A Reputation Management Approach for Resource Constrained Trustee Agents

Han Yu[1], **Chunyan Miao**[1], **Bo An**[2], **Cyril Leung**[3], **and Victor R. Lesser**[4]
[1]Nanyang Technological University, Singapore
[2]The Key Lab of Intelligent Information Processing,
Institute of Computing Technology, Chinese Academy of Sciences, China
[3]University of British Columbia, Canada [4]University of Massachusetts Amherst, USA
{*yuha0008, ascymiao*}*@ntu.edu.sg, boan@ict.ac.cn, cleung@ece.ubc.ca, lesser@cs.umass.edu*

Abstract

Trust is an important mechanism enabling agents to self-police open and dynamic multi-agent systems (ODMASs). Trusters evaluate the reputation of trustees based on their past observed performance, and use this information to guide their future interaction decisions. Existing trust models tend to concentrate trusters' interactions on a small number of highly reputable trustees to minimize risk exposure. When a trustee's servicing capacity is limited, such an approach may cause long delays for trusters and subsequently damage the reputation of trustees. To mitigate this problem, we propose a reputation management approach for trustee agents based on distributed constraint optimization. It helps a trustee to make situation-aware decisions on which incoming requests to serve and prevent the resulting reputation score from being affected by factors out of the trustee's control. The approach is evaluated through theoretical analysis and within a simulated, highly dynamic multi-agent environment. The results show that it can achieve close to optimally efficient utilization of the trustee agents' collective capacity in an ODMAS, promotes fair treatment of trustee agents based on their behavior, and significantly outperforms related work in enhancing social welfare.

1 Introduction

In environments that can be modeled as open multi-agent systems (ODMASs) (e.g., e-commerce systems, sensor networks, crowdsourcing systems), agents (both human and artificial) with diverse backgrounds may be self-interested and sometimes even malicious [Sabater and Sierra, 2005]. Agents in these ODMASs often need to rely on the services (e.g., expertise, resources) of other agents in order to achieve their goals. The interactions among agents, coupled with the uncertainty in other agents' behaviors, put their interest at risk. Throughout this paper, we refer to agents who provide services to other agents as *trustee agents* and agents who need to rely on them as *truster agents*.

Over the years, a number of computational models have been proposed to carry out trust management in these systems. In existing trust models, truster agents evaluate the interaction outcomes based on the quality of the results received. Usually, the timeliness of task completion is part of the quality metric [Jøsang *et al.*, 2007; Yu *et al.*, 2010]. A trustee's resources are assumed to be able to cope with the amount of requests effectively even as it becomes highly reputable (e.g., in a web-service system). Thus, existing trust models often direct truster agents to delegate tasks to the most reputable trustee agents they can find to improve the expected quality of the interaction results. Existing trust models are not specific on how trustee agents should respond to incoming requests. Since trustee agents are assumed not to be resource constrained, existing trust models expect them to accept all incoming requests (*accept-when-requested*) in order to maximize their own potential gain.

In applications where trustees are resource constrained (e.g., human workers in a crowdsourcing system, nodes in a sensor network), this approach may overwhelm reputable trustees with requests, causing long delays for trusters and subsequently causing them to give negative ratings to the trustees. In this case, the situation can be modeled as a *congestion game* [Monderer and Shapley, 1996] where the payoff for each truster depends not only on the trustworthiness of the trustees, but also on how many other trusters are delegating tasks to the same trustee as it is doing. Since the unsatisfied trusters may not be aware of other trusters' choices, they will negatively rate the performance of the trustee which is not fair to the trustee in this situation. This phenomenon has been shown in [Yu *et al.*, 2012] to negatively affect the social welfare of the system.

To enable existing trust models to achieve good performance in ODMASs with resource constrained trustees, we propose a Distributed Request Acceptance approach for Fair utilization of Trustees (DRAFT). The basic idea of DRAFT is illustrated in Figure 1. It can be visualized as a set of control valves that enables each individual trustee agent to dynamically determine how many new interaction requests of different types of tasks should be accepted at each time step using its local knowledge about its own situation. Such decisions strike a balance between maximizing the potential reward for the trustee agent and minimizing the delay experienced by the requesting truster agents.

