Maximizing Expected Impact in an Agent Reputation Network

Gavin Rens[1], Abhaya Nayak[2], and Thomas Meyer[1]

[1] University of Cape Town, South Africa
{grens,tmeyer}@cs.uct.ac.za
[2] Macquarie University, Sydney, Australia
abhaya.nayak@mq.edu.au

Abstract. Many multi-agent systems (MASs) are situated in stochastic environments. Some such systems that are based on the partially observable Markov decision process (POMDP) do not take the benevolence of other agents for granted. We propose a new POMDP-based framework which is general enough for the specification of a variety of stochastic MAS domains involving the impact of agents on each other's reputations. A unique feature of this framework is that actions are specified as either undirected (regular) or directed (towards a particular agent), and a new directed transition function is provided for modeling the effects of reputation in interactions. Assuming that an agent must maintain a good enough reputation to survive in the network, a planning algorithm is developed for an agent to select optimal actions in stochastic MASs. Preliminary evaluation is provided via an example specification and by determining the algorithm's complexity.

Keywords: Learning · Trust and reputation · Planning · Uncertainty · POMDP.

1 Introduction

Autonomous (synthetic) agents need to deal with questions of trust and reputation in diverse domains such as e-commerce platforms, crowdsourcing systems, online virtual worlds, and P2P file sharing systems (Yu et al., 2013; Pinyol and Sabater-Mir, 2013), wireless sensor networks (Momani and Challa, 2010), the semantic web (Artz and Gil, 2007) and distributed AI/multi-agent systems (Sabater and Sierra, 2005). Yet we see very few computational trust/reputation frameworks which can handle uncertainty in actions and observations in a principled way and which are general enough to be useful in several domains. A partially observable Markov decision process (POMDP) (Monahan, 1982; Lovejoy, 1991) is an abstract mathematical model for reasoning about the utility of sequences of actions in stochastic domains. Although its abstract nature allows it to be applied to various domains where sequential decision-making is required, a POMDP is typically used to model a single agent. We propose an extension to the POMDP, which has the potential to be quite generally applicable in stochastic multi-agent systems where trust and reputation are an issue.

We call the proposed model *Reputation Network POMDP* (*RepNet-POMDP* or simply *RepNet*).

As is done by Pinyol et al. (2012), we distinguish between the *image* one agent has in the eyes of another and the *reputation* with which an agent is associated, given its aggregated image in the eyes of a group of agents. The unique features of a RepNet are: (i) it distinguishes between undirected (regular) actions and directed actions (towards a particular agent), (ii) besides the regular state transition function, it has a directed transition function for modeling the effects of reputation in interactions and (iii) its definition (and usability) is arguably more intuitive than similar frameworks. Furthermore, we suggest methods for updating agents' image of each other, for learning action distributions of other agents, and for determining perceived reputations from images. We present a planning algorithm for an agent to select optimal actions in a network where reputation makes a difference.

Many multi-agent systems need a rich framework to represent uncertainty, especially when they are associated with the real world. What makes RepNets different from most other frameworks for multi-agent reasoning about trust and reputation is that uncertainty is treated as a primary concern. Many aspects of the network are modeled as probability distributions, for instance, state transitions, possible observations, possible actions, and possible states. With this richness in representation comes the usual higher complexity in reasoning.

We start by covering the relevant POMDP theory, for background. This is followed by a section formally defining the proposed framework, including a proposal for how reputation can be computed and learnt (updated) and an algorithm for determining the actions which lead to optimal benefit. Finally, before our concluding remarks, we briefly review the work most related to trust and reputation in groups of agents, especially where there is much uncertainty. This paper is accompanied by a technical report which can be found online at arXiv.org in primary category cs.AI with the same title as this paper.

