Proceedings of the 2014 Federated Conference on
Computer Science and Information Systems pp. 59–66
DOI: 10.15439/2014F482
ACSIS, Vol. 2

Using Fuzzy Logic and Q-Learning for Trust Modeling in Multi-agent Systems

Abdullah Aref
School of Electrical Engineering and Computer Science
Faculty of Engineering, University of Ottawa
Ottawa, Ontario, K1N 6N5, Canada

Thomas Tran
School of Electrical Engineering and Computer Science
Faculty of Engineering, University of Ottawa
Ottawa, Ontario, K1N 6N5, Canada

***Abstract*—Often in multi-agent systems, agents interact with other agents to fulfill their own goals. Trust is, therefore, considered essential to make such interactions effective. This work describes a trust model that augments fuzzy logic with Q-learning to help trust evaluating agents select beneficial trustees for interaction in uncertain, open, dynamic, and untrusted multi-agent systems. The performance of the proposed model is evaluated using simulation. The simulation results indicate that the proper augmentation of fuzzy subsystem to Q-learning can be useful for trust evaluating agents, and the resulting model can respond to dynamic changes in the environment.**

I. Introduction

A MULTI-AGENT Systems (MAS) involves multiple autonomous, self-interested, and goal-driven interacting intelligent agents [1]. An open MAS is a class of these systems in which agents can freely enter and leave at any time [2]. As each agent has only limited capabilities, it may need to rely on the services or resources from other agents in order to accomplish its goals [3]. Agents cannot assume that other agents share the same core beliefs about the system, or that other agents make accurate statements regarding their competencies and abilities. In addition, agents must accept the possibility that other agents may intentionally spread false information, or otherwise behaving in a harmful way, to achieve their own goals [1]. Therefore, agents should be equipped with a strong trust assessment model that is capable of maximizing the benefit, also referred to as utility gain (UG), of interacting with other agents. The estimation should be accurate enough that allows trust evaluating agents, also referred to as trustors (TRs), to identify the most beneficial trustee (TE) in their systems. The trust estimation model should consider all relevant factors, which affect the trust that an agent has about other agents. Failure to gather those factors would lead to compute a non-accurate trust value, which could explicitly affect agent's outcome [4]. Moreover, the model should dynamically update agents' belief sets to capture new characteristics of the environment, and should not rely on any centralized entities. Furthermore, the failure or takeover of any node must not lead to the failure of the whole system.

Trust has been defined in many ways in different domains [5]. For this work the definition used in [4] for trust in MASs, will be adapted. An agent's trustworthiness is considered as a measurement of the agent's possibility to do what it is supposed to do. In this work, we describe a trust model for MAS that combines the advantages of both: fuzzy logic and reinforcement learning for trust modeling in MAS. Moreover, we use a suspension technique in combination with reinforcement-learning to speedup the response of the model to dynamic changes in the system.

The paper is organized as follows: the related work is presented in section II followed by a general overview about fuzzy logic systems and reinforcement learning in section III. Section IV presents the details of the proposed model, while performance analysis is presented in section V. The last section presents conclusions and future work

II. Related Work

According to [3], most existing research on trust evaluation models can be divided into four main categories: direct trust evaluation models, that depends on past experience, indirect or reputation-based trust evaluation models, that depends on third-party testimonials from other agents in the same environment, socio-cognitive trust evaluation models, that depends on examining the social connections among agents to determine their trustworthiness, and organizational trust evaluation models, that depends on some organizational affiliations or endorsements issued by some trusted third-party to determine the trustworthiness of agents

FIRE [2] is a well-known decentralized trustworthiness estimation model for open MASs. The model categorizes trust components into direct experience called Interaction trust, Witness reputation, Role-based trust and Certified reputation. The model assumes that witnesses are honest and willing to cooperate and uses weighted summation to aggregate trust components.

Fuzzy logic offers the ability to handle uncertainty and imprecision effectively, and is therefore ideally suited to reasoning about trust [6]. Fuzzy inference copes with imprecise inputs and allows inference rules to be specified using imprecise linguistic terms, such as "very high" or "slightly low" [6].

FuzzyTrust [7] uses fuzzy logic inferences to estimate trust based on direct experience and witnesses testimonials taking into consideration uncertainties and incomplete information in a peer to peer system. The authors compare the performance of FuzzyTrust with the well known EigenTrust algorithm

[8], over the public domain transaction data from eBay, and demonstrated that it is more effective than EigenTrust.

A reinforcement learning (RL) based trustworthiness estimation model for buying and selling agents in an open, dynamic, uncertain and untrusted e-marketplace is described in [9] and further elaborated in [10], where buyers model the trustworthiness of the sellers as trustworthy, untrustworthy and neutral sellers. A buying agent chooses to purchase from a trustworthy seller. If no trustworthy seller is available, then a seller from the set of non-untrustworthy sellers is chosen. The seller's trustworthiness estimation is updated based on whether the seller meets the expected value for the demanded product with proper quality. A decentralized extension to the model used in [9] is describe in [11], [12], to enable indirect trustworthiness where advising agents are partitioned into trustworthy, untrustworthy and neutral sets to address buyers' subjectivity in opinions. However, the authors did not present any experimental results to justify their theoretical approach [13].

