

Protecting buying agents in e-marketplaces by direct experience trust modelling

Thomas Tran

Received: 16 May 2008 / Revised: 14 October 2008 / Accepted: 17 November 2008 /
Published online: 14 January 2009
© Springer-Verlag London Limited 2008

Abstract In this paper, we describe a framework for modelling the trustworthiness of sellers in the context of an electronic marketplace where multiple selling agents may offer the same good with different qualities and selling agents may alter the quality of their goods. We consider that there may be dishonest sellers in the market (for example, agents who offer goods with high quality and later offer the same goods with very low quality). In our approach, buying agents use a combination of reinforcement learning and trust modelling to enhance their knowledge about selling agents and hence their opportunities to purchase high value goods in the marketplace. This paper focuses on presenting the theoretical results demonstrating how the modelling of trust can protect buying agents from dishonest selling agents. The results show that our proposed buying agents will not be harmed infinitely by dishonest selling agents and therefore will not incur infinite loss, if they are cautious in setting their penalty factor. We also discuss the value of our particular model for trust, in contrast with related work and conclude with directions for future research.

Keywords Adaptive and learning agents · Trust modelling · Multi-agent scenarios · Electronic commerce applications

1 Introduction

In electronic marketplaces, modelling the trustworthiness of selling agents is a mechanism for determining whether to approach these agents in future interactions. In a realistic electronic marketplace where selling agents are able to alter both the price and quality of their goods over time and where buying agents can only evaluate the quality of a good after purchase, dishonest selling agents may exist. It is therefore critical for buying agents to develop

T. Tran (✉)
School of Information Technology and Engineering, University of Ottawa,
800 King Edward Avenue, Ottawa, ON, Canada K1N 6N5
e-mail: ttran@site.uottawa.ca

strategies to effectively model the behaviour the selling agents in the marketplace, in order to mitigate the possible harm inflicted by any dishonest sellers.

In our view, product quality should be a multi-faceted concept. That is, a buyer may judge the quality of a product it purchases based on a combination of various factors that are deemed important to the buyer. For a physical product, these factors are likely the product characteristics, whether the product condition is as advertised, whether it is delivered on time, the level of its after-sale technical supports and customer services, etc. For an information good, these factors are perhaps the server-supported download rate, whether or not the information is free of error, the presentation of information (e.g., plain text or text combined with graphics, etc.), the relevance, the timeliness, and the completeness of information,¹ etc. Thus, an MP3 player that is shipped on time to the buyer and is accompanied by excellent technical supports should be considered as having better quality than one that is delivered late and has poor technical supports. Similarly, a file that can be downloaded quickly and contains relevant information should have higher quality than one whose content is only partially relevant and whose server does not support high speed internet access. As noted later in Sect. 3.4, although product quality is viewed as a multi-faceted concept, it can be represented by a single numerical value. This can be achieved by computing product quality as a weighted sum of the quality factors.

This paper presents a framework for modelling electronic marketplaces and for directing the processing of buying agents in those marketplaces. In particular, instead of simply presenting a strategy for modelling the trustworthiness of selling agents, we focus on presenting the theoretical results that show that this strategy will prevent infinite harm from dishonest sellers. An agent that can provide this kind of “protection” will indeed be providing satisfaction for its user.² The model that we present is one where if a buyer is careful in setting the penalty factor that it applies to adjust the trust ratings of sellers in the marketplace, then the maximum loss that buyer can incur from non-cooperative transactions from sellers is in fact bounded above by a constant term. This therefore ensures that infinite loss cannot occur. Without an effective mechanism for detecting and avoiding dishonest sellers, a buyer may repeatedly purchase undesirable goods from the same dishonest seller and constantly increase its total loss. Our framework is designed to avoid this situation.

Note that our proposed approach to modelling the trustworthiness of sellers is one where each buying agent determines how trustworthy a selling agent is, based on its own past experiences with that agent. We do not rely on ratings provided by other buyers in the marketplace. As such, when we refer to the trustworthiness of a seller, we mean the impression formed of that seller in the past, by that same buyer.

1.1 Direct experience as valuable

Modeling the trustworthiness of selling agents on the basis of direct experience has been acknowledged by several researchers as an important component of a comprehensive trust modeling system. Fullam and Barber [10] has recently conducted research into how best to use a balance of experience and reputation in the judgment of selling agents’ trustworthiness. This work makes clear that it is sometimes beneficial to rely more on experience, thus necessitating an effective method for modeling trustworthiness of agents on this basis alone.

¹ Relevance of information means the extent to which the information is appropriate and helpful; timeliness indicates the extent to which the information is sufficiently up-to-date; and completeness refers to the extent to which the information is not partially missing and is of sufficient breadth and depth for the user’s purpose.

² The human user (or owner) of a buying agent should receive greater satisfaction if the agent is protected from the dishonest selling agents in the market.

Huynh, Jennings, and Shadbolt [11] proposed FIRE, a framework for trust that incorporates direct trust as well as role-based trust and reputation. Here, the direct trust model of Sabater [23] is proposed as the method for capturing this element of the overall calculation, with some adjustment to consider more carefully the decay of trust values over time. As discussed later in Sect. 5, the model we present in this paper has some advantages over that of Sabater. In this respect, it may provide a stronger basis for the direct trust evaluation, for systems such as FIRE.

In addition, some researchers have argued for the value of an agent learning without sharing information (e.g., [24]), as way to guard against processing incorrect or deceptive information provided by others. In our market model, buyers learn to select sellers based on their own experiences and do not communicate with other buyers in the market. In this way, buyers can act independently and anonymously without being affected by communication delays (due to other buyers being busy), the failure of some key buyer (whose buying policy influences other buyers), or the reliability of the information received from other buyers.

1.2 An overview of our approach for using direct experience

Our model for evaluating the trustworthiness of sellers has some elements that are especially valuable. The modelling of trustworthiness first of all allows buying agents to optimize their expected product values by selecting appropriate sellers to do business with among the set of trustworthy sellers and avoiding the untrustworthy ones. This strategy should increase the buyer's chances of purchasing high quality goods and reduce the risk of receiving low value ones, and therefore bring better satisfaction to the buyer. This is in contrast to other approaches for the design of intelligent agents in electronic marketplaces (e.g., [5, 30]), that do not attempt to explicitly model the trustworthiness of sellers, to drive the buying decisions. In our model, a buyer decides which seller to interact with by narrowing the search to a list of sellers deemed to be trustworthy; sellers who are not trustworthy are either in the untrustworthy set (and are therefore excluded from consideration), or may be consulted in an exploration phase, if they are neither trustworthy nor untrustworthy. There is therefore significant benefit to including the three distinct sets of trustability for sellers. With our examination of protection from infinite harm, we have shown that trust can in fact be modelled in such a way that it ultimately provides very useful information for buyers.

Along with our modelling of trust, our approach provides a mechanism for new trustworthy sellers to enter the marketplace. In fact, since sellers may be learning to improve their profits, some non-trustworthy sellers may have reasonably adjusted their prices and greatly improved the quality of their goods, and thus should be reconsidered as trustworthy sellers. Our proposed buying algorithm accounts for these possibilities by letting a buyer b explore the marketplace with probability ρ to discover new trustworthy sellers.

A proposed buyer, in its exploitation time, will choose a seller from the set of trustworthy sellers. Since this trustworthy set is usually quite smaller (in terms of cardinality) than the set of all sellers in the market, the proposed buying algorithm should reduce computational time, and accordingly result in improved time-performance for the buyer (compared to the case where the buyer has to consider all possible sellers in the market). This is especially important in those application domains where the buyer is required to calculate a suitable seller within a constrained time frame.

This paper provides a theoretical exploration of desirable values for λ , the penalty factor for non-cooperative agents. As such, we demonstrate how penalty factors can be effectively incorporated into the modelling of trust. In addition, we show how trust modelling can be effective in environments where some users are actually attempting to cheat the buyers.

One of the most important issues for any model of trust is how to adjust it. A good adjusting (or updating) scheme should base its updating conditions on appropriate factors in order to achieve reasonable modelling of trust; yet the updating scheme should not be too complicated to implement. The proposed trust mechanism uses a feasible updating scheme³ the operation of which is based on the comparison of the true value of the good purchased by a buyer and the product value demanded by that buyer. These updating conditions depend on three factors, namely (i) the price of the good, (ii) the quality of the good, and (iii) the relative importance between price and quality as viewed by the buyer. Since these are the most significant factors for the buyer to determine the value of the good purchased, we believe that our updating scheme is appropriate for market settings and therefore provides reasonably accurate modelling of sellers' trustworthiness.

The remaining of this paper is organized as follows: Sect. 2 provides background information on reinforcement learning which is incorporated as a learning element into our approach. Section 3 describes our proposed algorithms for buying and selling agents, respectively. Section 4, the focus of our paper, presents the theoretical results which show that our proposed model can protect buying agents from infinite harm caused by dishonest selling agents. Section 5 discusses the value of our model in the context of related work. Section 6 suggests possible directions for future research, and Sect. 7 concludes the paper.

2 Reinforcement learning

As detailed below in Sect. 3, our proposed approach integrates reinforcement learning into trust modelling.⁴ Therefore, this section is devoted to provide readers with sufficient background information on reinforcement learning. We first introduce the reinforcement learning problem followed by a number of illustrating examples to clarify some technical terms. We then present the formal definitions and notations used in the description of the reinforcement learning methods. Finally, we briefly discuss different well-known reinforcement learning methods, including a detailed description of the one which we integrate into our trust model (and which we compared our proposed approach to, as mentioned in the foot note above). Our main references for this section are [22, 26].

2.1 The reinforcement learning problem

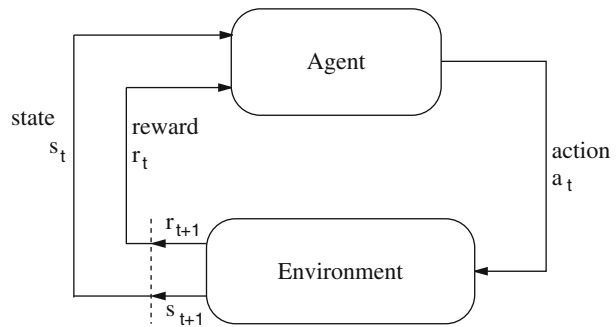
The reinforcement learning problem is the problem of learning from interaction to achieve a goal. In this problem, an agent observes a current state s of the environment, performs an action a on the environment, and receives a feedback r from the environment. This feedback is also called *reward*, or *reinforcement*. The goal of the agent is to maximize the cumulative reward it receives in the long run.

More specifically, the agent and its environment interact at each of a sequence of discrete time steps, $t = 1, 2, 3, \dots$. At each time step t , the agent receives some representation of the environmental state $s_t \in S$, where S is the set of possible states. On that basis, the agent selects and performs an action $a_t \in A$, where A is the set of possible actions. One time step

³ See Eqs. (12) and (13) in Sect. 3.

⁴ We have demonstrated, in [28], the value of this integration approach in comparison to the approach that relies on reinforcement learning alone, without modelling trust. Different from [28], the main focus of this paper is to present the theoretical results that show that our approach can protect buying agents from being harmed infinitely by dishonest sellers.

Fig. 1 The agent–environment interaction in the reinforcement learning problem



later, as the consequence of its action, the agent receives a numerical reward $r_{t+1} \in \mathbb{R}$, and finds itself in a new state s_{t+1} . Figure 1 illustrates the agent–environment interaction.

At each time step, the agent implements a policy π , which is a mapping from the current state into the desirable action to be performed in that state (i.e., $\pi : S \mapsto A$). Solving a reinforcement learning problem means finding a policy that maximizes the reward that the agent receives over the long run. Such a policy is called the optimal policy.

2.2 Examples

This section clarifies the terms *environmental state*, *actions* and *rewards* used in the reinforcement learning problem description above by presenting a number of illustrating examples.

2.2.1 Game playing

A reinforcement learning agent can be used to help a chess player to improve his play. At each time step, the environmental state can be the current configuration of all the pieces on the chessboard. The actions are all the possible moves. The rewards may be zero for most of the time and +1 when the player wins. Alternatively, we may give a reward of −1 for every time step until the game ends. This will encourage the agent to win as quickly as possible.

2.2.2 Bio-reactor

Consider a bio-reactor which is essentially a large vat of nutrients and bacteria used to produce a useful chemical. The rate at which the useful chemical is produced depends on moment-by-moment temperatures and stirring rates for the bio-reactor. Reinforcement learning can be applied to determine the optimal temperature and stirring rate at each time step. The states in this case are likely to be sensor readings of thermocouple plus symbolic inputs representing the ingredients in the vat and the target chemical. The actions may be target temperatures and target stirring rates that are passed to low-level control systems that, in turn, activate heating elements and motors to obtain the targets. The rewards may be moment-by-moment measures of the rate at which the useful chemical is produced. We notice that in this example each state is a vector (or list) of sensor readings and symbolic inputs, and each action is also a vector composed of target temperature and stirring rate. It is typical for a reinforcement learning problem to have states and actions represented as vectors. Rewards, however, should always be single numerical values.