2   Background - The POMDP

A POMDP is a tuple $\langle S, A, T, \Omega, O, R, b^0 \rangle$, where S is a finite set of states, A is a finite set of actions, T is a transition function s.t. $T(s, a, s')$ is the probability that a executed in s will take an agent to s', Ω is a finite set of observations, O is an observation function s.t. $O(a, o, s)$ is the probability that observation o due to action a is perceived in s, R is a reward function s.t. $R(s, a, s')$ is the (immediate) reward the agent experiences when it executes action a in state s and ends up in state s', and $b^0 \in \Delta(S)$ is the initial probability distribution over states when the agent is deployed.[3]

In POMDP theory, a probability distribution over states is called a *belief state*. Formally, a belief state is a function $b : S \to [0, 1]$ such that $\sum_{s \in S} b(s) = 1$. The *state estimation* function is used by an agent to update its belief state. An

---

[3] $\Delta(X)$ is the set of probability distributions over elements in X.

agent with current belief state b_{cur} will have new belief state $b_{new} = SE(a, o, b_{cur})$ after performing action a and making observation o, where $SE(a, o, b_{cur}) := \{(s', p) \mid s' \in S\}$ such that

$$p = \frac{O(a, o, s') \sum_{s \in S} T(s, a, s') b_{cur}(s)}{P(o \mid a, b_{cur})}$$

is the probability of perceiving o in the belief state 'reached' after performing a in b ($P(o \mid a, b) := \sum_{s' \in S} O(a, o, s') \sum_{s \in S} T(s, a, s') b(s)$ is a normalizing factor).

The expected reward – with respect to some belief state b – an agent gets for executing action a is computed as $\sum_{s \in S} R(a, s) b(s)$.

An agent should maximize the rewards it expects to receive in the future. Rewards farther into the future are conventionally 'discounted' because, intuitively, situations in the distant future are less likely to be experienced. Hence, a *discount factor* $\gamma \in [0, 1]$ is applied when determining the value of sequences of actions. The optimal value of belief state b given an horizon of k steps is defined as the optimal belief state value function V^*:

$$V^*(b, k) := \max_{a \in A} \Big\{ \sum_{s \in S} R(a, s) b(s) + \gamma \sum_{o \in \Omega} P(o \mid a, b) V^*(SE(a, o, b), k - 1) \Big\},$$

$$V^*(b, 1) := \max_{a \in A} \Big\{ \sum_{s \in S} R(a, s) b(s) \Big\}.$$

3   RepNet-POMDP - a Proposal

We shall first introduce the basic structure of a RepNet-POMDP, then discuss matters relating to image and reputation, and finally, develop a definition (and thus an algorithm) for computing optimal behaviour in RepNets. The accompanying technical report contains two proposed instantiations of the image update function, and explains how action distributions can be learnt.

3.1   The Basis

The components of the RepNet structure will first be introduced briefly, followed by a detailed discussion of each component. A RepNet-POMDP is defined as a pair of tuples $\langle System, Agents \rangle$. *System* specifies the aspects of the network that apply to all agents; global knowledge shared by all agents.

$$System := \langle G, S, A, \Omega, I, U \rangle,$$

where

- G is a finite set of agents $\{g, h, i, \dots\}$.
- S is a finite set of states.
- A is the union of finite disjoint sets of *directed* actions A^d and *undirected* actions A^u.

– Ω is a finite set of observations.
– $I : G \times S \times G \times S \times A \to [-1, 1]$ is an *impact* function s.t. $I(g, s, h, s', a)$ is the impact on g in s due to h in s' performing action a.
– $U : [0, 1] \times [-1, 1] \times [-1, 1] \to [-1, 1]$ is an *image update* function used by agents when updating their image profiles s.t. $U(\alpha, r, i)$ is the new image level given learning rate α, current image level r and current impact i.

Agents specifies the names and subjective knowledge of the individual agents; individual identifiers and beliefs per agent.

$$Agents := \langle \{T_g\}, \{DT_g\}, \{O_g\}, \{AD_g^0\}, \{Img_g^0\}, \{B_g^0\} \rangle, \text{[4]}$$