Theoretical analysis and extensive simulations demonstrated that DRAFT helps trustworthy trustee agents maintain

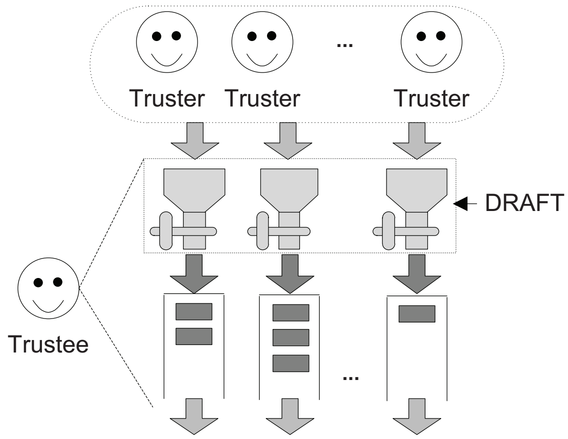


Figure 1: The basic idea of DRAFT.

good reputation values regardless of truster agents' choices for interaction partners. If all trustee agents adopt DRAFT, tasks can be allocated efficiently among trustee agents based on their reputation without the overhead of coordination among truster agents. The social welfare of an ODMAS can also be significantly enhanced compared to using the prevailing *accept-when-requested* approach.

2 Motivating Example and Related Work

The level of congestion in a trustee's pending task queues is determined by both the rate of arrival of task requests and the rate at which the trustee serves them. While the serving rate is out of the control of trust models, the arrival rate is affected by the Trust-aware Interaction Decision-making (TID) approach used. The most commonly used TID approach is a greedy approach - each truster selects the most reputable trustee it can find to serve its requests with a predetermined percentage of interactions reserved for trying out unfamiliar trustees [Jøsang and Ismail, 2002; Teacy *et al.*, 2005; Weng *et al.*, 2010].

Such an approach is employed by e-commerce systems such as Taobao.com. The sellers (trustees) on Taobao.com are mainly small and medium businesses which have limited resources in terms of staff and capital. Reputation is an important factor determining the amount of business for each seller on Taobao.com. As the competition among the sellers is fierce, sellers take their reputation very seriously and are generally reluctant to decline incoming orders. The situation is similar to the problem formulated in this paper. As recently in 2012, reports from China about over-worked sellers on Taobao.com dying of exhaustion have surfaced [Zhang, 2012]. The common features of such cases are: 1) the victims are young (in their 20s); 2) the e-shops under their management are highly reputable on Taobao.com; 3) they receive large number of orders; 4) their e-shops are small and medium sized enterprizes which require them to personally handle most business activities; and 5) they misjudged their capacity of serving incoming orders effectively. Although these are extreme cases, they demonstrate the importance of protecting the wellbeing of resource constrained trustees.

Motivated by these observations, we propose DRAFT in this paper. To the best of our knowledge, it is the first approach designed to help resource constrained trustee agents determine how to react to incoming task requests to protect their reputation through minimizing the delay experienced by trusters.

3 System Model with Resource Constrained Trustees

In this section, we formally define a system model in which trustee agents are resource constrained and the delays experienced by truster agents are partially due to their collective task delegation choices. In such a system, the trustee agents can be viewed as the base set of congestible resources. A truster agent's payoff is affected by which trustee agents it delegates tasks to and how many other truster agents are delegating tasks to the same set of trustee agents.

In a practical system, a trustee i may only be qualified to perform up to C different types of tasks. In computational trust literature, the trust evidence of a trustee agent i in serving different types of task requests are often kept separately (e.g., a trustee agent can have a high reputation in selling T-shirts but a low reputation in selling gloves, although it is able to sell both items). This is referred to as the *context* of the trust evidence evidence [Weng *et al.*, 2010] and denoted as $c \in (1, \ldots, C)$. The maximum utility for successfully completing one task of each type is represented by the vector $\vec{G} = (G^c_{max})$.The general effort (e.g., cost, physical or mental exertion) required to complete one task of each type is denoted as $\vec{e} = (e^c)$.The vector $\vec{Q}_i(t) = (Q^c_i(t)), Q^c_i(t) = 0$ represents the number of tasks stored in queues for different types of task of trustee i at the beginning of time step t. Due to resource constraints, i is able to expend an effort of up to e^{max}_i to serve the tasks in its task queues per time step. The value of e^{max}_i depends on an agent's ability and willingness and may differ from agent to agent.