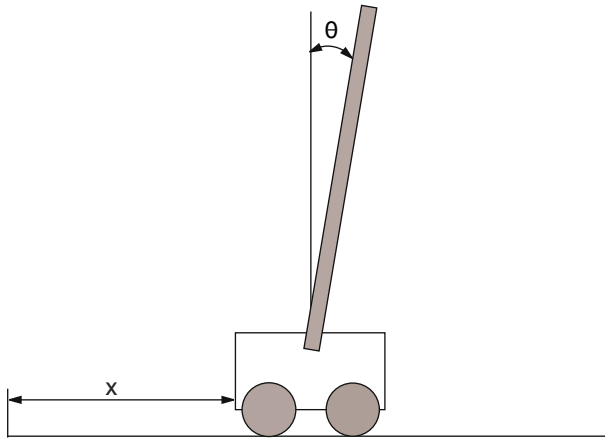


Fig. 2 The *inverted pendulum* (also known as the *pole balancing*) problem

2.2.3 Recycling robot

Consider a mobile robot whose job is to collect empty pop cans in an office environment. The robot has sensors to detect cans, an arm with gripper to pick up a can to place it in an onboard bin, as well as a navigation system to help it move around. It operates on a rechargeable battery. A reinforcement learning agent can be used to make high-level decisions on how to search for cans. Basically, the agent has to decide which of the following three actions the robot should perform: (i) actively searching for a can for a certain period of time, (ii) remaining stationary to wait for someone to bring it a can, or (iii) heading back to his home base to recharge its battery. At each time step, the environmental state should be the state of the battery. The rewards can be zero most of the time, $+1$ when the robot is able to collect an empty can, and -1 if the battery goes all the way down. In this example, the reinforcement learning agent is not the entire robot, and the environmental states describe conditions within the robot, not the external environment. Reinforcement learning is used for the robot to make decisions on which action to take at each state, in order to maximize the cumulative reward the robot receives over the long run.

2.2.4 Inverted pendulum

The setup for a famous reinforcement learning problem known as *inverted pendulum* or *pole balancing* is shown in Fig. 2.

The goal of the inverted pendulum problem is to apply forces on a cart moving along a track so that the pole hinged from the cart will not fall over. A failure is said to occur if the pole falls past a given angle from vertical, or if the cart reaches an end of the track. The pole is reset to vertical after each failure. For this problem, a state may consist of the distance x and the angle θ , which are continuous variables. The actions are clearly *jerk left* or *jerk right*, which are discrete. The rewards could be $+1$ for a time step during which a failure did not occur. Alternatively, we could choose the rewards to be -1 for each failure and zero at all other times.

2.3 Definitions and notations

This subsection presents some necessary definitions and notations, which will be used in the description of the reinforcement learning methods in Sect. 2.4.

Let $r_{t+1}, r_{t+2}, r_{t+3}, \dots$, be the sequence of rewards received after time step t . We define the *return*, R_t , to be

$$R_t = r_{t+1} + r_{t+2} + r_{t+3} + \dots + r_T, \quad (1)$$

where T is the final time step. This definition makes sense in applications in which there is a natural notion of the final time step, i.e., when the agent–environment interaction breaks naturally into subsequences called *episodes*. In cases where the agent–environment interaction does not break into episodes but goes on continually without limit, we define the return R_t to be

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots = \sum_{k=0}^{\infty} (\gamma^k r_{t+k+1}), \quad (2)$$

where γ is called the *discount rate* ($0 \leq \gamma \leq 1$). The discount rate γ determines the present value of future rewards. That is, a reward received k time steps in the future is worth only γ^{k-1} times what it would be worth if it were received immediately.

The goal of the agent in the reinforcement learning problem is to maximize the *expected return*, denoted by $E\{R_t\}$.

We define the *value of a state s under policy π* , denoted by $V^\pi(s)$, to be the expected return when starting in s and following π thereafter,

$$V^\pi(s) = E_\pi\{R_t | s_t = s\}. \quad (3)$$

The function V^π is called the *state value function* under policy π .

Similarly, we define the *value of taking action a in state s under policy π* , denoted by $Q^\pi(s, a)$, to be the expected return starting from s , taking action a , and thereafter following policy π ,

$$Q^\pi(s, a) = E_\pi\{R_t | s_t = s, a_t = a\}. \quad (4)$$

The function Q^π is called the *state-action value function* under policy π .

Let $P_{ss'}^a$ denote the probability that the environment changes to a new state s' when the agent executes action a in state s . $P_{ss'}^a$ is called the *transition probability*. Let $R_{ss'}^a$ denote the expected value of the next reward received when the agent executes action a in state s and the environment changes to the next state s' . $R_{ss'}^a$ is called the *expected immediate reward*. The quantities $P_{ss'}^a$ and $R_{ss'}^a$ are used to specify the most important aspects of the agent's environment.

2.4 Reinforcement learning methods

Any method that is suited for solving the reinforcement learning problem described in Sect. 2.1 is considered as a reinforcement learning method. There are three well-known, fundamental classes of algorithms for solving the reinforcement learning problem, namely dynamic programming, Monte Carlo, and temporal-difference learning methods. We shall briefly discuss the dynamic programming and Monte Carlo methods, but shall describe the temporal-difference method in greater detail since this is the method that we integrate as a learning

```

Initialize  $Q(s, a)$  arbitrarily
Initialize  $s$ 
Repeat
    Choose  $a$  to perform in  $s$  to maximize  $Q(s, a)$ :
         $a \leftarrow \arg \max_a Q(s, a)$ 
    Take action  $a$ , observe reward  $r$  and transition to new state  $s'$ 
    Update  $Q(s, a)$ :
         $Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$ 
        // where  $\alpha$  is called the learning rate ( $0 \leq \alpha \leq 1$ ) and
        //  $\gamma$  is the discount rate ( $0 \leq \gamma \leq 1$ )
     $s \leftarrow s'$ 
Until  $s$  is terminal

```

Fig. 3 A TD learning algorithm called *Q-learning*

element into our trust model (and also the method that we compared our proposed approach to in [28]).

Dynamic programming (DP) refers to a collection of algorithms that can be used to discover optimal policies for the reinforcement learning problem. DP algorithms are well developed mathematically, but require a complete model of the environment, given by a set of transition probabilities, $P_{ss'}^a$, and a set of expected immediate rewards, $R_{ss'}^a$. In general, a DP algorithm iteratively computes the state value function $V(s)$ with the maximum one taken over all actions.⁵ It then derives the required policy $\pi(s)$ based on this optimal value function.

Monte Carlo (MC) method represent the algorithms that solve the reinforcement learning problem based on averaging sample returns. Unlike DP algorithms, MC algorithms do not require a complete model of the environment. They only need *experience*, that is, sample sequences of states, actions, and rewards from an online or simulated interaction with the environment. MC algorithm are defined only for episodic tasks, in which the agent–environment interaction is divided into episodes. It is only upon the completion of an episode that the value functions, namely $V(s)$ or $Q(s, a)$, and the policies are updated. MC algorithms are thus incremental in an episode-by-episode sense, and not in a step-by-step sense.

Like MC algorithms, temporal-difference (TD) learning algorithms can learn directly from experience without a model of the environment. However, unlike MC algorithms, which must wait until the end of an episode to update the value function (only then is the return R_t known), TD algorithms only need to wait until the next time step. TD algorithms are thus incremental in a step-by-step sense.

One of the most widely used TD algorithms is known as the *Q-learning algorithm*, as illustrated in Fig. 3. 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.

In our proposed algorithms described in Sect. 3, we make use of a slightly different form of TD learning as follows: Given a state s , our agent will select an action that maximizes the state value $V(s)$. While interacting with the environment, the agent attempts to make its

⁵ From now on we shall simplify the function notations $V^\pi(s)$ and $Q^\pi(s, a)$ defined in Sect. 2.3 above as $V(s)$ and $Q(s, a)$, respectively; and shall explicitly mention the policy π when needed.

Fig. 4 The TD learning algorithm used in our approach

```

Initialize  $V(s)$  for all  $s$ 
Repeat forever
  Choose  $a$  to perform in  $s$  to maximize  $V(s)$ :
     $a \leftarrow \arg \max_a V(s)$ 
  Take action  $a$ , observe transition from state  $s$  to state  $s'$  with
  the corresponding state values being  $V(s)$  and  $V(s')$ 
  Update  $V(s)$ :
     $V(s) \leftarrow V(s) + \alpha[V(s') - V(s)]$ 
    // where  $\alpha$  is the learning rate ( $0 \leq \alpha \leq 1$ )
   $s \leftarrow s'$ 

```

estimated state value become more and more accurate. To do this, the agent adjusts the value of the earlier state to be closer to the value of the later state, using an updating rule. Figure 4 illustrates this learning process with the updating rule shown in Eq. (5), where the learning rate α is initially set to 1 and then reduced over time to a small positive fraction.⁶

3 The proposed algorithms

This section presents our agent market model and proposes the learning algorithms for buying agents in electronic marketplaces, based on trust modelling and reinforcement learning. We also describe the processes followed by the selling agents in the marketplace, to complete the context. In addition, we discuss the roles of the parameters used in the proposed algorithms, with some guidelines for calculating or choosing their values.

3.1 The agent market model

We model the agent environment as an open marketplace populated with economic software agents that we classify as *buyers* and *sellers*. These are self-interested agents whose goal is to maximize their own benefit. Our market environment is rooted in an information delivery infrastructure such as the Internet, which provides agents with free access to all other agents. The process of buying and selling goods is realized via a *contract-net* like mechanism [7, 25], which consists of three elementary phases: (i) A buyer announces its request for a good. (ii) Sellers submit bids for delivering such goods. (iii) The buyer evaluates the submitted bids and selects a suitable seller. The buyer then pays the chosen seller and receives the good from that seller. Thus, the buying and selling process can be viewed as an *auction* where a seller is said to be *winning the auction* if it is able to sell its good to the buyer. Figure 5 illustrates the three basic phases of this process.

We assume that the quality of a good offered by different sellers may not be the same, a seller may alter the quality of its goods, and there may be dishonest sellers in the market. We also assume that a buyer can examine the quality of the good it purchases only after it receives that good from the selected seller. Each buyer has some way to evaluate the good it purchases, based on the price and the quality of that good.

⁶ Decreasing the learning rate α over time helps the learning method to converge [22, 26].

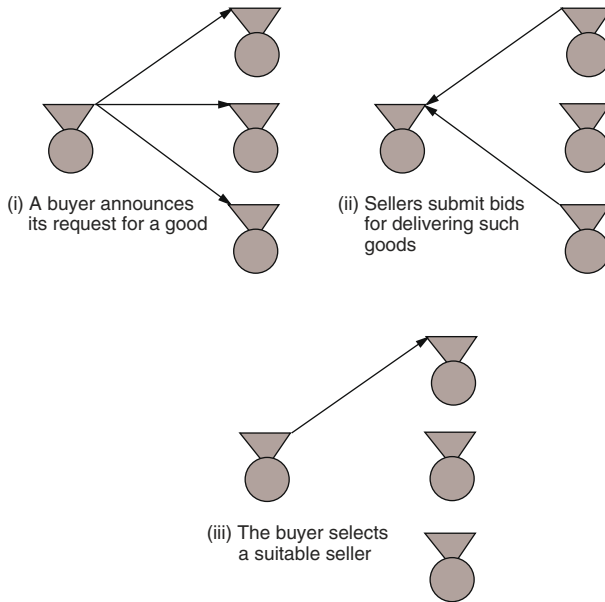


Fig. 5 Three basic phases of the buying and selling process

3.2 Buying algorithm

Consider the scenario where a buyer b announces its request for some good g . Let G be the set of goods, P be the set of prices, and S be the set of all sellers in the marketplace. G , P , and S are finite sets.

Buyer b models the trustworthiness of all sellers in the market using function $r^b : S \mapsto (-1, 1)$, which is called the *trust function* of b . Initially, buyer b sets the *trust rating* $r^b(s) = 0$ for every seller $s \in S$. After each transaction with a seller s , buyer b will update (increase or decrease) $r^b(s)$ depending on whether or not s satisfies b in the transaction. A seller s is considered *trustworthy* by buyer b if $r^b(s) \geq \Omega$, where Ω is buyer b 's *trustworthiness threshold* ($0 < \Omega < 1$). A seller s is considered *untrustworthy* by buyer b if $r^b(s) \leq \theta$, where θ is buyer b 's *untrustworthiness threshold* ($-1 < \theta < 0$). A seller s with $\theta < r^b(s) < \Omega$ is neither trustworthy nor untrustworthy to buyer b . In other words, b does not have enough information to decide on the trustworthiness of s . Let S_t^b and S_{ut}^b be the sets of trustworthy and untrustworthy sellers to buyer b respectively, i.e.,

$$S_t^b = \{s \in S \mid r^b(s) \geq \Omega\} \subseteq S, \quad (6)$$

and

$$S_{ut}^b = \{s \in S \mid r^b(s) \leq \theta\} \subseteq S. \quad (7)$$

Buyer b will focus its business on the trustworthy sellers and stay away from the untrustworthy ones.