where

– $T_g : S \times A^u \times S \to [0, 1]$ is the *transition* function of agent g.
– $DT_g : S \times A^d \times [-1, 1] \times S \to [0, 1]$ is the *directed transition* function of agent g s.t. $DT_g(s, a, r, s')$ is the probability that agent g executing an action a in state s (directed towards some agent h) will take g to state s', while g believes that agent h perceives g's reputation to be at level r. $DT_g(s, a, r, s') = P(s' \mid g, s, a, r)$, hence $\sum_{s' \in S} DT_g(s, a, r, s') = 1$, given some current state s, some reputation level r and some directed action a of g.
– O_g is g's observation function s.t. $O_g(a, o, s)$ is the probability that observation o due to action a is perceived by g in s.
– $AD_g^0 : G \times S \to \Delta(A)$ is agent g's initial *action distribution* providing g with a probability distribution over actions for each agent in each state.
– $Img_g^0 : G \times G \to [-1, 1]$ is g's initial image profile. $Img_g(h, i)$ is agent h's image of agent i, according to g.
– $B_g^0 : G \to \Delta(S)$ is g's initial mapping from agents to belief states.

The agents in G are thought of as forming a linked group who can influence each other positively or negatively and who cannot be influenced by agents outside the network. It is assumed that all action execution is synchronous, that is, one agent executes one action if and only if all agents execute one action. All actions are assumed to have an equal duration and to finish before the next actions are executed. The immediate effects of actions are also assumed to have occurred before the next actions. That is, for every agent's action, the successor state is reached, reputation updated and impact completed before the next round of actions is executed. We shall call these rounds of actions *steps*.

All agents share knowledge of who the agents in the network are, what the set of possible states is (S), what actions can possibly be performed (A), impact of actions (I), image update function (U), the set of possible observations (Ω) and the likelihoods of perceiving them in various conditions. All other components of the structure relate to individual agents and how they model some aspect of the network: dynamics of their actions (T_g and DT_g) and observations (O_g),

---

[4] Notation $\{X_g\}$ is shorthand for $\{X_g \mid g \in G\}$. That is, there is a function X for each agent in G.

likelihood of actions of other agents (AD_g), beliefs about reputation (Img_g) and their initial belief states (B_g).

In this formalism, only the action distributions (AD_g), image profiles (Img_g) and set of belief states (B_g) change. All other models remain fixed.

An agent should maintain an image profile for all other agents in the network in order to guide its own behaviour. An image profile is an assignment of image levels between every ordered pair of agents. For instance, if (according to g) h's image of i $(Img_g(i, h))$ is, on average, low, g should avoid interactions with i if g has a good image of h $(Img_g(h, g))$. Note that agents' multi-lateral image is not common knowledge in the network. Hence, each agent has only an *opinion* about each pair of agent's image as deemed by each other agent. In general, it is not necessary that $Img_g(g, h) = Img_h(g, h)$. Nor is it necessary that $Img_g(h, i) = Img_g(i, h)$.

$Img_g(h, i)$ changes as agent g learns how agent i 'treats' its network neighbour h. Agent g uses U to manage the update of its levels of reputation as deemed by other agents. An agent needs to have a strategy how to build up its image profile of each other agent. How should an agent adjust its opinion of another agent if it keeps on being impressed by that agent? Formally, there is a maximum image level of 1. Should a well-behaved agent get an image of 1 after relatively few actions and then have its image stay at 1? Or should such a 'good' agent's image gradually approach 1? If so, how gradually? We decided to define the image update function U common to all agents for the sake of simplicity, while introducing the RepNet-POMDP framework. This assumption can easily be relaxed by defining U differently for each agent.

Actually, we define directed transitions to be conditioned on reputation (derived from images): Suppose g wants to trade with h. Agent g could perform a `tradeWith_h` action. But if h deems g's reputation to be low, h would not want to trade with g. This is an example where the effect of an action by one agent (g) depends on its level of reputation as perceived by the recipient of the action (h). Note that it does not make sense to condition the transition probability on the reputation level of the recipient as perceived by the actor (h's reputation as perceived by g in this example): The effect of an action by g should have nothing to do with h's image levels, given the *action is already committed to* by g. However, the effect of an action committed to (especially one directed towards a particular agent) may well depend on the actor's (g's) reputation levels; h may react (effect of the action) differently depending on g's reputation.