Recent survey such as [3][14] provide more insight on existing work in the field of MAS trust modeling.

III. Basic Concepts

A. Fuzzy Logic System (FLS)

FLSs have been extensively applied with success in many diverse application areas due to their similarity to human reasoning, and their simplicity [15]. An FLS provides a nonlinear mapping of input data vector into a scalar output. Such system maps crisp inputs into crisp outputs. It has four components: fuzzy logic rules, fuzzifier , an inference engine, and defuzzifier [16].

The main idea is that, the sets are based on the concept of a membership function (MFs), that defines the level to which a fuzzy variable is a member of a set. One represents full membership, whereas zero represents no membership; in other words, sets used for expressing input and output parameters are fuzzy [6]. An MF provides a measure of the level of similarity of an ingredient to the fuzzy subset. It is necessary to note that in fuzzy logic an ingredient can reside in more than one set to varying levels of association, which can't happen in crisp set theory. Triangular, trapezoidal, piecewise linear and Gaussian, are commonly used shapes for MFs [16].

Rules may be implemented by experts or can be derived from numerical data. In either case, fuzzy rules are represented as a collection of IF- THEN statements. MFs map input values into the interval [0,1] by the process known as "fuzzification" [6]. The fuzzifier maps crisp inputs into fuzzy sets, to stimulate rules which are in terms of linguistic variables. Fuzzy logic rules define the relationship between inputs and output. The inference engine, handles the way in which rules are combined. The conclusion membership levels are aggregated by superimposing the resultant membership curves. In many applications, crisp numbers must be collected at the output of an FLS. The defuzzifier maps output sets into crisp numbers[16]. Figure 1 presents the general architecture of an FLS.

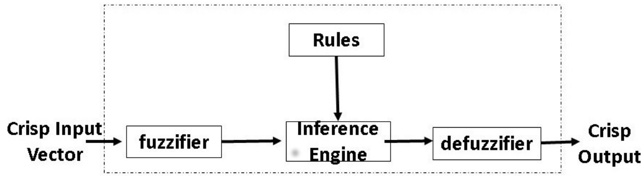

Fig. 1. Fuzzy Logic System [16]

During fuzzy inference, for each fuzzy rule, the inference engine determines the membership level for each input. Then measures the degree of relevance for each rule based on membership levels of inputs and the connectives (such as AND, OR) used with inputs in the rule. After that, the engine drives the output based on the calculated degree of relevance and the defined fuzzy set for the output variable in the rule[17].

Mamdani min-max method [18] is a well known direct inference method. where the degree of membership of rule conclusions is clipped at a level determined by the minimum of the maximum membership values of the intersections of the fuzzy value antecedent and input pairs. This ensures that the degree of membership in the inputs is reflected in the output [6]. In this work, Mamdani's method is used.

The centroid defuzzification method is an appealing defuzzification method [17]. The centroid method takes the center of gravity of the final fuzzy space in order to produce an output sensitive to all rules. In this work, the centroid defuzzification method is used.

B. Reinforcement learning

The reinforcement learning problem is the problem of learning from interaction to achieve a goal. In this problem, an agent observes a current state s of the environment, performs an action an on the environment, and receives a feedback r from the environment (reward, or reinforcement). The goal of the agent is to maximize the cumulative reward it receives in the end [10].

Temporal-difference (TD) learning algorithms can learn directly from experience without a model of the environment. TD algorithms do not require an accurate model of the environment and are incremental in a systematic sense [10]. One of the most widely used TD algorithms is known as the Q-learning algorithm. Q-learning works by learning an action-value function based on the interactions of an agent with the environment and the instantaneous reward it receives. For a state s, the Q-learning algorithm chooses an action a to perform such that the state-action value $Q(s, a)$ is maximized. If performing action a in state s produces a reward r and a transition to state s', then the corresponding state-action value $Q(s, a)$ is updated accordingly. State s is now replaced by s' and the process is repeated until reaching the terminal state [10]. The detailed mathematical foundation and formulation, as well as the core algorithm of Q-learning, can be found in [19] therefore it is not repeated here.

Q-learning is an attractive method of learning because of the simplicity of the computational demands per step and also because of proof of convergence to a global optimum, avoiding all local optima, as long as the Markov Decision Process (MDP) requirement is met; that is the next state depends only on the current state and the taken action (it is worth noting that the MDP requirement applies to all RL methods) [15].

IV. Using Fuzzy Logic and Q-Learning for Trust Modeling in Multi-agent Systems

In this section, we propose the use of Fuzzy Logic and Q-Learning for Trust Modeling in Multi-agent Systems (FQT) as an improvement over RL based trust estimation by incorporating fuzzy subsystems to perform human-like decisions.

A. Overview

According to the proposed model, TRs classify TEs into three non-overlapping sets. The first set includes trustworthy TEs, the second set contains untrustworthy TEs and the third set includes neutral (neither trustworthy nor untrustworthy) TEs. Additionally, TRs classify witnesses in a similar way. If a TR is not satisfied by the interaction with a TE, the TR suspends the use of that TE for incoming transactions for a while. TRs suspend witnesses in a similar way.