Buyer b estimates the expected value of the goods it purchases using the *expected value function* $f^b : G \times P \times S \mapsto \mathbb{R}$. Hence, the real number $f^b(g, p, s)$ represents buyer b 's expected value of buying good g at price p from seller s .

Since multiple sellers may offer good g with different qualities and a seller may alter the quality of its goods, buyer b puts more confidence in the trustworthy sellers. Thus, it chooses among the trustworthy sellers in S_t^b a seller \hat{s} that offers good g at price p with maximum expected value:

$$\hat{s} = \arg \max_{s \in S_t^b} f^b(g, p, s), \quad (8)$$

where \arg is an operator such that $\arg f^b(g, p, s)$ returns s .

If no sellers in S_t^b submit bids for delivering g (or if $S_t^b = \emptyset$), then buyer b will have to choose a seller \hat{s} from the non-trustworthy sellers, provided that \hat{s} is not an untrustworthy seller:

$$\hat{s} = \arg \max_{s \in (S - (S_t^b \cup S_{ut}^b))} f^b(g, p, s). \quad (9)$$

In addition, with a small probability ρ , buyer b chooses to explore (rather than exploit) the marketplace by randomly selecting a seller $\hat{s} \in (S - S_{ut}^b)$. This gives buyer b an opportunity to discover new trustworthy sellers. Initially, the value of ρ should be set to 1, then decreased over time to some fixed minimum value determined by b .⁷

After paying seller \hat{s} and receiving good g , buyer b can examine the quality $q \in Q$ of good g , where Q is a finite set of real values representing product qualities. It then calculates the true value of good g using the *true product value function* $v^b : G \times P \times Q \mapsto \mathbb{R}$. For instance, if buyer b considers the quality of good g to be twice more important than its price, it may set $v^b(g, p, q) = 2q - p$.

The expected value function f^b is now incrementally learned in a reinforcement learning framework:

$$\Delta = v^b(g, p, q) - f^b(g, p, \hat{s}), \quad (10)$$

$$f^b(g, p, \hat{s}) \leftarrow f^b(g, p, \hat{s}) + \alpha \Delta, \quad (11)$$

where α is called the *learning rate* ($0 \leq \alpha \leq 1$). Similar to ρ , the learning rate α should initially be set to a starting value of 1 and then reduced over time to a fixed minimum value chosen depending on individual buyers.⁸

Thus, if $\Delta = v^b(g, p, q) - f^b(g, p, \hat{s}) \geq 0$ then $f^b(g, p, \hat{s})$ is updated with the same or a greater value than before. This means that seller \hat{s} has a good chance to be chosen by buyer b again if it continues offering good g at price p in the next auction. Conversely, if $\Delta < 0$ then $f^b(g, p, \hat{s})$ is updated with a smaller value than before. This implies that seller \hat{s} may not be selected by buyer b in the next auction if it continues selling good g at price p .

In addition to updating the expected value function, the trust rating $r^b(\hat{s})$ of seller \hat{s} also needs to be updated. Let $\vartheta^b(g) \in \mathbb{R}$ be the product value that buyer b demands for good g .⁹ In other words, the demanded product value $\vartheta^b(g)$ is buyer b 's threshold for the true product value $v^b(g, p, q)$. We use a trust updating scheme motivated by that proposed in [33] as follows:

⁷ The decrease in ρ reflects a decreasing need to explore the marketplace in order to build up the trustworthy set.

⁸ Reducing α over time is a standard method that allows reinforcement learning to converge and reflects a decreasing need to adjust previous f^b values, as learning becomes more extensive.

⁹ Note that it is necessary to tie the adjustment of trust to the buyer's demanded product value and not simply rely on its modelling of the expected value of the good, since: a buyer may in fact expect the value of the good to increase, from the time when that good was previously purchased, but if this value is still below what the buyer demands, the seller should not be labelled yet as trustworthy, for the buyer.

If $v^b(g, p, q) - \vartheta^b(g) \geq 0$, that is, if seller \hat{s} offers good g with value greater than or equal to the value demanded by buyer b , then its trust rating $r^b(\hat{s})$ is increased by

$$r^b(\hat{s}) \leftarrow \begin{cases} r^b(\hat{s}) + \mu(1 - r^b(\hat{s})) & \text{if } r^b(\hat{s}) \geq 0, \\ r^b(\hat{s}) + \mu(1 + r^b(\hat{s})) & \text{if } r^b(\hat{s}) < 0, \end{cases} \quad (12)$$

where μ is a positive factor called the *cooperation factor*¹⁰ ($\mu > 0$).

Otherwise, if $v^b(g, p, q) - \vartheta^b(g) < 0$, that is, if seller \hat{s} sells good g with value less than that demanded by buyer b , then its trust rating $r^b(\hat{s})$ is decreased by

$$r^b(\hat{s}) \leftarrow \begin{cases} r^b(\hat{s}) + \nu(1 - r^b(\hat{s})) & \text{if } r^b(\hat{s}) \geq 0, \\ r^b(\hat{s}) + \nu(1 + r^b(\hat{s})) & \text{if } r^b(\hat{s}) < 0, \end{cases} \quad (13)$$

where ν is a negative factor called the *non-cooperation factor*¹¹ ($\nu < 0$).

The set of trustworthy sellers to buyer b now needs to be updated based on the new trust rating $r^b(\hat{s})$, as in one of the following two cases:

- If $(\hat{s} \in S_t^b)$ and $(r^b(\hat{s}) < \Omega)$ then buyer b no longer considers \hat{s} as a trustworthy seller, i.e.,

$$S_t^b \leftarrow S_t^b - \{\hat{s}\}. \quad (14)$$

- If $(\hat{s} \notin S_t^b)$ and $(r^b(\hat{s}) \geq \Omega)$ then buyer b now considers \hat{s} as a trustworthy seller, i.e.,

$$S_t^b \leftarrow S_t^b \cup \{\hat{s}\}. \quad (15)$$

Finally, the set of untrustworthy sellers also needs to be updated:

1. If $(\hat{s} \notin S_{ut}^b)$ and $(r^b(\hat{s}) \leq \theta)$ then buyer b now considers \hat{s} as an untrustworthy seller, i.e.,

$$S_{ut}^b \leftarrow S_{ut}^b \cup \{\hat{s}\}. \quad (16)$$

Setting μ and ν

The co-operation and non-cooperation factors, μ and ν , are used to adjust the trust ratings of sellers once the buyer has examined the quality of the good purchased.

To protect itself from dishonest sellers, buyer b may require $|\nu| > |\mu|$ to implement the traditional assumption that trust should be difficult to build up, but easy to tear down. Moreover, buyer b may vary μ and ν as increasing functions of the true product value v^b to reflect the common idea that a transaction with higher value should be more appreciated than a lower one (i.e., the trust rating of a seller that offers higher true product value should be better increased and vice versa).

In particular, we propose the following equations for the calculation of μ and ν . If $v^b(g, p, q) - \vartheta^b(g) \geq 0$, we define the cooperation factor μ as

$$\mu = \begin{cases} \frac{v^b(g, p, q) - \vartheta^b(g)}{\Delta v^b} & \text{if } \frac{v^b(g, p, q) - \vartheta^b(g)}{\Delta v^b} > \mu_{\min}, \\ \mu_{\min} & \text{otherwise,} \end{cases} \quad (17)$$

¹⁰ Buyer b will consider seller \hat{s} as being *cooperative* if the good \hat{s} sells to b has value greater than or equal to that demanded by b .

¹¹ Buyer b will consider seller s as being *non-cooperative* if the good \hat{s} sells to b has value less than that demanded by b .

where $\Delta v^b = v_{\max}^b - v_{\min}^b$ with v_{\max}^b and v_{\min}^b being the maximum and minimum value of the true product value function $v^b(g, p, q)$.¹² We prevent μ from becoming zero when $v^b(g, p, q) = \vartheta^b(g)$ by using the value μ_{\min} .

However, if $v^b(g, p, q) - \vartheta^b(g) < 0$, we define the non-cooperation factor ν as

$$\nu = \lambda \left(\frac{v^b(g, p, q) - \vartheta^b(g)}{\Delta v^b} \right), \quad (18)$$

where λ is called the *penalty factor* ($\lambda > 1$) to implement the above mentioned idea that $|\nu|$ should be greater than $|\mu|$. If applying Eq. (13) using ν as defined in (18) results in the updated value $r^b(\hat{s}) \leq -1$, that is, seller \hat{s} is too non-cooperative, then buyer b will place \hat{s} in the untrustworthy set S_{ut}^b by setting $r^b(\hat{s}) = \theta$. Note that the use of an untrustworthy set and the requirement that the buyer select a non-untrustworthy seller when it does business in effect allows the buyer to avoid very undesirable sellers, after a loss. Sellers who have only disappointed the buyer in a modest way can still be selected in the future, but they will have to compete with other sellers who may be considerably more attractive to the buyer.

Table 1 lists the symbols used in this section, for quick reference.

Note that this model for making buying decisions in electronic marketplaces has been compared to ones that rely on recursive modelling of other agents and has been shown to be preferable in terms of user's satisfaction and computational time [29]. We have also demonstrated experimentally the value of this model in comparison to one that relies on reinforcement learning alone, without modelling trust (see [28] for details). For example, we can demonstrate that buyers using our proposed model can more effectively avoid doing business with a class of dishonest sellers who try to attract buyers with high quality goods and then cheat them with really low quality ones. Figure 6 below shows the buying activity in the marketplace when buyers use reinforcement learning alone (a) compared to the inclusion of our proposed trust modelling (b).

3.3 Selling algorithm

The sellers in our marketplace produce various versions of each good g , tailored to meet the needs of different buyers. In each auction, the seller must choose a price p greater than or equal to cost to sell good g to buyer b such that its expected profit is maximized. Sellers have the latitude of adjusting both the price and quality of the good being offered, depending on whether they have been unsuccessful in making sales or even if they have been successful for a number of auctions and want to attempt to earn a greater profit. We have also developed algorithms for selling agents in electronic marketplaces that employ reinforcement learning to maximize their expected profits. Note that our algorithms for buying agents to model the trustworthiness of sellers in an effort to make effective purchasing decisions do not rely on any particular algorithms being used by sellers; the adjustments are made on the basis of the satisfaction assessed, once a purchase has been made.

We include below a summary of the model for sellers' algorithms, to demonstrate one systematic way that sellers may in fact be operating within the marketplace, when their trustworthiness is modelled by the buyers.

Consider the scenario where a seller $s \in S$ has to decide on the price to sell some good g to a buyer b . Let B be the (finite) set of buyers in the marketplace, and let function $h^s : G \times P \times B \mapsto \mathbb{R}$ estimate the expected profit for seller s . Thus, the real number

¹² v_{\max}^b and v_{\min}^b are derived from the maximum and minimum elements of the finite sets P and Q . See Sect. 3.4 for more detail.

Table 1 Symbols used in the proposed buying algorithm

| Symbol | Meaning |
|------------------|---|
| G | Finite set of goods |
| S | Finite set of sellers |
| B | Finite set of buyers |
| P | Finite set of prices |
| Q | Finite set of qualities |
| $r^b(s)$ | Trust rating assigned to seller s by buyer b |
| Ω | Trustworthiness threshold ($0 < \Omega < 1$) |
| θ | Untrustworthiness threshold ($-1 < \theta < 0$) |
| S_t^b | Set of trustworthy sellers to buyer b |
| S_{ut}^b | Set of untrustworthy sellers to buyer b |
| $f^b(g, p, s)$ | Buyer b 's expected value of buying good g at price p from seller s |
| $v^b(g, p, q)$ | True value of good g to buyer b , with p and q being the price and quality of g , respectively |
| α | Learning rate ($0 \leq \alpha \leq 1$) |
| ρ | Exploration probability ($0 \leq \rho \leq 1$) |
| $\vartheta^b(g)$ | Product value demanded by buyer b for good g |
| μ | Cooperation factor ($\mu > 0$) |
| ν | Non-cooperation factor ($\nu < 0$) |
| Δv^b | $\Delta v^b = v_{\max}^b - v_{\min}^b$, where v_{\max}^b and v_{\min}^b are the maximum and minimum value of $v^b(g, p, q)$, respectively |
| λ | Penalty factor ($\lambda > 1$) |

$h^s(g, p, b)$ represents the expected profit for seller s if it sells good g at price p to buyer b . Let $c^s(g, b)$ be the cost of seller s to produce good g for buyer b . Note that seller s may produce various versions of good g , which are tailored to meet the needs of different buyers. Seller s will choose a price \hat{p} greater than or equal to cost $c^s(g, b)$ to sell good g to buyer b such that its expected profit is maximized:

$$\hat{p} = \arg \max_{\substack{p \in P \\ p \geq c^s(g, b)}} h^s(g, p, b), \quad (19)$$

where in this case \arg is an operator such that $\arg h^s(g, p, b)$ returns p .

