Privacy Loss in Multiagent Planning

A Classical Definition with Illustration

Roman van der Krogt*
Cork Constraint Computation Centre
Department of Computer Science
University College Cork, Cork, Ireland
roman@4c.ucc.ie

ABSTRACT

Privacy is often cited as the main reason to adopt a multiagent approach for a certain problem. This also holds true for multiagent planning. Still, papers on multiagent planning hardly ever make explicit in what ways their systems protect their users' privacy, nor do they give a quantitative analysis. The reason for this is that a theory of privacy loss in multiagent planning is virtually non-existent so far.

This paper proposes a measure for privacy loss based on Shannon's theory of information. To illustrate our approach, we apply this metric to an existing multiagent planning system to assess its merits when it comes to privacy on two domains. For this, we compare its plans to centrally generated solutions (by a trusted third party) for the same problems. The results clearly establish the need for such an analysis: even though the multiagent planner seemingly exchanges little information, its overall performance with respect to privacy is less than that of the centralised system (not taking into account the privacy loss with respect to the central planner, of course).

Categories and Subject Descriptors

I.2.8 [**Problem Solving, Control Methods, and Search**]: Plan execution, formation, and generation; I.2.11 [**Distributed Artificial Intelligence**]: Intelligent Agents, Multiagent Systems

General Terms

Algorithms

Keywords

Multiagent Planning, Privacy

*Roman van der Krogt is supported by an Irish Research Council for Science Engineering and Technology (IRCSET) Postdoctoral Fellowship.

1. INTRODUCTION

The literature names a great many reasons why to pursue multiagent planning. One of the reasons that often comes up is that of *privacy*. Especially in circumstances where the agents represent (possibly competing) companies, sharing data with other parties is considered undesirable. At the same time, it is well recognised that cooperation may be mutually beneficial to all parties. An example of this is the reduction of *empty rides* in transportation. Transportation companies are often faced with deliveries of cargo from point A to point B, but find themselves without a matching order for the return. Often, another company might be in the same situation in the opposite direction: it has a load from B' to A', which are close to respectively points B and A. By cooperating on the two tasks, they can spare themselves the empty return rides, which is beneficial for both parties (and the environment). However, the companies are also each others competitor. As such, they have a natural tendency to distrust one another. By revealing too much information, a company may find itself in a situation where the competitor has such knowledge of the order books and cost structures that it can consistently undercut bids on tenders.

Multiagent planning is one of the tools that can help agents in a situation such as described above. It offers a way to cooperate while being in control of which information is shared and with whom. Yet it is impossible to cooperate without sharing *any* information. At the very least, a pair of cooperating agents has to agree on which subtasks are being carried out by one agent on behalf of the other [13]. Several approaches, such as (Generalised) Partial Global Planning (GPGP; see, for example, [2]), go even further and share detailed parts of their plans. In the latter approach, more information is exchanged. Clearly, this must lead to a poorer performance when it comes to privacy. This raises the obvious question of how to measure this performance. Just how much privacy is lost by exchanging certain information? How can we evaluate which method is better then another when privacy is concerned?

This paper introduces a measure for privacy loss in multiagent planning based on Shannon's Theory of Information [8]. We show how the concept of "uncertainty" that underpins Shannon's work can be interpreted in the context of plans, and how we can derive a measure from this for the information that is gained during negotiations on plan construction or coordination. We then apply this measure to an existing multiagent planner, called MPOPR [13], and show its privacy

behaviour on two domains, compared to a centralised system using a trusted third party. In the latter case, agents extract information about the other agents' plans from the plan they are given from the central planner.

The remainder of the paper is organised as follows. In the next section we will introduce our measure for privacy loss. Then we introduce the multiagent planning system that we applied our work to and the two domains for which we did our analysis. After examining our findings, we discuss some related work from the field of distributed constraint satisfaction. Finally, we draw some initial conclusions and expand on future work.

2. CHARACTERISING PRIVACY LOSS IN MULTIAGENT PLANNING

If we want to speak about a loss in privacy, we have to obtain a measure to compare different cases. Information theory can give us this measure. The key ideas that we introduce here come from Shannon [8]. Whereas Shannon followed a rigorous route in deriving his famous function, we follow the more intuitive explanation given by Schneider [7] in setting out the background.

2.1 Information and Uncertainty

Information is closely linked to uncertainty. Suppose we have M differently coloured balls in a hat, and we intend to randomly draw one. Now we have a certain degree of *uncertainty* regarding the colour of the ball we will draw. When we draw a ball, we get some *information* (on the colour of this ball) and our uncertainty (regarding the colour of this ball) *decreases*. Shannon's work gives an answer to the questions of how to measure this uncertainty. If we assume an equal probability for all of the balls, we would like to say that we have an "uncertainty of M colours". However, we would like our measure of uncertainty to be additive, which leads to the following formula for the uncertainty H:

$$H = \log_2(M) \tag{1}$$

If we intend to draw a second ball there are two situations: either we put the drawn ball back or not:

1. If we return the ball that was drawn, we again have an uncertainty of $\log_2(M)$. Thus, the total uncertainty that we have in drawing two balls is $2\log_2(M) = \log_2(M^2)$ in this case.