TRs use Q-Learning to estimate the trustworthiness for TEs based on direct experience (DT). For those TEs that are not categorized as untrustworthy, the calculated DT is used as an input to the direct trust fuzzy subsystem, together with suspension period and the average of time-decayed utility gain within the last G interaction with the TE. The defuzzified output of this fuzzy subsystem is the fuzzy direct experience (FDT) of trustworthiness estimation.

For those TEs that are not categorized as untrustworthy, TRs consult witnesses for their testimonials about TEs. This is known as indirect trust (IT). Then information from both sources (direct experience and testimony of witnesses) are combined to compute total trust estimation (TT).

TRs request TEs to bid for coming transactions. The calculated TT is used as an input to the TE selection fuzzy subsystem (TSF). The second input is the difference between the bid value of the TE and the average bidding of all TEs for the same transaction. The third input is the difference between the average of time-decayed utility gain within the last H interaction with the TE and the average of time-decayed utility gain within the last H interaction with all TEs. The TR selects the TE that maximizes the outcome of TSF.

Figure 2 present the general architecture of the proposed model.

B. Fuzzy Direct Trustworthiness Estimation FDT (TR,TE):

In the proposed model, TRs use Q-learning to estimate the direct trust of TEs in a way similar to the process in [10]. If the TR is satisfied by the interaction with the TE, Eq. (1) is used to update the credibility of the TE as viewed by the TR.

$$DT_i(TR, TE) = DT_{i-1}(TR, TE) + \alpha(1 - |DT_{i-1}(TR, TE)|) \tag{1}$$

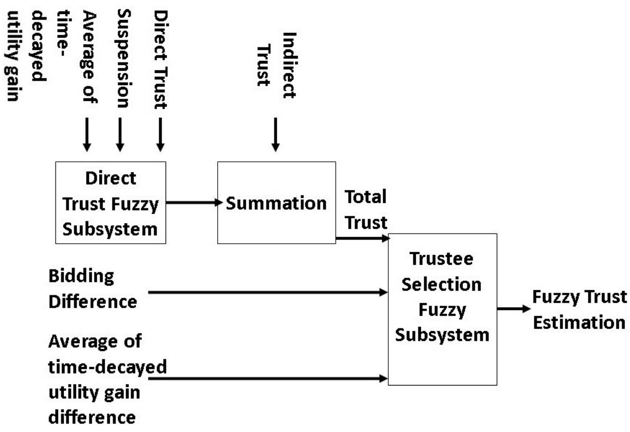

Fig. 2. Architecture of FQT

Here DT$_i$ (TR, TE) is the direct trust estimation of the TE by the TR at time i. The value of DT (TR, TE) varies from -1 to 1. A TE is considered trustworthy if the trustworthiness estimation is above an honesty threshold (HT). The TE is considered untrustworthy if the trustworthiness estimation value falls below a fraudulent threshold (FT). TEs with trustworthiness estimation values between the two thresholds are considered neutral. The cooperation factor α is positive ($1 > \alpha > 0$) and the initial value of the direct trustworthy estimation is set to zero. TR will consider TE as being cooperative if the resulting UG of the transaction is greater than or equal the TR's satisfactory threshold.

If the TR is not satisfied by the interaction with the TE, Eq. (2) is used to update the credibility of the TE as viewed by TR.

$$DT_i(TR, TE) = DT_{i-1}(TR, TE) + \beta(1 - |DT_{i-1}(TR, TE)|) \tag{2}$$

Here β is a negative factor called the non-cooperation factor ($0 > \beta > -1$). TR will consider TE as being non-cooperative if the resulting UG of the transaction is less than the TR's satisfactory threshold. [10] described mathematical formulas to calculate the cooperation and non-cooperation factors in the context of an e-marketplace; however, we believe that those factors are application dependent and should be set by each agent independently. In general, we agree with [10] that the factors should be related to the value gain of the transaction.

Furthermore, the TR suspends the use of the TE for a period of time determined by equation (3)

$$SUS_i(TE) = SUS_{i-1}(TE) + BSI * IV \tag{3}$$

Where SUS$_i$(TE) is the suspension penalty associated with TE at time instant i. Basic Suspension Interval (BSI) is application dependent, it could be days in e-marketplace or seconds in a robotics system that has a short life time and Interaction Value (IV) indicates how much the TR values the interaction, not the actual utility gain of the interaction.

The value $SUS_i(TE)$ decreases with time if ther is no dissatisfactory transaction. That is $SUS_i(TE) = SUS_{i-1}(TE) - 1$, but can't be less than 0.Therefore, a large value of $SUS_i(TE)$ for a TE means a recently misbehaved TE

In the proposed model, each trust evaluation agent uses a direct trust fuzzy subsystem to find the Fuzzy Direct Trustworthiness Estimation (FDT). We define three input parameters for the fuzzy engine of the trust model and one output parameter; the input parameters are:

- The calculated DT using Q-Learning, calculated in equations 1 and 2. This parameter represents the long term relationship between the TR and the TE. A TE with a large value of DT means a TE that used to be cooperative for a relatively large number of transactions
- The suspension period of the TE, calculated in equation 3. This parameter is used to address the short term relationship between the TR and the TE. It helps the TR to address a recently malfunctioning TE that used to be honest for a relatively large number of transactions.
- AUG'$_t$The average of time-decayed utility gain within the last G interactions with the TE, calculated in equation 4. UG is the net benefit that the TR achieves from the transaction. Time decaying is used to to emphasize that UG from recent transaction weigh more compared to UG from old transactions if they have the same absolute value.