The expected profit function h^s is learned incrementally using reinforcement learning:

$$h^s(g, p, b) \leftarrow h^s(g, p, b) + \alpha(\phi^s(g, p, b) - h^s(g, p, b)), \quad (20)$$

where $\phi^s(g, p, b)$ is the actual profit of seller s if it sells good g at price p to buyer b , and is defined as follows:

$$\phi^s(g, p, b) = \begin{cases} p - c^s(g, b) & \text{if seller } s \text{ wins the auction,} \\ 0 & \text{otherwise.} \end{cases} \quad (21)$$

Thus, if seller s does not win the auction then $(\phi^s(g, p, b) - h^s(g, p, b))$ is negative, and by (20), $h^s(g, p, b)$ is updated with a smaller value than before. This means that price \hat{p}

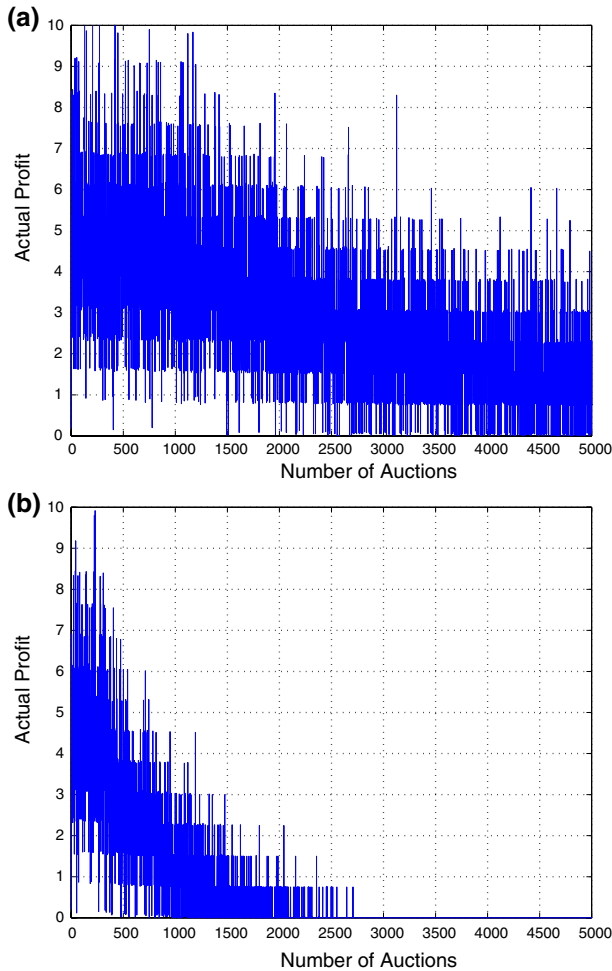


Fig. 6 Graphs of profit values made by the dishonest sellers from a buyer using reinforcement learning alone and not modelling sellers' trustworthiness **(a)**, and from a buyer following the proposed buying algorithm **(b)**

will probably not be chosen again to sell good g to buyer b in future auctions, but rather some lower price will. Conversely, if seller s wins the auction then price \hat{p} will probably be re-selected in future auctions.

If seller s succeeded in selling good g to buyer b once, but subsequently fails for a number of auctions, say for m consecutive auctions (where m is seller s specific constant), then it may not only be because s has set a too high price for good g , but probably also because the quality of g does not meet buyer b 's expectation. Thus, in addition to lowering the price via Eq. (20), seller s may optionally add more value (quality) to g by increasing its production cost:¹³

$$c^s(g, b) \leftarrow (1 + Inc)c^s(g, b), \quad (22)$$

where Inc is seller s specific constant called the *quality increasing factor*.

¹³ This supports the common assumption that it costs more to produce high quality goods.

In contrast, if seller s is successful in selling good g to buyer b for n consecutive auctions, it may optionally reduce the quality of good g , and thus try to further increase its future profit:

$$c^s(g, b) \leftarrow (1 - Dec)c^s(g, b), \quad (23)$$

where Dec is seller s specific constant called the *quality decreasing factor*.

3.4 Discussion on parameters

In this subsection we discuss the roles of several parameters used in our proposed algorithms and provide some general guidelines for calculating or choosing their values.

3.4.1 Trustworthiness threshold Ω

The trustworthiness threshold Ω ($0 < \Omega < 1$) is a buyer b 's specific constant, which buyer b uses to determine whether it should consider a seller s as a trustworthy seller. Consequently, the stricter (or more conservative) buyer b is, the higher value it would choose for Ω . In addition, the less trustworthy the market environment is, the higher the value b should set Ω to. As the range of Ω is $(0, 1)$, a buyer b of moderate strictness acting in a market of medium trustworthiness should probably choose Ω to be 0.50.

3.4.2 Untrustworthiness threshold θ

The untrustworthiness threshold θ ($-1 < \theta < 0$) is also a buyer's subjective and specific constant. A buyer b uses this constant to determine whether a seller s should be rated as an untrustworthy seller. How this threshold should be set depends on each individual buyer and its experience with the specific marketplace in consideration. A more tolerant buyer interested in exploring its marketplace and participating in a reasonably trustworthy market would likely choose a low value for θ . In contrast, a more conservative buyer operating in a less trustworthy marketplace would have the tendency to set a higher value for θ . Note that buyer b may receive unsatisfactory goods more often if it sets a too low value for θ , since in this case dishonest sellers may not be placed in the untrustworthy set as they should be. However, a too high value of θ set by buyer b may result in more and more sellers being transported into the untrustworthy set, with the extreme case where all sellers in the market may be severely treated as untrustworthy sellers. Of course in this case, buyer b would have to reset its buying algorithm with an adjusted (lower) value for θ and then execute the algorithm over again. Thus, a buyer should experience its specific market and consider its own characteristic (e.g., being conservative or exploratory) to adjust the value of θ accordingly, to reach a value that it is happy with. In addition, since the protection mechanism of the proposed algorithm suggests buyer b not to choose any sellers from the untrustworthy set, buyer b should set θ sufficiently low to avoid the situation where it may mistakenly or unfairly place a seller s in the untrustworthy set without having enough evidence of s 's being non-cooperative. A sufficiently low value of θ also provides those sellers, who are willing to improve the quality of their products, with an opportunity to make satisfactory offers to buyer b . Taking these reasons into account, we suggest that the values of θ should be chosen from the range $[-0.9, -0.7]$.

3.4.3 True product value function v^b

Each buyer b has its own way to evaluate the good it purchases using the true product value function v^b . Basically, v^b is a function of the price p that buyer b pays for the good, and also of the quality q that b examines the good after receiving it from the seller. Buyer b formulates v^b based on its opinion of the relative importance of these two factors. For example, if b considers quality to be more important than price, it may set $v^b = aq - p$ with $a > 1$.

As mentioned in Sect. 1, although the product quality q is represented by a single numerical value, it could be a multi-faceted concept. That is, buyer b may judge the quality of a product based on a combination of various factors such as physical product characteristics, whether the product is delivered on time, and whether the product is supported after sale, etc. As such, buyer b may calculate q as a weighted sum of these factors.

Since p and q are elements in the finite sets of prices and quality values respectively, there exist the maximum and minimum values of the true product value function v^b , i.e., v_{\max}^b and v_{\min}^b . If we continue with the above example, then $v_{\max}^b = aq_{\max} - p_{\min}$ and $v_{\min}^b = aq_{\min} - p_{\max}$. The existence of v_{\max}^b and v_{\min}^b justifies their use in Eqs. (17) and (18).

3.4.4 Demanded product value ϑ^b

After a transaction with a seller s , buyer b needs to decide if it should increase (or decrease) the trust rating of s , based on whether or not the true value v^b of the good offered by s meets buyer b 's demanded product value ϑ^b . In other words, the demanded product value ϑ^b serves as buyer b 's threshold for the true product value v^b . Let us give an example of how a buyer b may calculate ϑ^b . For a particular good g , buyer b should have in its mind the lowest quality q_{low}^b that it would like g at least to have, and the highest price p_{high}^b that it would agree to pay for that quality. Buyer b then can calculate the demanded product value ϑ^b based on q_{low}^b and p_{high}^b using the true product value function v^b . If we reuse the above example of v^b then buyer b will calculate its demanded product value as follows: $\vartheta^b = aq_{\text{low}}^b - p_{\text{high}}^b$. This choice of ϑ^b means that in order to satisfy buyer b 's demand, seller s will have to offer good g with quality greater than q_{low}^b if it intends to sell good g to buyer b at price greater than p_{high}^b .

3.4.5 Exploration probability ρ

The exploration probability ρ allows buyer b to discover new trustworthy sellers by, at probability ρ , considering choosing a seller from the set of non-untrustworthy sellers (rather than the smaller set of trustworthy sellers). That is, in addition to the trustworthy sellers, buyer b will also consider the sellers that are neither trustworthy nor untrustworthy. Since these are the sellers whose trustworthiness buyer b has not yet had enough information to decide on, some of them may have the potential of becoming trustworthy sellers. Certainly, buyer b needs to explore at probability $\rho = 1$ at the beginning, because at this point b does not have trustworthiness information of any seller in the market and its set of trustworthy sellers is still empty. However, as b is able to build up some members for its trustworthy set after a number of transactions, it should exploit the market more and explore it less. That means that b should gradually decrease ρ over time down to some fixed, minimum value ρ_{\min} . Of course, there is a trade off in choosing a value for ρ_{\min} . The higher ρ_{\min} , the more opportunities to explore but the fewer chances to exploit. In marketplaces where new sellers often enter the market, ρ_{\min} may be set to as high a value as 0.3 (i.e., buyer b will explore the market 30%

of the time). Nevertheless, in marketplaces where new sellers rarely join the market, ρ_{\min} should be set to low value (e.g., from 0.05 to 0.10).

3.4.6 Learning rate α

As suggested by its name, the learning rate α influences the rate buyer b learns its expected value function f^b , and the rate seller s learns its expected profit function h^s . Let us just look at the case of buyer b since the case of seller s is similar. Initially, buyer b just stores some initialized, incorrect values of f^b in its internal database. Thus, it needs to quickly update those initialized values with the actual ones by setting $\alpha = 1$ at the beginning. Over time, as b has roughly learned what the values of f^b should be, only small fractions of the current values of f^b need to be used for adjusting the previous values. That is, α should be reduced over time. In fact, reducing α over time will help the reinforcement learning method to converge [22, 26].

4 Theoretical results

In our proposed buying algorithm, reinforcement learning and trust modelling are integrated and used as two layers of learning to enhance buyers' performance. To properly study the protection capability of this mechanism, we need to address the question of whether or not the proposed algorithm can protect a buyer from being harmed infinitely by a dishonest seller, i.e., whether or not it can protect a buyer from purchasing low value goods repeatedly many times from a dishonest seller. In other words, we would like to investigate the worst case scenario of a proposed buyer caused by a dishonest seller.

We will present, in this section, the theoretical results to show that a buyer designed according to our proposed model can indeed avoid infinite harm from a dishonest seller. We will proceed as follows:

- First, to lay the necessary ground of terminology, we will define the concepts of *gain* and *loss* of a buyer b in a transaction with a seller s by evaluating if the true value of the good that buyer b purchases from s is greater than the product value that buyer b demands for the good. We will also define the concept of *being better off* of a buyer b after a series of transactions with a seller s by comparing whether the total gain of buyer b in these transactions is greater than its total loss.
- Then, we will consider a situation where a seller s , who has behaved non-cooperatively in its previous transaction with a buyer b resulting in a decrease in its trust rating, now tries to be cooperative in order to increase its trust rating back to the previous value. We will prove (in Proposition 1) that the gain of buyer b in the subsequent transaction is greater than its loss in the previous transaction (i.e., b is better off after these two transactions) if b properly sets its penalty factor λ .
- After that we will generalize the result of Proposition 1 in Corollary 1, which shows how to set the penalty factor λ such that Proposition 1 holds regardless of how badly seller s might have behaved non-cooperatively in its previous transaction, according to the situation mentioned above.
- Finally, using the result of Corollary 1 we will prove (in Proposition 2) that the maximum loss (or harm) that a dishonest seller s can ever cause to a buyer b , who follows our proposed buying algorithm and sets its penalty factor λ according to Corollary 1, is bounded above by a constant term. In addition, we will provide an example to demonstrate that this

constant is relatively small and acceptable in practice. This final result guarantees that a buyer following our proposed algorithm will not be harmed infinitely by a dishonest seller, and therefore will not incur infinite loss.

