A Fair Reputation-based Incentive Mechanism for Cooperative Crowd Sensing

Rawan F. El Khatib[1], Nizar Zorba[1], and Hossam S. Hassanein[2]

[1]Department of Electrical Engineering, Qatar University, Doha, Qatar
[2]School of Computing, Queen's University, Kingston, Canada
{rawan.elkhatib, nizarz}@qu.edu.qa, hossam@cs.queensu.ca

***Abstract*—Crowd Sensing (CS) is a paradigm empowered by the pervasiveness of mobile smart devices, in which crowds of device owners cooperate to provide information about their surrounding environment. In this paper, we introduce the Data and Participant Assessment and Remuneration Scheme (DPARS) for cooperative CS applications. DPARS implements a three-stage procedure to estimate a fair reputation-based payoff for CS participants. We achieve this by first applying a consensus-based outlier detection technique on the received data. The output of this technique is used to statistically evaluate participants' reputations based on the Dirichlet process. Consequently, a fair payoff for every participant is determined by treating participants as coalitions of players in a cooperative game. Performance results indicate that our proposed scheme efficiently detects misbehaving participants, and decreases the amount of incentives allocated to them.**

I. INTRODUCTION

The ubiquitous presence of sensor-enriched smart devices has given rise to a promising sensing paradigm called *Crowd Sensing* (CS), in which ordinary citizens cooperate to collect and upload data about their surrounding environment. The collected data is processed to extract crowd intelligence information, and used to facilitate the provision of a wide array of services. These services include environmental monitoring, e-health care, urban dynamics sensing and crisis control, among others [1].

Broadly speaking, a CS application organizer is interested in monitoring an environmental phenomenon by collecting data from the crowd. The organizer typically begins by specifying multiple Points of Interest (PoIs), from which data is to be collected. The organizer also specifies the sensing duration, which is divided into equal-length sensing cycles depending on the application [2]. In order to ensure the sensing service quality, it is customary to accept a large number of data instances from different participants at every PoI, so that the actual value of the environmental variable is accurately deduced. At the end of the sensing duration, participants are compensated with monetary rewards according to an appropriate incentive scheme [3].

The inherent openness of these human-centric applications poses several challenges to the veracity of the collected data. On one hand, malicious participants may deliberately contribute erroneous data instances to hamper end-service delivery, or may even provide irrelevant and fake data instances to obtain monetary gain without actually performing any sensing tasks. On the other hand, corrupted data can be a result of inadequate sensor calibration and/or communication-related failures without deliberate participant interference., or can be a result of improper device placement (e.g., placing a cellphone inside one's pocket when monitoring noise levels). Given that the ability of the CS organizer to provide end-services is strongly dependent on the CS participants and the quality of the contributed data, we recognize that there is a pressing need to employ techniques to diagnose and overcome these situations.

More specifically, the challenges faced by the organizer are threefold. Firstly, the organizer must estimate the goodness of every data instance within the context of all the data received from the PoI, given that the ground-truth measurement is unknown to the organizer. Secondly, these goodness estimations must be used in a framework that allows the organizer to analyze and predict participants' behavior in the long run, and decide whether a participant systematically contributes corrupted data. And lastly, these estimations must be taken into consideration when determining the incentive payoff, such that the remuneration per participant is proportional to the value of his/her contribution to the CS system.

In this paper, we propose the Data and Participant Assessment and Remuneration Scheme (DPARS), which encompasses solutions to the above enlisted challenges. DPARS consists of three main stages: data assessment, participant assessment and incentive determination. In the first stage, DPARS uses a consensus-based outlier detection technique to assess the goodness of every data instant, and assigns it a score relative to all data received from the same PoI. This score is used in the second stage to gauge participants' behavior by incorporating a statistical reputation management system based on the Dirichlet Process [4]. Lastly, the incentive determination stage draws on concepts from cooperative game theory, where participants are treated as coalitions that cooperate to increase the utility of the organizer. These contributions are estimated based on the first two stages, and the Shapley value [5] is used to determine the fair remuneration per participant.