2. If we do not return the ball, we obtain an uncertainty for the second ball of $\log_2(M-1)$. Hence, the reduction in uncertainty, or the information gained, by drawing a ball in this situation is $\log_2(M) - \log_2(M-1) = \log_2(\frac{M}{M-1})$.

So far, we have considered the situation where each outcome (i.e. each colour) had equal probability. But what if there are fewer colours then balls, with some colours more likely than others? First, let us rearrange Equation 1 in terms of the probability $P = \frac{1}{M}$ that any colour is drawn:

$$\begin{aligned} H &= \log_2(M) \\ &= -log_2(\frac{1}{M}) \\ &= -log_2(P) \end{aligned} \tag{2}$$

Now, let P_i be the probability of drawing colour i, with $\sum_{i=1}^{M} P_i = 1$. The "surprisal" [11] of drawing the i^{th} colour is defined by analogy with $-\log_2(P)$ to be

$$u_i = -\log_2(P_i) \tag{3}$$

In the generalised case, uncertainty is the average surprisal for the infinite string of colours drawn (returning each ball before drawing a new one). For a string of finite length N, with colour i appearing N_i times, this average is

$$\sum_{i=1}^{M} \frac{N_i}{N} u_i \tag{4}$$

For an infinite string, the frequency $\frac{N_i}{N}$ approaches P_i, the probability of drawing this colour. The average surprisal would now be:

$$\sum_{i=1}^{M} P_i u_i \tag{5}$$

Substituting for the surprisal (cf. Equation 3), we get Shannon's general formula for uncertainty:

$$H = -\sum_{i=1}^{M} P_i \log_2(P_i) \tag{6}$$

Notice that the unit for uncertainty is bits per symbol. The H function forms a symmetrical (multidimensional) curve that peaks when all symbols are equally likely and falls towards zero when one of the symbols becomes dominant.

At the start of this section, we said that information can be considered to be the decrease in uncertainty. Using Equation 6, we can express information. Information relates to communication and uncertainty as follows: Suppose we have an uncertainty H_{before} before an event (such as the transmission of a message) and that uncertainty after the event is H_{after}. Then the information that was gained in the event equals

$$R = H_{before} - H_{after} \tag{7}$$

This is what Shannon calls the *rate of information transmission*. Thus, information always relates two points in time, and the uncertainties at those times.

2.2 Uncertainty with respect to Plans

Having established a general measure of information and uncertainty in the previous subsection, how are we to apply this to planning? If we go back to the classical definition of planning, we consider a plan Δ to be a (totally-ordered) sequence of actions from a set $O = \{\overline{o}_1, \overline{o}_2, \ldots \overline{o}_m\}$ of possible actions that brings about a state change from the current state I to some goal state G. Hence, a plan is a sequence $o_1 \cdot o_2 \cdots o_n$, where $o_i \in O$ is the action executed at time point i. Under the assumption that the occurrence of each o_i is independent from the sequence $o_1 \cdot o_2 \cdots o_{i-1}$ of previous

actions, the uncertainty about which action o_j is executed at time step j, follows straightforward from Equation 6:[1]

$$-\sum_{i=1}^{|O|} P_i \log_2(P_i)$$

Here, analogously to Equation 6, P_i is the probability of action $\overline{o}_i$ being executed. Unfortunately, we cannot establish the probability of a given action. In some cases, however, we have access to previous plans, from which we can estimate the probability by the frequency F_i that an action $\overline{o}_i$ occurred in the past. This leads to the following formula for the uncertainty $H_{\overline{o}}$ regarding the execution of an action at a certain time step:

$$H_{\overline{o}} = -\sum_{i=1}^{|O|} F_i \log_2(F_i) \quad (8)$$

Notice that this is the average surprisal for encountering a certain action in any timestep. A plan consists of a number of actions, however, for successive timesteps. Because of the additive property of the uncertainty measure, we can extend Equation 8 to the uncertainty $H_{\Delta,T}$ of a a plan Δ over a certain timewindow $[1 \dots T]$ as follows:

$$\begin{aligned} H_{\Delta,T} &= T \times H_{\overline{o}} \\ &= -T \sum_{i=1}^{|O|} F_i \log_2(F_i) \end{aligned} \quad (9)$$

2.3 Privacy Loss

Now that we have defined the concept of uncertainty in the context of planning, we can define the privacy loss that agents incur by exchanging information with other agents while constructing a multiagent plan. Privacy relates to the amount of information other entities have about you. In the case of multiagent planning, it relates to the information other agents have about your plan.