For coherency, we will make use of in this section the notations that are introduced in Sect. 3 to describe our proposed trust mechanism.

Definition 4.1 A buyer b is said to gain in a transaction with a seller s if it purchases from s some good g with value v^b greater than or equal to its demanded product value ϑ^b .

The difference $(v^b - \vartheta^b)$ is called the gain of b in the transaction.¹⁴ In this case, seller s is said to be cooperative with buyer b , value v^b is called seller s 's cooperative value, and the transaction is called the cooperative transaction.

Definition 4.2 A buyer b is said to lose in a transaction with a seller s if it purchases from s some good g with value v^b less than its demanded product value ϑ^b .

The difference $(\vartheta^b - v^b)$ is called the loss of b in the transaction. In this case, seller s is said to be non-cooperative with buyer b , value v^b is called seller s 's non-cooperative value, and the transaction is called the non-cooperative transaction.

Definition 4.3 A buyer b is said to be better off after a series of transactions with a seller s if its total gain is greater than its total loss in these transactions.

We now present and prove our first proposition.

Proposition 4.1 *Let s be a seller who is cooperative with a buyer b in order to get back to its previous trust rating after a non-cooperative transaction. Buyer b will be better off if it sets*

$$\lambda > \frac{1}{1 - \left(\frac{\vartheta^b - v_0^b}{\Delta v^b}\right)}, \quad (24)$$

where, as described in Sect. 3.2, λ is the penalty factor, ϑ^b is buyer b 's demanded product value, v_0^b is seller s 's non-cooperative value, and $\Delta v^b = v_{\max}^b - v_{\min}^b$ with v_{\max} and v_{\min} being the maximum and minimum product values, respectively.

Proof Consider the scenario where a proposed buyer b purchases from a seller s some good g with value v_0^b less than b 's demanded product value ϑ^b . Let $a = \vartheta^b - v_0^b$ ($a > 0$) be the loss of b in this non-cooperative transaction with s .

According to the proposed trust mechanism, b will decrease the trust rating of s using Eq. (13). Let $r_0^b(s)$ and $r_1^b(s)$ be the trust ratings of s before and after the transaction, respectively.

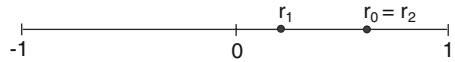
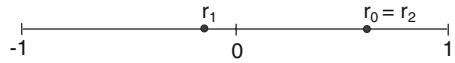
We will show that when s is cooperative in the next transaction by offering good g with value $v_1^b \geq \vartheta^b$ to get back to its previous trust rating $r_0^b(s)$, the gain $(v_1^b - \vartheta^b)$ of b in this transaction will be greater than its loss a in the previous transaction if b sets λ as in (24).

Case 1: $r_0 \geq 0$

We first consider the case where $r_0^b(s) \geq 0$. By Eqs. (13) and (18) we have

$$\begin{aligned} r_1^b(s) &= r_0^b(s) + v(1 - r_0^b(s)) \\ &= r_0^b(s) - \frac{\lambda a}{\Delta v^b} (1 - r_0^b(s)). \end{aligned} \quad (25)$$

¹⁴ No loss is also considered as a gain.

Fig. 7 The case of $r_1 \geq 0$ **Fig. 8** The case of $r_1 < 0$ 

To make our notations simpler without loss of generality, from now on we will drop the superscript b and variable s in the notations.

Case 1.1: $r_1 \geq 0$

Let's consider the case of $r_1 \geq 0$. The situation is illustrated in Fig. 7.

By offering value $v_1 > \vartheta$, the trust rating of s is increased to r_2 . According to Eqs. (12) and (17) we have

$$\begin{aligned} r_2 &= r_1 + \mu(1 - r_1) \\ &= r_1 + \frac{v_1 - \vartheta}{\Delta v}(1 - r_1) \\ &= r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) + \frac{v_1 - \vartheta}{\Delta v} \left(1 - \left(r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) \right) \right). \end{aligned} \quad (26)$$

In order for s to get back to its previous trustworthiness position r_0 we must have $r_2 = r_0$, that is

$$\begin{aligned} r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) + \frac{v_1 - \vartheta}{\Delta v} \left(1 - \left(r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) \right) \right) &= r_0, \\ \frac{v_1 - \vartheta}{\Delta v} \left(1 - \left(r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) \right) \right) &= \frac{\lambda a}{\Delta v}(1 - r_0), \\ (v_1 - \vartheta) \left(1 - \left(r_0 + \frac{\lambda a}{\Delta v}(1 - r_0) \right) \right) &= \lambda a(1 - r_0), \end{aligned} \quad (27)$$

$$v_1 - \vartheta = \frac{\lambda a(1 - r_0)}{(1 - r_0) + \frac{\lambda a}{\Delta v}(1 - r_0)},$$

$$v_1 - \vartheta = \frac{\lambda a}{1 + \frac{\lambda a}{\Delta v}},$$

$$v_1 - \vartheta > a \text{ iff}$$

$$\begin{aligned} \frac{\lambda a}{1 + \frac{\lambda a}{\Delta v}} &> a, \\ \lambda &> 1 + \frac{\lambda a}{\Delta v}, \\ \lambda - \frac{\lambda a}{\Delta v} &> 1, \\ \lambda &> \frac{1}{1 - \frac{a}{\Delta v}}. \end{aligned} \quad (28)$$

Case 1.2: $r_1 < 0$

For the case of $r_1 < 0$, the situation is shown in Fig. 8.

By Eqs. (12) and (17) we have

$$\begin{aligned} r_2 &= r_1 + \mu(1 + r_1) \\ &= r_1 + \frac{v_1 - \vartheta}{\Delta v}(1 + r_1) \\ &= r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) + \frac{v_1 - \vartheta}{\Delta v} \left(1 + \left(r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) \right) \right). \end{aligned} \quad (29)$$

Seller s gets back to its previous trustworthiness position r_0 iff $r_2 = r_0$. That is,

$$\begin{aligned} r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) + \frac{v_1 - \vartheta}{\Delta v} \left(1 + \left(r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) \right) \right) &= r_0, \\ \frac{v_1 - \vartheta}{\Delta v} \left(1 + \left(r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) \right) \right) &= \frac{\lambda a}{\Delta v}(1 - r_0), \\ (v_1 - \vartheta) \left(1 + \left(r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) \right) \right) &= \lambda a(1 - r_0), \\ v_1 - \vartheta &= \frac{\lambda a(1 - r_0)}{1 + r_0 - \frac{\lambda a}{\Delta v}(1 - r_0)}, \end{aligned} \quad (30)$$

$v_1 - \vartheta > a$ iff

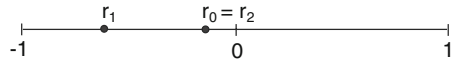
$$\begin{aligned} \frac{\lambda a(1 - r_0)}{1 + r_0 - \frac{\lambda a}{\Delta v}(1 - r_0)} &> a, \\ \lambda a(1 - r_0) &> a \left(1 + r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) \right), \\ \lambda(1 - r_0) + \frac{\lambda a}{\Delta v}(1 - r_0) &> 1 + r_0, \\ \lambda + \frac{\lambda a}{\Delta v} &> \frac{1 + r_0}{1 - r_0}, \\ \lambda &> \frac{\frac{1 + r_0}{1 - r_0}}{1 + \frac{a}{\Delta v}}. \end{aligned} \quad (31)$$

Since $r_1 < 0$ we have

$$\begin{aligned} r_0 - \frac{\lambda a}{\Delta v}(1 - r_0) &< 0, \\ r_0 - \frac{\lambda a}{\Delta v} + \frac{\lambda a}{\Delta v}r_0 &< 0, \\ r_0 \left(1 + \frac{\lambda a}{\Delta v} \right) &< \frac{\lambda a}{\Delta v}, \\ r_0 &< \frac{\frac{\lambda a}{\Delta v}}{1 + \frac{\lambda a}{\Delta v}}, \\ r_0 &< \frac{\lambda a}{\Delta v + \lambda a}. \end{aligned} \quad (32)$$

Using (32) we have

$$\frac{\frac{1 + r_0}{1 - r_0}}{1 + \frac{a}{\Delta v}} < \frac{\frac{1 + \frac{\lambda a}{\Delta v + \lambda a}}{1 - \frac{\lambda a}{\Delta v + \lambda a}}}{1 + \frac{a}{\Delta v}} = \frac{\frac{\Delta v + 2\lambda a}{\Delta v}}{1 + \frac{a}{\Delta v}} = \frac{\Delta v + 2\lambda a}{\Delta v + a}. \quad (33)$$

Fig. 9 The case of $r_0 < 0$ 

Thus, (31) holds if

$$\begin{aligned}
 \lambda &> \frac{\Delta v + 2\lambda a}{\Delta v + a}, \\
 \lambda(\Delta v + a) - 2\lambda a &> \Delta v, \\
 \lambda(\Delta v - a) &> \Delta v, \\
 \lambda &> \frac{\Delta v}{\Delta v - a}, \\
 \lambda &> \frac{1}{1 - \frac{a}{\Delta v}}.
 \end{aligned} \tag{34}$$

Combining (28) and (34), for both cases of r_1 we have

$$\lambda > \frac{1}{1 - \frac{a}{\Delta v}}. \tag{35}$$

Case 2: $r_0 < 0$

Let us now consider the case where $r_0 < 0$ (Fig. 9).

As mentioned before, because of seller s 's non-cooperative transaction, buyer b will decrease the trust rating of s down to r_1 ($-1 < r_1 < 0$). By (13) and (18) we have

$$\begin{aligned}
 r_1 &= r_0 + v(1 + r_0) \\
 &= r_0 - \frac{\lambda a}{\Delta v}(1 + r_0).
 \end{aligned} \tag{36}$$

By offering high value $v_1 > \vartheta$, the trust rating of s is increased to r_2 . Using (12) and (17) we have

$$\begin{aligned}
 r_2 &= r_1 + \mu(1 + r_1) \\
 &= r_1 + \frac{v_1 - \vartheta}{\Delta v}(1 + r_1) \\
 &= r_0 - \frac{\lambda a}{\Delta v}(1 + r_0) + \frac{v_1 - \vartheta}{\Delta v} \left(1 + r_0 - \frac{\lambda a}{\Delta v}(1 + r_0) \right).
 \end{aligned} \tag{37}$$

In order for s to get back to its previous trustworthiness position r_0 , we set $r_2 = r_0$, that is

$$\begin{aligned}
 r_0 - \frac{\lambda a}{\Delta v}(1 + r_0) + \frac{v_1 - \vartheta}{\Delta v} \left(1 + r_0 - \frac{\lambda a}{\Delta v}(1 + r_0) \right) &= r_0, \\
 \frac{v_1 - \vartheta}{\Delta v} \left(1 + r_0 - \frac{\lambda a}{\Delta v}(1 + r_0) \right) &= \frac{\lambda a}{\Delta v}(1 + r_0), \\
 v_1 - \vartheta &= \frac{\lambda a(1 + r_0)}{(1 + r_0) - \frac{\lambda a}{\Delta v}(1 + r_0)}, \\
 v_1 - \vartheta &= \frac{\lambda a}{1 - \frac{\lambda a}{\Delta v}},
 \end{aligned} \tag{38}$$

$v_1 - \vartheta > a$ iff

$$\frac{\lambda a}{1 - \frac{\lambda a}{\Delta v}} > a. \tag{39}$$

Since $r_1 \in (-1, 0)$, we have

$$\begin{aligned} r_0 - \frac{\lambda a}{\Delta v}(1 + r_0) &> -1, \\ \frac{\lambda a}{\Delta v}(1 + r_0) &< 1 + r_0, \\ \frac{\lambda a}{\Delta v} &< 1, \\ 1 - \frac{\lambda a}{\Delta v} &> 0. \end{aligned} \quad (40)$$

Using (40), (39) is equivalent to

$$\begin{aligned} \lambda a &> a \left(1 - \frac{\lambda a}{\Delta v} \right), \\ \lambda &> 1 - \frac{\lambda a}{\Delta v}, \\ \lambda \left(1 + \frac{a}{\Delta v} \right) &> 1, \\ \lambda &> \frac{1}{1 + \frac{a}{\Delta v}}. \end{aligned} \quad (41)$$

Because

$$\frac{1}{1 - \frac{a}{\Delta v}} > \frac{1}{1 + \frac{a}{\Delta v}} \quad (42)$$

(41) will certainly hold if (35) holds. In other words, for both cases $r_0 \geq 0$ and $r_0 < 0$, buyer b will be better off in the transactions with seller s if b sets the penalty factor

$$\lambda > \frac{1}{1 - \frac{a}{\Delta v}}. \quad (43)$$

The proof is therefore complete. \square

Derived from Proposition 4.1, the following corollary shows how to set λ so that Proposition 4.1 holds regardless of seller s 's non-cooperative values.