The remainder of the paper is organized as follows. In section II, we discuss related work. In section III, we introduce DPARS and present the three stages of our scheme in detail. Section IV presents and discusses simulation results. Finally,

978-1-5386-4727-1/18/$31.00 ©2018 IEEE

section V presents our conclusions.

II. Related Work

There exists a plethora of literature devoted to incentive design and reputation management for ad hoc networks in general, and CS systems in particular.

In [6], the authors proposed a reputation-based scheme to evaluate CS participants. Participants were clustered in three different groups based on their provided data measurements as compared to the ground-truth. The group of participants with the highest reputation was considered to be the winning group, and its contributed data was sent to the server. Accordingly, participants' reputations are updated using positive and negative scores based on their group. Another work [7] presented a robust reputation system that efficiently detects different kinds of corrupted participants, including collusion and on-off attacks. The proposed system was shown to improve the accuracy of data aggregation process at the server. However, these works lacked the implementation of an incentive mechanism based on the reputation values.

In terms of incentive design, the work in [3] focused on recurrent CS applications, where every round is considered as a single-shot game. The proposed incentive mechanism encouraged participants cooperation to sell their sensory data instances. Simulations showed that the scheme minimized the number of participants in every round, while controlling sample price and improving dropout rates. On the other hand, the authors in [8] proposed to promote global cooperation between participants by leveraging social ties. In particular, participants are motivated to increase their payoff by impacting the behavior of friends and socially-related participants. The proposed incentive mechanism was shown to achieve cooperation on a low budget, while providing a near optimal solution. Although similar works increase the level of cooperation between participants, they fail to take behavioral factors, such as reputation, into account.

The closest work to ours is [9], where the authors proposed to hire mobile trusted participants who provide ground-truth measurements for an increased amount of payoff. The ground-truths are compared to data provided from other participants, and used to build their reputation scores based on a subjective-logic belief model. The reputation scores were used to calculate the incentive payoff for every participant. As will be shown, our work is different because it does not require expensive ground-truth knowledge to estimate the truthfulness of data. Furthermore, we appeal to a more elaborate method for determining the individual payoff based on cooperative game theory.

III. DPARS Overview

In this section, we begin by introducing the key concepts and the system model. Then, we present the three stages of DPARS in detail: the data assessment stage, the participant assessment stage, and the incentive determination stage.

A. Preliminaries

We adopt a general CS system model, where the organizer divides the area of interest into M PoIs, and the entire sensing duration into equal-length epochs denoted by e. At every PoI m, there exists a set of participants $\mathcal{L}_m = \{P_{m,i} : i \in [1, 2, ..., L_m]\}$, where $P_{m,i}$ is the notation for participant i at PoI m. At each epoch e , the organizer receives a collection of data instances from $\mathcal{L}_m$ denoted by $\mathcal{X}_{e,m} = \{x_{e,m,i} : i \in [1, 2, ..., L_m]\}$, where $x_{e,m,i}$ denotes the data instance from participant $P_{m,i}$ at epoch e. At the end of the entire sensing duration, there exists a total incentive budget B. Every PoI is allocated a portion of B denoted by b_m, such that $\sum_{m=1}^{M} b_m = B$. Moreover, b_m is further divided among the set $\mathcal{L}_m$, where each participant's share is $b_{m,i}$, under the condition that $\sum_{i=1}^{L_m} b_{m,i} = b_m$.

DPARS works as follows. The first stage takes as inputs the set $\mathcal{X}_{e,m}$ from all the PoIs, and performs a consensus-based outlier detection algorithm on each set separately. Every data instance $x_{e,m,i}$ is assigned a score $s_{e,m,i} \in [0, 1]$. This score is an indication of the level of goodness of $x_{e,m,i}$ relative to the other data instances from the same PoI. At the next stage, DPARS statistically evaluates every participant using an accumulation of the scores based on the Dirichlet process. The output of this stage is an assigned reputation $R_i \in [0, 1]$ that serves as a measure of the participant's behavior in the long run. Finally, the last stage allocates fair portions of the incentive budget B, such that participants are rewarded based on their cooperation to increase the organizer utility. In the following subsections, we present in details the operation of the three stages of DPARS.