Consider the case of two agents, entering into negotiations for some aspect of their planning problems. Before the negotiations, the uncertainty with regard to the other agent's plan is governed by Equation 9. During their negotiations, agents may learn about certain aspects of the other agent's plan. In particular, they may receive information about certain actions (not) being executed at a certain timepoint. Let $O_i \subseteq O$ be the set of actions that are known to be possible at timestep i, and let $F_i|_{O_i}$ be the restriction of F_i to this set O_i.[2] Then the uncertainty *after* the negotiations equals

$$H_{\Delta,T} = -\sum_{t=1}^{T} \sum_{i=1}^{|O_i|} F_i|_{O_i} \log_2(F_i|_{O_i}) \quad (10)$$

Notice how this reduces to Equation 9 when $O_i = O$ for all $1 \leq i \leq T$.

A few issues arise, however, that prevent us from simply combining Equation 7 with Equations 9 and 10. Firstly, the uncertainty with respect to a plan $H_{\Delta,T}$ depends on the length T of that plan. However, in general agents do not know the length of the other agent's plan. This gives rise to a number of different measures of privacy loss for different choices of estimating the length of the plan, as we see below. Secondly, because of synchronisation issues, agents may have to wait in their plan, not executing any actions at all. We can resolve this by introducing an additional *idle* action to the set O of allowed actions. This action has neither preconditions nor effects and is used to pad agents' plans, so that all plans of length T have T actions. We define $O^+ = O \cup \{idle\}$ to denote the expanded set of actions.

As indicated, different estimates of the plan length result in different measures of privacy loss. The basic equation from which these different measures derive is the following:

$$\begin{aligned} R(T) = &-T \sum_{i=1}^{|O^+|} F_i \log_2(F_i) + \\ &\sum_{t=1}^{T} \sum_{i=1}^{|O_i|} F_i|_{O_i} \log_2(F_i|_{O_i}) \end{aligned} \quad (11)$$

This equation is derived from Equation 7, substituting Equation 9 for H_{before} and Equation 10 for H_{after} and taking into account the *idle* action.

To define the different measures of privacy loss, consider a group of agents $A = \{a_1, a_2, \dots, a_N\}$ with respective plans Δ_{a_i} for $1 \leq i \leq N$. If we assume that the agents do not share their knowledge about other agents' plans, a loss of privacy always occurs between a pair of agents. Let the *target agent* τ be the owner of the plan Δ_τ under consideration, and the *invading agent* ι, $\iota \neq \tau$, be the agent that has gained some knowledge about Δ_τ. We can now distinguish the following measures of privacy loss:

internal The *internal privacy loss* can only be measured by the target agent. It takes into account the true length of Δ_τ: $R_{\text{internal}} = R(|\Delta_\tau|)$.

maximum The *maximum privacy loss* assumes that Δ_τ ends at the final timestep about which some information was gained : $R_{\text{maximum}} = R(\max\{i \,|\, O_i \neq O^+\})$.

estimated The *estimated privacy loss* assumes that $|\Delta_\tau| = |\Delta_\iota|$, unless information was gained about a later timestep: $R_{\text{estimated}} = R(\max\{|\Delta_\iota|\} \cup \{i \,|\, O_i \neq O^+\})$.

theoretical The *theoretical privacy loss* assumes that the plan is as long as that of the longest plan in the group of agents: $R_{\text{theoretical}} = R(max\{\Delta_a \,|\, a \in A\})$. It can only be computed by an omniscient entity.

The difference between these measures lies in the estimation of the length of the target plan. Obviously, this is an important aspect, as it has a profound impact. However, the invading agent has no way to determine this exact length, and has to make an estimate. The *internal* privacy loss is, in some sense, the *true* privacy loss, as it takes into account the actual length of the plan Δ_τ. The target agent can use this

[1] This assumption is, of course, too strong in practice. See the discussion on how to relax it.

[2] Thus, $F_i|_{O_i} = \frac{N_i}{\sum_{o_j \in O_i} N_j}$, where N_k is the number of times action o_k was encountered in the plans sampled to estimate P_i.

measure to evaluate its own plan with respect to privacy. At the other end of the scale, we have the *theoretical* loss. This is the loss over the entire horizon of the planning episode. It cannot be computed by any of the agents (as none of them know the exact lengths of the other agents' plans), but it can be used by developers to assess the privacy implications of their systems. The *maximum* and *estimated* privacy losses can be used by invading agents to estimate what information they have gained. The maximum loss takes an optimistic view, and assumes that the last timestep of which information was gained is also the last step of the target plan. The estimated loss takes a more moderate view, and uses the length of the invading agent's plan instead, if it is longer. Hence, it is an estimate of the theoretical loss, and is a more accurate measure of privacy loss over the whole episode than the maximum loss. In our next section, we show some examples of the different measures.

3. EXPERIMENTAL SETUP

To illustrate the metric that we propose, we apply it to evaluate the privacy aspect of a multiagent planner. For this analysis, we have used the results for the benchmark set of the MPOPR planning system [12, 13]. This set consists of multiagent versions of the logistics domain from the plan repair benchmark set of the GPG system [3], as well as a multiagent version of the gripper domain. We focus our attention on the latter domain, as it was constructed to investigate scalability issues and hence has a wider range of problems than the former set.