Among the dimensions commonly used to evaluate the trustworthiness of a trustee agent, timeliness of serving requests is unique. It is affected not only by the ability and willingness of the trustee agent, but also by the collective task delegation decisions made by the truster agents. For each type of tasks, there is a generally accepted maximum elapse time represented by the vector $\vec{T} = (T^c)$. The actual gain for i for completing a task depends on whether the result is rated successful by the requesting agent:

$$g^c_i(t) = \begin{cases} G^c_{max}, & \text{if interaction is successful} \\ 0 & \text{, Otherwise} \end{cases} \tag{1}$$

The interaction is deemed successful only if the result produced by i is correct and received by the requesting agent within the expected elapse time.

At each time step t, the incoming task requests received by i are denoted as $\vec{\lambda}(i) = (\lambda^c_i(t))$.. The exact number depends on the collective task delegation decisions made by the truster agents in the ODMAS which may be influenced by i's reputation. The reputation of i in each context at any given time t, which can be assumed to be common knowledge, is represented by $\gamma^c_i(t)$. As our focus in this paper is on decision-making by trustee agents, we do not discuss how i's reputation is evaluated. Instead, we assume the existence of a reputation model such as [Jøsang and Ismail, 2002] which produces $\gamma_i^c(t)$ in the range of [0, 1] with 1 denoting completely trustworthy and 0 denoting not trustworthy at all.

4 The *DRAFT* Approach

The common goal of multi-agent trust models is to maximize the social welfare (which is the summation of utility derived from all interactions) in the system. We define a utility function $u_i^c(\mu)$ to represent the utility derived by trustee agent i from completing μ tasks of type c. $u_i^c(\mu)$ will be defined in later sections of this paper. Then, the social welfare of an ODMAS is the summation of $u_i^c(\mu)$ over all i and c. When trustees are resource constrained, the optimization objective can be expressed as:

$$\text{Maximize:} \quad \frac{1}{T}\sum_{t=0}^{T}\sum_{i,c} u_i^c(\mu_i^c(t)) \tag{2}$$

$$\text{Subject to:} \quad \sum_{c}(\mu_i^c(t) \cdot e^c) \leqslant e_i^{max} \tag{3}$$

where the vector $(\mu_i^c(t))$ denotes the actual number of tasks of each type completed by i at t. It depends on the availability of resource at trustee agent i's disposal as well as its trustworthiness.

In systems where $e_i^{max} \to \infty$, Constraint (3) is redundant (i.e., trustees are not resource constrained). In this case, greedy task delegation decisions made by truster agents, combined with the *accept-when-requested* approach by trustee agents can maximize (2). However, under our system model, which is similar to a congestion game, the overall level of congestion in the system must be minimized in order to maximize (2). The queuing dynamic of $Q_i^c(t)$ is:

$$Q_i^c(t) \leftarrow Q_i^c(t-1) + A_i^c(t-1) - \mu_i^c(t-1) \tag{4}$$

where $A_i^c(t)$ is the number of new type c task requests accepted by i at time step t; $\mu_i^c(t)$ is the actual number of type c tasks completed by i during time step t. $\mu_i^c(t)$ is determined entirely at the discretion of i. The objective of DRAFT is to help i determine the appropriate value of $A_i^c(t)$ at each time step t so that i would not be overwhelmed by incoming requests.

Although the decisions are made for individual trustees, DRAFT needs to ensure the collective decision also maximizes (2). In order to do so, the first challenge is to quantify the level of congestion in the system. We adopt the *Lyapunov functions* [Neely, 2010] to measure congestion. They are scalar functions which are easy to construct and interpret. Based on this concept, we define the overall level of task queue congestion in an ODMAS at any t as:

$$L(t) = \sum_{i,c}(Q_i^c(t))^2 \tag{5}$$

A small value of $L(t)$ indicates that all $Q_i^c(t)$ are having a low level of congestion.

$L(t)$ can be trivially minimized by making trustee agents reject all incoming requests and, therefore, keeping all $Q_i^c(t) = 0$ for all the time. However, this is not a desirable mode of operation for any ODMAS. Instead, we want to limit the growth of the overall level of congestion while filling in spare capacities whenever they become available with new requests (if there are enough new requests from trusters). To tackle this challenge, DRAFT minimizes the *Lyapunov drift* [Neely, 2010] which is expressed as a conditional expectation:

$$\Delta(t) = \mathbb{E}\{L(t+1) - L(t) \mid \boldsymbol{Q}(t)\} \tag{6}$$

where $\boldsymbol{Q}(t) = (\vec{Q}_1(t), \ldots, \vec{Q}_C(t))$.The smaller the value of $\Delta(t)$, the more efficient the utilization of the collective resources possessed by trustee agents.