B. Data Assessment

As stated earlier, CS systems are greatly susceptible to the reception of corrupted data as a result of their openness. Hence, the organizer must employ a strategy to detect corrupted data instances and compromised participants, in order to mitigate their negative effect on end-service delivery. The goal of the data assessment stage is to produce the score $s_{e,m,i}$ for every data instance, where the score is an indication of its consistency with other data instances from the same PoI. It is worth mentioning here that in practical systems, the organizer performs this step without knowledge about the ground-truth, i.e., the real value of the environmental variable is not available at the organizer side, hence making the data assessment even more challenging.

Generally, techniques that aim at detecting anomalous data instances within a data sample are referred to as *outlier detection techniques* [10]. These can be broadly classified in two categories: parametric and nonparametric outlier detection. Parametric techniques estimate the amount of deviation from a predefined model that describes the underlying physical phenomenon. For example, in a temperature monitoring framework, each data instance is individually compared to a spatio-temporal model of temperature distribution, and classified as normal or abnormal. While these techniques do not require a certain number of data instances for accurate estimation, prior knowledge of the model is hard to obtain in most cases. On the contrary, nonparametric models operate only using the gathered data instances, without reliance on prior spatio-temporal models or ground-truths. We focus our attention on nonparametric consensus-based outlier detection techniques, where the algorithm establishes a rule of consistency and evaluates all data instances accordingly.

There is an extensive amount of research investigating consensus-based outlier detection techniques. From which, we choose to implement the *Local Outlier Factor* (LOF) technique [11]. Our motivation to implement this method among all others is twofold: a) it is computationally efficient, and b) it offers a flexible continuous score rather than a binary one, where data instances are categorized as either normal or outlier. The concept of LOF is based on the relative local density of points, where the output is an LOF score for every data instance based on a single parameter k. To understand how the LOF technique works, we first present definitions used in calculating the LOF, where it is understood that a *point* is a data instance in our context. For simplicity, we enlist all definitions for two arbitrary points i and j.

- The distance between points i and j ($d(i,j)$): the euclidean distance between points i and j.
- The k-distance of point i (d_i^k): the distance between point i to its kth closest neighbor.
- The k-neighborhood of point i (N_i^k): the set of all points whose distance from i is less or equal to d_i^k.
- The k-reachability distance of i to j ($\rho_{i\to j}^k$): the maximum of the distance between i and j, and j's k-distance. Mathematically:

$$\rho_{i\to j}^k = max\{d(i,j), d_j^k\} \tag{1}$$

- The k-local reachability density of point i (LRD_i^k): the inverse of the average of reachability distances in i's neighborhood. Mathematically:

$$LRD_i^k = \frac{|N_i^k|}{\sum_{\forall j\in N_i^k} \rho_{i\to j}^k} \tag{2}$$

- The k-local outlier factor of point i (LOF_i^k): the average ratio of the k-local reachability density of point i and its k neighbors. Mathematically:

$$LOF_i^k = \frac{\sum_{\forall j\in N_i^k} LRD_j^k}{LRD_i^k \cdot |N_i^k|} \tag{3}$$

In this manner, the LOF is a measure of the local reachability density of a given point compared to its k-neighborhood. Intuitively, if a point is an outlier, it will have low local reachability density compared to its k neighbors. This subsequently means that its LOF score will be high. Specifically, $LOF_i^k \in [0,\infty)$, where $LOF_i^k \approx 1$ indicates an inlier, and $LOF_i^k \gg 1$ indicates an outlier. In our context, the LOF procedure is applied to every $\mathcal{X}_{e,m}$ set, where every data instance receives an LOF score denoted by $LOF_{e,m,i}^k$. We note that this is dependent on our choice of the parameter k. To improve the performance, we repeat the procedure on