3.1 The System and the Datasets

The idea behind the MPOPR system is to combine a dynamic planning method for each agent with an auction for delegating (sub)tasks. The system consists of a number of agents that first concurrently plan for a single goal. As some goals may involve subgoals that the agent cannot achieve itself, each agent comes equipped with some high-level information about the services of others. They can use this information to reason about which subgoals they should auction. After the first planning phase, the agents take part in an auction (if there is any) to exchange some of these unattainable subgoals. Then, they apply a plan repair technique to add another goal to their plan, and take part in the next auction. They continue to alternatingly perform these steps of adapting a plan using plan repair and taking part in an auction until a complete and valid plan is computed. When an agent gets a task assigned on which others depend, a heuristic is employed that lets the agent schedule it early in its plan to prevent cyclic dependencies.

Notice that the information exchanged is very little: the auctioneer only specifies a fact that it wants to see achieved, the bidders issue a single value as their bid, and the winner communicates when it starts working on the goal and and at which point it is finished. No information is exchanged about actions that are being undertaken, as is for example done in the partial global planning approach [2]. This may lead one to assume that the system has favourable privacy properties. Indeed, this is what the authors suggest [13]: "*However, it allows us to create valid multiagent plans without exchanging details about the plans, which is very important for self-interested agents.*"

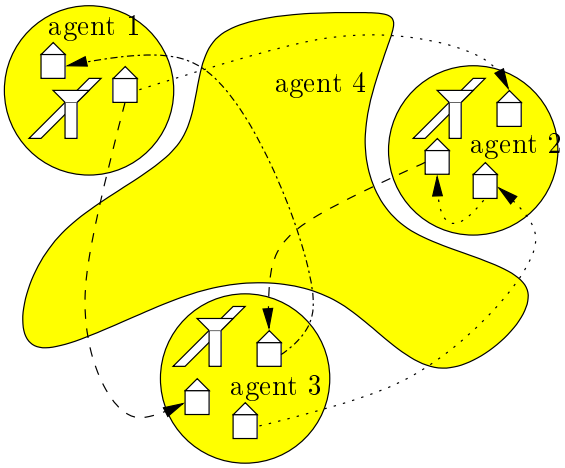

Figure 1: The multiagent logistics domain.

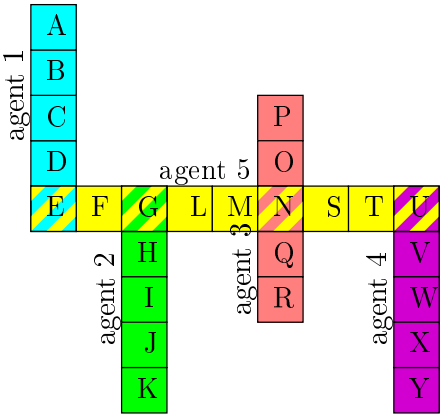

Figure 2: Graphical representation of the robots domain. Depicted are the 25 rooms, as well as the areas in which each agent may travel.

The system is evaluated on two sets of benchmark problems. In [13], the authors present a multiagent variant on the well-known logistics domain. As can be seen in Figure 1, the standard problem is divided over a number of agents: one for each city, responsible for the intra-city transport of goods, and an additional agent that handles the inter-city transport. For our analysis, we have used the B benchmark set, which consists of 45 plan repair problems.[3]

In [12], an additional domain is introduced called robots, cf. Figure 2. This is a multiagent variant on the gripper domain, in which a moving robot has to deliver objects. In the multiagent variant, each agent is confined to a single corridor and the agents have to coordinate to bring objects between corridors. This benchmark set consists of a number of random problems generated for different problem sizes (measured by the total number of goals). For our analysis, we have averaged the results over 3 problems of each size.

[3]Note, however, that we did not employ a plan repair method to solve these problems. Rather we computed solutions from scratch, without taking into consideration a previously existing plan.

3.2 Evaluation

We evaluate the privacy aspect of the MPOPR system by comparing it to a centralised generation of plans by the VHPOP [16] system.[4] To do so, we compute the information that an agent can gather from the final plan that is computed. This information is present in the form of inter-agent dependencies. Such dependencies exist when one agent achieves a subgoal for another agent. Moreover, information can be gained by relating this information. For example, in the logistics domain, if an agent is informed that particular subgoal of moving a package within a city is started at time i and finishes at time $i+2$, we can not only infer that a *load* action takes place at time i and an *unload* takes place at time $i+2$, but due to the workings of the domain, we can also deduct that a *move* action is executed at step $i+1$. In this case, the uncertainty for three timesteps of the plan is removed, which is the information we gained. For one agent, this does not hold, however: the inter-city transport agent in the logistics domain controls two airplanes. Hence, for every timestep, there is uncertainty about *two* actions. And if we look at the dependencies, we see that still a large degree of uncertainty remains. For example, if we can infer a *load* action for a certain timestep, we do not know which of the two airplanes is involved. Thus, the uncertainty in this case is only slightly reduced. A more elaborate example is given in the sidebar.

For the purpose of this evaluation, we have assumed that the probabilities of each action is equal. While this is not completely accurate (*move* actions occur slightly more often in both domains) the analysis still gives a good feeling for the privacy loss.