$$AUG'_t = \frac{\sum_{j=1}^{G} e^{-\lambda \Delta \mathrm{T}_j} UG_j}{G} \tag{4}$$

Here ΔT$_j$ = Current Time - Time of transaction j, λ is the decaying factor, UG$_j$is the utility gain for transaction j with the TE being evaluated, and G is the size of the historical window considered for calculating AUG'$_t$

The input parameters should be fuzzified before being used in the engine. We define the FDT as the defuzzified output. The individual "if…then" rules for driving the FDT is of the kind "if DT is HIGH and the SUS is LOW, and the AUG'$_t$ is HIGH then the FDT is VERY HIGH". This rule intuitively states that if the estimated direct trust is high and the suspension period is low, and the average utility gained by interacting with this TE is HIGH then the TE is expected to be honest and the transaction result is expected to be very high based on local experience of TR.

In the proposed model, we use the rules presented in Table I. Since each of the input parameters can be categorized as being Low (L), Medium (M), and High (H) and the output parameter can be categorized as being Very Low (VL), Low (L), Medium (M), Very High (VH), and High (H). We use a Mamdani min-max approach of inference and the centroid technique for the defuzzification

In table I, we insisted that a recently suspended TE will have a low direct trust value. The idea is that a TR will stop interacting with a misbehaving TE immediately, and wait utile it is clear whether this misbehaviour is accidental or it is a behavioural change. Because suspension is temporary,

TABLE I
DIRECT TRUST FUZZY SUBSYSTEM RULES

Rule	DT	Suspension	AUG'	Output
1	L	L	L	L
2	M	L	L	L
3	H	L	L	M
4	L	M	L	L
5	M	M	L	L
6	H	M	L	M
7	L	H	L	VL
8	M	H	L	VL
9	H	H	L	VL
10	L	L	M	L
11	M	L	M	L
12	H	L	M	H
13	L	M	M	L
14	M	M	M	L
15	H	M	M	H
16	L	H	M	VL
17	M	H	M	VL
18	H	H	M	VL
19	L	L	H	L
20	M	L	H	M
21	H	L	H	VH
22	L	M	H	L
23	M	M	H	L
24	H	M	H	H
25	L	H	H	VL
26	M	H	H	VL
27	H	H	H	VL

and because TR uses information from witnesses, the effect of accidental misbehavior will phase out, but the effect of a behaviour change will be magnified

C. Indirect Trustworthiness Estimation IT(TR, TE):

To estimate indirect trust, a TR consults other witnesses who interacted previously with the TE. To reduce the effect of fraudulent witnesses, a TR excludes reports from any witness where the mean of the differences between the witness's trustworthiness estimation and the TR's trustworthiness estimation of TEs other than the one under consideration is above the witnesses differences threshold (WDT). An honest witness (WT) reports its testimony (RT) about a TE as

$$RT(WT,\, TE) = FDT(WT,\, TE) \tag{5}$$

where FDT (WT, TE) is the WT fuzzy direct experience of trustworthiness estimation of the TE.

A TR will calculate the indirect trust (IT) component as

$$IT(TR,\, TE) = \frac{\sum_{k=1}^{N} weight_k * RT(WT_k,\, TE)}{N} \tag{6}$$

where N is the number of consulted witnesses. RT (WT$_k$, TE) is the testimony of witness k about TE, and weight$_k$ is the weight assigned by the TR to testimony of WT$_k$. The calculation of the weight factor, or the adaptation of a calculation technique from the literature, is considered a future work.

TRs track the credibility of their witnesses. Each TR updates its rating for the witnesses after each interaction as follows

- If the transaction was satisfactory for the TR and the witness WT had recommended TE or

- If the transaction was NOT satisfactory and WT's opinion was "not recommend".

Then the trustworthiness estimation of WT is incremented as in equation 7

$$DT(TR, WT) = DT(TR, WT) + \gamma(1 - |DT(TR, WT)|) \quad (7)$$

- Otherwise, the trustworthiness estimation of WT is decremented as in equation 7

$$DT(TR, WT) = DT(TR, WT) + \zeta(1 - |DT(TR, WT)|) \quad (8)$$

where γ and ζ are positive and negative factors respectively and chosen by the TR as cooperation and noncooperation factors. The value of DT (TR, WT) varies from -1 to 1. A witness is considered trustworthy if the trustworthiness estimation is above the witnesses' honesty threshold (WHT). A witness is considered untrustworthy if the trustworthiness estimation falls below the witnesses' fraudulence threshold (WFT). Witnesses with trustworthiness estimation values in between the two thresholds are considered neutral.

When a TR wants to interact with a TE at instant i, the TR avoids any WT that is untrustworthy.

D. Total Trustworthiness Estimation TT(TR,TE):

The proposed trust model takes into consideration TRs' direct trust of TE(s), testimonials of witnesses, and credibility of witnesses. Therefore, the total trust estimate can be calculated using Eq. (9)

$$TT(TR, TE) = x * FDT(TR, TE) + (1-x) * IT(TR, TE) \quad (9)$$

Here FDT(TR,TE) is the fuzzy direct experience estimation component of the TR for the TE, IT(TR,TE) is the indirect trust estimation component of the TR for the TE and x is a positive factor, chosen by the TR, which determines the weight of each component in the model.