Algorithm 1 The LOF algorithm

Input: $\mathbf{k} = [k_1, k_2, ..., k_J]$, $\mathcal{X}_{e,m}, \forall m \in [1,2,...,M]$
Output: $s_{e,m,i}, \forall m \in [1,2,...,M], \forall i \in [1,2,...,L_m]$

for $m \leftarrow 1, M$ **do**
  for $i \leftarrow 1, L_m$ **do**
    for $j \leftarrow 1, J$ **do**
      $LOF_{e,m,i}^{k_j} = \frac{\sum_{\forall j\in N_i^k} LRD_j^k}{LRD_i^k \times |N_i^k|}$
    end for
    $\omega_{e,m,i} = max\{LOF_{e,m,i}^{k_j} : j \in [1,2,...,J]\}$
    $\hat{\omega}_{e,m,i} = max\{0, \omega_{e,m,i} - 1\}$
    $\bar{\omega}_{e,m,i} = max\{0, 0.5F(\hat{\omega}_{e,m,i}) - 1\}$
    $s_{e,m,i} = 1 - \bar{\omega}_{e,m,i}$
  end for
end for

different values $\mathbf{k} = [k_1, k_2, ..., k_J]$. We choose the final LOF as the maximum and denote it by $\omega_{e,m,i}$:

$$\omega_{e,m,i} = max\{LOF_{e,m,i}^{k_j} : j \in [1,2,...,J]\} \tag{4}$$

At this point, we have a mathematically solid measure of the outlierness of every data instance as compared to all data instances in its PoI. A final step in this stage is to convert this measure into a score $s_{e,m,i} \in [0,1]$. In order to do this, we adopt the regularization and normalization procedure proposed in [12]. The goal of this procedure is to translate the LOF values from $[1,\infty)$ into the range $[0,1]$. Specifically, the regularization step transforms $s_{e,m,i}$ into the interval $[0,\infty)$, as follows:

$$\hat{\omega}_{e,m,i} = max\{0, \omega_{m,e,i} - 1\} \tag{5}$$

Hence, inliers will now have an LOF value around 0, whereas outliers will have much larger than 1. The next step is to normalize the scores. We apply a statistical normalization procedure as it *increases the contrast between inlier and outlier scores* [12]. To do so, we map our LOF scores into a Gaussian distribution whose mean μ is the mean of all regularized LOF scores, and variance σ is the variance of all regularized LOF scores $F(\hat{\omega}_{e,m,i}) = N(\mu,\sigma)$. The normalized CDF scores are obtained by:

$$\bar{\omega}_{e,m,i} = max\{0, 0.5F(\hat{\omega}_{e,m,i}) - 1\} \tag{6}$$

Now, we have a regularized and normalized LOF for every received data instance as a measure of its degree of outlierness. Since we defined the score $s_{e,m,i}$ as a measure of goodness, then it is:

$$s_{e,m,i} = 1 - \bar{\omega}_{e,m,i} \tag{7}$$

The procedure is applied to all PoI data sets at every epoch, as shown in algorithm 1 in detail. In the next subsection, we will show how we will utilize this instantaneous data instance goodness score in the evaluation of the overall participant behavior.

C. Participant Assessment

In the data assessment stage, the goal was to obtain scores that measure the level of the relative goodness of the received data instance. Notably, these scores are instantaneous by nature, and they do not contain accumulative information about a participant in particular. Therefore, there is a need for a more comprehensive view in our scheme that offers objective scores representative of every participant behavior. Towards this end, we employ a *reputation management* mechanism in this stage. The goal is to produce a *reputation value* $R_i \in [0,1]$ that represents the degree of reliability of a given participant.

There are numerous reputation management mechanisms proposed in previous works, among which is a prevalent category based on Bayesian statistics [4]. In this category, a random variable follows a certain likelihood function F_s, and a dependent variable is assumed to have a prior distribution F_r. Then, the posterior distribution $F_{r,p}$ can be found from the well-known Bayes' theorem. Furthermore, if the posterior distribution belongs to the same class as the prior, this distribution is called the *conjugate prior* of the likelihood function F_s. The importance of this result is the algebraic convenience of obtaining a closed-form expression of the posterior that it directly related to the prior distribution.

