Research on Plan Recognition Based on Misleading Action Processing

Zengyu Cai*, Yuan Feng, Yong Gan, Ran Zhang and Jianwei Zhang

School of Computer and Communication Engineering, Zhengzhou University of Light Industry, ZhengZhou 450002, China

Abstract: As the application field of plan recognition expands continuously, the traditional plan recognition methods have hindered the application of plan recognition in hostile environment, which ignored the defects of misleading action. In this paper, a plan recognition algorithm based on the process of misleading action is proposed under the framework of plan knowledge graph. The method makes it possible to infer the action that is not observed by introducing the concept of the original reliability into the plan knowledge graph, and makes it possible to process the misleading action and wrong operation of the hostile agent by introducing the recognition process of the misleading action, which the method based on plan knowledge graph cannot achieved. The experimental results showed the algorithm in this paper greatly improved the accuracy of plan recognition by the process of the misleading action and it was with good performance. The algorithm has great significance for the application of plan recognition in the adversarial environment such as the intrusion detection, tactical planning, intelligence games etc.

Keywords: Plan recognition, adversarial, misleading action, reliability.

1. INTRODUCTION

Plan recognition refers to the process of deducing the goal of an agent or its planning process from the observed actions of an agent. It is an important branch of artificial intelligence with a wide application field. The early plan recognition approaches applied in areas like story understanding [1], psychological model [2], natural language fragments understanding [3] and intelligent computer system interface [4], etc. In these approaches, there are several typical ones. Kautz's universal framework of plan recognition was based on circumscription and minimum cardinality [5]. Charniak applied Bayesian network to plan recognition, and proposed a plan recognition model based on the Bayesian network, which made the plan recognition method have a wider range of applications [6]. Jiang proposed a plan recognition algorithm [7] based on the Plan Knowledge Graph (PKG). This plan recognition algorithm was based on PKG transforming plan problems into graph search problems. So the recognition was more efficient and the recognition results could explain the observations effectively, which were consistent with Kautz's method. However, it was more complex and couldn't predict the unobserved and future actions.

Recently, the study of plan recognition gradually focuses on applications in more complex adversarial environment. The typical ones are like robot soccer, intrusion detection, military command, intelligent game, etc. The traditional planning methods, however, were more focused on the researches on cooperative recognition and keyhole recognition and they assumed that every action performed by the agent was necessary to reach goals, namely there was no wrong action and misleading action. In addition, the agent action was completely observable and the observed action was completely reliable plan recognition. But in the adversarial field, the above assumptions aren't often valid. It's difficult for the traditional planning recognition algorithm [1-6] to meet the needs of adversarial environment. For adversarial planing recognition, as the observing agent and the observed agent are in a hostility state, the observed agent has three kinds of actions: (1) the necessary action to achieve its goal; (2) the wrong action; (3) the misleading action to let the opposition side make wrong judgments. But it is obviously unreasonable for the three kinds of actions to act as the recognition basis. For example, in the process of intrusion detection, the detection system observes several actions of the attacker, but it can't be sure that these actions all have intrusive motives. Some actions are likely to be caused by wrong operations, even the confusing actions of the attacker to mislead the detection system. For these confusing actions, appropriate methods must be taken to judge and make correct recognition.

Although there have been several researches [8-11] on the plan recognition in adversarial environment, but the researches mainly focus on the observation of the incompleteness of action and the forecast of future action. The research on the misleading action is limited to discussing whether choosing action sequence through reliability is optional planning thought [12] or not, which is not combined with the specific planning recognition algorithm. On the basis of the analysis of the plan recognition based on the PKG, the Misleading Action Processing (MAP) plan recognition algorithm is proposed in this paper. Compared with the existing methods, the new algorithm can not only deduce the unobserved action, but also can identify and delete the observed misleading action, which further improves the recognition accuracy.

The structure of the whole paper is as follows. Part 1 is a introduction. Section 2 introduces the background knowledge of the content of this paper. Section 3 presents a plan recognition algorithm based on the MAP. Section 4 presents a comparison of the experimental results with the existing algorithms. The paper concludes with a brief summary of results.

