DRAMa: An IoT-enabled Distributed Reputation Agent Model

Kalliopi Kravari
School of Informatics
Aristotle University of Thessaloniki
Thessaloniki, GR-54124
Greece
kkravari@csd.auth.gr

Maria Ustyantseva
School of Informatics
Aristotle University of Thessaloniki
Thessaloniki, GR-54124
Greece
mariusty@csd.auth.gr

Nick Bassiliades
School of Informatics
Aristotle University of Thessaloniki
Thessaloniki, GR-54124
Greece
nbassili@csd.auth.gr

ABSTRACT

The Internet of Things is a network of objects, called Things, which can interact with the environment or other Things, with no human intervention. At the same time, multi-agent systems are considered a modern medium of communication and interaction with limited or no human intervention. Hence, combining agent technology with the Internet of Things seems promising. Yet, the open, distributed and heterogeneous environment raises important challenges, such as trustworthiness among the various devices and participants. Hence, distributed reputation models inevitably attract more attention. In this paper we propose DRAMa, a distributed reputation model that provides a novel mechanism, combining reputation with risk and reward. DRAMa attempts to reverse the traditional view of reputation models by involving auctions in the decision making procedures. Finally, an evaluation of the model demonstrates the added value of the approach.

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence** → **Distributed artificial intelligence**; *Multi-agent systems*; Cooperation and coordination

KEYWORDS

Intelligent Multi-agent Systems, Internet of Things, Agent Reputation, Distributed Trust Reputation Model

1 INTRODUCTION

The Internet of Things (IoT)[1] is a network of physical or virtual objects, called Things, able to interact with the environment or other Things [1]. At the same time, intelligent Agents (IAs) are capable of autonomously representing people, devices or services, ensuring optimal performance, flexibility and trustworthiness in interactions [2]. Hence, combining agent technology with the IoT seems more than promising. Yet, the open, distributed and heterogeneous environment raises important challenges, such as intelligence and trustworthiness among the various types of devices and participants. Things acting in such an open and risky environment will have to make the appropriate decisions about who or what to trust and eventually interact with [4]. Hence, distributed reputation models attract more attention in an attempt to provide a supportive mean for realistic and quite safe interactions in the era of the IoT, a vital issue for its success.

In this paper we propose DRAMa, a distributed reputation model, appropriate for the IoT. DRAMa combines reputation with risk and reward in a novel more personalized and intuitive mechanism. In this context, the more well rated a party is (higher reputation) the more reward it receives, encouraging honest and high-valued behaviour. Furthermore, it involves the popular challenge-response-contract scheme by providing an auction mechanism that supports decision making while eliminating the need for locating ratings. Actually, DRAMa is designed under a reverse to traditional reputation models perspective. Parties adopting DRAMa broadcast their need for a specific task contractor, waiting for them to respond while they minimize their own effort. As a result, the party will locate all available and willing to interact parties, choosing eventually the more appropriate among them. Even IoT devices with limited capabilities can use this procedure. Moreover, an evaluation of the reputation model is presented.

The rest of the paper is organized as follows. Section 2 presents a brief overview of auctions and the Contract Net Protocol. Section 3 presents DRAMa and its contribution. In Section 4, DRAMa's evaluation is presented, demonstrating the performance and the added value of the approach. Section 5 discusses related work, and Section 6 concludes with final remarks and directions for future work.

2 REACHING AGREEMENTS

The capabilities of negotiation and argumentation are central to the ability of an agent to reach agreements and act properly.

2.1 Auctions

Auctions are a special form of negotiation between an auctioneer and a number of bidders. The auctioneer desires to maximize the price of a good while bidders desire to minimize it. There are plenty of auction types used for allocating a single item [7]. We are interested in first-price actions since they are one-shot auctions similar to the procedure of task allocation that studies the proposed DRAMa model. In this auction type, all bidders

SETN '18, July 9–15, 2018, Rio Patras, Greece

ACM ISBN 978-1-4503-6433-1/18/07…$15.00
https://doi.org/10.1145/3200947.3201048.

simultaneously submit sealed bids so that no bidder knows the bid of others. The first-price sealed-bid auction includes four rounds. The auctioneer initializes the auction (1st round), bidders submit a sealed bid for the good (2nd round), then good is allocated to the agent that made the highest bid (3rd round) and the winner pays the price it submitted, which is the highest bid (4th round).