Corollary 4.1 *Let s be a seller who is cooperative with a buyer b in order to get back to its previous trust rating after a non-cooperative transaction. For all non-cooperative values v of seller s , buyer b will be better off if it sets*

$$\lambda > \frac{1}{1 - \left(\frac{\vartheta - v_{\min}}{\Delta v} \right)} \quad (44)$$

where, as described in Sect. 3.2, λ is the penalty factor, ϑ is buyer b 's demanded product value, and $\Delta v = v_{\max} - v_{\min}$ with v_{\max} and v_{\min} being the maximum and minimum product values, respectively.

Proof For all non-cooperative values v , we have $\vartheta - v_{\min} \geq \vartheta - v$, and therefore

$$\frac{1}{1 - \left(\frac{\vartheta - v_{\min}}{\Delta v} \right)} \geq \frac{1}{1 - \left(\frac{\vartheta - v}{\Delta v} \right)}. \quad (45)$$

It follows that for all non-cooperative values v , Proposition 4.1 holds if b sets λ as in (44), and the corollary follows directly. \square

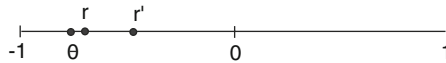


Fig. 10 Buyer b 's loss will be reduced if seller s decides to cooperate even in order to be non-cooperative in following transactions

In practice, we find that a buyer b is sufficiently satisfied when it sets

$$\lambda \approx \frac{1}{1 - \left(\frac{\vartheta - v_{\min}}{\Delta v}\right)}. \quad (46)$$

We now define the concept of a *cautious* buyer b as one who sets its penalty factor λ as in (44). We then present and prove our final result in Proposition 4.2.

Definition 4.4 A buyer b is said to be cautious if it sets

$$\lambda > \frac{1}{1 - \left(\frac{\vartheta - v_{\min}}{\Delta v}\right)}, \quad (47)$$

where, as described in Sect. 3.2, λ is the penalty factor, ϑ is buyer b 's demanded product value, and $\Delta v = v_{\max} - v_{\min}$ with v_{\max} and v_{\min} being the maximum and minimum product values, respectively.

Proposition 4.2 *The maximum loss of a cautious buyer b is bounded above by*

$$\frac{|\theta|(v_{\max} - \vartheta)}{1 + \theta} + (\vartheta - v_{\min}) \quad (48)$$

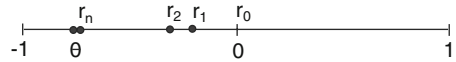
where, as described in Sect. 3.2, θ is buyer b 's untrustworthiness threshold, ϑ is buyer b 's demanded product value, and v_{\max} is the maximum product value.

Proof First, we notice that the loss of buyer b will be reduced when a seller s decides to be cooperative in a transaction, even in order to further behave non-cooperatively in the following transactions. In particular, let r be the current trust rating of s . Suppose somehow s realizes that its trust rating r is close to b 's untrustworthiness threshold θ and therefore decides to alternate between cooperative and non-cooperative actions, in order to avoid being placed in the untrustworthy set. Let r' be the trust rating of s resulting from its cooperative behaviour. The situation is illustrated in Fig. 10.

Let d be the gain of b in that cooperative transaction. Let l be the loss of b when s alternatively does not cooperate resulting in its trust rating being moved back to r . Since b is cautious, by Corollary 4.1, $d > l$. Hence, the loss so far of b is reduced by $(d - l)$ after these two transactions with s . It can therefore be inferred that the more s continues to alternate between cooperative and non-cooperative actions, the more the loss of b will be reduced.

It is now clear that the greatest loss of a buyer b is caused by a seller s who is continuously non-cooperative until its trust rating is arbitrarily close to the untrustworthiness threshold θ , when it performs the final non-cooperative transaction with minimum value v_{\min} , resulting in buyer b 's loss $(\vartheta - v_{\min})$, the greatest loss that b may incur in a transaction.

Let s be such a seller with initial trust rating $r_0 = 0$. Let v_0, v_1, \dots, v_{n-1} be the non-cooperative values that s continuously offers until its trust rating is arbitrarily close to θ . Let r_1, r_2, \dots, r_n be the trust ratings of s resulting from the offerings of v_0, v_1, \dots, v_{n-1} , respectively. So, r_n approaches θ from the right and $r_n \approx \theta$ (Fig. 11).

Fig. 11 Seller s is consecutively non-cooperative

Let L be the total loss of be caused by this series of non-cooperative transactions. Then,

$$L = \sum_{i=0}^{n-1} (\vartheta - v_i). \quad (49)$$

By (13) and (18) we have

$$\begin{aligned} r_1 &= 0 + \lambda \left(\frac{v_0 - \vartheta}{\Delta v} \right) (1 - 0) = -\lambda \left(\frac{\vartheta - v_0}{\Delta v} \right) (1 + 0), \\ r_1 &< -\lambda \left(\frac{\vartheta - v_0}{\Delta v} \right) (1 + \theta), \\ r_2 &= r_1 + \lambda \left(\frac{v_1 - \vartheta}{\Delta v} \right) (1 + r_1) = r_1 - \lambda \left(\frac{\vartheta - v_1}{\Delta v} \right) (1 + r_1), \\ r_2 &< -\lambda \left(\frac{\vartheta - v_0}{\Delta v} \right) (1 + \theta) - \lambda \left(\frac{\vartheta - v_1}{\Delta v} \right) (1 + \theta), \\ r_2 &< -\lambda \left(\frac{1 + \theta}{\Delta v} \right) [(\vartheta - v_0) + (\vartheta - v_1)], \\ r_3 &= r_2 + \lambda \left(\frac{v_2 - \vartheta}{\Delta v} \right) (1 + r_2) = r_2 - \lambda \left(\frac{\vartheta - v_2}{\Delta v} \right) (1 + r_2), \\ r_3 &< -\lambda \left(\frac{1 + \theta}{\Delta v} \right) [(\vartheta - v_0) + (\vartheta - v_1)] - \lambda \left(\frac{\vartheta - v_2}{\Delta v} \right) (1 + \theta), \\ r_3 &< -\lambda \left(\frac{1 + \theta}{\Delta v} \right) [(\vartheta - v_0) + (\vartheta - v_1) + (\vartheta - v_2)], \\ &\vdots \\ r_n &< -\lambda \left(\frac{1 + \theta}{\Delta v} \right) [(\vartheta - v_0) + (\vartheta - v_1) + (\vartheta - v_2) + \dots + (\vartheta - v_{n-1})], \\ r_n &< -\lambda \left(\frac{1 + \theta}{\Delta v} \right) L \quad (\text{using (49)}). \end{aligned} \quad (50)$$

Since r_n approaches θ from the right and $r_n \approx \theta$, (50) gives

$$\begin{aligned} -\lambda \left(\frac{1 + \theta}{\Delta v} \right) L &> \theta, \\ \lambda \left(\frac{1 + \theta}{\Delta v} \right) L &< -\theta = |\theta|, \\ L &< \frac{|\theta| \Delta v}{\lambda(1 + \theta)}. \end{aligned} \quad (51)$$

Since b is a cautious buyer, by Definition 4.4 we have

$$\lambda > \frac{1}{1 - \left(\frac{\vartheta - v_{\min}}{\Delta v} \right)}. \quad (52)$$

We notice that

$$\begin{aligned}
 \frac{1}{1 - (\frac{\vartheta - v_{\min}}{\Delta v})} &= \frac{1}{\frac{\Delta v - \vartheta + v_{\min}}{\Delta v}} \\
 &= \frac{\Delta v}{v_{\max} - v_{\min} - \vartheta + v_{\min}} \\
 &= \frac{\Delta v}{v_{\max} - \vartheta}.
 \end{aligned} \tag{53}$$

So, (52) and (53) give us

$$\lambda > \frac{\Delta v}{v_{\max} - \vartheta}. \tag{54}$$

Thus, (51) becomes

$$\begin{aligned}
 L &< \frac{|\theta| \Delta v}{\left(\frac{\Delta v}{v_{\max} - \vartheta} \right) (1 + \theta)}, \\
 L &< \frac{|\theta| \Delta v (v_{\max} - \vartheta)}{\Delta v (1 + \theta)}, \\
 L &< \frac{|\theta| (v_{\max} - \vartheta)}{1 + \theta}.
 \end{aligned} \tag{55}$$

Adding the final loss of $(\vartheta - v_{\min})$, the maximum loss of b is bounded above by

$$\frac{|\theta| (v_{\max} - \vartheta)}{1 + \theta} + (\vartheta - v_{\min}). \tag{56}$$

The proof is therefore complete. \square

Thus, letting b be a cautious buyer as defined in Definition 4.4, we have shown that the worst case scenario of b in dealing with a non-cooperative seller s is where s continuously performs non-cooperative transactions until its trust rating $r^b(s)$ is arbitrarily close to the untrustworthiness threshold θ , when it performs the final non-cooperative transaction with minimum value v_{\min} resulting in buyer b 's loss of $(\vartheta - v_{\min})$. The maximum loss of b in this worst case scenario is, however, bounded above by the constant term shown in (56).

As a side note, the time it takes a non-cooperative seller s to cause the maximum loss to a buyer b can be computed as the time of one non-cooperative transaction times the number of non-cooperative transactions. In general, the number of non-cooperative transactions performed by a seller s depends on the non-cooperative behaviour of that specific seller s . That is, it depends on the non-cooperative values that seller s decides to offer to buyer b . Recall that by (13) and (18), the lower the non-cooperative values are, the faster s is placed in the untrustworthy set, i.e., the fewer the number of non-cooperative transactions are performed. The time of one transaction is composed of communication time and computation time. The communication time accounts for the time needed for network communication between the buyer and sellers in the market (e.g., the buyer broadcasting its request to sellers, the sellers responding with their bids, the buyer contacting its chosen seller, etc.). Clearly, the communication time depends on the properties of the particular network underlying the marketplace in consideration. The computation time accounts for the time needed by the buyer to compute the seller that it chooses to purchase the good from according to its buying algorithm, and the time required by the seller to compute its offer to the buyer according to its selling algorithm. For example, the time complexity of our proposed buying algorithm is

$O(|S|)$, since a buyer b only needs to examine the set of sellers S to compute a suitable seller; whereas the time complexity of our proposed selling algorithm is $O(|P|)$, where $|P|$ is the cardinality of the set of prices, since a seller s will search this set for a price that maximizes its expected profit. In addition, after completing one transaction, seller s may have to wait for an arbitrary period of time before it is chosen again by buyer b for the next transaction. We can reasonably consider this waiting time as part of seller s 's one-transaction time. Thus, if we let n be the number of non-cooperative transactions, T_{commu} be the communication time, T_{compu} be the computation time, and T_{waiting} be the waiting time of seller s between two transactions, then the time it takes a non-cooperative s to cause the maximum loss to a buyer b is $n \times (T_{\text{commu}} + T_{\text{compu}} + T_{\text{waiting}})$. Note that among these terms, only T_{compu} is known based on the buying and selling algorithms followed by buyer b and seller s respectively, while other terms depend on specific settings or uncertain factors and are therefore unknown in general. Consequently, this paper focuses on presenting a more important and significant result regarding the maximum loss that a buyer may incur and only discusses the time issue as a side note here.

An example

Let us look at a concrete example to see how large the upper bound given in (56) can be.

Consider a cautious buyer b whose demanded product value is chosen as the midpoint between the maximum and minimum product values:

$$\vartheta = \frac{v_{\max} + v_{\min}}{2}. \quad (57)$$

Suppose that buyer b decides to set the untrustworthiness threshold θ to be -0.8 ($\theta = -0.8$). As discussed in Sect. 3.4, if b chooses θ to be too low, dishonest sellers may not be placed in the untrustworthy set as they should be, resulting in b 's frequently purchasing unsatisfactory value goods. In contrast, if buyer b sets θ to be too high, more sellers may be placed in the set of untrustworthy sellers, with the extreme case where all sellers in the market may be treated as untrustworthy sellers. Moreover, due to the fact that b will not re-select untrustworthy sellers to do business with (according to the proposed algorithm), θ should be set sufficiently low in order for b to avoid the situation where it may carelessly place a seller s in the untrustworthy set without having enough evidence of s 's being non-cooperative. This setting also gives those sellers who are willing to improve the quality of their products an opportunity to make good offers to b . Considering these reasons, we suggest that θ should take values in the range $[-0.9, -0.7]$.