Logistics

Notice that in the logistics domain, interactions only occur between the inter-city agents and the intra-city agent. We therefore look at the average privacy loss that the inter-city agent incurs with respect to each of the other agents, and the average privacy loss by the intra-city agents to the inter-city agent. Figure 3 shows the *theoretical* privacy loss that the intra-city agents incur with respect to the intra-city agent for the 45 different problems in this set. As we can see, the privacy loss is worse for the MPOPR system when compared to the centralised result. The main reason for this is that the centralised approach combines more orders. The anti-cycle heuristic employed by MPOPR has the effect that its plans usually contain consecutive *load*, *drive*, and *unload* actions for tasks that it agrees to do for another agent. As discussed before, this leads to a decrease in uncertainty for three timesteps. Compare this the the central approach, which often has (also more efficient) sequences as *load*, *load*,*drive*, *unload*, *unload*. Depending on the order of the loads and unloads, the *drive* action may or may not be deduced, which on average leads to a lower loss in privacy.

Figure 4 shows a quite different picture for the inter-city agent. As we noted earlier in this section, the information that agents can gain from interacting with this agent is lower, because of the two airplanes this agent can use. As a result, the loss in privacy is very little for both methods.

[4]The reason for choosing VHPOP is that MPOPR is ultimately derived from this planner.

Example

Consider the following problem in the logistics domain: There are two cities, A and B, each with a post office po_X and an airport ap_X. Agent a_A handles the truck in city A and has a package p_1 to transport between po_A and ap_A. Agent a_B is responsible for the truck in B. It has an order to transport p_2 from po_A to po_B. A third agent a_{AB} owns the airplane to transport goods between the airports in cities. Notice that the transportation of p_2 requires all three agents to cooperate. For sake of simplicity, assume that the truck in A is located at the post office, the one in B is at the airport, and the airplane is in city A.

Central case

The central planner could produce the following plan (each action is preceded by its timestep and the executing agent in parentheses):

1 (a_A): load p_1	7 (a_{AB}): fly from A to B
2 (a_A): load p_2	8 (a_{AB}): unload p_2
3 (a_A): drive po_A to ap_A	9 (a_B): load p_2
4 (a_A): unload p_1	10 (a_B): drive ap_B to po_B
5 (a_A): unload p_2	11 (a_B): unload p_2
6 (a_{AB}): load p_2	

Agent B is informed of its plan (i.e. the final 3 actions), as well as the fact that a_A undertakes its part of the task between timesteps 2 and 5, and a_{AB} between time steps 6 and 8. From this, a_B can infer the *load* and *unload* actions in the plan of a_A. The *minimal* privacy loss is computed over the first 5 steps, as step 5 is the latest time step of which a_B learns information. The minimal privacy loss of a_A with regard to a_B is therefore $-5\log_2(\frac{1}{4}) + 3\log_2(\frac{1}{4})$, or 40% ($\frac{-5\log_2(\frac{1}{4})+3\log_2(\frac{1}{4})}{-5log_2(\frac{1}{4})}$). The *estimated* loss of a_{AB} with respect to a_B is $-11\log_2(\frac{1}{8}) + (3\log_2(\frac{1}{2}) + 8\log_2(\frac{1}{8}))$ or 18%. For 3 steps (the *load*, *fly* and *unload* actions), we now know the type of action, but not which plane. This reduces the uncertainty for these three steps from 8 possible actions (4 for each plane), to just two.

Distributed case

MPOPR produces the following plan:

1 (a_A): load p_2	8 (a_{AB}): load p_2
2 (a_A): drive po_A to ap_A	9 (a_{AB}): fly from A to B
3 (a_A): unload p_2	10 (a_{AB}): unload p_2
4 (a_A): drive ap_A to po_A	11 (a_B): load p_2
5 (a_A): load p_1	12 (a_B): drive ap_B to po_B
6 (a_A): drive po_A to ap_A	13 (a_B): unload p_2
7 (a_A): unload p_1	

The privacy losses now are as follows. Agent a_B now knows 3 out of 3 actions of agent a_A when computing the minimal privacy loss (it can infer the *move* action from the time steps of the *load* and *unload* actions): $-3\log(\frac{1}{4}) + 0$, or 100%. The estimated loss of a_{AB} is slightly less in this scenario, as the length of the plan has increased to 13: $-13\log_2(\frac{1}{8}) + (3\log_2(\frac{1}{2}) + 10\log_2(\frac{1}{8}))$, or 15%.