2.2 The Contract Net Protocol

Agents via task sharing allocate tasks that they cannot handle to others. A well-known task-sharing protocol is the Contract Net Protocol (CNP) introduced by Smith [8]. It consists of a collection of agents, acting as nodes that form the contract net. Each node on the network can act as manager or contractor. Each time a node gets a composite or a hard to handle task, it breaks the problem down into sub-tasks and announces the sub-task to the contract net acting as a manager. Bids are then received from potential contractors and the winning contractors are awarded the tasks. CNP consists of five stages [6]. In the first stage (*Recognition*), an agent realizes either it does not have the capability to fulfil its goal or it would prefer to allocate it. Then (*Announcement*) it sends out a task announcement that includes a specification encoding the task description and constraints. Next (*Bidding*), agents that receive the announcement decide whether they should bid for the task or not, depending on their capabilities. Hence, the contract is awarded to the agent with the most promising bid (*Awarding*). This may involve the generation of further contract nets in the form of sub-contracting to complete the task (*Expediting*).

3 DRAMa

The proposed DRAMa model is a distributed reputation model that can be used in many IoT fields [5].

3.1 Model Abstract Architecture

DRAMa has no centralized authority; each agent use the model and run its auctions. It supports initiator and participant agents, similar to CNP's manager and contractors, while it provides a six-step interaction protocol, using the principles of the sealed first-price auctions to form its auction mechanism.

Fig. 1 presents the architecture of the DRAMa model. As soon as an initiator agent enters the environment, initializes the number of tasks that it wants to propose to others (step 1). For practical reasons, the number of tasks range from 0 to 10, where 0 stands for no desired tasks. For each task, initiator broadcasts a call for proposal (CFP) that includes the task description and a maximum price (step 2). This price (the reward) is proportionate to task complexity. The more complex a task is the more valued it is. The calculation formula depends on initiator's private strategy.

Broadcasting a CFP instead of seeking for witness reports could limit the disadvantage of locating ratings. The rating records could always be there but usually they are unreachable since various agents may join or leave the system at any time. On the other hand, sometimes there is a large amount of available ratings but taking all of them into account has significant computational cost. Broadcast allows agents to reach easily available agents.

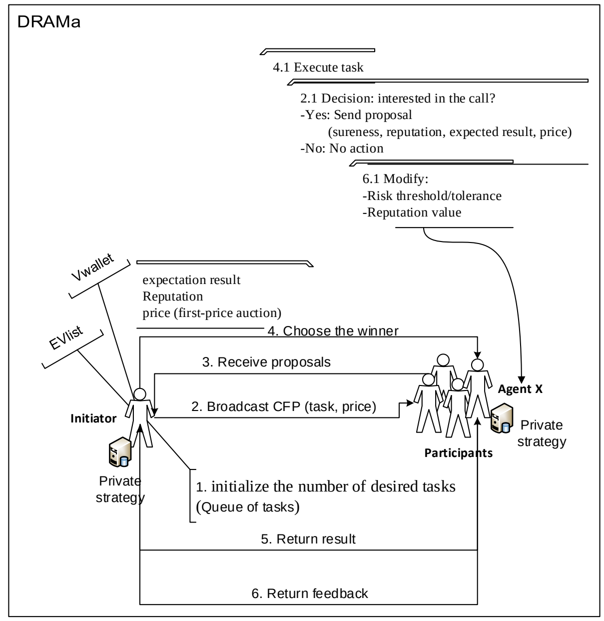


Figure 1: DRAMa abstract architecture.

After the broadcast, participants that receive the CFP call make a decision upon that (substep 2.1), depending on their personal strategy that involves parameters such as the reward (price), their reputation, the expected result and how sure they are for that (see subsection 3.2.2). If they are not interested, they ignore the call. Otherwise, they send back a proposal containing values for their reputation, sureness, expected result and their desired price. This price should be equivalent or lower than the price included in the CFP call. This way, the initiator will not pay more than the amount it is willing to pay while it could even save some credits.

Meanwhile, the initiator waits for a limited time and receives proposals (step 3). If it receives no proposals, it puts the task back in queue. When the initiator runs the same auction procedure for the rest tasks, returns to that task and repeats the procedure. An initiator can repeat the procedure until there is no task left in its queue. Following step 3, the initiator evaluates the received proposals and choose an appropriate, sending an inform message to that agent (step 4). Once again the evaluation formula depends on its private strategy and it is mainly based on the expected result, the agent's reputation and the proposed price.