Continuing with the example, assume s' is a state in which g gets what it wanted out of a trade with h, and s is a state in which g is ready to trade. Then $DT(s, \texttt{tradeWith\_}h, -0.6, s')$ might equal 0.1 due to h's inferred unwillingness to trade with g due to g's current bad reputation (-0.6) as deemed by h. On the other hand, $DT(s, \texttt{tradeWith\_}h, 0.6, s')$ might equal 0.9 due to g's high esteem (0.6) as deemed by g and thus inferred willingness to trade with g.

It makes sense to talk about the impact/effect of an action (a) from one state (s) to another (s'): $I(g, s, h, s', a)$. Suppose you (agent g) are in state s where you have a high energy level, and I (agent h) am in state s' where I have a low

energy level. If you push me (action a), you can push harder than usual and I can recover or defend myself less than usual. The impact of the action is thus dependent on both the state in which the action is executed and the state being impinged upon.

It is assumed that every agent g has some (probabilistic) idea about what actions its neighbours will perform in a given state. As stated in the definition of the RepNet-POMDP structure, $AD_g(h, s)$ is a distribution over the actions in A that h could take when in state s. It is debatable how realistic it would be to have a single action model $AD(h, s)$ common to all agents. Every agent g perceives different signals, depending on their state. Every agent would thus learn a different action distribution for its neighbours. One method for learning these distributions is proposed later in this section.

The other component of the structure which changes is B_g; every agent (g) maintains a probability distribution over states for every agent in G (including itself). That is, for every agent g, its belief state for every agent h $(B_g(h))$ is maintained and updated. In other words, every agent maintains a belief state representing 'where' it thinks the other agents (incl. itself) are. As actions are performed, every g updates these distributions of itself and its neighbours. In POMDP theory, probability distributions over states are called *belief states*. B_g changes via 'normal' state estimation as in regular POMDP theory. The state estimation (update) theory for RepNet-POMDPs is explained in the next subsection.

3.2   Image and Reputation in RepNets

There are many ways in which an agent can compute reputations, given the components of a RepNet-POMDP. In this section, we investigate one approach.

The notions of trust and reputation are typically considered to be different (Artz and Gil, 2007; Seymour and Peterson, 2009; Momani and Challa, 2010): reputation informs trust with reputation itself being an agregation of indirect information, namely, evalution of an agent which has propagated through a network. However, it has also been argued that trust and reputation are not easily distinguished (Pinyol et al., 2012, Sect. 7.1, e.g.). At a basic level, *image* is an opinion of another agent which is built up from direct experience with that agent (Conte and Paolucci, 2002; Pinyol et al., 2012). Direct experience or direct interaction with an agent is sometimes viewed as a kind of reputation, however, we feel it is useful to distinguish image as a personal experience, reputation as propagated image valuations, and trust as reputation plus other sources of information concerning an agent's willingness to be vulnerable when dealing with the trustee.

Recall that $AD_g(h, s)$ is the probability distribution over actions g believes h executes in s. In other words, $AD_g(h, s)(a)$ is the probability of a being executed by h in s according to g. The image that an agent i in state s^i has of an agent h in state s^h can be modeled as

$$\sum_{a \in A} \left[\delta AD_g(i, s^i)(a) I(h, s^h, i, s^i, a) + (1 - \delta) AD_g(h, s^h)(a) I(i, s^i, h, s^h, a) \right],$$

where $\delta \in [0, 1]$ trades off the importance of the impacts on h and impacts due to h. As δ tends to 1, more importance is given to how i treats h and less to how h treats i. Recall that B_g is the set of current belief states of all agents in the network, according to g. Hence, $B_g(i)$ is a belief state, and $B_g(i)(s)$ is the probability of i being in s, according to g. For better readability, we might denote $B_g(i)$ as b_i^g. Agent g perceives at some instant that i's image of h is

$$Image_g(h, i, B_g) := \sum_{s^h \in S} b_h^g(s^h) \sum_{s^i \in S} b_i^g(s^i) \sum_{a \in A} \left[\delta AD_g(i, s^i)(a) I(h, s^h, i, s^i, a) \right.$$
$$\left. + (1 - \delta) AD_g(h, s^h)(a) I(i, s^i, h, s^h, a) \right]. \tag{1}$$

In (1), the uncertainty of agents h and i's states are taken into account. Note that this perceived image is independent of g's state.