The Dirichlet Process (DP) is a probability distribution whose range is itself a set of probability distributions. Furthermore, it is the conjugate prior of the general infinite-dimensional discrete distribution [4]. Therefore, we propose to use the DP as our distribution. Specifically, we think of our score $s_{e,m,i}$ as a discrete random variable with infinite possible values in $[0,1]$. Therefore, the reputation is assumed to have a prior DP distribution. After every observed score, we can update our assumption about the reputation. Here, the update becomes a simple parameter change using the successive observations to estimate the posterior distribution of the reputation, as we will show shortly.

Let the DP represent the prior distribution of participant's i reputation, denoted by $D_i(\alpha_i)$ where α_i is a distribution shape parameter called the base measure. Let the score $s_{e,m,i}$ represent evidence from which the reputation distribution shall be updated. From [4], we know that the posterior distribution of $D_i(\alpha_i)$ after an observation $s_{e,m,i}$, is also a DP where the base measure is found to be: $\alpha_i + \delta(s_{e,m,i})$. Here, $\delta(s_{e,m,i})$ is the unit impulse function existing at $s_{e,m,i}$ only.

Consequently, the posterior probability of a new observation $s_{E+1,m,i}$ given all previous observations until epoch E, is found as:

$$P(s_{E+1,m,i}|s_{1:E,m,i}) = \frac{\alpha_i + \sum_{e=1}^{E} \delta(s_{e,m,i})}{\alpha_i + E} \qquad (8)$$

In this context, we define a participant's reputation R_i as the expected future behavior of participant i given all observations $s_{e,m,i}$. Mathematically, this can be written as the expectation of the random variable $s_{e,m,i}$:

$$R_i = \frac{\alpha_i + 2\sum_{e=1}^{E} s_{e,m,i}}{2(\alpha_i + E)} \qquad (9)$$

Note that now we have constructed a long term view of each participant's behavior from successive instantaneous scores. At every epoch, the reputation of each participant's will be updated, and will eventually converge to a value in $[0,1]$ that is representative of the behavioral pattern. In particular, when there are no observations about participant i ($E = 0$), the value of R_i will be 0.5. This value is intuitive since at the beginning, we assume that the participant has the same probability of being normal or compromised. As more observations are collected about participant i, R_i will approach its real value. For a participant whose contributions are mostly rated $s_{e,m,i} = 0$, as $E \to \infty$, $R_i \to 0$, and vice versa.

D. Incentive Determination

In the incentive determination stage, the aim is to find the amount of fair payoff for every participant. Here, fairness means that every participant's remuneration should be proportional to its contribution to the organizer utility. We achieve this objective by using the *Shapley value* from cooperative game theory [5]. In a cooperative game, players form coalitions to obtain a certain gain, and the Shapley value determines the unique distribution of the total gain, such that it is fair and efficient. Specifically, for a game with G players, the Shapley value payoff for player g is:

$$\phi_g = \sum_{C \subseteq G \setminus g} \frac{|C|!(G - |C| - 1)!}{G!} (v(C \cup g) - v(C)) \qquad (10)$$

where C is a coalition from G without player g, and $v(.)$ is a valuation function that measures the utility of a coalition of players depending on the nature of the game, such that $v(\emptyset) = 0$. Clearly, the Shapley value can be interpreted as the average marginal contribution of player g over all the possible permutations in which the coalition can be formed.