Simply balancing the workload on trustee agents can produce small $\Delta(t)$ values. However, as trustees may have differing trustworthiness, such an approach gives no guarantee on the quality of the results produced. The reward that can be derived from delegating tasks to trustee agents following a given plan should be maximized.

DRAFT addresses this challenge by formulating a combined optimization objective as a reward-minus-drift expression:

$$\rho \times reward(t) - \Delta(t) \tag{7}$$

$\rho > 0$ is a chosen constant that affects the relative emphasis given to the reward and the drift respectively.

$reward(t)$ is defined as the total utility that can be derived from new tasks accepted by trustees at t. Since both $reward(t)$ and $\Delta(t)$ partially depends on the behavior of the trustees which cannot be known with certainty, their values can only be estimated. The expected reward from a request acceptance decision can be expressed by:

$$reward(t) = u_i^c(A_i^c(t)) = A_i^c(t) \cdot g_i^c \tag{8}$$

In the long run, the actual utility g_i^c is related to the probability of i producing a correct result within the stipulated deadline. Although this probability cannot be definitively known, it can be approximated using i's reputation in performing each type of tasks which can be known by agents in a community including i itself. Therefore,

$$u_i^c(A_i^c(t)) = A_i^c(t) \cdot \gamma_i^c(t) \cdot G_{max}^c \tag{9}$$

At each t, the *Lyapunov drift* $\Delta(t)$ is affected by the existing tasks in the task queues and the new tasks admitted into each queue. Thus, it can be estimated by the expressions:

$$\Delta(t) = Q_i^c(t) \cdot A_i^c(t) \tag{10}$$

Therefore, the optimization objective for DRAFT is:

$$\text{Maximize:} \quad \frac{1}{T}\sum_{t=0}^{T}\sum_{i,c}[\rho \cdot u_i^c(A_i^c(t))] \tag{11}$$

$$\text{Subject to:} \quad \sum_{c}(A_i^c(t) \cdot e^c) \leqslant e_i^{max} \text{ for all } i \tag{12}$$

$$0 \leqslant A_i^c(t) \leqslant \lambda_i^c(t) \text{for all } i \text{ and } c \tag{13}$$

By substituting (9) into (11), we have

$$\begin{aligned}
&\frac{1}{T}\sum_{t=0}^{T}\sum_{i,c}[\rho \cdot A_i^c(t) \cdot \gamma_i^c(t) \cdot G_{max}^c - Q_i^c(t) \cdot A_i^c(t)] \\
&= \frac{1}{T}\sum_{t=0}^{T}\sum_{i,c}[A_i^c(t) \cdot a_i^c(t)]
\end{aligned} \tag{14}$$

Algorithm 1 DRAFT

Require: $a_i^c(t)$ values for all c in trustee agent i, the incoming requests $\lambda_i^c(t)$ for all c at i, and e_i^{max}.

$e_i(t) = e_i^{max}$
for each $Q_i^c(t)$ in i in descending order of its $\frac{a_i^c(t)}{e^c}$ **do**
    if $\frac{a_i^c(t)}{e^c} > 0$ **then**
        if $\lambda_i^c(t) \cdot e^c \leqslant e_i(t)$ **then**
            $A_i^c(t) = \lambda_i^c(t)$
        else
            $A_i^c(t) = \lfloor \frac{e_i(t)}{e^c} \rfloor$
        end if
        $e_i(t) \leftarrow e_i(t) - A_i^c(t) \cdot e^c$
    else
        $A_i^c(t) = 0$
    end if
end for
Reject unaccepted tasks
Return($A_i^c(t)$ for each $Q_i^c(t)$ in i)

where $a_i^c(t) = \rho \cdot \gamma_i^c(t) \cdot G_{max}^c - Q_i^c(t)$ is defined as the availability score of each task queue of i at t. DRAFT helps a trustee agent come up with a task request acceptance plan $(A_i^c(t))$ about how many new tasks of different types it should accept at each time step based on an agent's current situation which is represented by the 3-tuple $\langle (a_i^c(t)), (\lambda_i^c(t)), e_i^{max} \rangle$. In order to maximize (14), DRAFT proceeds as illustrated in Algorithm 1. In essence, the higher the payoff per unit effort for a task τ^c, the higher the reputation of i in performing tasks of type c, and the more spare capacity i currently has in accommodating more requests for performing tasks of type c, the more likely τ^c will be accepted by DRAFT on behalf of i. In the case where not all incoming requests are accepted, DRAFT will inform the requesting trusters so that they can look for other alternatives.