Now we can define the new image of one agent h with respect to another i, as perceived by a particular agent g:

$$U(\alpha, Img_g(h, i), Image_g(h, i, B_g)),$$

where α is a learning rate supplied from outside. Just as SE updates an agent's belief state, the *image expectation* function $IE(g, Img_g, \alpha, B_g) := Img_g'$ updates an agent's image profile. That is, given g's set of belief states B_g, for all $h, i \in G$,

$$Img_g'(h, i) = U(\alpha, Img_g(h, i), Image_g(h, i, B_g)).$$

An agent g could form its opinion about h in at least three ways: (1) by observing how other agents treat h, (2) by observing how h treats other agents and (3) by noting other agents' opinion of h. But g must also consider the reasons for actions and opinions: Agent i might perform an action with a negative impact on h because i believes h has a bad reputation or simply because i is bad. We define reputation as

$$RepOf_g(h) := \frac{1}{|G|} \left[Img_g(h, g) + \sum_{i \in G, i \neq g} Img_g(h, i) \times Img_g(i, g) \right].$$

Here, we have assumed that it does not make sense to weight $Img_g(h, g)$ by $Img_g(g, g)$ because it makes no sense to weight one's opinion about h's image by one's opinion of one's own image. Hence, $Img_g(h, g)$ is implicitly weighted by 1.

To get a better feel for the behaviour of $RepOf_g(h)$, consider the following table. The third column can be thought of as the real reputation of h according to what g believes i thinks of h. Notice that if i has a negative image level, then g regards h in an opposite way to how i regards h. That is, an agent with a negative image is regarded as purposefully misleading. Moreover, the degree of negativity of i's image gives the degree to which g regards h in the opposite light. If h has zero image according to i, then g cannot assign a reputation to h, no matter what g thinks of i. Similarly, if i has zero image, then g cannot assign any reputation to h, no matter what i thinks of h. When i has an image close to zero

$Img_g(h,i)$	$Img_g(i,g)$	$Img_g(h,i) \times Img_g(i,g)$
-0.5	-0.5	0.25
0.0	-0.5	0.0
0.5	-0.5	-0.25
-0.5	0.0	0.0
0.0	0.0	0.0
0.5	0.0	0.0
-0.5	0.5	-0.25
0.0	0.5	0.0
0.5	0.5	0.25

(according to g), it is like assigning very little informational value to what i says. And if i has high image (close to 1), then i's opinions are highly trusted, but if i has low image (close to -1), then i's opinions are highly distrusted, informing g to believe the complete opposite of what i proclaims.

The question is, Where did g get i's image? What if, for instance, g hears from trusted agent j that i's reputation/image is opposite to what g thought of i's reputation? There are several ways to remedy this uncertainty. One way is to iteratively seek and weight the reputation of agents to some 'depth'. This method has technical difficulties which we do not try to address now. The simple approach above partly solves the problem in two ways. (1) i's reputation is only one of all the reputations considered by g, and g takes the *average* of all agents' opinions of g to come to a conclusion of what to think of h (h's reputation according to g). (2) Reputation is also informed by actual activity, as perceived by each agent g. Hence, every agent builds up more accurate opinions of other agents, according to their *activities* (not only what others say about others). Activities inform image and image informs reputation.

Suppose a step has just occurred and g perceives o. Let b_h^g be the belief state g assigns to h. The new belief state of some agent g according to itself is defined by

$$OSE(a,o,(b_g^g)_{cur}) := \{(s',p) \mid s' \in S\} \tag{2}$$

such that

$$p = \frac{O(a,o,s') \sum_{s \in S} T_g^{du}(s,a,s',g)(b_h^g)_{cur}(s)}{P(o \mid a,(b_g^g)_{cur})}$$

and, recalling that A^d and A^u are the sets of directed, respectively, undirected actions,

$$T_g^{du}(s,a,s',h) := \begin{cases} DT(s,a,RepOf_g(h),s') & \text{if } a \in A^d \\ T(s,a,s') & \text{if } a \in A^u \end{cases}$$

and $P(o \mid a,b)$ is a normalizing constant. We call definition (2) the *objective state estimation* function (of agent g).