2. RESEARCH BACKGROUND

2.1. Definition of Plan and Plan Recognition

McDermott and James Hemdeler thought a plan is devising the sequence of actions for an agent [13]. We define it as a set of actions that can achieve the goals of a problem. The planning problem references a STRIPS-like domain (a set of operators), a set of objects, a set of propositions (literals) called the initial conditions and a set of problem goals which are propositions that are required to be true at the end of a plan. The planning can find a valid plan. That is a set of actions and specified time steps in which each is to be carried out. A valid plan must make all the problem goals true at the final time step. In the intrusion prevention, intelligent planning can give a reasonable response action on the basis of the targets that have been identified of the intruders and the next action protected.

Plan recognition involves inferring the intention of an agent from a set of observed actions [14]. In intrusion detection, the plan recognition can determine if there is invasion or threaten and can predict the next possible action of the attacker.

2.2. Overview of the Plan Recognition Methods

There are a lot of plan recognition application fields, the characteristics of the observed agent is not the same in different application fields. According to the characteristics and attributes of the observed agent, plan recognition can be divided into many types. The common classification methods are as follows.

(1) The function of the observed agent in plan recognition is one of the most commonly used plan recognition classification methods. According to the function of the observed agent, plan recognition can be divided into keyhole recognition, cooperative recognition and hostile recognition.

- Keyhole recognition: Agent doesn’t care or doesn't know the recognizer is watching its action. In the recognition process, agent will not provide help for the recognizer, also it won't hinder the recognition. Keyhole recognition is mainly used in areas like production monitoring, intelligent user interface, etc.
- Cooperative recognition: Agent actively cooperates with the recognition. Agent acts to make the recognizer understand. Cooperative recognition is mainly used in areas like robot soccer, story understanding, etc;
- dversarial recognition: Agent poses a threat to the recognizer and destroys the normal planning of the recognizer. And agent will also prevent or interfere with the recognition. Opponents plan recognition is applied in adversarial environments like intrusion detection, military command, etc.

(2) According to whether the plan recognition has misleading action or not, plan recognition can be divided into no misleading recognition and misleading recognition.

- No misleading planning: Each action of the Agent recognized in the process of planning is necessary to reach goals.
- Misleading planning: The Agent recognized in the process of planning carries out some wrong actions which are caused by the limited ability of the agent itself, or because the agent specifically act to interfere with the recognition.

To be convenient, most plan recognition methods at present are under the assumption that the recognition of planning is correct. This paper mainly studies misleading planning.

(3) According to whether the action sequences can be completely observed or not, plan recognition can be divided into completely observable recognition and partially observable recognition.

- Completely observable recognition: The recognizer can observe all the agent’s actions and the action execution order.
- Partially observable recognition: This may be due to the recognizer has missed some actions of the agent, or the action itself can not be observed. The recognizer can not observe all the agent’s actions. In these cases, the effect of the action is usually used to make recognition. The completely observable plan recognition is relatively simpler than partially observable plan recognition.

Usually researchers suppose that the action sequence is completely observable to reduce the difficulty of recognition. However, in real life, the action sequence can not be completely observable in many cases, especially in the cases where the recognizing party and the recognized party are in a hostile state. It’s impossible to get all the action information of the other party. Therefore, the partially observable plan recognition has higher research value. This paper belongs to the partially observable plan recognition.

2.3. Plan Recognition Methods Based on PKG

Jiang developed a new plan representation approach based on PKG [7], which transformed plan problems into graph search problems. So the recognition was more efficient and the recognition results were consistent with Kautz's method. At the same time, it could explain the observations effectively. However, it couldn’t predict the unobserved and future actions.

The PKG is an acyclic AND/OR graph G = (V, E) where V and E denote the set of vertices and edges respectively, in which vertices denote plans (events) and edges denote the supporting relation between nodes [7]. AND vertices present that they are component nodes of their parents. The children and their parents have relationships of the whole and the part which is presented by arc lines in the graph. OR vertices present that their parents and they have relationships of abstraction and specialization. All nodes are joined by edges which are used to connect the parent and its children.

Plan recognition algorithms based on PKG choose candidate plans by computing probability of every event in real world. Two kinds of data are needed in computing the probability. Firstly, the probability of every event in the real world; secondly, the probability of one event induced by another event (also called supporting degree). The supporting degree means the probability of a plan (event) induced by another plan (event). There are only two relationships between events, abstraction and specialization or whole and part. Supporting degree under the two relationships is simply prescribed as follows: the supporting degree of the appearance of specialization plan to the abstraction plan is 1; the sum supporting degree of the appearance of all part plans to the whole plan is 1. See details in [7].