The agent that was accepted for the task, runs the appropriate procedures executing the task (substep 4.1) and returns the result to the initiator (step 5). The initiator then returns a feedback, equal to *task price/10*, stating how much satisfied it was with the task execution (step 6). If the agent succeeded, this amount will be added to its current reputation value whereas if it failed it would be subtracted. Hence, the reputation is related to task complexity and agent's success or failure. As a result, the participant agent, according to its strategy, modifies its risk threshold as well as its reputation value (substep 6.1 - see subsection 3.2.2). Finally, the initiator moves to the next task in queue, repeating the procedure.

3.2 Agent Roles

Both initiators and especially participants vary depending on their personal strategy that allow them to take more or less risk.

3.2.1 Initiators

An initiator has to run auctions in order to locate partners that will perform its desired tasks. It poses a virtual wallet, called Vwallet (Fig. 1), which contains its credits (virtual money). Actually, both initiators and participants have a Vwallet but it is more important for initiators. If an agent's Vwallet is empty it cannot pay and, thus, act as initiator. Hence, it should first act as participant in other agents' auctions in order to earn some credits.

Furthermore, it has a list, called EVlist, which contains its evaluation criteria (used at step 6), namely validity, completeness, correctness and response time [8]. Validity is the degree that an agent is sincere and credible, namely it believes what it says and it believes is true in the world respectively. Completeness rates dishonest and fraud behavior. Correctness refers to the final provided task with respect to the specifications. Response time refers to the time that an agent needs in order to complete the task. The rating values vary from 0 (% - terrible) to 100 (% - perfect) The final evaluation value comes from a weighted sum.

3.2.2 Participants

This study includes three categories of participants, namely Normal, Risky and Prudent. The *Normal* agents prefer tasks that have prices from *0.5*reputation* to *1.5*reputation*. They do not have risk coefficients and they are not interested in tasks that have price greater than *1.5*reputation*. They are 100% sure in success (sureness value) if the price is lower than *0.5*reputation*.

Risky and Prudent newcomer agents have zero risk level. These agents prefer tasks with prices between *0.5*reputation* and *1.5*reputation*. *Risky* agents are not interested in tasks that have prices lower than *0.5*reputation* whereas *Prudent* agents are not interested in tasks that have prices greater than *1.5*reputation*. After a proposal rejection, a *Risky* agent increases its risk level (maximum value100) about 10 points. If its proposal is accepted but the initiator is not satisfied with the result, it decreases its risk level about 5 points. In any other case, the risk level remains constant. On the other hand, each time, an initiator is satisfied with the result, the *Prudent* agent increases its risk level about 5 points whereas if it is not, the agent decreases its risk level about 5 points. In cases of proposal rejection, the risk level remains constant. Actually, risk influences the agent's sureness about its success, determining the sureness value that it sends to the initiator at the proposal step (steps 2.1 & 3).

The prices are related to the reputation in the sense that each agent values its value in order to optimize its profit. The price offered by the initiator is related to the complexity of the task since it will pay more for a more complex and time consuming task. Table 1 presents the relation among *Sureness*, price and reputation, indicating the decision-making mechanism. The sureness, price and risk values range from 0 to 100 whereas "-" means that the agent is not interested in this task. P stands for price value and R stands for reputation value.

Table 1: Agents sureness logic (P: price, R: reputation)

Agent/ sureness	Normal	Risky	Prudent
P<=R*0.5	100	-	100
R*0.5<P<R	P	100	P + Risk
R<P<R*1.5	R*2 - P	100	P - Risk
P>= R*1.5	-	P – R + Risk	-

The sureness value calculation depends on the degree that an agent is willing to take the specific risk. Its private strategy determines what is more important for the agent, competing for highly priced tasks with the risk of failure and reputation decrease or trying to earn moderate amounts without great risk.