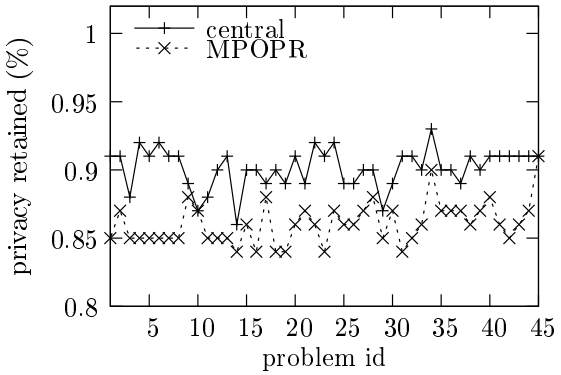

Figure 3: Average (theoretical) privacy loss of the intra-city (trucking) agents

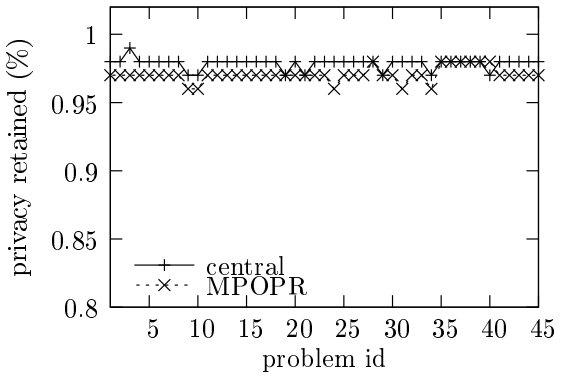

Figure 4: Average (theoretical) privacy loss of the inter-city (flying) agent

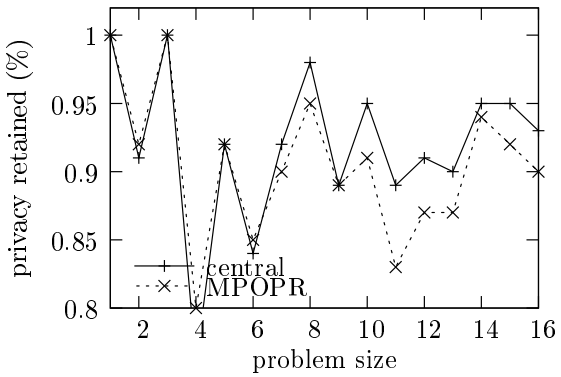

Figure 5: Average internal privacy loss agents 1-4

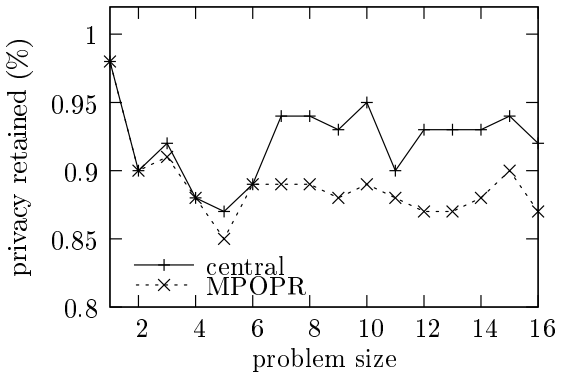

Figure 6: Average estimated privacy loss agents 1-4

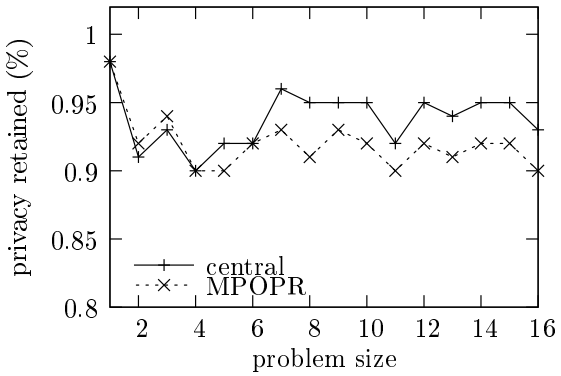

Figure 7: Average theoretical privacy loss agents 1-4

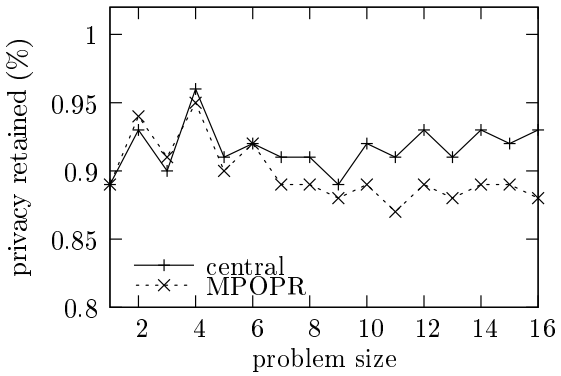

Figure 8: Average estimated privacy loss of agent 5

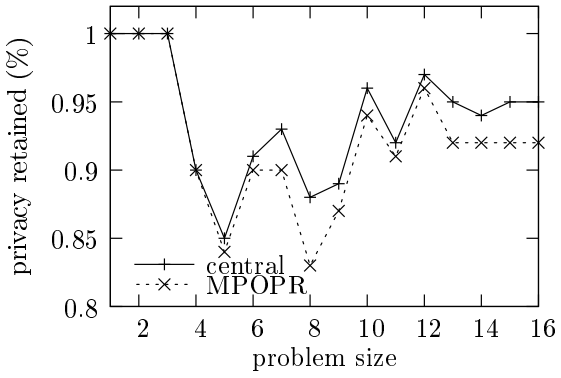

Figure 9: Average internal privacy loss of agent 5

Here, too, MPOPR fares worse than the centralised method on most problems.