3. PLAN RECOGNITION ALGORITHMS BASED ON MISLEADING ACTION PROCESSING

3.1. Basic Concepts

In order to narrate conveniently, this paper presents the following definitions:

Definition 1: Misleading action: In adversarial domains, the adversaries are aware of the presence of the recognition process. So they actively try to avoid being observed. The adversary may choose actions that are hard to detect, or add some purposeless actions to disturb the recognizer’ s observation, or itself might have wrong actions during the plan execution. All of these actions are called misleading actions. The problem of recognizing plans in such settings is termed as adversarial plan recognition. The misleading action is defined in reference [15] as that in addition to actions directly contributing to achieving a goal, and an attacker can take actions to mislead plan recognition, or to exploit some of its weaknesses.

Definition 2: Original reliability of action: The degree of an action being as the basis of recognition is called the original reliability. And it is also called action reliability. The original reliability of “action a” can be denoted by P0(a). Normally the original reliability of each action is an empirical value set according to the misleading performance of action in practical operation. It is determined by domain experts, and its essence is the probability of this action being not observed.

Definition 3: Reliability of planning or intentions: In PKG, a non-leaf node usually represents a plan or an intention. The degree of the node being as the basis of recognition is called the original reliability of a plan or an intention. And it is also called action reliability. The reliability of “plan A” can be denoted by P(A). The original reliability of a plan or an intention is determined by calculating the original reliability of each child node.

Definition 4: Support degree: The possibility of another plan B’ s occurrence after the occurrence of plan A’ s occurrence is the support degree A to B, which denoted by P’ (B/A). It is the weight of edge in PKG.

Definition 5: Direct observed action: If an original action is seen as the observed action, we called the action as “directly observed action”. And it is denoted as obs(a) equals True.

Definition 6: Direct observed plan: If one descendant node of plan A is a direct observation action, we call this plan A as “direct observation plan”. And it is denoted as obs (A) equals True.

Definition 7: Threshold: Denoted by ϕ, it is a fixed value given by a plan recognition algorithm, between 0 and 1. If the reliability of a plan or intention is less than one numerical value, it is believed that this plan or intention won’t happen. This numerical value is called threshold and is denoted by ϕ in this paper.

3.2. Changing the PKG

In this paper, it uses the PKG to store plan library, which is similar to that in reference [7]. It is also a directed acyclic AND/OR graph. There are three types of nodes, “OR”, “AND” and “LEAF” nodes which are depicted by circles, rectangles and triangles respectively in Fig. (**1**). The original reliability is added to the node to express the probability of action. These nodes have attributes including name, time-slice and reliability, which are depicted as (name, time-slice, reliability). The AND node presents the whole-part relation that its children nodes are its component events. OR node presents abstract-specific relation that its children nodes are its specialization. LEAF node corresponds directly to primitive action. AND nodes are represented by an indirect arc across the lines connecting the parent node to its children, and OR nodes do not have this arc. The leaf nodes in the tree are referred as primitive actions.

The directed edges connect child nodes to their parents. The direction of arrows is from parents to their children. There are two types of edges, “AND” and “OR”. The k-joint line points k subsequent child nodes from a parent node. It uses the k-joint line to present the whole-part relationships. The weights on the edges present the supporting degrees. For comparing with the methods in Reference [7] and [14] easily, we use the same expression with them. We assume that all the supporting degrees of a specialization node to its abstraction nodes are equal 1. It is omitted in Fig. (**1**). All of the weights in Fig. (**1**) are the supporting degrees of whole-part relationships.

In this paper, we use simple hierarchical plans, as most plan recognition work does. We assume that attackers have a plan library that provides recipes for achieving goals.