DRAMa provides a simple formula for the *Predicted result*, the result that an agent expects to provide (the degree of its success). This formula takes into account the reputation value r, the sureness value s and a weight factor R that normalizes the output, ranging between 0 and 100. We propose, based on performance experiments, the equation *r*0.7+s*0.1+R*0.2* (1) for normal and prudent agents and for risky agents the equation *r*0.6+s*0.2+R*0.2* (2). Equation 2 indicates that risky agents are more sure about themselves while risk is not a prohibitive factor for them. Any other weight combination would be acceptable.

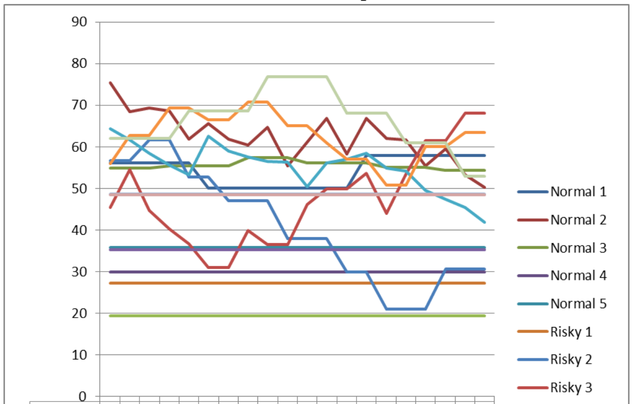
4 EVALUATION

We implemented DRAMa in Jason [9], a tool for creating MASs. The testbed environment is a MAS consisting of initiators and participants. In order to reduce the environment complexity, it is assumed that there is only one type of service in the testbed. Fig. 2 presents one of the experiments populated with mixed type agents, including 5 normal agents, 5 risky and 5 prudent. It demonstrates their reputation distribution throughout 82 auctions.

The reputation of normal agents does not change much, mainly due to their strategy that encourages them to propose for average-prices tasks without taking great risk. Yet, it seems that this behavior leads to reputation loss. Similarly, only prudent agents with high reputation values win and take tasks. Prudent agent eventually lose their reputation since they earn a small amount of credit for average-priced tasks, pretty much like normal agents do. However, prudent agents show smoother value changes, due to their conservative behavior. Yet, these agents start to lose reputation earlier than normal agents. This can be interpreted considering that they act more conservatively than normal agents.

The case of risky agents is a bit different, since all agents act as participants winning tasks. Inevitably, reputation values change over time and this is the rule for all agents. Risky agents have a more complicated private strategy that encourages them to target highly-priced tasks, taking the necessary risk. Risk computation and its role in sureness calculation is important here, making risky agents more confident and impatience. This behavior let risky agents to earn more credits for each successfully executed task. Yet, the same behavior makes them lose more each time they fail.

This experiment demonstrated that DRAMa allows agents to locate promising partners without the need of complication estimation mechanisms. All tasks were assigned to agents while dishonest and bad behavior lead to reputation decrease.



Normal 1	56	56	56	56	56	50	50	50	50	50	50	50	50	58	58	58	58	58	58	58
Normal 2	75	69	69	69	62	66	62	60	65	55	61	67	58	67	62	62	56	60	53	50
Normal 3	55	55	55	56	56	56	56	57	57	57	56	56	56	56	55	55	55	54	54	54
Normal 4	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
Normal 5	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
Risky 1	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27
Risky 2	57	57	62	62	53	53	47	47	47	38	38	38	30	30	21	21	21	31	31	31
Risky 3	45	55	45	40	37	31	31	40	37	37	46	50	50	54	44	53	62	62	68	68
Risky 4	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19
Risky 5	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35
Prudent 1	64	62	59	56	53	63	59	58	57	56	51	56	57	58	55	54	50	47	45	42
Prudent 2	56	63	63	69	69	66	66	71	71	65	65	61	57	57	51	51	60	60	63	63
Prudent 3	49	49	49	49	49	49	49	49	49	49	49	49	49	49	49	49	49	49	49	49
Prudent 4	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
Prudent 5	62	62	62	62	69	69	69	69	77	77	77	77	68	68	68	61	61	61	53	53

Figure 2: Reputation in a mixed agent system.

5 RELATED WORK

Various trust and reputation models have been proposed [3], among them the authors of this paper [11]. One of the first models that studies social disciplines was Social Regret [10], a reputation system incorporates the notion of social graph. Social Regret groups agents with frequent interactions and considers each one of these groups as a single source of reputation values. Hence, only the most representative agent within each group is asked for information. A heuristic is used in order to find groups and select the best agent to ask. CRM (Comprehensive Reputation Model) [12] is a typical distributed reputation model where ratings are obtained either from an agent's interaction history or collected from others in the form of ratings. CRM is a probabilistic-based model, taking into account the number of interactions between agents, the timely relevance of provided information and the confidence of reporting agents on the provided data.