Robots

In the robots domain, we also have a central agent that interacts with all other agents. The corridors all intersect with the central corridor, and with no others. Hence, we look at the agent controlling the central corridor and the average of the other agents. However, we not only look at the theoretical privacy loss, we shall also look into the internal privacy loss and the estimated loss. For this domain, the graphs display the average values obtained over 3 different problems instances of a given size (in number of goals to be achieved)

The average loss that the agents incur from interacting with the agent in the central corridor is shown in Figure 5 (internal privacy loss), Figure 6 (estimated loss) and Figure 7 (theoretical loss). Again, we observe that MPOPR often gives a greater loss in privacy than the centralised method. Interestingly, if we look at the average theoretical privacy loss (and to a lesser extent the estimated loss), this seems to remain relatively stable. We conjecture that this is due to the fact that while the plans do get bigger (as there are more goals to achieve), there are also more opportunities for learning about the other agent's actions. Together these two effects counter each other, leading to the stable behaviour we see. This behaviour is not noticeable in the internal privacy loss. Often one of the agents only has one or two packages to handle. If this involves a coordination with the central agent, almost all of the plan is exposed. However, since the central agent does not know the lengths of the plans, the estimate does not reflect this. The observer simply doesn't know that it has learned so much information.

Figures 8 and 9 show the privacy loss for the central agent. As once can see, the privacy loss for this agent is comparable to that of the other agents, although slightly lower. Since the agent interacts with many other agents they gain less information about its plan then *vice versa*. This is a matter of its plan being bigger on average.

3.3 Preliminary Conclusions

While the aim of this section is mainly to illustrate the metrics that we propose, we can draw some initial conclusions from the results. First of all, it underlines the necessity for a metric for privacy loss. A system that was designed to leak little information, turns out to be actually worse when it comes to privacy than a system in which a plan is computed centrally and then distributed. Even though we singled out the MPOPR planning system, other multiagent systems have neither undertaken a privacy analysis. Rather, they rely on enforcing certain properties that are thought to provide privacy for their users. Unfortunately, good intentions do not make for a secure system, as we have seen from this analysis.

A recommendation that can be drawn from these experiments for a privacy aware planner would be to allow for a certain degree of slack and randomisation. Another option would be the interleaving of actions. The MPOPR system enforces neither of those aspects, whereas the central planning has many tasks interleaved, which improves privacy. However, in light of the previous paragraph, we cannot simply assume that adhering to these properties would make for a secure system. A privacy analysis remains indispensable.

4. RELATED WORK

Recently, researchers in the field of Distributed Constraint Optimisation (DCOP) and Distributed Constraint Satisfaction (DisCSPs) have started to propose metrics for analysis of privacy loss in such systems. Most of the work in this area focuses on distributed meeting scheduling. In this type of problems, a number of agents has to schedule a number of meetings. Each meeting requires a certain set of agents to be present, and each agent has preferences or costs attached to timeslots and locations. Silaghi and Faltings [9] use a measure of privacy to drive their algorithm. Each agent has certain costs associated with the revelation of whether some tuple of values is feasible. During the exchange of messages, agents have to pay this cost if some tuple is fully determined by the other agents. Negotiations are terminated if the cost of revealing a certain tuple is greater than the potential reward for collaborating. A similar model is the basis of Silaghi and Mitra [10]. However, the privacy metric here is the size of the smallest coalition necessary to deduce an agent's costs for certain tuples. Wallace and Freuder [15] consider a measure of privacy loss that is very close to ours. Their work, like ours, is based on information entropy. However, the application of information theory is more straight-forward as they consider the uncertainty of each of the variables in the constraint satisfaction problem, rather than having to apply it to an additional datastructure (i.e. the plan) as we do. Recent work by Maheswaran *et al.* [4] proposes a general quantitative framework to analyse privacy loss. The three earlier approaches can be seen as specific instances of this framework.

Beyond DCOP and DisCSP, research on privacy is undertaken in the agent community. This includes work on cryptographic techniques, secure auctions and randomisation (see e.g. work by Brandt [1], Van Otterloo [14] and Naor [5]). Of particular interest to planning is the work on randomisation (e.g. Paruchi *et al.* [6] and Van Otterloo [14]). These approaches assumes that actions and behaviours can be observed. By choosing actions in a randomised fashion (e.g. using policies with a high entropy) agents can try to provide minimal information on their preferences, while still attaining their goals.

5. DISCUSSION AND EXTENSIONS

Although privacy is an issue that is often mentioned in work on multiagent planning (as well as multiagent approaches to different problems), heretofore this notion was neither made explicit, nor analysed. The present work shows how Shannon's Information Theory can be applied to (classical) planning to derive meaningful definitions of concepts such as uncertainty, information and privacy loss. As an illustration of our work, we applied it to an existing multiagent planning system to evaluate its performance with regard to privacy, compared to a central solution with a trusted third party. This clearly established the need for such an analysis: even though the multiagent planner seemingly exchanged little information, its overall performance with respect to privacy was less than that of the centralised system.