To determine the fair incentive payoff for every participant, we implement a two-step procedure. In the first step, we treat the set of all PoIs as a coalition of M players, and use the Shapley value to divide B into fair payoffs for every PoI b_m. In the second step, we view the set of all participants at PoI m as a coalition of L_m players, and further divide b_m into fair portions for every participant $b_{m,i}$. Specifically, the fair payoff for a PoI m is:

$$\phi_m = \sum_{C \subseteq M \setminus m} \frac{|C|!(M - |C| - 1)!}{M!} \cdot (v_{PoI}(C \cup m) - v_{PoI}(C)) \qquad (11)$$

where $v_{PoI}(C)$ is a valuation function that measures the organizer utility achieved by the cooperation of all the PoIs in C. We define it as:

$$v_{PoI}(C) = e^{\gamma I_m} \sum_{m \in C} \hat{R}_m \qquad (12)$$

where $\hat{R}_m$ is the sum of the individual participant reputations in PoI m, I_m is the number of participants in m whose reputation is larger than a threshold R_{th}, and $\gamma \in [0,1]$

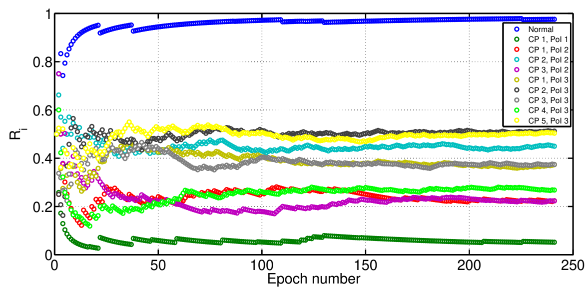

Fig. 1: Evolution of R_i for some participants.

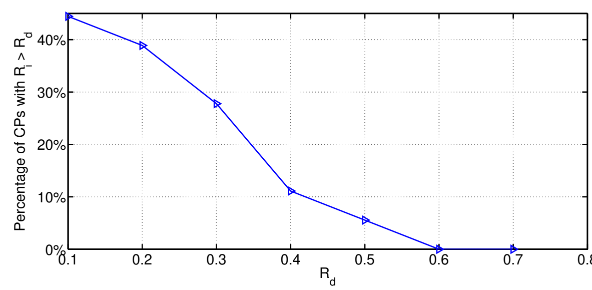

Fig. 2: Percentage of CPs with $R_i > R_d$.

is an adjusting factor. We refer to the multiplicative term $e^{\gamma I_m}$ as the Synergy Factor (SF). Here, *synergy* refers to the ability of a coalition to cooperate and create a whole whose utility is greater than the sum of the individuals. In other words, the SF is a boosting factor dependent on a coalition's potential to produce trusted knowledge. The logic behind this is that the organizer's utility is increased when participants provide consistent data about the monitored environmental variable, and this is reflected in the number of reputable participants I_m. Thus, the valuation function rewards PoIs with reputable participants, while it penalizes PoIs with under-achieving participants. Given the Shapley value for each PoI m, the fair PoI level payoff is:

$$b_m = \frac{\phi_m B}{\sum_{m=1}^{M} \phi_m} \tag{13}$$

Having found the allocated budget for every PoI, we use the Shapley value in a second step to further divide it among $\mathcal{L}_m$, by applying Eq. (10) on L_m participants and using a participant valuation function as follows:

$$v_{Part}(C) = \sum_{i \in C} \frac{{R_i}^2}{\sum_{i \in C} R_i} \tag{14}$$

In this manner, the payoff for every participant is a function of the reputation of the entire PoI (Eq. (12)) and its own reputation level among the coalition of participants at m. As in the previous step, reputable participants receive higher rewards than under-achieving participants. The exact value is found as:

$$b_{m,i} = \frac{\phi_{m,i} b_m}{\sum_{i=1}^{L_m} \phi_{m,i}} \tag{15}$$

IV. Performance Evaluation

In this section, we first introduce our simulation setup environment and parameters, then present performance evaluation results.