E. Trustee Selection

In the proposed model, TRs request TEs to bid for the coming interactioneach. Each TR uses a fuzzy engine to select a profitable TE. We define three input parameters for the fuzzy engine of the trust model and one output parameter; the input parameters are

- The total trust estimation (TT), as calculated by combining information from direct experience and testimony of witnesses, detail calculations described later in this section.
- Bidding Difference (BD): The difference between the promised UG, i.e. bidding value, of the TE (B_t) and the average bidding values of all TEs bidding for the same transaction.This parameter is used to differentiate a TE that promises high UG while the average promise is relatively low, from one that promises high UG while almost every TE promises high UG. In both cases, the TE promises high UG. but this value in more important in the first case compared to the second case

$$BD = B_t - \frac{\sum_{l=1}^{M} B_l}{M} \quad (10)$$

- Average UG Difference (DAUG'$_t$): The difference between the average of time-decayed UG within the last H interactions with the TE and the average of time-decayed utility gain within the last H interactions with all TEs. Time decaying is used to emphasize that recent transaction weighs more compared to old transactions if they have the same value of UG.

$$DAUG'_t = \frac{\sum_{p=1}^{H} e^{-\lambda \Delta \mathrm{T}} UG_p}{H} - \frac{\sum_{q=1}^{H} e^{-\lambda \Delta \mathrm{T}} \bar{UG}_q}{H} \quad (11)$$

Here ΔT = Current Time - Time of the transaction. $UG_p$is the utility gain for transaction p with the TE being evaluated, $\bar{UG}_q$ is the utility gain for transaction q, regardless of the TE, and H is the size of the historical window considered for calculating AUG'$_t$

The input parameters should be fuzzified before being used in the engine. We define the fuzzy estimated utility gain (FUG) as the defuzzified output parameter. The individual "if…then" rules for driving the fuzzy estimated utility gain FUG is of the kind "if the estimated total trust is HIGH and the DAUG'$_t$ is HIGH and the BD is HIGH then the Fuzzy UG is VERY HIGH". This rule intuitively states that if the estimated trust is high and utility gained by interacting with this TE is higher than the overall average utility gain, and the TE is promising higher utility gain compared to other bidding TEs, then the TE is expected to be honest and the transaction result is expected to be very high.

In the proposed model, we use the rules presented in Table II for TSF. Since each of the input parameters can be categorized as being Low (L), Medium (M), and High (H) and the output parameter can be categorized as being Very Low (VL), Low (L), Medium (M), Very High (VH), and High (H). Here, again, we use a Mamdani min-max approach to inference and the centroid technique for the defuzzification.

TR evaluates the trustworthiness of TEs that are not untrustworthy. TEs whose trustworthiness cannot be determined (due to no available rating) are placed in the Unknown Trust (UT) set. Those, whose trustworthiness has been determined, are placed in the Known Trust (KT) set. On one side, selecting a TE from the set KT is likely to give a more predictable value for the expected UG. However, the TR has not learnt enough about the TE population, therefore, it may get a non-optimal performance. On the other side, selecting a TE from the set UT allows TR to explore more about the TE population, although it may risk losing utility if it encounters a bad TE [2]. To encourage honest bidding when selecting a TE from UT set, a random TE with the second highest bidding value is selected.

Obviously, if one of the two sets is empty, TR can only select from the other set. Otherwise, it needs to determine

TABLE II
TRUSTEE SELECTION RULES

Rule	TT	BD	DAUG'$_t$	Output
1	L	L	L	VL
2	M	L	L	L
3	H	L	L	M
4	L	M	L	L
5	M	M	L	M
6	H	M	L	M
7	L	H	L	L
8	M	H	L	M
9	H	H	L	M
10	L	L	M	L
11	M	L	M	M
12	H	L	M	M
13	L	M	M	L
14	M	M	M	M
15	H	M	M	M
16	L	H	M	M
17	M	H	M	H
18	H	H	M	H
19	L	L	H	L
20	M	L	H	M
21	H	L	H	H
22	L	M	H	L
23	M	M	H	M
24	H	M	H	H
25	L	H	H	M
26	M	H	H	H
27	H	H	H	VH

TABLE III
VALUES OF USED PARAMETERS

Parameter	Value
Total number of Trustees	10
Total Number of trustors	100
Number of Good Trustees	2
Number of Bad Trustees	3
Number of Ordinary Trustees	3
Number of Intermittent Trustees	2
Number of Categories for Trustees	4
Maximum utility gain	10
trustee cooperation factor	0.1
trustee non-cooperation factor	-0.3
Witnesses cooperation factor	0.1
Witnesses non-cooperation factor	-0.3
Direct trust fraction	0.5
Degree of decay	0.1
Trustees' honesty threshold	0.5
Trustees' fraudulent threshold	-0.5
Witnesses' honesty threshold	0.5
Witnesses' fraudulence threshold	-0.5
Trustor satisfactory threshold	0
Witnesses differences threshold	0.5

which action it should take. The exploit-vs explore dilemma can be addressed by using the Boltzmann exploration strategy [20]. Using this strategy, an agent tends to explore its environment first and then gradually move towards exploitation when it learns more about the environment. When exploiting, TR selects the TE with the highest FUG.