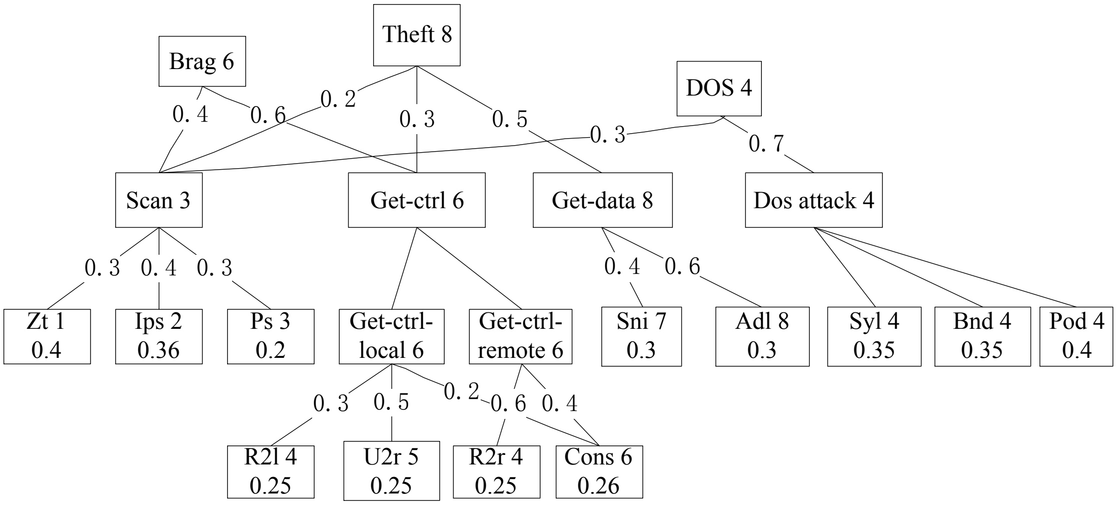

Fig. (1). The PKG of simple intrusion domain. The numbers on the nodes represent time-slice and Original reliability of those nodes.

The Probabilistic Hostile Agent Task Tracker(PHATT) [16] is based on a model of the execution of simple hierarchical plans rather than plans as formal models. This paper makes use of the PHATT plan library and predigests it simultaneously. This plan library described using changed PKG shown in Fig. (**1**).

In the library, there are three top-goals of attacker as Brag, Theft, and Dos. The Brag plan includes two steps: (scan) and (get-ctrl). The theft plan includes three steps: (scan), (get-ctrl), and (get-data). The Dos plan includes two steps: (scan) and (dos-attack). And these steps have their respective sub-plans. As space is limited, these sub-plans are not interpreted further. The orders of the four steps are partial ordered relations. Ordering constraints within a method are represented by time-slice. For example, the hacker must scan before it get-ctrl.

3.3. Supporting Degree

In PKG, it considers the influences of the children to their parents but not the parents to their children. We define the supporting degrees as follows.

(1) Effects of the parts on the whole.

In an availed plan, that if any part of a plan happens, the plan may take place. If all parts of a plan happen, the plan is sure to take place. The value of a part to its parent is determined by domain experts. We add constraints that sum of all supporting degree of part plans to the whole plan equals 1. For example, an and-node B and its n children nodes A_i, i =1, 2 , ⋯ , n, we set

$$P'(B/A_i) \leq 1 \text{ And} \qquad (1)$$

(2) Effects of the specialization on the abstraction

It is obvious that if a specialization plan happens, one of its abstract plans is sure to take place. So we define that supporting degree of specialization plans to the abstraction plan is equal 1. For example, an or-node B and its arbitrary child node Ai, we set

$$P'(B/A_i) = 1 \qquad (2)$$

3.4. Calculation of Reliability

According to the observations, we might find many plan sets satisfying the conditions, but the possibility of their occurrence in the real world is not the same. This is mainly due to that the importance of the observations in the plans satisfying the conditions is different. Due to the different importance, the support degree of the observations for the possibility of the plan' s occurrence is also different. For example, the event "**get-ctrl**" is very important in the plan "**Brag**" in Fig. (**1**). So the possibility of being **Brag** after the occurrence of the plan "**get-ctr**l" is 0.6. However, in the same event, because it is not so important in the plan "**Theft**", the possibility of the occurrence of the plan "Theft" after the occurrence of the plan "get-ctrl" is only 0.3. By using support degree to estimate the possibility of the occurrence of each plan, we can find the optimal solution. Here, the support degree refers to the possibility of one plan(event)'s occurrence making another occur.

In order to make the results more in line with the objective conditions, the provisions in this paper when calculating the possibility of the occurrence of a plan are:

(1). The possibility of the occurrence of the observed action is P(A)=1. The possibility of all the other plans "a_i" is initialized as $P0(a_i)$ which is the original reliability of this node.

