of the requestor and the worker in different scenarios

e) Average payoff per scenario: Fig. 8 illustrates the average payoff obtained by the requestor and the worker after running 50 game rounds. As we can see, AC and RC scenarios are win-win cases for the requestor and the worker. Although the RC worker performed more **I** action leading to a cost for the requestor, Fig. 8 indicates that their interactions

still output positive payoffs, i.e., the selfish worker is not considered as malicious. On the other hand, in RS and AM scenarios, the requestor and the worker both receive negative payoffs, but it is obvious that the worker loses much more, which means the requestor is able to minimize its loss when playing with a malicious worker. From the above performance evaluation based on changes in PBP values, the occurrence rate of game states, and average payoff in different scenarios, we notice that the proposed model encourages cooperation between the requestor and the worker, reduces the loss and the cost of the requestor if facing malicious worker, and penalizes the malicious worker. Moreover, the changes in PBP values become stabilized within 50 game rounds, which also results in the true types of the worker being identified accurately.

D. Comparative analysis with other approaches

In this subsection, we compare our work with approaches presented in [22] and [23] (thereafter referred as "QL" and "CJAL") to demonstrate the ability of the proposed model in presence of RS worker with a favorable past history. QL approach allows the players to learn the optimal action in a particular state by maximizing the expected payoff, and CJAL approach proposes that players learn the action frequencies of others conditioned on the modeling player's own action, which is called conditional joint action. We choose these two approaches since comparing our work with other game theoretical trust models remains demanding due to the variety of game formulations and payoff matrix, and these two approaches are reputable learning schemes that enable modeling the behavior of opponent players to optimize player's payoff. We involve RS scenario and a favorable past history for having a complex initialization of the game. QL and CJAL approaches both require adaptations to be simulated with a Crowd-IoT context, we thus retain the same parameter settings given in Table VI.

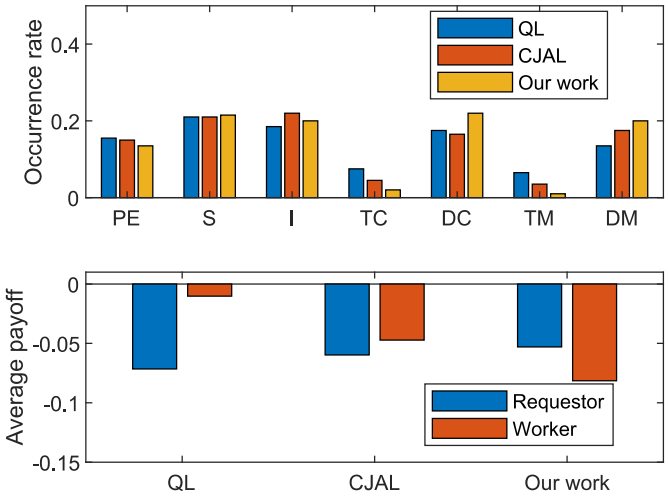

Fig. 9: Comparison between different approaches based on occurrence rate and average payoff in RS scenario with Favorable history

As shown in Fig. 9, in presence of RS malicious behavior, the proposed model outperforms the other two approaches in the average payoff obtained. By reviewing the occurrence rates of the three approaches, we can notice that the requestor of QL approach performs more action **T** as the **TC** and **TM** states are reached more. On the other hand, the action **D** is much less performed compared with CJAL and our work, particularly when the malicious worker performs action **M**. This also explains that the requestor of QL receives the worst average payoff among the three approaches, and its worker obtains a very small negative payoff, nearly no loss caused. As for CJAL, as the requestor of CJAL approach performs less **T** and more **D** actions, the misbehaving of RS worker does not create much damage to the requestor, and thus its loss is reduced. On the other hand, the success rate of misbehaving for the RS worker becomes smaller, and consequently, the malicious worker will receive a lower negative payoff compared with the QL worker, which means its misbehavior is punished. Besides, it can be seen in our work, the requestor's average payoff is higher than the malicious worker, but QL and CJAL approaches cannot address the issue where the payoff of the requestor is lower than that of the malicious worker. As a malicious attacker with favorable history corresponds to the newcomer attack behavior discussed in [18], where the attacker benefits from refreshing its historical record, our work shows resilience against this attack type.

VI. Conclusion and Future Work

To overcome several limitations of the current works and address misbehavior in Crowd-IoT, we presented, in this paper, a Stochastic Bayesian Game (SBG) where the requestor and the worker are regarded as heterogeneous Crowd-IoT participants with an asymmetric payoff matrix, and the selfish action and the malicious action can be distinguished. More importantly, the complex behavioral schemes of the worker, i.e., strategies, are considered in the SBG by applying the BAR threat model. We also involved these strategies in the simulation to assess the performance of the proposed model. From the observation in Section V through the simulation results of the performance evaluation of the proposed model under various scenarios and the comparative analysis with the other two approaches from the literature, we can notice that workers' behavioral types can be deduced accurately, and the requestor can perform optimal action by maximizing its long-term payoff. Therefore, the Altruistic and Rational worker performing cooperate can receive a positive payoff and the Byzantine worker's action can be tracked and punished. Through a comparison with two other approaches based on the most complex worker's strategy and history setting (RS worker and favorable history), our proposed model using HBA to specify the requestor's strategies outperforms other approaches in the average payoff obtained, and the numerical results also show the resilience of our work when dealing with the malicious worker whose history has been refreshed. Furthermore, as the worker also has the purpose of gaining a higher payoff, the optimal strategy when facing a requestor controlled by HBA is to perform

cooperate as much as possible, even though the successful attack allows the malicious worker to obtain the maximal gain in our designed payoff matrix. This also signifies that the cooperativeness between the requestor and the worker is encouraged in our proposed model. As discussed in [18], more complex behavioral schemes, especially malicious ones, can be considered as worker strategies in future work. We also plan to conduct the implementation with IoT devices in order to test the proposed model within a real Crowd-IoT environment.