The new belief state of h ($\neq g$) according to g is defined by

$$SSE(g,h,o,(b_h^g)_{cur}) := \{(s',p) \mid s' \in S\} \tag{3}$$

such that

$$p = \frac{\sum_{a \in A} O(a, o, s') \sum_{s \in S} T_g^{du}(s, a, s', h)(b_h^g)_{cur}(s) AD_g(h, s)(a)}{P(o \mid AD_g, (b_h^g)_{cur})}$$

and $P(o \mid AD_g, b)$ is a normalizing constant. We call definition (3) the *subjective state estimation* function (of agent h other than g).

An agent g uses (3) when updating its belief about other agents' belief states, else g uses (2) to update its own belief state. Let $BSE(g, a, o, B_g)$ be the set of belief states of all agents (from g's perspective) after the next step, determined from the current set of belief states B_g, given agent g executed a and perceived o. That is, $BSE(g, a, o, B_g)$ is defined as

$$\{(h, b_h^g) \mid h \in G, h \neq g, b_h^g = SSE(g, h, o, B_g(h))\} \cup \{(g, OSE(a, o, B_g(g)))\}.$$

3.3   Optimal Behaviour in RepNets

Advancement of an agent in RepNet-POMDPs is measured by the total impact on the agent. An agent might want to maximize the network's (positive) impact on it after several steps in the system.

An agent g currently in s^g can predict agent h's action a performed in s^h to cause an instantaneous impact on it (g). The *expected* instantaneous impact of h in s^h on g in s^g is calculated as

$$\sum_{a \in A} AD_g(h, s^h)(a) I(g, s^g, h, s^h, a).$$

We abbreviate the un-normalized perceived impact on g in s^g by the network neighbours

$$\sum_{h \in G, h \neq g} \sum_{s^h \in S} B_g(h)(s^h) \sum_{a \in A} I(g, s^g, h, s^h, a) AD_g(h, s^h)(a)$$

as $PIN(g, s^g, AD_g, B_g)$. That is, $PIN(g, s^g, B_g)$ is the expected impact other agents have on g in a particular state, with respect to the neighbours' possible actions and their belief states.

The self impact on g executing a in s^g is defined as $I(g, s^g, g, s^g, a)$. Here we take it that every agent knows which action it performs. We thus define the total perceived impact on g by the network as

$$PI_{tot}(g, a, B_g) := \frac{1}{|G|} \sum_{s^g \in S} B_g(g)(s^g) \big(PIN(g, s^g, AD_g, B_g) + I(g, s^g, g, s^g, a) \big).$$

Intuitively, an agent g can choose its next action so as to maximize the total impact all agents will have on it in the future. Then the optimal impact function

w.r.t. g over the next k steps is defined as

$$OI(g, AD_g, Img_g, B_g, k) := \max_{a \in A} \big\{ PI_{tot}(g, a, B_g)$$
$$+ \gamma \sum_{o \in \Omega} P(o \mid a, B_g) OI(g, AD'_g, Img'_g, B'_g, k-1) \big\},$$
$$OI(g, AD_g, Img_g, B_g, 1) := \max_{a \in A} \big\{ PI_{tot}(g, a, B_g) \big\},$$

where AD'_g is $ADE(g, o, AD_g)$, Img'_g is $IE(g, Img_g, \alpha, B_g)$ and B'_g is $BSE(g, a, o, B_g)$.

4   Related Work

We mention two frameworks that include notions of quantitative uncertainty, but that do not involve the POMDP model:

Yu and Singh (2002) develop an (uncertain) evidential model of reputation management based on the Dempster-Shafer theory. A limitation of this approach is that it models only the uncertainty in the services received and in the trustworthiness of neighbours who provide referrals. It does not model dynamical systems, nor does it allow for stochastic actions and observations.

Pinyol et al. (2012) propose an integration of a cognitive reputation model, called *Repage*, into a BDI agent. They define a many-sorted, first-order, probabilistic logic capable of capturing the semantics of Repage information. The logic includes predicates for belief, image and reputation. With their logic, Pinyol et al. (2012) can specify capabilities or services that our framework cannot. On the other hand, their Repage + BDI architecture cannot model noisy observations or uncertainty in state (belief states). Whereas the approach of Yu and Singh (2002) keeps a history of a fixed number of interactions to determine "local belief" and fixed depth "referral chains", our framework is more flexible. Admittedly, our framework may require a learning rate for maintaining 'local belief' (which we call *image*).