Social Regret, similarly to DRAMa, make use of social disciplines, although it rather attempts to heuristically reduce the number of queries to be done in order to locate ratings. DRAMa uses the notion of social contracts that eliminates the need for locating ratings. CRM is a hybrid model that has a complicating mechanism for locating ratings. DRAMa provides a simpler and quite effective way in order to bring in touch potential partners. Only DRAMa defines the notion of auction initiator that possess a specific amount of credits. This mechanism limits fraud behavior, encouraging agents not only to ask for help but also to give. Additionally, DRAMa allows agents to have a "value", a metric of how good they are, letting them gain what they deserve.

6 CONCLUSIONS AND FUTURE WORK

This paper presented DRAMa, a distributed reputation model, appropriate for the IoT. DRAMa combines reputation with risk and reward. Locating ratings consuming great amount of time and resources is limited in DRAMa since it proposed an auction mechanism for call broadcasting. On the other hand, it uses well known parameters, such as completeness and correctness for proposal and result evaluation. DRAMa encourages honest behavior and allows agents to evaluate their added value. Hence, the more well rated a party is, the more reward it receives.

As for future directions, we plan to further improve DRAMa by adopting some kind of learning, since agents with low reputation values sometimes are unable to win tasks. Our intention is to provide a mechanism that could let agents to be more flexible with risk. At the same direction, we plan to include not only strict rules for good and bad results but also a possible neutral window that will protect the reputation value from sharp fluctuations.

ACKNOWLEDGMENT

The Postdoctoral Research was implemented through an IKY scholarship funded by the "Strengthening Post-Academic Researchers / Researchers" Act from the resources of the OP "Human Resources Development, Education and Lifelong Learning" priority axis 6,8,9 and co-funded by The European Social Fund - the ESF and the Greek government.

REFERENCES

[1] Roy Want, Bill N. Schilit, and Scott Jenson. 2015. Enabling the Internet of Things. *Computer* 48, 1 (2015), 28–35.

[2] Michael Gelfond and Yulia Kahl. 2014. The Answer-Set Programming Paradigm. Knowledge Representation, Reasoning, and the Design of Intelligent Agents (2014), 114–130.

[3] Zheng Yan, Peng Zhang, and Athanasios V. Vasilakos. 2014. A survey on trust management for Internet of Things. *Journal of Network and Computer Applications* 42 (2014), 120–134.

[4] Andrew Whitmore, Anurag Agarwal, and Li Da Xu. 2014. The Internet of Things—A survey of topics and trends. *Information Systems Frontiers* 17, 2 (December 2014), 261–274.

[5] Shancang Li, Li Da Xu, and Shanshan Zhao. 2014. The internet of things: a survey. *Information Systems Frontiers* 17, 2 (2014), 243–259.

[6] Smith. 1980. The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. *IEEE Transactions on Computers* C-29, 12 (1980), 1104–1113.

[7] Paul Klemperer. 1999. Auction Theory: A Guide to the Literature. SSRN *Electronic Journal* (1999).

[8] Ferry Hendrikx, Kris Bubendorfer, and Ryan Chard. 2015. Reputation systems: A survey and taxonomy. Journal of Parallel and Distributed Computing 75 (2015), 184–197.

[9] Anon. Jason Interpreter. *Wiley Series in Agent Technology Programming Multi-Agent Systems in AgentSpeak usingJason*, 67–100.

[10] Jordi Sabater and Carles Sierra. 2001. Social ReGreT, a reputation model based on social relations. *ACM SIGecom Exchanges* 3, 1 (January 2001), 44–56.

[11] Kalliopi Kravari and Nick Bassiliades. 2016. DISARM: A social distributed agent reputation model based on defeasible logic. *Journal of Systems and Software* 117 (2016), 130–152.

[12] Babak Khosravifar, Jamal Bentahar, Maziar Gomrokchi, and Rafiul Alam. 2012. CRM: An efficient trust and reputation model for agent computing. *Knowledge-Based Systems* 30 (2012), 1–16.