Acknowledgments

This research was partially funded by the Federation Charles Hermite and by the ANR grant MATCHES, ref ANR-18-CE40-0019.

References

[1] Jeff Howe et al. "The rise of crowdsourcing". In: *Wired magazine* 14.6 (2006), pp. 1–4.

[2] Kenneth Li Minn Ang, Jasmine Kah Phooi Seng, and Ericmoore Ngharamike. "Towards crowdsourcing internet of things (crowd-iot): Architectures, security and applications". In: *Future Internet* 14.2 (2022), p. 49.

[3] Aleksandr Ometov et al. "Challenges of multi-factor authentication for securing advanced IoT applications". In: *IEEE Network* 33.2 (2019), pp. 82–88.

[4] *UC San Diego team's effort in DARPA's shredder challenge derailed by sabotage*. URL: https://jacobsschool.ucsd.edu/news/release/1150. (accessed: 03.02.2023).

[5] Pintu Kumar Sadhu, Venkata P Yanambaka, and Ahmed Abdelgawad. "Internet of Things: Security and Solutions Survey". In: *Sensors* 22.19 (2022), p. 7433.

[6] Arbia Riahi Sfar et al. "A roadmap for security challenges in the Internet of Things". In: *Digital Communications and Networks* 4.2 (2018), pp. 118–137.

[7] Bacem Mbarek, Mouzhi Ge, and Tomás Pitner. "Trust-based authentication for smart home systems". In: *Wireless Personal Communications* 117 (2021), pp. 2157–2172.

[8] Yosra Ben Saied et al. "Trust management system design for the Internet of Things: A context-aware and multi-service approach". In: *Computers & Security* 39 (2013), pp. 351–365.

[9] Runbo Su et al. "PDTM: Phase-based dynamic trust management for Internet of things". In: *2021 International Conference on Computer Communications and Networks (ICCCN)*. IEEE. 2021, pp. 1–7.

[10] Victor Naroditskiy et al. "Crowdsourcing contest dilemma". In: *Journal of The Royal Society Interface* 11.99 (2014), p. 20140532.

[11] Qin Hu et al. "Solving the crowdsourcing dilemma using the zero-determinant strategies". In: *IEEE Transactions on Information Forensics and Security* 15 (2020), pp. 1778–1789.

[12] Chuanxiu Chi et al. "Multistrategy repeated game-based mobile crowdsourcing incentive mechanism for mobile edge computing in Internet of Things". In: *Wireless Communications and Mobile Computing* 2021 (2021), pp. 1–18.

[13] Kun Wang et al. "Toward trustworthy crowdsourcing in the social internet of things". In: *IEEE Wireless Communications* 23.5 (2016), pp. 30–36.

[14] Stefano V Albrecht, Jacob W Crandall, and Subramanian Ramamoorthy. "Belief and truth in hypothesised behaviours". In: *Artificial Intelligence* 235 (2016), pp. 63–94.

[15] Stefano Vittorino Albrecht. "Utilising policy types for effective ad hoc coordination in multiagent systems". PhD thesis. The University of Edinburgh, 2015.

[16] Amitanand S Aiyer et al. "BAR fault tolerance for cooperative services". In: *Proceedings of the twentieth ACM symposium on Operating systems principles*. 2005, pp. 45–58.

[17] Amira Bradai, Walid Ben-Ameur, and Hossam Afifi. "Byzantine resistant reputation-based trust management". In: *9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing*. IEEE. 2013, pp. 269–278.

[18] Runbo Su et al. "Ensuring Trustworthiness in IoIT/AIoT: A Phase-Based Approach". In: *IEEE Internet of Things Magazine* 5.2 (2022), pp. 84–88.

[19] Benny Vejlgaard et al. "Coverage and capacity analysis of sigfox, lora, gprs, and nb-iot". In: *2017 IEEE 85th vehicular technology conference (VTC Spring)*. IEEE. 2017, pp. 1–5.

[20] Edgar Saavedra et al. "A Universal Testbed for IoT Wireless Technologies: Abstracting Latency, Error Rate and Stability from the IoT Protocol and Hardware Platform". In: *Sensors* 22.11 (2022), p. 4159.

[21] Arbia Riahi Sfar et al. "A game theoretic approach for privacy preserving model in IoT-based transportation". In: *IEEE Transactions on Intelligent Transportation Systems* 20.12 (2019), pp. 4405–4414.

[22] Montdher Alabadi and Zafer Albayrak. "Q-learning for securing cyber-physical systems: a survey". In: *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*. IEEE. 2020, pp. 1–13.

[23] Dipyaman Banerjee and Sandip Sen. "Reaching pareto-optimality in prisoner's dilemma using conditional joint action learning". In: *Autonomous Agents and Multi-Agent Systems* 15 (2007), pp. 91–108.