We mention two frameworks that employ POMDPs as a component, but cannot be specified as a POMDP or variant thereof:

Regan et al. (2005) aim to construct a principled framework, called *Advisor-POMDP*, for buyers to choose the best seller based on some measure of reputation in a market consisting of autonomous agents: a model for collecting and using reputation is developed using a POMDP.

SALE POMDP (Irissappane et al., 2014) is an extension of Advisor-POMDP: It can deal with the seller selection problem by reasoning about advisor quality and/or trustworthiness and selectively querying for information to finally selects a seller with high quality.

The major difference between Advisor- and SALE POMDP on the one hand, and RepNets on the other, is that a RepNet has a model for every agent in the network and every agent has a (subjective) view on every other agent's belief state and action likelihood. Advisor- and SALE POMDP do not allow for the modeling of common information shared by all agents in the network, and hence

cannot cater for agents having views of other agents' belief state and action likelihood.

Finally, regarding related work, we mention frameworks that are variants of the POMDP (and thus most related to the present work).

Decentralized POMDPs (DEC-POMDPs) (Bernstein et al., 2000) are concerned more with effective collaboration in noisy environment than with self-advancement in a network of potentially unfriendly strangers.

Interactive POMDPs (I-POMDPs) (Gmytrasiewicz and Doshi, 2005) are for specifying and reasoning about multiple agents, where willingness to cooperate is not assumed. Whereas DEC-POMDP agents do not have a model for every other agent's belief state and action likelihood, I-POMDP agents maintain a model of every other agent (incl. their own). A model that agent g has of agent h is not necessarily the model h has of itself.

I-POMDPs and DEC-POMDPs do not have a notion for trust, reputation or image. Seymour and Peterson Seymour and Peterson (2009) introduce notions of trust to the I-POMDP, which they call trust-based I-POMDP (TI-POMDP). A trust model τ_i (similar to our Img_g) is maintained as part of agent states. They define transition functions from environment states to environment states (via composite actions), which reportedly includes changes to τ_i. However, there are several inconsistencies in the presentation of their framework (which we cannot discuss due to limited space); it is thus hard to compare RepNets to TI-POMDPs. Nonetheless, RepNets are arguably more understandable model structures than I-POMDP and TI-POMDPs, and as with I-POMDPs, the factored formulation of RepNets reduces the curse of dimensionality.

5 Summary and Future Work

This paper presented a new framework, called RepNet-POMDP, for agents in a network of self-interested agents to make considered decisions. The framework deals with several kinds of uncertainty and facilitates agents in determining the reputation of other agents. An algorithm was provided for an agent to look ahead several steps in order to choose actions in a way that will influence its reputation so as to maximize the network's positive impact on the agent. We aimed to make the framework easily understandable and generally applicable in systems of multiple, self-interested agents where partial observability and stochasticity of actions are problems. To illustrate the usefulness of RepNet-POMDPs, the accompanying technical report has an example where we set up a trading domain for a network of four agents.

There are of course many more possible instantiations of U. Just as the reward function in POMDPs needs to be tailored for the domain, U needs to be tailored for the RepNet-POMDP domain. Choosing between difference, saturation or some other update function will be domain dependent. The accompanying technical report provides a detailed discussion.

In the accompanying technical report, we present one way in which an agent can incrementally learn what actions other agents are likely to perform. That

is, we propose how some agent g can learn the probability of some agent h performing some action in some state. The agent learns by Bayesian conditioning on observations.