Using Proposition 4.2, the maximum loss that buyer b may incur in dealing with a seller s is bounded by

$$\frac{|-0.8|(v_{\max} - \vartheta)}{1 - 0.8} + (\vartheta - v_{\min}) = 4(v_{\max} - \vartheta) + (\vartheta - v_{\min}). \quad (58)$$

Since

$$\begin{aligned} v_{\max} - \vartheta &= v_{\max} - \frac{v_{\max} + v_{\min}}{2} \\ &= \frac{v_{\max} - v_{\min}}{2} \end{aligned} \quad (59)$$

and

$$\begin{aligned} \vartheta - v_{\min} &= \frac{v_{\max} + v_{\min}}{2} - v_{\min} \\ &= \frac{v_{\max} - v_{\min}}{2} \end{aligned} \quad (60)$$

we have

$$v_{\max} - \vartheta = \vartheta - v_{\min}. \quad (61)$$

Thus, the upper bound in (58) becomes

$$5(\vartheta - v_{\min}), \quad (62)$$

which is five times the loss that buyer b may incur in one transaction. In other words, the worst case scenario of buyer b dealing with any seller s is no more than five disappointing transactions and this is for a malicious seller who wants to cheat as many times as possible. Note that if buyer b is more cautious and set the untrustworthiness threshold θ to be -0.7 (instead of -0.8 as in this example), then the maximum loss that buyer b may incur when dealing with a dishonest seller s is bounded above by only $3.3(\vartheta - v_{\min})$ (instead of $5(\vartheta - v_{\min})$). Thus, a software buying agent following our proposed algorithm will not perform any worse than what we do in our everyday lives. By common sense and our self-protection nature, we usually do not return to a store after being disappointed by that store a number of times.

5 Related work

The model presented in this paper is intended to be used to evaluate the trustworthiness of selling agents, on the basis of direct experience. The value of this particular approach in providing buying agents with the protection from infinite harm from selling agents has been demonstrated, through theoretical analysis of the model.

In this section, we discuss, in the context of related work, the value of having buyers make purchasing decisions based on their own individual models of the sellers in the marketplace, and highlight the merit of this kind of approach to trust modeling.

5.1 Further comparison with competing models

The specific design of our model also contrasts favourably with other approaches in the literature.

First of all, we are different from approaches for designing electronic marketplaces via recursive modelling, without an explicit model of trust, such as that of Vidal and Durfee [30]. In this approach, buyers learn to avoid sellers who have disappointed them in the past, by modelling other agents. Although they acknowledge the importance of dealing with dishonest sellers in the marketplace, there is no explicit strategy to identify these particular sellers. Moreover, those agents that keep deep recursive models of others suffer from the associated computational costs of maintaining these models. In addition, in Vidal and Durfee's work, the selling agents only alter product prices and not product quality, so the marketplace being modelled is not as flexible as ours.

Our particular model for trust is also distinct from those developed by other researchers. In contrast to [9, 23], by introducing the trustworthiness and untrustworthiness thresholds (Ω and θ), our trust mechanism allows a buyer to partition the set of all sellers in the market into three disjoint subsets, namely the trustworthy sellers, the untrustworthy sellers, and the neither trustworthy nor untrustworthy sellers (i.e., those sellers that the buyer has not yet had enough information to decide on their trustworthiness). As a result, the buyer can quickly select a trustworthy seller while avoiding the untrustworthy ones, whenever a purchase decision needs to be made. Also, it can have easy access to the sellers whose trustworthiness it has not yet decided on in order to discover new trustworthy sellers. We believe that this design is

especially suitable for electronic marketplaces where the rate of transactions between trading agents is usually high.

Our trust mechanism is motivated by [33]. As in [33], our updating scheme encourages sellers to offer high value goods and discourages them from offering low value ones, by making use of the cooperation and non-cooperation factors (μ and ν), respectively. However, in contrast with [33], we do not keep these factors fixed, but take one step further to vary them as increasing functions of the true product value (see Eqs. 17 and 18) in order to implement the common idea that a transaction with higher value should be more appreciated than a lower one and vice versa. In other words, the trust rating of a seller who offers higher value goods should be better increased, and the trust rating of a seller who greatly disappoints should be more seriously decreased. We also implement the traditional assumption that trust should be difficult to build up but easy to tear down by introducing the penalty factor $\lambda > 1$, by which we penalize a non-cooperative seller λ times greater than we would award a cooperative one.

The approach of Bakos and Dellarocas [1] to modelling trust is similar to ours, in that buying agents may rate sellers based on the quality of the good received after purchase. Still, there are important differences between Bakos and Dellarocas' model and ours: (i) Their buyers' valuations of goods are based on quality alone whereas our buyers' evaluations of goods are based on price, quality, and the relative importance between price and quality as viewed personally by each buyer. (ii) Their buyers rate sellers using a binary decision. This does not distinguish between sellers offering very high quality goods and those offering barely high quality goods. Similarly, sellers offering extremely low quality goods and those offering just low quality goods receive the same rating. Our buyers, in contrast, calculate the ratings of sellers using Eqs. (12), (13), (17) and (18), which implement the common idea that higher value transactions should be more appreciated than lower ones.

As for the examination of untrustworthy agents in a marketplace and the need to protect buyers from harm, other researchers have certainly discussed the relative advantages of modelling trust in order to improve the decision making of buying agents. Breban and Vassileva [3], working with the trust evolution model of Jonker and Treuer [13], allow agents to progressively learn which agents are the best ones to join their coalition. Sabater and Sierra [23] present a rich model of reputation, allowing for the integration of various factors being modelled, towards a single reputation rating, to demonstrate that the evaluations can provide significant detail, towards purchasing decisions. Yu and Singh [33] believe it is useful to take advantage of ratings offered by other agents in a designated neighbourhood, which may improve the evaluation of selling agents by potential buyers. In addition, other researchers have clarified the importance of modelling reputation, saying that reputation can be used to "impose appropriate constraints" [12] or "the formalization of reputation is an excellent place to begin in formalizing trust" [4]. Yet, none of these researchers has specifically proved how the setting of trust values in their models can provide protection from harm, one of the main results of our work.

Moreover, some researchers have investigated how agents can develop strategies to deliberately deceive other agents in the society [8]. This research reinforces the need to develop strategies to also mitigate against possible harm from these deceptive agents - the theme of our research.

Also related to our research are a number of other models that assist buyers in making purchase decisions from a product perspective. Wong and Lam [32] propose a learning framework to automatically extract and summarize product features from multiple online auction sites for buyers. Leung, Chan and Chung [20] present a collaborative filtering approach based on fuzzy association rules and multiple-level similarity to recommend potential products to

buyers. In contrast, our proposed model incorporates reinforcement learning and trust modelling to recommend the best possible sellers for buyers and provides the buyers with a mechanism that can protect them from being harmed infinitely by dishonest sellers in the market.

5.2 Other approaches for protecting from harm

It is also important to acknowledge competing approaches towards the protection of buying agents from dishonest or unreliable behaviour of selling agents. Various researchers have begun to study the provision of security guarantees within electronic marketplaces. For example, Kerr and Cohen [18] develop a security framework in an environment where selling agents operate with units of trust that are risked when entering into transactions with buying agents. In this system, when buying agents are disappointed, their reporting causes selling agents to lose trust units and thus eventually be unable to engage in transactions. In [17], it is shown that under certain conditions this system causes rational selling agents to simply behave honestly, because this is the profit maximizing strategy.

Barber and Kim [2] have also examined the extent to which trust and reputation systems provide a level of security for buying agents in electronic marketplaces. In their approach, unreliable agents can be isolated over time, in a process of belief revision. Deceptive agents can be detected and avoided. In this sense, trust and reputation modeling serve to enforce a kind of “soft security” within the system. This work reinforces the value of trust modeling to protect agents from harm, but it does not explicitly evaluate the degree of harm that can be inflicted by malicious agents within their particular modeling of trust and reputation.

Various researchers have investigated the use of social networks of agents to share reputation ratings of selling agents (e.g., [23,33]). Of particular importance is the work of Jurca and Faltings [14,15], since it examines more closely the promotion of honest reporting amongst agents, with the use of incentives. As long as agents report the behaviour of selling agents honestly, a buying agent will operate with reliable information about selling agents, in order to avoid the most untrustworthy ones. This work therefore also provides a level of security for the buying agents. Experiments in this paper show the advantages of being honest and the effectiveness of the incentive mechanism. While the research does not explicitly measure the degree of harm that buying agents can endure from dishonest selling agents, it should still be possible for these buying agents to avoid untrustworthy selling agents, provided that others in the community have had experience with these agents and their behaviour has been consistent, over time. We discuss further the value of employing a social network for reputation reporting, as future work, in Sect. 6.

The approach of using incentive mechanisms in electronic marketplaces in order to promote honesty is in fact a valuable direction for providing protection from harm. If selling agents can be motivated to act honestly, then buying agents can avoid untrustworthy sellers. The challenge of ensuring honesty amongst both buying agents and selling agents in electronic marketplaces is currently being examined by Zhang [34,35]. In this framework, selling agents model the trustworthiness of buying agents by virtue of the extent to which they are respected in their social networks and deliver greater rewards to more trustworthy buying agents. This promotes buyer honesty. Buying agents use social networks to detect unreliable selling agents as well. Experiments show that selling agents that offer rewards to buyers in this framework will enjoy greater profits, as their reputation is promoted within the social network, over time. This research is currently being extended to provide theoretical proofs of the robustness of the incentive mechanism.

Although our modelling of trust focuses on the individual dimension of trustworthiness, we examine for future research a possible market environment where the social dimension of reputation is needed to allow an agent to compute reputation ratings based not only on its own but also on others' experience. This is discussed further in Sect. 6.

6 Future directions

6.1 Application to mobile e-commerce

Since more and more people nowadays use mobile devices such as personal digital assistants or mobile phones, mobilizing online trading is a research direction that has been gaining great interest from researchers in both academia and industry. Chiang and Liao [6] proposed an agent-based framework, implemented using JADE-LEAP¹⁵ tools, to assist mobile shoppers in making purchase decisions. Keegan and O'Hare introduced Easishop [16], a mobile commerce cross merchant product comparison system, to support mobile shoppers. Rosendahl et al. [21] suggested an architecture that provides mobile users with different options to access a mobile home automation service. Although most existing mobile e-commerce systems introduce mobile agent technology to somewhat overcome the limitations of current mobile devices, they are usually stand-alone applications without an interface with other systems and often require costly long-standing connection of mobile devices with the Internet. In contrast, we have proposed a feasible mobile agent-based architecture for e-commerce that allows users of mobile devices to conduct business anytime and anywhere, as well as enables inter-operability among multiple mobile e-commerce systems and significantly reduces the expensive connection cost [31]. Nevertheless, the mobile agents proposed in this architecture are still rather simple and do not have any learning capabilities. As the next research step, we would like to incorporate the learning agents described in this work (together with the mechanism to protect from infinite harm) into the proposed mobile architecture.

Indeed, the application area of mobile devices is one where direct experience trust models are needed, because bandwidth and time constraint issues make it prohibitive to be consulting other agents in order to evaluate trustworthiness. This would therefore be a rich environment in which to explore the merit of our particular direct experience trust model. In addition, using our model in this setting would serve to provide a more valuable framework for users engaging in e-commerce using mobile devices, offering more protection against harm from other agents.

6.2 Marketplaces involving social networking

In Sect. 5, we discussed the merit of developing a model that relies on direct experience in order to judge the trustworthiness of selling agents. We mentioned that other researchers have proposed models that operate in marketplaces where reputation ratings of selling agents are shared between buying agents. In order to accommodate marketplaces of this kind, it would be valuable for us to explore as well an extension of our research that can make use of trust ratings of sellers being provided to buying agents by other buying agents in the marketplace (known as advisors). The ideal would be to make use of the direct experience model described in this paper and to then integrate this with a method for reasoning about trust ratings provided by the social network.

¹⁵ Java Agent Development Environment—Lightweight Extensible Agent Platform.

Researchers such as Huynh et al. [11], Fullam and Barber [10], Sabater and Sierra [23] and Zhang [34] have all developed frameworks that allow for direct experience ratings to be incorporated into the reasoning about the reputation of sellers. What would be especially interesting to explore is how to measure the extent that harm can be inflicted by dishonest selling agents, in cases where direct experience is limited and ratings provided by the social network would play a larger role. One challenge that arises in this scenario is coping with possible collusion between advisors and selling agents (thus misrepresenting the trust rating of certain selling agents). Buying agents need to effectively model both the advisors and the selling agents, in order to avoid disappointment from untrustworthy sellers. Indeed, various researchers have developed methods for effectively addressing the issue of untrustworthy advisors [27,36].