5 Analysis

In this section, we envision a situation where all trustee agents in an ODMAS adopt DRAFT and analyze the impact on the size of the task queues and the social welfare.

Assume there are positive constants B, M and ρ such that the *reward-minus-drift* expression in (8) satisfies:

$$\rho \times reward(t) - \Delta(t) \geqslant \rho U^{opt} + M \sum_{i,c} Q_i^c(t) - B \quad (15)$$

where U^{opt} is the social welfare produced by the theoretical optimal solution for (11). Taking the expectations over the distribution of $\boldsymbol{Q}(t)$ on both sides of (15), we have:

$$\begin{aligned} &\rho \sum_{i,c} \mathbb{E}\{u_i^c(A_i^c(t))\} - \mathbb{E}\{L(\boldsymbol{Q}(t+1)) - L(\boldsymbol{Q}(t))\} \\ &\geqslant \rho U^{opt} + M \sum_{i,c} \mathbb{E}\{Q_i^c(t)\} - B \end{aligned} \quad (16)$$

which holds for all time steps t. Summing both sides over $t \in \{0, 1, \ldots, T-1\}$, we have:

$$\begin{aligned} &\rho \sum_{t=0}^{T-1} \sum_{i,c} \mathbb{E}\{u_i^c(A_i^c(t))\} - \mathbb{E}\{L(\boldsymbol{Q}(T)) - L(\boldsymbol{Q}(0))\} \\ &\geqslant \rho T U^{opt} + M \sum_{t=0}^{T-1} \mathbb{E}\{Q_i^c(t)\} - BT \end{aligned} \quad (17)$$

Since $u_i^c(\cdot) \geqslant 0$ and $L(\cdot) \geqslant 0$, and suppose the potential utility reward for completing $A_i^c(t)$ tasks are bounded by $\sum_{i,c} u_i^c(A_i^c(t)) \leqslant G_{max}$ (where $G_{max} = \sum_{i,c} [A_i^c(t) \cdot G_{max}^c]$ can be achieved when all new tasks are completed successfully within their respective deadlines), by re-arranging the terms in (17) and dividing both sides by TM, the upper bound on the sizes of the task queues in the MAS is:

$$\begin{aligned} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i,c} \mathbb{E}\{Q_i^c(t)\} &\leqslant \frac{B + \rho G_{max} - \rho U^{opt}}{M} \\ &\leqslant \frac{B + \rho Gmax}{M} \end{aligned} \quad (18)$$

Similarly, by rearranging (17) and dividing both sides by $T\rho$, the lower bound on the social welfare produced by agents in the MAS is:

$$\begin{aligned} &\frac{1}{T} \sum_{t=0}^{T-1} \sum_{i,c} \mathbb{E}\{u_i^c(A_i^c(t))\} \\ &\geqslant U^{opt} + \frac{M}{T\rho} \sum_{t=0}^{T-1} \sum_{i,c} \mathbb{E}\{Q_i^c(t)\} - \frac{B}{\rho} \\ &\geqslant U^{opt} - \frac{B}{\rho} \end{aligned} \quad (19)$$

From the above analysis, it can be deduced that if the condition in (15) can be fulfilled (which can be done through careful choice of the values for B, M and ρ), then, based on (18), a theoretical upper bound exists for all pending request queues for all trustee agents over the long run if the agents follow the recommendations made by DRAFT. This ensures that the request queue lengths will not keep on increasing and the trustee agents always can stop the growth of their request queues so their perceived quality of service can be maintained.

In addition, based on (19), the time averaged social welfare achieved in a given ODMAS through DRAFT can approach that achieved by the theoretical optimal solution within B/ρ in the long run. By increasing ρ, the social welfare produced by DRAFT can be made closer to the optimal social welfare. However, increasing ρ also causes the upper bound to the pending request queue lengths to rise according to (18), thereby increasing the expected time taken to complete a request. Due to the physical limitations of the trustees in a realistic system, if the increase in the value of ρ causes the expected completion time of tasks to start exceeding the stipulated deadlines, social welfare will start to decrease as the trusters' level of satisfaction decreases. Setting the value of ρ arbitrarily high will not make the social welfare produced by

DRAFT be indefinitely close the optimal. Thus, the trade-off between quality and the timeliness in receiving services only exists within a limited range of the ρ value. The actual range depends on the physical limitations of the trustees in each given system. On the other hand, in an ODMAS where trustee agents adopt the accept-when-requested approach, these upper and lower bounds cannot be guaranteed.