(2). Plan A is a component of plan "B_i, i=1,⋯,n" and the reliability of A is P(A), the possibility of the occurrence of "B_i" under the condition is :

$$P'(B_i/A)=P(B_i)+P'(B_i/A)*P(A) \qquad (3)$$

(3). Plan B is a abstract parent of plan A.The possibility of the occurrence of plan A making plan B occur is:

$$P'(B/A)=\text{Max}\ \{P(B),\ P(A)\} \qquad (4)$$

3.5. Plan Recognition Algorithms Based on MAP

Since node deleting algorithm and MAP algorithm will be used in our plan recognition algorithm, we' ll introduce the two algorithms firstly. The function of node deleting algorithm is to judge if there is parent node of the given nodes in PKG. If there is, no deletion; if there is not, delete this node, recursively use this algorithm and delete its child node. The function of MAP algorithm is to judge if the action is misleading or ordinary according to the given nodes of PKG. And it does special operation according to the reliability and direct observations of this node. The descriptions of node deletion algorithm and MAP algorithm are as follows:

Algorithm 1: function delete node (LRS, n) // node deletion algorithm

If in LRS, n has no parent node, then delete n;

As for each child node c of n's, delete the side from n to , recursively transfer, delete (LRS, c), otherwise keep n;

Algorithm 2: function misleading action process (LRS, n): //misleading action recognition and processing algorithm

If Obs(n)=false, and $P(n)<\psi$ (ψ is threshold)，then n is misleading plan that adds to recognition algorithm, delete (LRS, n)，make deletion to m ,all child nodes of n, from n to side m, and delete (LRS, m)

If Obs(n)=true，and $P(n)<\psi$ (ψ is threshold), then n is parent node of the specific misleading action that is observed or the observed misleading action, delet (LRS, n), make deletion to m ,all child nodes of n, from n to side m, and delete (LRS, m)

If Obs(n)=false, and $P(n)<\psi$ (ψ is threshold), then it is believed that n is unobserved action and plan that already happened.

If n is or-node, then make delete (LRS, m) to the child nodes of n except all other nodes m that have the largest reliability.

If n is and-node, then make nothing.

If Obs(n)=true，and $P(n)<\psi$ (ψ is threshold)，then it is believed that n is parent node of the observed conventional action.

If n is or-node, then make delete (LRS, m) to the child nodes of n except all other nodes m that have the largest reliability.

If n is and-node, then make nothing.

With the two algorithms above, we can do the plan recognition based on MAP. The basic idea of MAP algorithm is first searching the PKG from the bottom to top according to the observed action, determine the reliability of the occurrence of each plan and action, and get the intermediate solution graph; then from the top of PKG to the bottom, delete misleading actions, and get the final solution graph. The algorithm descriptions are as follows:

Algorithm 3: misleading action recognition and processing algorithm

1. The original MSG (Medium solution graph) is empty;

2. As for all original action nodes m (leaf child node), in MSG (Medium solution graph), if m is observed action, and P(m)=1,obs (m) =true, or the rest nodes are P(m)= P0(m), obs (m) =false;

3. As for all nodes in MSG, find parent node n of all nodes from m in knowledge graph.

If n is concrete parent node of m, then P(n)=max{ P(n), P(m)};

If n is integral parent node of m, then P(n)= P(n)+ P(m)×P'（n/m);

If n is not in MSG, then Obs(n)= Obs(m), add P(n) to MSG; Otherwise Obs(n)= Obs(m) +Obs(n);

4. Act like this until to the top node, then get MSG;

5. LRS is initialized as MRS, do as follows to all nodes of LRS n from the top: misleading action process(LRS，n);

6. In the leaf nodes of LRS, if Obs(n)=false, then the action is unobserved, make P(n) equal to the probability maximum of its parent nodes.

7. Do as follows to all actions Oi of the observation action set: if Oi is not in the solution graph, the Oi is misleading action, and add it to misleading action set M.

8. Return LRS and M.

Consider the following observations: (zt, ips, sni, pod), it indicates the intrusion intentions based on PKG as Fig. (**1**) using our recognize algorithm. The MSG is gotten as Fig. (**2**).