One interesting subtopic to investigate is how to make use of a classification into sets of untrustworthy, trustworthy and neither trustworthy nor untrustworthy, when modeling the advisors. If this is done effectively with thresholds, in a method similar to the one proposed in this paper for sellers, then buying agents should enjoy a level of protection from harm from advisors, and by implication, from selling agents as well. We have taken some first steps to map out this model in [19]. Further research is necessary in order to determine how to best combine the modeling of advisors with the modeling of sellers.

6.3 Integrating the factor of time

One specific topic to examine more closely, as we extend our study of trust mechanisms, is to determine the value of modelling the time of transaction with an agent. Jonker and Treur [13] propose a formal trust model in which their trust update function makes use of an inflation parameter to implement the fact that recent experiences are more important than older ones. Our current thought is that in market environments, it is the value of a transaction rather than its recency that should matter. That is, in our view, an agent a_1 should not forget the fact that an agent a_2 was not cooperative with it in a one-million-dollar transaction in the past, and consider agent a_2 as being trustworthy just because a_2 has recently been cooperative with a_1 in a few one-dollar transactions. However, we would like to study the pros and cons of this model more carefully in future work.

6.4 Further analysis of sellers' behaviours

For future work, it would also be worthwhile to explore some extensions to the modelling of trust, for effective buying decisions. For example, since trust ratings are adjusted after each purchase of a good, it would be possible for the buyers to maintain and analyse further the statistics gathered about specific selling instances. Some possible uses of these analyses would be: (i) To deepen the model of the trustworthiness of a particular seller. For example, the buyer may determine that this seller is untrustworthy with respect to only certain kinds of goods. (ii) To extrapolate a model of the trustworthiness of all sellers with respect to a particular type of good. Perhaps certain goods are inherently more prone to dishonest representation. (iii) To isolate classes of sellers who exhibit similar behaviour in the marketplace.

6.5 Possible enhancement of the selling algorithm

While this paper focuses on the merit of our proposed buying algorithm and the extent to which it protects from harmful sellers, we believe that there is also benefit in revising our proposed algorithm for sellers, to make it more efficient, towards a more valuable overall framework for designing electronic marketplaces.

In the proposed selling algorithm (Sect. 3.3), a seller s keeps track of buyers' behaviours by using variable b in its expected profit function $h^s(g, p, b)$. Since multiple buyers may have different opinions in evaluating the goods they purchase (by adopting different true product value functions v^b), seller s obviously needs to track the behaviours of buyers. If s does not do so, then a failure in selling good g at price p to some buyer b will falsely keep it from making the same offer to other buyers, including those buyers who value g in a different way than b does, and therefore may be happy to purchase g at price p from s . However, this modelling of buyers' behaviours may not be necessary when buyers in the market have similar evaluation functions for the goods they purchase.

The above discussion suggests that it is possible to explore an additional version of the proposed selling algorithm in which a seller divides buyers into groups that use similar true product value functions and keeps track of groups of buyers' behaviours, instead of individual buyers' behaviours. Suppose, a seller s divides buyers into n groups, namely $\Gamma_1, \Gamma_2, \dots, \Gamma_n$; then this additional version can be obtained from the proposed selling algorithm described in Sect. 3.3 by substituting b with Γ_i in the expected profit function $h^s(g, p, b)$, the production cost function $c^s(g, b)$, and the actual profit function $\phi^s(g, p, b)$, where Γ_i is the group that buyer b belongs to.

We believe that the approach for sellers to model groups of buyers' behaviours should have at least the following advantages:

- Since the number of buyer groups may be considerably less than the number of all buyers in the market, this approach should reduce the size of a seller's internal database and accordingly the search space, resulting in improved performance when the seller searches for an optimal price to sell some good.
- More importantly, this approach allows a seller to significantly reduce the number of customized versions of a good to be maintained. That is, instead of producing so many versions of a good tailored to meet specific demands of all individual buyers in the market, the seller now only needs to produce a much smaller number of versions of the good, which are customized for the different groups of buyers.

However, this approach also presents several issues to be addressed:

- How to measure the similarity between buyers' true product value functions (especially when the buyers may not explicitly let sellers know their functions) in order to classify buyers into groups.
- How to update the models of groups of buyers appropriately over time.
- How to detect if any buyer has changed the way it evaluates the goods it purchases (i.e., changed its true product value function) and therefore should be removed from its current group and placed in another group, resulting in both groups being updated.
- Finally, a formal analysis is needed to justify the approach, considering its advantages and the complexity in addressing the above issues.

7 Conclusion

We have presented algorithms for buying agents in electronic marketplaces where sellers may alter the quality of their goods and where buyers will be disappointed if the products received fail to meet a certain standard of quality. Since dishonest sellers are allowed to exist in the marketplace (for example, offering high quality goods and then later offering very low quality goods), it is important for buyers to develop strategies for modelling their trust in sellers, in such a way that sellers who offend the most are penalized the most, while still

allowing sellers who have delivered disappointing goods to redeem themselves in the future and be selected for business. This is achieved in our model by the setting of the trustworthiness and untrustworthiness thresholds, the use of the trustworthy and untrustworthy sets, and the introduction of the co-operation and non-cooperation factors, to adjust the trust ratings of sellers. In addition, we have shown that with the use of a penalty factor, set in an appropriate way, it is possible for buyers to enjoy a certain level of protection against dishonest sellers. Our proposal for buying agents to protect themselves from infinite harm by selling agents is therefore to have them track the trustworthiness of sellers and to make adjustments, but to do so according to the formulae of the model presented in this paper.

Our particular model for trust is a mechanism for allowing agents to model other agents in the community, and by so doing, to determine how other users may be adversely affecting their own user. The agents who choose to model trust can then, in turn, engender the trust of their users, by providing better satisfaction to these users, with their buying decisions. In addition, our framework for modelling trust is designed to be user specific, sensitive to the true value function and demanded product value expressed by the user and the thresholds set to be generous or severe in penalizing disappointing selling agents. Most importantly, there is a mechanism for adjusting penalty factors to ensure protection for the users, regardless of their preferences for price and quality and their tolerance for lacklustre goods.

Acknowledgments This research is partially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. Bakos Y, Dellarocas C (2003) Cooperation without enforcement? A comparative analysis of litigation and online reputation as quality assurance mechanisms. In: MIT Sloan School of Management, Working Paper 4295-03
2. Barber KS, Kim J (2002) Soft security: isolating unreliable agents from society. In: Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02) Workshop on Trust, Reputation and Security: Theories and Practice
3. Breban S, Vassileva J (2002) Using inter-agent trust relationships for efficient coalition formation. In: Proceedings of the Fifteenth Conference of the Canadian Society for Computational Studies of Intelligence, pp 221–236
4. Carter J, Ghorbani A, Marsh S (2002) Architectural components of information-sharing societies. *Comput Intell Spec Issue Agent Technol Electron Commer* 18(4):638–655
5. Chavez A, Maes P (1996) Kasbah: an agent marketplace for buying and selling goods. In: Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology
6. Chiang H, Liao Y (2004) An agent-based framework for impulse-induced mobile shopping. In: Proceedings of the Fourth International Conference on Computer and Information Technology, pp 509–514
7. Davis R, Smith RG (1983) Negotiation as a metaphor for distributed problem solving. *Artif Intell* 20(1):63–109
8. de Rosi F, Carofiglio V, Grassano G, Castelfranchi C (2003) Can computers deliberately deceive? A simulation tool and its application to turing's imitation game. *Comput Intell* 19(3):235–263
9. Esfandiari B, Chandrasekharan S (2001) On how agents make friends: mechanisms for trust acquisition. In: Proceedings of the Fifth International Conference on Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies, pp 27–34
10. Fullam KK, Barber KS (2007) Dynamically learning sources of trust information: experience vs. reputation. In: Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-07)
11. Huynh TD, Jennings NR, Shadbolt NR (2006) An integrated trust and reputation model for open multi-agent systems. *J Auton Agent Multi Agent Syst* 13:119–154
12. Jones S, Marsh S (1997) Human-computer-human interaction: trust in CSCW. *SIGCHI* 29(3)

13. Jonker C, Treur J (1999) Formal analysis of models for the dynamics of trust based on experiences. In: *Proceedings of the Fourth International Conference on Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies*, pp 81–94
14. Jurca R, Faltings B (2003) An incentive-compatible reputation mechanism. In: *Proceedings of the IEEE Conference on E-Commerce*
15. Jurca R, Faltings B (2002) Towards incentive-compatible reputation management. In: *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02) Workshop on Deception, Fraud and Trust in Agent Societies*
16. Keegan S, O'Hare G M P (2004) Easishop: delivering cross merchant product comparison shopping for the mobile user. In: *Proceedings of the Fifteenth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC-04)*, pp 1860–1864
17. Kerr R, Cohen R (2007) Towards provably secure trust and reputation systems in e-marketplaces. In: *Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-07)*
18. Kerr R, Cohen R (2006) Modeling trust using transactional, numerical units. In: *Proceedings of the Conference on Privacy, Security and Trust (PST-06)*
19. Regan K, Tran T, Cohen R (2005) Sharing models of sellers amongst buying agents in electronic marketplaces. In: *Proceedings of the 10th International Conference on User Modeling (UM-05) Workshop on Decentralized, Agent Based and Social Approaches to User Modelling*
20. Leung CW, Chan SC, Chung F (2006) A collaborative filtering framework based on fuzzy association rules and multiple-level similarity. *Knowl Inform Syst* 10(3):357–381
21. Rosendahl A, Hampe J F, Botterweck G (2007) Mobile home automation—merging mobile value added services and home automation technologies. In: *Proceedings of the Sixth International Conference on Mobile Business (ICMB-07)*
22. Russell S, Norvig P (1995) *Artificial intelligence: a modern approach*. Prentice Hall, Englewood Cliffs
23. Sabater J, Sierra C (2001) REGRET: a reputation model for gregarious societies. In: *Proceedings of the Fifth International Conference on Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies*, pp 61–69
24. Sen S, Sekaran M, Hale J (1994) Learning to coordinate without sharing information. In: *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pp 426–431
25. Smith RG (1980) The contract net protocol: high level communication and control in a distributed problem solver. *IEEE Trans Comput* C-29(12):1104–1113
26. Sutton R, Barto A (1998) *Reinforcement learning: an introduction*. MIT Press, Cambridge
27. Teacy WTL, Patel J, Jennings NR, Luck M (2005) Coping with inaccurate reputation sources: experimental analysis of a probabilistic trust model. In: *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*
28. Tran T, Cohen R (2004a) 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 Multi Agent Systems (AAMAS-04)*, vol 2, pp 828–835
29. Tran T, Cohen R (2004b) The advantages of designing adaptive business agents using reputation modelling compared to the approach of recursive modelling. *Comput Intell Spec Issue Bus Agents Semant Web* 20(4):532–547
30. Vidal JM, Durfee EH (1996) The impact of nested agent models in an information economy. In: *Proceedings of the Second International Conference on Multi-Agent Systems*, pp 377–384
31. Weng Z, Tran T (2007) A mobile intelligent agent-based architecture for e-business. *Int J Inform Technol Web Eng* 2(4):63–80
32. Wong T, Lam W (2008) Learning to extract and summarize hot item features from multiple auction web sites. *Knowl Inform Syst* 14(2):143–160
33. Yu B, Singh MP (2000) A social mechanism of reputation management in electronic communities. In: Klusch M, Kerschberg L (eds) *Cooperative information agents IV, Lecture Notes in Artificial Intelligence*, vol. 1860. Springer, Berlin, pp 154–165
34. Zhang J, Cohen R (2007a) An incentive mechanism for eliciting fair ratings of sellers in e-marketplaces. In: *Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-07)*
35. Zhang J, Cohen R (2007b) Design of a mechanism for promoting honesty in e-marketplaces. In: *Proceedings of the Twenty Second National Conference on Artificial Intelligence (AAAI-07)*
36. Zhang J, Cohen R (2007c) Evaluating the trustworthiness of advice about selling agents in e-marketplaces: a personalized approach. *Electronic Commerce Research and Applications*

Author Biography



Thomas Tran is an Assistant Professor in the School of Information Technology and Engineering at the University of Ottawa. He is also a member of the Canadian Society for the Computational Studies of Intelligence (CSCSI), the American Association for Artificial Intelligence (AAAI), and the IEEE. Thomas received his Ph.D. from the University of Waterloo in 2004, and was the recipient of the Governor General's Gold Medal at the Convocation Ceremony. His research interests include Artificial Intelligence (AI), Electronic Commerce, Autonomous Agents, Multi-Agent Systems, Reinforcement Learning, Trust and Reputation Modelling, Recommender Systems, Architecture for Mobile e-business, Agent Negotiation, Mechanism Design, and Applications of AI.