6 Empirical Evaluation

As the problem in this study is relatively new, there is no existing dataset that can be used to evaluate DRAFT. In addition, real world data are useful for designing a realistic experiment environment, but the behavior patterns of the trustees are fixed for which we do not have ground truth. In order to comprehensively evaluate DRAFT under different circumstances, and to provide more flexible control of trustee agents' behavior, we implement it within a simulated ODMAS environment based on our system model. Our hypotheses in this section are:

- *Hypothesis 1:* A trustworthy trustee agent can better mitigate the adverse effect of reputation damage by using the DRAFT approach than using the *accept-when-requested* approach.
- *Hypothesis 2:* The social welfare of an ODMAS can be improved through the use of the DRAFT approach.

6.1 Experiment Design

The trustee agent population consists of 100 agents belonging to four groups with different behavior patterns in our experiments. They are labeled as:

1. *Hon*: honest trustee agents who return high quality task results randomly 90% of the time on average;
2. *MH*: moderately honest trustee agents who return high quality task results randomly 70% of the time on average;
3. *MM*: moderately malicious trustee agents who return high quality task results randomly 30% of the time on average;
4. *Mal*: malicious trustee agents who return high quality task results randomly 10% of the time on average.

The number of them adopting each of the four different behavior patterns is varied in each experiment to simulate different trustee agent population configurations. A trustee agent population configuration is denoted as $HonX$. It represents a trustee agent population consisting of $X/2\%$ *Hon* trustee agents, $X/2\%$ *MH* trustee agents, $(100-X)/2\%$ *MM* trustee agents, and $(100-X)/2\%$ *Mal* trustee agents. The average maximum effort each type of trustee agents can expend per time step is shown in Table 1. More trustworthy trustee agents can complete less number of tasks in each time step than less trustworthy ones in order to maintain the quality of their work.

Table 1: The e_i^{max} values for trustee agent in each group.

	Hon	*MH*	*MM*	*Mal*
e_i^{max}	25	30	35	40

Table 2: The properties of the types of tasks in the study.

c	*1*	2	3	4	5
G^c_{max}	5	4	3	2	1
e^c	5	4	3	2	1
T^c	1	2	2	3	3

Five categories of tasks are available for trustee agents to serve. Their properties used in the experiments are listed in Table 2. 1,000 truster agents equipped with the Beta Reputation System (BRS) [Jøsang and Ismail, 2002] for evaluating the reputations of the trustee agents are included in the simulation. Each truster agent randomly use 15% of its time for exploration while spending the rest of the time exploiting known reputable trustee agents. During exploitation rounds, truster agents require a trustee agent to have a reputation of over 2/3 in order to consider it as a candidate. At each time step, a number of requests for each type of task are sent out to the trustee agents based on their reputations by the truster agents (assuming truster agents do not intentionally distort their reputation ratings about the trustee agents). The average aggregate workload in the system is similar to that in the Amazon's Mechanical Turk crowdsourcing system as reported in [Ipeirotis, 2010] and [Ross *et al.*, 2010]

Two simulated ODMASs are run in parallel. In one of them, the trustee agents adopt the traditional *accept-when-requested* approach for handling incoming task requests. In the other, the trustee agents adopt the proposed DRAFT approach for handling incoming task requests. The results from these two sets of experiments are labeled as *TRD* and *DRAFT* respectively in the following figures. If no trustee agent is willing to accept a task request under *DRAFT* in a particular time step, the truster agent will attempt to propose the same task in the following time steps. Each simulation is run for 1,000 time steps and is repeated 10 times to reduce the effect of random variations.