As Fig. (**2**) shows, that the hacker is engaged in DOS has very high probability (0.93). And both Theft and Brag intention have the low probability, they are (0.52) and (0.45) respectively. These results in this step can explain observations and predict future actions. But as the misleading actions are not considered and processed, the nodes which have low probability in MSG are many. After the misleading action process in step 5, we get the final solution graph, as Fig. (**3**) shows.

The IGP is gotten as f Fig. (**3**) according IIG in Fig. (**2**). As Fig. (**3**) shows, the unobserved action or future action ps is indicated with high probability (0.76). And the misleading action sin is not in the GP. It is deleted as misleading action because its ancestor nodes' probability is all below the given threshold 0.6. So our algorithm can not only indicate unobserved actions and predict future action of attackers but also process misleading action which satisfies the requirements of network security environment. It lays a foundation for intrusion response and opposition.

4. EXPERIMENTAL RESULTS AND THE DISCUSSION

4.1. Experimental Results

We have achieved the plan recognition algorithm based on MAP using JAVA, and the test platform is Red Hat Linux 9.0. CPU is Intel Pentium processor E5300 2.6 G, 2 G memory. In this paper, the network security domain knowledge base is used as the test data. The threshold value is set as 0.60 to each field. System input is plan knowledge base for storing the invasion PKG, the events captured in network and their time step. Output is the possibility of the occurrence of various plans (events) in invasion PKG.

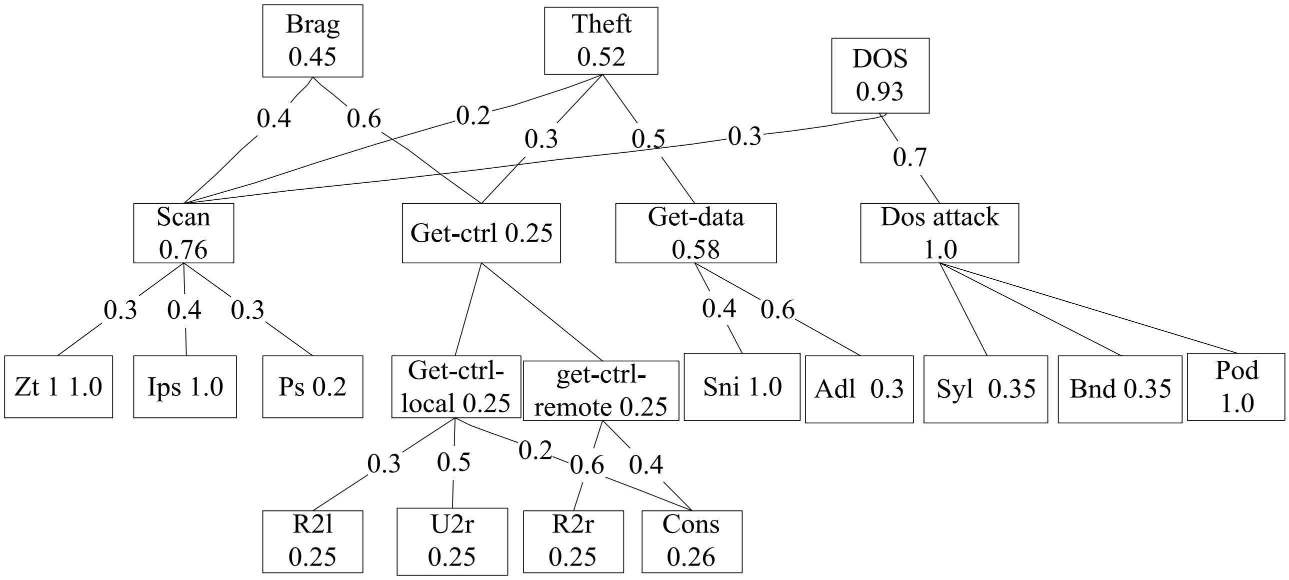

Fig. (2). The MSG of simple intrusion domain with observed actions (zt, ips, sni, pod). The numbers on the reliability of those nodes.