V. Performance Evaluation

It is often difficult to find suitable real world data set for comprehensive evaluation of trust models, since the effectiveness of various trust models needs to be assessed under different environmental conditions and misbehaviors [3]. Therefore, in trust modeling for MASs research field, most of the existing trust models are assessed using simulation or synthetic data [3]. One of the most popular simulation test-beds for trust models is the agent reputation and trust (ART) test-bed proposed in [21]. However, even this test-bed does not claim to be able to simulate all experimental conditions of interest. For this reason, many researchers design their own simulation environments when assessing the performance of their proposed trust models [3].

A. Simulation Environment

We use simulation to evaluate the performance of the proposed model for distributed, multi-agent environment using the discrete-event multi-agent simulation toolkit MASON [22] with TEs, that provide services, and TRs, that consume services. For the Fuzzy subsystems, we used the jFuzzyLogic Java package [23]. As with [2], we assume that the performance of a TE in a particular service is independent from that in another service. Therefore, without loss of generality, and in order to reduce the complexity of the simulation environment, it is assumed that there is only one type of service in the system simulated and all TEs offer the same service with, possibly, different performance. In order to study the performance of the proposed trust model for TE selection, we compare the proposed model with the well known FIRE trust model [2]. Each simulation experiment is repeated 10 times with different seed values for the random number generators, and the average of the 10 experiments is presented as the simulation result. Network communication effects are not considered in this simulation. Each agent can reach each other agent. The simulation step is used as the time value for interactions. Transactions that take place in the same simulation step are considered simultaneous. Locating TEs and witnesses are not part of the proposed model; therefore, TRs locate TEs and witnesses through the system. TRs evaluate the trustworthiness of the TE(s), and then selects one to interact with.

Having selected a TE, the TR then interact with the selected TE and gains some utility from the transaction (UG). The value of UG is in $[-10, 10]$ and depends on the level of performance of the TE in that transaction. A TE can serve many users at a time. A TR does not always use the service in every round. The probability it needs and requests the service, called its activity level, is selected randomly when the agent is created.

After each transaction, the TR updates the credibility of the TE participated in the transaction. It is assumed that TEs may be selfish, liars, non cooperative or simply malfunctioning. In order to compare our work with FIRE [HuynhJS2006], honest witnesses assumed.

TEs can be in one of four types: good, ordinary, bad, and intermittent. Each of them, except the last, has a mean level of performance. The actual performance follows a normal distribution around this mean which is in the reange of (5,10] for good TEs, [0, 5] for ordinary TEs and [-10,0) for bad

TABLE IV
DIRECT TRUST FUZZY SUBSYSTEM INPUT AND OUTPUT MFS

	DT	Suspension	AUG'	Output
VL	- -	- -	- -	(PWL) -1.0, -0.5, -0.3
L	(PWL) -1.0,-0.5,0.01	(PWL) 0.0,0.25,0.5	(PWL) -10, -0.1, 1.0	(TR) -0.35, -0.1, 0.1
M	(TR) 0.0, 0.2, 0.5	(TR) 0.4, 0.7, 1.0	(TR) 0, 2.5, 5	(TR) 0.0, 0.2, 0.4
H	(PWL) 0.4, 0.7, 1	(PWL) 0.9, 1, 999	(PWL) 4, 6, 10	(PWL) 0.3, 0.5, 0.6
VH	- -	- -	- -	(PWL) 0.5, 0.7, 1.0
legend	PWL: piece-wise linear		TR:triangular	

TABLE V
TRUSTEE SELECTION INPUT AND OUTPUT MFS

	TT	BD	DAUG'$_t$	Output
VL	- -	- -	- -	(PWL) 0.0, 0.25, 0.4
L	(PWL) -1.0,-0.1,0.1	(PWL) -20.0, -0.1, 0.1	(PWL) -20, -0.1, 0.1	(TR) 0.3, 0.5, 0.9
M	(TR) 0.0, 0.2, 0.6	(TR) 0.0, 1.0, 3.0	(TR) -0.1, 0.5, 2	(TR) 0.8, 1.0, 1.2
H	(PWL) 0.5, 0.7, 1	(PWL) 2, 4, 20	(PWL) 1.5, 2.5, 20	(PWL) 1.1, 1.4, 1.6
VH	- -	- -	- -	(PWL) 1.5, 1.6, 2.0
legend	PWL: piece-wise linear		TR:triangular	

TEs. Intermittent trstees, on the other hand, yield (random) performance levels in the range [-10, 10] and they can result in positive UG some times and negative UG other times.

Since agents are owned and controlled by various stakeholders, the performance of an agent may not be consistent over time. A TE may change its behavior. In this simulation study, the performance of a TE can be changed by a randomly selected amount with a probability selected randomly when the agent is created. When bidding, an honest (good or ordinary) TE bids its utility gain value, this value is considered the value of the transaction with the corresponding TR. A bad (unhonest) TE bids a positive value for its utility gain, but the utility gain that the corresponding TR can get is the true utility gain that the bad TE can afford (negative value).