A. Simulation Setup

We conduct simulation experiments to evaluate the performance of DPARS. We study the performance of our scheme in a cooperative noise monitoring CS application using MATLAB. We assume we have $M = 3$ PoIs, at each one there exists 25 participants. Each PoI corresponds to a location where there are different nearby conditions, and different noise levels. We generate random data based on the measurements provided in [13]. Let the 3 PoIs correspond to vehicle traffic, construction and quiet places, respectively. We generate data instances at each PoI from normal distributions with means 75.4, 73, and 67.7 dBA, where dBA is the unit of noise levels measured using the A-(frequency) weighted network, respectively. We divide our participants into two categories: normal and corrupted. For normal participants, we set the variance of the distribution to 0.2 at every PoI. For Corrupted Participants (CPs), we deliberately tamper with the data instances of a randomly selected set of participants. We assume there exists 1, 3 and 5 corrupted participants at the PoIs, respectively. For the evaluation of the data at each epoch, we set $k = [3, 4, 5]$. Furthermore, let $\alpha_i = 1$ for all participants, and we update the reputation value after each epoch.

B. Simulation Results

We begin by studying the evolution of the reputation value for some participants in Fig. 1, where we plotted the reputation per epoch for a normal participant chosen randomly, and all the CPs in the 3 PoIs. Firstly, we note that the normal participant starts with $R_i = 0.5$ as all other participants, but quickly converges to $R_i = 1$. Secondly, note that all CPs tend to fluctuate in their reputations, as a result of the random fabricated data. In general, all reputation values converge around epoch $e = 100$. More specifically, we note that CP 1 from PoI 1 has the lowest reputation among all CPs, because it is in a high-achieving PoI and the outliers are very easy to detect. Whereas the set of CPs from PoI 2 and 3 have varying reputations in a slightly higher range. This is because when the number of outliers increase, they might be mistaken for a group of inliers by themselves, hence giving higher scores and reputation values. However, all these CPs converge to a maximum reputation of 0.5, which is considered to be the borderline in our reputation management scheme.

To verify this result, in Fig. 2 we have plotted the percentage of CPs with varying decision threshold for the reputation, denoted by R_d. From the Figure, we notice that the average drops to around 10% at $R_d = 0.5$. This verifies that our proposed scheme detects corrupted data and participants efficiently.

In Fig. 3, we plot the fraction of the allocated budget for every PoI for different values of the adjusting parameter γ, where $R_{th} = 0.6$. Note that for $\gamma = 0$, the allocation of

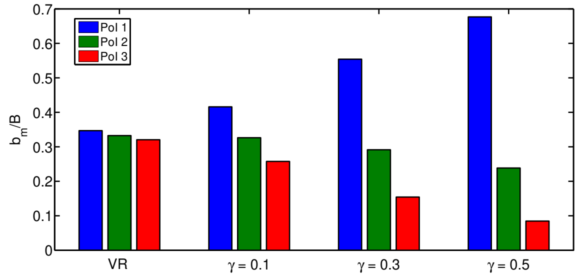

Fig. 3: Fraction of allocated budget per PoI.

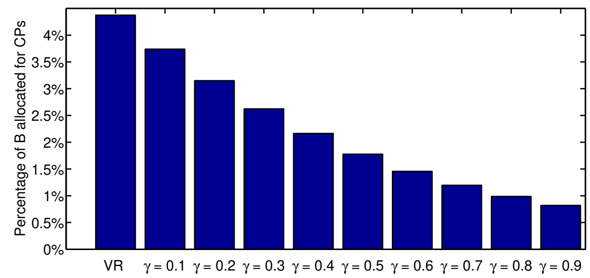

Fig. 4: Percentage of B allocated for CPs.

the budget degenerates to the **Variable Reward** (VR) scheme in [9], where participants are rewarded according to their relative reputations, without considering the cooperation or synergy between participants. For $\gamma > 0$, the distribution of the allocated budget among different PoIs becomes increasingly sparse. This is achieved by allocating a larger payoff for highly-reputable PoIs, and less for under-achieving PoIs. In our simulation, it is obvious that PoI 1 achieves the highest synergy as all participants except 1 are reputable, as opposed to PoI 3. As we have mentioned earlier, the logic behind the SF is that the organizer has high trust in the aggregated data. The exact amount of reward and penalty is controlled by γ.