There are many ways to define image and reputation. The framework could also have been made more sophisticated, requiring a trust layer. These considerations are left for future work. We did not discuss how one could, in the RepNet framework, distinguish between a new agent whose image or reputation is unknown and a well-known agent with an average image ($RepOf_g(h) \approx 0$). This is not a failing of the RepNet framework, but discuss it is beyond the scope of this paper. We have assumed that $RepOf_g$ is the weighted average of all image opinions, but it might be more reasonable for g to take others' opinions less seriously as it builds confidence about images of agents over time via actual activities perceived. Another aspect of RepNet that needs improvement is reputation (trust) in *context*. The present framework assumes a global context, wherein the learnt reputation of an agent is applicable at all times / in all states. This approach is appropriate only in very narrow or specific domains where there are few degrees of freedom in context. Typically though, context will change, and applicability of reputation of agents will change with context. See, for instance, the work of Nayak (2012) and Pinyol et al. (2012).

Asking for trust/reputation opinions from trusted agents is not possible in RepNet-POMDPs. All reputation info comes through observation; none through direct communication. If the image profiles were part of the states (as in I-POMDPs (Gmytrasiewicz and Doshi, 2005)), 'asking' might be modeled. But this would play into the curse of dimensionality; a bad thing.

Clearly, the planning algorithm presented here is highly intractable. Approximate methods for solving large POMDPs could also be looked at to make Rep-Nets practical (Gmytrasiewicz and Doshi, 2009; Irissappane et al., 2014). The accompanying technical report provides an evaluation of the *optimal impact* algorithm in terms of computational complexity.

Bibliography

Artz, D. and Gil, Y. (2007). A survey of trust in computer science and the semantic web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):58–71.

Bernstein, D., Zilberstein, S., and Immerman, N. (2000). The complexity of decentralized control of markov decision processes. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, UAI'00, pages 32–37, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Conte, R. and Paolucci, M. (2002). *Reputation in artificial societies: Social beliefs for social order.* Kluwer, Dordrecht.

Gmytrasiewicz, P. and Doshi, P. (2005). A framework for sequential planning in multi-agent settings. *J. Artif. Intell. Research*, 24(1):49–79.

Gmytrasiewicz, P. and Doshi, P. (2009). Monte carlo sampling methods for approximating interactive pomdps. *J. Artif. Intell. Research*, 34:297–337.

Irissappane, A., Oliehoek, F., and Zhang, J. (2014). A POMDP based approach to optimally select sellers in electronic marketplaces. In *Proceedings of the Thirteenth Intl. Conf. on Autonomous Agents and Multiagent Systems*, AAMAS-14, pages 1329–1336, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Lovejoy, W. (1991). A survey of algorithmic methods for partially observed Markov decision processes. *Annals of Operations Research*, 28:47–66.

Momani, M. and Challa, S. (2010). Survey of trust models in different network domains. *Ad hoc, Sensor and Ubiquitous Computing*, 1(3):1–19.

Monahan, G. (1982). A survey of partially observable Markov decision processes: Theory, models, and algorithms. *Management Science*, 28(1):1–16.

Nayak, A. (2012). *Trust in Context*, pages 517–529. LNAI. Springer Berlin Heidelberg, Berlin, Heidelberg.

Pinyol, I. and Sabater-Mir, J. (2013). Computational trust and reputation models for open multi-agent systems: a review. *Artificial Intelligence Review*, 40:1–25.

Pinyol, I., Sabater-Mir, J.and Dellunde, P., and Paolucci, M. (2012). Reputation-based decisions for logic-based cognitive agents. *Autonomous Agents and Multi-Agent Systems*, 24(1):175–216.

Regan, K., Cohen, R., and Poupart, P. (2005). The advisor-pomdp: A principled approach to trust through reputation in electronic markets. *Conference on Privacy Security and Trust*, 1.

Sabater, J. and Sierra, C. (2005). Review on computational trust and reputation models. *Artif. Intell. Review*, 24:33–60.

Seymour, R. and Peterson, G. (2009). A trust-based multiagent system. In *Proceedings of Intl. Conf. on Computational Science and Engineering*, pages 109–116. IEEE.

Yu, B. and Singh, M. (2002). An evidential model of distributed reputation management. In *Proceedings of the First Intl. Conf. on Autonomous Agents*

and Multiagent Systems, AAMAS-02, pages 294–301, New York, NY, USA. ACM.

Yu, H., Shen, Z., Leung, C., Miao, C., and Lesser, V. (2013). A survey of multi-agent trust management systems. *IEEE Access*, 1:35–50.