Table III presents the number of agents, and other parameters used in the proposed model and those used in the environment.

The membership functions for the input, output parameters used for direct trust fuzzy subsystem in our evaluation are summarized in Table IV, the membership functions for the input, and output parameters used for the TE selection fuzzy subsystem in our evaluation are summarized in Table V.

B. Experimental results

1) Performance in a static environment: The first thing to test is whether the proposed model helps TRs select profitable TEs (i.e. those yielding positive UG) from the population and, by so doing, helps them gain better utility than when using FIRE trust model. In this section, we use a static environment, which means that each TR attempt to make a transaction each step, and witnesses and TEs do not change their honesty levels.

Figure 3 describes the average UG per transaction as the number of transactions increases from 5 to 50 in the static environment. The charted UG is calculated as the averaged value for 10 different runs of the experiment. For each run, the summation of UG that all TRs accumulated at the end of each fifth simulation step is divided by the number of TRs (note that in the static environment, each TR interact in each simulation step.). The figure shows that selecting providers using the proposed model perform closely to FIRE despite the fact that FIRE make use of rule-based trust that can't be assumed to be available all the time. Moreover, the performance of both

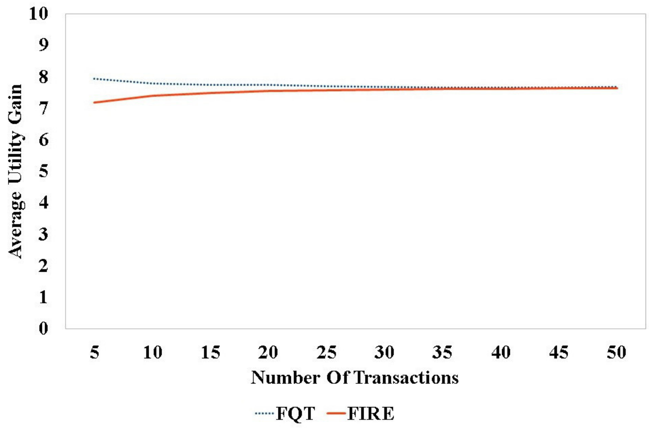

Fig. 3. Performance in Static Environment

models stabilize after a while. For FIRE, this is consistent with the results obtained in [2]. This stabilization in the performance of the two models indicates that they both learned to interact with the most beneficial TEs in the system

2) Performance in a dynamic environment: A trust model designed for MAS should be able to function properly in a dynamic environment. In this section we test the performance of the proposed model in a changing environment, as described below. As with static environment, we compare the performance of the proposed model with the case of using FIRE.

Specifically, the same experiments will be run, but with each of the following conditions: each TE may alter its average level of performance at maximum 1.0 UG unit with a probability of 0.10 each simulation step. A TR uses the service with probability in the range [0.25 - 1.0], intermittent TEs flip their honesty randomly and TEs may leave the system and new TEs may join the system with probability 0.5

Figure 4 describes the average UG per transaction as the number of transactions increases from 5 to 50 in the dynamic environment. The charted average UG is calculated as the averaged value for 10 different runs of the experiment. For each run, the summation of UG that all TRs accumulated when the total number of transaction in the system equals a multiple of five of the number of TRs is divided by the number of TRs. This value is averaged for 10 different runs of the experiment.

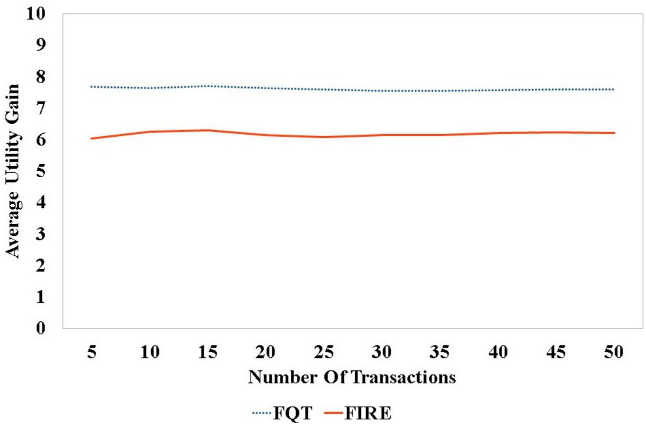

Fig. 4. Performance in Dynamic Environment

The figure shows that selecting TEs using the proposed model performs consistently better than using FIRE in terms of UG in a dynamic environment, which indicates that the proposed model responds better to dynamic changes compared to FIRE. Moreover, the performance of both models stabilize after a while. For FIRE, this is consistent with the results obtained in [2]. However, FIRE is not able to respond to the dynamics of the system as fast as the proposed model. This is due to the use of the fuzzy subsystems and due to picking the TE with the second highest bid.