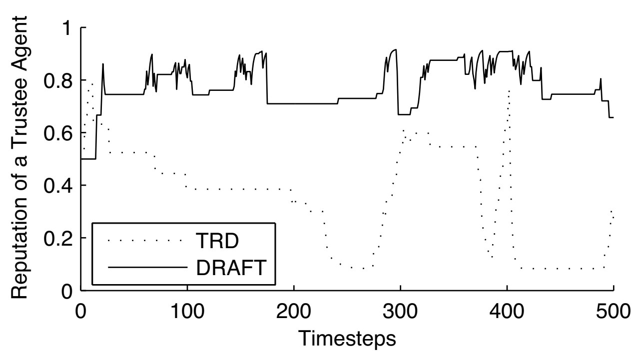
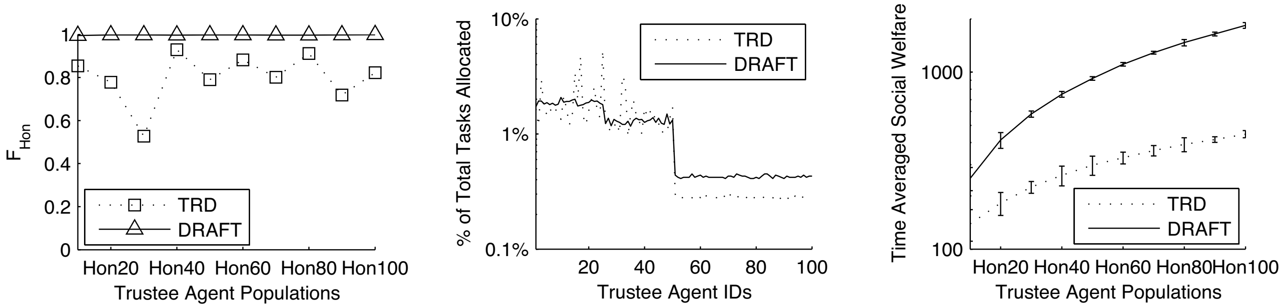


Figure 2: Change of reputation of a Hon trustee agent under different approaches over 500 time steps.



(a) Task Distribution Fairness F_{Hon} (b) Task Allocation Snapshot for *Hon50* (c) Time Averaged Social Welfare

Figure 3: Comparing the Performance of the DRAFT Approach against the traditional *accept-when-requested* Approach.

6.2 Analysis of Results

Hypothesis 1

Figure 2 shows a snapshot of the change in the reputation evaluation of a trustee agent i which belongs to the Hon group (the ground truth trustworthiness is 0.9) over 500 time steps under both *TRD* and *DRAFT*. It can be seen that under *TRD*, the agent's reputation fluctuates significantly. The sequence of event is: during earlier phase of interactions, i was discovered by a few truster agents through exploration and its good performance had gained it a high reputation (about 0.8) compared to other trustee agents. This attracted many other truster agents to request i's services. The sudden influx of requests to i resulted in long backlog in its task queues. Then, i kept working on these tasks and producing high quality results. However, as most of them were completed with long delays, it received a large number of negative ratings from unsatisfied truster agents and its reputation dropped to a very low level, indicating loss of trust from others. This happens without i changing its behavior. As its reputation dropped, the number of requests received by i also decreased. Over time, the backlog in i's task queues had gradually been worked off. After that, i's high quality service was rediscovered by a few truster agents through exploration. Then, a similar sequence of events occurred, causing another round of fluctuation in its reputation.

On the other hand, the reputation of i is closer to the ground truth under *DRAFT* than under *TRD*. By dynamically determining when to be greedy, i was protected from the adverse effect of reputation damage caused by truster agents seeking higher expected utility in an uncoordinated fashion.

Hypothesis 2

Overall, the social equity among the group of the most trustworthy trustee agents (Hon) can be gauged through the fairness index (F_{Hon}) [Jain *et al.*, 1984; Yu *et al.*, 2012]. If $F_{Hon} = 1$ (perfect equality), all *Hon* trustee agents complete exactly equal number of tasks; if $F_{Hon} = 0$ (most unfair), one trustee agent completes all the tasks and all others complete no task.

From Figure 3(a), it can be shown that F_{Hon} under *DRAFT* is consistently close to 1 (its value varies between 0.996 and 0.999), whereas F_{Hon} under *TRD* fluctuates erratically over different trustee agent population configurations. Although in some cases, *TRD* can also lead to a highly fair distribution of tasks among trustee agents. However, such performance cannot be consistently achieved.

A snapshot of the percentage of all tasks delegated to trustee agents under agent population configuration *Hon50* is shown in Figure 3(b). Under *TRD*, clear peaks representing high concentration of tasks on a few trustee agents can be seen; whereas the distribution of tasks is smoother under *DRAFT* which represents more efficient utilization of the trustee agents' capacities. In addition, DRAFT helps an ODMAS ensure that: 1) more trustworthy agents receive more tasks than less trustworthy ones, and 2) similarly trustworthy agents receive similar amounts of tasks.