A number of extensions to this work seem obvious. Firstly, it seems reasonable to consider past actions when it comes to establishing the uncertainty of a certain timepoint. For example, in the logistics domain, we often see the following sequence of actions: *load* (a package into a truck), *move* (the truck to the package's destination) and *unload* (the package from the truck). Hence, if we know that at time t a *load* action takes place, the probability (or estimated frequency) of a *move* action at time $t+1$ is slightly higher. Work on the entropy of Markov processes can be of help here. For example, for a first-order Markov source $\mathcal{S}$ (where the probability of a symbol is dependent only upon the immediately preceding one), the entropy rate is

$$H(\mathcal{S}) = -\sum_i P_i \sum_j P_i(j) \log_2 P_i(j)$$

where $P_i(j)$ is the probability of j given that i was the preceding symbol. Such an extension would bring our work closer to the model of planning in conditional and conformant planning approaches.

Another extension is to consider more than just the type of action. Our model is limited in the sense that we do not distinguish between knowing that a *move* action takes place, and knowing the precise locations. A step in this direction was made in the *logistics* domains, where we allowed uncertainty to exist with regard to which plane was used by the intercity transport agent. However, there may be domains where such a distinction is vital and requires a more rigorous assessment then our current model allows for.

Thirdly, our work is based on the classical definition of plans. Over the years, more advanced definitions have emerged, allowing for parallel execution of actions and durative actions. Obviously, to assess the privacy issues of modern planners built on these richer formalisms, the definitions of uncertainty, information and privacy loss will have to be extended. It is not immediately clear how our work can be generalised to these situations.

Finally, this work allows new multiagent planning systems to be designed. Whereas before, privacy was mentioned as a driving force but not explicitly taken into account, the existence of a metric for privacy loss can direct research to new algorithms that are optimised for privacy. This would also entail research into the relation between privacy loss and other factors such as optimality and search efficiency.

6. REFERENCES

[1] F. Brandt. Fully private auctions in a constant number of rounds. In *Proceedings of the Seventh Annual Conference on Financial Cryptography (FC-03)*, pages 223–238, 2003.

[2] K. S. Decker and J. Li. Coordinating mutually exclusive resources using GPGP. *Autonomous Agents and Multi-Agent Systems*, 3(2):113–157, 2000.

[3] A. Gerevini and I. Serina. Fast plan adaptation through planning graphs: Local and systematic search techniques. In *Proc. of the Fifth International Conference on Artificial Intelligence Planning Systems (AIPS-00)*, pages 112–121, 2000.

[4] R. T. Maheswaran, J. P. Pearce, E. Bowring, P. Varakantham, and M. Tambe. Privacy loss in distributed constraint reasoning: A quantitative framework for analysis and its applications. *Autonomous Agents and Multi-Agent Systems*, 13(1):27–60, 2006.

[5] M. Naor, B. Pinkas, and R. Sumner. Privacy preserving auctions and mechanism design. In *Proceedings of the first ACM conference on Electronic Commerce*, pages 129–139, 1999.

[6] P. Paruchi, M. Tambe, D. Dine, S. Kraus, and F. Ordonez. Safety in multiagent systems via policy randomization. In *AAMAS workshop on Safety and Security in Multiagent Systems*, 2005.

[7] T. D. Schneider. Information theory primer, v2.60. `http://www.lecb.ncifcrf.gov/~toms/paper/primer`, 2007.

[8] C. Shannon. A mathematical theory of communication. *Bell System Tech. J.*, 27:379–423,623–656, 1948.

[9] M. Silaghi and B. Faltings. A comparison of distributed constraint satisfaction approaches with respect to privacy. In *Proceedings of the Third workshop on Distributed Constraints Reasoning (DCR-02)*, pages 147–155, 2002.

[10] M. Silaghi and D. Mitra. Distributed constraint satisfaction and optimization with privacy enforcement. In *Proc. of the 2004 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT-04)*, pages 531–535, 2004.

[11] M. Tribus. *Thermostatics and thermodynamics*. D. van Nostrand Company, Inc., Princeton, NJ, 1961.

[12] R. van der Krogt. *Plan Repair in Single-Agent and Multi-Agent Systems*. PhD thesis, Delft University of Technology, Delft, The Netherlands, 2005.

[13] R. van der Krogt and M. de Weerdt. Coordination through plan repair. In *MICAI 2005: Advances in Artificial Intelligence*, pages 264–274, 2005.

[14] S. van Otterloo. The value of privacy: optimal strategies for privacy minded agents. In *Proc. of the Fourth Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS-05)*, pages 1015–1022, 2005.

[15] R. J. Wallace and E. C. Freuder. Constraint-based reasoning and privacy/efficiency tradeoffs in multi-agent problem solving. *Artificial Intelligence*, 161:209–227, 2005.

[16] H. L. S. Younes and R. G. Simmons. VHPOP: Versatile heuristic partial order planner. *Journal of AI Research*, 20:405–430, 2003.