VI. Conclusion and Future Work

In this paper, we presented a trust model for MASs that combines the use of Q-learning to estimate trustworthiness and incorporate two fuzzy subsystems for TE selection to enhance the utility gain estimation. The presented model allows direct and indirect sources of trust information to be integrated to provide a collective trust estimation. In addition, the proposed model incorporates fuzzy subsystems to account for suspension periods, average utility gain, bidding differences, and the relative average utility gain of a TE compared to the overall utility gain. The proposed model has been simulated using MASON with the use of the jFuzzyLogic package. The results indicate that the model can help TRs enhance their utility gain and that the proposed model can respond better to dynamic changes in the environment. In short, we believe the proposed model can provide a trust measure that is sufficiently useful to be used in MASs

Dynamically determining parameter values for the fuzzy subsystems, enabling TEs to actively promote their honesty, bootstrapping trust for new TEs and using Q-learning to dynamically select the proper action in each rule of the fuzzy subsystem are considered as future work.

References

[1] C. Burnett, T. J. Norman, and K. Sycara, "Trust decision-making in multi-agent systems," in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume One*, ser. IJCAI'11. AAAI Press, 2011, pp. 115–120.

[2] T. D. Huynh, N. R. Jennings, and N. R. Shadbolt, "An integrated trust and reputation model for open multi-agent systems," *Autonomous Agents and Multi-Agent Systems*, vol. 13, no. 2, pp. 119–154, Sep. 2006.

[3] H. Yu, Z. Shen, C. Leung, C. Miao, and V. Lesser, "A survey of multi-agent trust management systems," *Access, IEEE*, vol. 1, pp. 35–50, 2013.

[4] B. Khosravifar, J. Bentahar, M. Gomrokchi, and R. Alam, "Crm: An efficient trust and reputation model for agent computing," *Know.-Based Syst.*, vol. 30, pp. 1–16, Jun. 2012.

[5] S. D. Ramchurn, D. Huynh, and N. R. Jennings, "Trust in multi-agent systems," *Knowl. Eng. Rev.*, vol. 19, no. 1, pp. 1–25, Mar. 2004.

[6] N. Griffiths, K.-M. Chao, and M. Younas, "Fuzzy trust for peer-to-peer systems," in *Distributed Computing Systems Workshops, 2006. ICDCS Workshops 2006. 26th IEEE International Conference on*, July 2006, pp. 73–73.

[7] S. Song, K. Hwang, R. Zhou, and Y.-K. Kwok, "Trusted p2p transactions with fuzzy reputation aggregation," *Internet Computing, IEEE*, vol. 9, no. 6, pp. 24–34, Nov 2005.

[8] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in *Proceedings of the 12th International Conference on World Wide Web*, ser. WWW '03. New York, NY, USA: ACM, 2003, pp. 640–651.

[9] T. Tran and R. Cohen, "Improving user satisfaction in agent-based electronic marketplaces by reputation modelling and adjustable product quality," in *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 2*, ser. AAMAS '04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 828–835.

[10] T. Tran, "Protecting buying agents in e-marketplaces by direct experience trust modelling," *Knowledge and Information Systems*, vol. 22, no. 1, pp. 65–100, 2010.

[11] K. Regan and R. Cohen, "Indirect reputation assessment for adaptive buying agents in electronic markets," *Business Agents and the Semantic Web workshop*, vol. 1, 2005.

[12] K. Regan, R. Cohen, and T. Tran, "Sharing models of sellers amongst buying agents in electronic marketplaces," *Decentralized Agent Based and Social Approaches to User Modelling workshop*, vol. 1, 2005.

[13] S. Beldona, "Reputation based buyer strategies for seller selection in electronic markets," Ph.D. dissertation, Electrical Engineering & Computer Science, University of Kansas, 2008.

[14] I. Pinyol and J. Sabater-Mir, "Computational trust and reputation models for open multi-agent systems: A review," *Artif. Intell. Rev.*, vol. 40, no. 1, pp. 1–25, Jun. 2013.

[15] S. Georgoulas, K. Moessner, A. Mansour, M. Pissarides, and P. Spapis, "A fuzzy reinforcement learning approach for pre-congestion notification based admission control," in *Proceedings of the 6th IFIP WG 6.6 International Autonomous Infrastructure, Management, and Security Conference on Dependable Networks and Services*, ser. AIMS'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 26–37.

[16] J. Mendel, "Fuzzy logic systems for engineering: a tutorial," *Proceedings of the IEEE*, vol. 83, no. 3, pp. 345–377, Mar 1995.

[17] C. Pappis and C. Siettos, "Fuzzy reasoning," in *Search Methodologies*, E. Burke and G. Kendall, Eds. Springer US, 2005, pp. 437–474.

[18] E. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1 – 13, 1975.

[19] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.

[20] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Int. Res.*, vol. 4, no. 1, pp. 237–285, May 1996.

[21] K. K. Fullam, T. B. Klos, G. Muller, J. Sabater, A. Schlosser, Z. Topol, K. S. Barber, J. S. Rosenschein, L. Vercouter, and M. Voss, "A specification of the agent reputation and trust (art) testbed: Experimentation and competition for trust in agent societies," in *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, ser. AAMAS '05. New York, NY, USA: ACM, 2005, pp. 512–518.

[22] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan, "Mason: A multiagent simulation environment," *Simulation*, vol. 81, no. 7, pp. 517–527, Jul. 2005.

[23] P. Cingolani and J. Alcala-Fdez, "jfuzzylogic: a robust and flexible fuzzy-logic inference system language implementation," in *Fuzzy Systems (FUZZ-IEEE), 2012 IEEE International Conference on*, June 2012, pp. 1–8.