Figure 3(c) shows the time averaged social welfare achieved by given trustee agent populations under TRD and *DRAFT*. By reducing the adverse effect of reputation damage through efficient utilization of trustee agents' capacities, the ODMAS equipped DRAFT consistently achieved significantly higher social welfare than the ODMAS with trustee agents use the traditional *accept-when-requested* approach.

7 Conclusions

In this paper, we propose DRAFT to enable trustee agents to manage their own reputation through situation-aware management of their workloads. With DRAFT, individual trustee agents can benefit from less fluctuation in their reputation values and more equitable access to task requests from truster agents. Opportunities are passed up by trustee agents who are busy and picked up by agents who have more spare capacities while taking into account of their reputation. Thus, the collective capacity of the trustee agent population can be efficiently utilized which increases the social welfare of the ODMAS over the long run.

As DRAFT can complement existing trust models without requiring them to be modified, it can be easily incorporated into systems which already include trust management mechanisms. In this way, it enables a system and its members to reap the aforementioned benefits without truster agents having to significantly change their mode of operation or to incur extra costs for the underlying system to be modified. This makes it a potentially attractive option for deployed systems (e.g., e-commerce systems, crowdsourcing systems) to enhance user experience and overall system performance.

Acknowledgements

This research is supported in part by Interactive and Digital Media Programme Office (IDMPO), National Research Foundation (NRF) hosted at Media Development Authority (MDA) of Singapore under Grant No.: MDA/IDM/2012/8/8-2 VOL 01, and the National Natural Science Foundation of China (No.61202212).

References

[Ipeirotis, 2010] Panagiotis G. Ipeirotis. Analyzing the amazon mechanical turk marketplace. *XRDS: Crossroads, The ACM Magazine for Students*, 17(2):16–21, 2010.

[Jain *et al.*, 1984] Rajendra K. Jain, Dah-Ming W. Chiu, and William R. Hawe. *A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems*. Digital Equipment Corporation, 1984.

[Jøsang and Ismail, 2002] Audun Jøsang and Roslan Ismail. The beta reputation system. In *Proceedings of the 15th Bled Electronic Commerce Conference*, pages 41–55, 2002.

[Jøsang *et al.*, 2007] Audun Jøsang, Roslan Ismail, and Colin Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 43(2):618–644, 2007.

[Monderer and Shapley, 1996] D. Monderer and L.S. Shapley. Potential games. *Games and Economic Behavior*, 14:124–143, 1996.

[Neely, 2010] Michael J. Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan and Claypool Publishers, 2010.

[Ross *et al.*, 2010] Joel Ross, Andrew Zaldivar, Lilly Irani, and Bill Tomlinson. Who are the turkers? worker demographics in amazon mechanical turk. In *Proceedings of the 28th ACM Conference on Human Factors in Computing (CHI)*, pages 2863–2872, 2010.

[Sabater and Sierra, 2005] Jordi Sabater and Carles Sierra. Review on computational trust and reputation models. *Artificial Intelligence Review*, 24(1):33–60, 2005.

[Teacy *et al.*, 2005] W. T. Luke Teacy, Jigar Patel, Nicholas R. Jennings, and Michael Luck. Coping with inaccurate reputation sources: Experimental analysis of a probabilistic trust model. In *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'05)*, pages 997–1004, 2005.

[Weng *et al.*, 2010] Jianshu Weng, Zhiqi Shen, Chunyan Miao, Angela Goh, and Cyril Leung. Credibility: How agents can handle unfair third-party testimonies in computational trust models. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 22(9):1286–1298, 2010.

[Yu *et al.*, 2010] Han Yu, Zhiqi Shen, Chunyan Miao, Cyril Leung, and Dusit Niyato. A survey of trust and reputation management systems in wireless communications. *Proceedings of the IEEE*, 98(10):1755–1772, 2010.

[Yu *et al.*, 2012] Han Yu, Zhiqi Shen, Chunyan Miao, and Bo An. Challenges and opportunities for trust management in crowdsourcing. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (IAT)*, pages 486–493, 2012.

[Zhang, 2012] Dan Zhang. Taobao seller dies of exhaustion. *CCTV.com*, Dec 2012.