Finally, in Fig. 4, we show the percentage of B that was allocated to all the CPs in the CS applications. Again, we compare the performance of our scheme to the VR scheme in [9] by letting $\gamma = 0$. We note that as increase γ, the total allocated budget for CPs decreases gradually, as the reputable participants receive higher rewards. Hence, our scheme minimizes the amount of B allocated for CPs. Additionally, the organizer can control the amount of penalty by changing γ, depending on the application requirements.

V. CONCLUSIONS

In this paper, we proposed the Data and Participant Assessment and Remuneration Scheme (DPARS) for cooperative crowd sensing applications. The DPARS aims to address three main challenges: to evaluate the goodness of every received data instance without prior knowledge about the ground-truth, to construct a long-term view of each participant's behavior using the data good estimations, and determine the fair payoff for every participant in the CS application. In the first stage of DPARS, the data is assessed using a consensus-based outlier detection mechanism, where each data instance receives a score of goodness. In the second stage, the reputation of each user is updated using the data scores based on the Dirichlet process. And finally, the last stage distributes the total budget among the PoIs and the participant using the Shapley value, based on a valuation function of the total reputations of the PoIs and participants. Simulations results show that our scheme detects misbehaving participants and minimizes the amount of incentive allocated to them.

ACKNOWLEDGMENT

This work was made possible by NPRP grant NPRP 9-185-2-096 from the Qatar National Research Fund (a member of The Qatar Foundation). The statements made herein are solely the responsibility of the authors.

REFERENCES

[1] B. Guo et al., "Mobile crowd sensing and computing: The review of an emerging human-powered sensing paradigm," *ACM Comput. Surv.*, vol. 48, no. 1, pp. 1-31, Sep. 2015.

[2] R. F. ElKhatib et al., "Multi-tasking for cost-efficient mobile crowd-sensing under uniformity constraints," in *Proc. of IEEE International Communications Conference (ICC)*, Kansas City, May 2018.

[3] L. G. Jaimes et al., "A cooperative incentive mechanism for recurrent crowd sensing," in *Proc. of SoutheastCon 2015*, Fort Lauderdale, FL, 2015.

[4] T. S. Ferguson, "A bayesian analysis of some nonparametric problems," in *The Annals of Statistics*, vol. 1, no. 2, pp. 209-230, Mar. 1973.

[5] L.S. Shapley, "A value for n-person games." in *Annals of Mathematics Studies*, no. 28, pp. 307-317, 1953.

[6] M. A. Alswailim et al., "A reputation system to evaluate participants for participatory sensing," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, Washington, DC, 2016.

[7] H. Mousa et al., "A reputation system resilient against colluding and malicious adversaries in mobile participatory sensing applications," in *Proc. of IEEE Annual Consumer Communications and Networking Conference (CCNC)*, Las Vegas, NV, 2017.

[8] G. Yang et al., "Promoting cooperation by the social incentive mechanism in mobile crowdsensing," in *IEEE Communications Magazine*, vol. 55, no. 3, pp. 86-92, Mar. 2017.

[9] F. Restuccia and S. K. Das, "FIDES: A trust-based framework for secure user incentivization in participatory sensing," in *Proc. of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, Sydney, NSW, 2014.

[10] S. Azmy et al., "Robust Quality Metric for Scarce Mobile Crowd-Sensing Scenarios," in *Proc. of IEEE International Communications Conference (ICC)*, Kansas City, May 2018.

[11] M. M. Breunig et al., "LOF: identifying density-based local outliers," in *Proc. of the ACM SIGMOD International Conference on Management of Data*, Dallas, TX, 2000.

[12] H.-P. Kriegel et al., "Interpreting and unifying outlier scores," in *Proc. of the SIAM International Conference on Data Mining*, Arizona, 2011.

[13] T. P. McAlexander et al., "Street-level noise in an urban setting: assessment and contribution to personal exposure," in *Environmental Health*, vol. 14, no. 1, Feb. 2015.