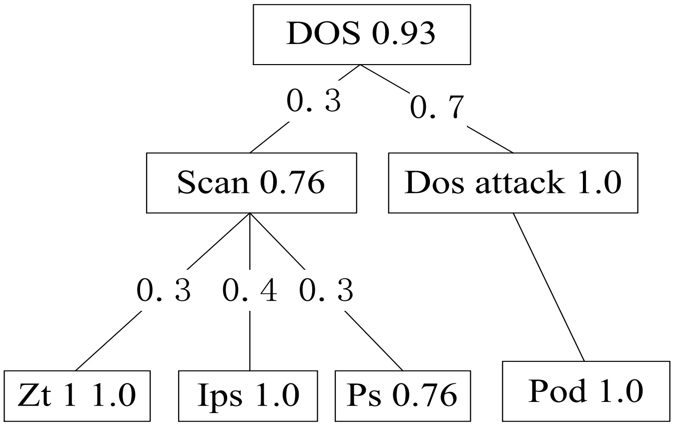

Fig. (3). The SG of simple intrusion domain with observed actions (zt, ips, sni, pod). The numbers on the reliability of those nodes.

4.1.1. Speed Test

Firstly the running speeds of the PKG algorithms, Predicting Future Actions(PFA) algorithm in [3] and MAP algorithm in this paper are tested. As the running time is too short to get the accurate running time, it got the running time by use the average value of running 10000 times. The number of observed actions is fixed as 5, and increase the number of nodes in PKG gradually. To studying the influence of the node number in PKG to recognition time, we get the experimental results as shown in Fig. (**4**). It can be seen from the Fig. (**4**) that the plan recognition time and the knowledge graph scale showed a linear relationship. Plan recognition algorithm of the PKG used the shortest time, the reason is that it only needed to search the PKG from bottom-up, but the other two need twice. So the time used by this algorithm is about half of the other two algorithms. The algorithm of this paper used a little less time compared with the PFA Algorithm in [3], which is because the PFA Algorithm need to calculate the credibility of the nodes both from bottom-up and top-down. In this paper, the possibilities of nodes are generated in once search of plan graph which reduced the workload of calculation.

4.1.2. Recognition Results

To compare results rationality of three algorithms, it uses action sequences (zp, ips, ps, pob) , (zp, ips, r2r) and (zp, ips, r2l, pod) as observed actions to do plan recognition. The results are shown in Table **1**.

(zp, ips, ps, pob) is the sequence of actions for DOS planning without misleading action and unobserved actions. In this case, the plan recognition results of the three algorithms are exactly consistent, which are DOS planning and its sub-plans including dos-attack and scan.

(zp,ips,r2r) is the incomplete action sequence of Brag with the unobserved action, yet without misleading action. In this case, the attack intention recognition results obtained by

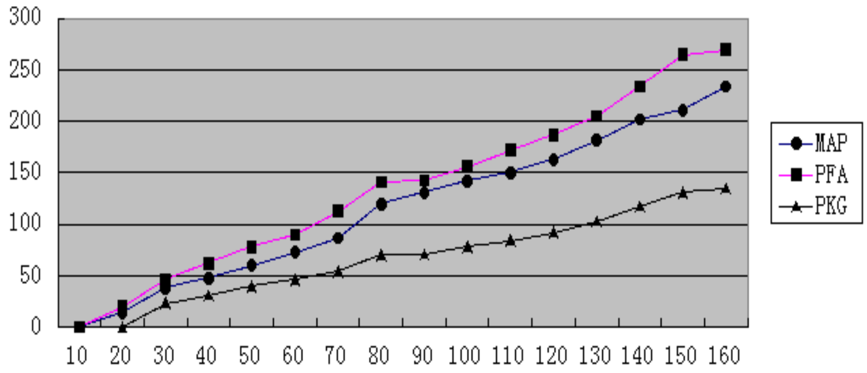

Fig. (4). Invasion recognition time-consuming experiment contrast among the PKG algorithm, PFA Algorithm and the MAP algorithm.

Table 1. Contrast Test Results (Network Security Environment).

Observed Action Sequence	Plan Recognition Results			The Recognition of Misleading Action Identification And The Speculation of Unobserved Action	
	PKG Algorithm	PFA Algorithm	MAP Algorithm	PFA Algorithm	MAP Algorithm
zp,ips,ps,pob	Dos,1 Dos-attack,1 Scan,1	Dos,1 Dos-attack,1 Scan,1	Dos,1 Dos-attack,1 Scan,1	——	——
zp,ips,r2r	Scan,0.70 Brag,0.64 Get-ctrl,0.6 Ctrl-romote,0.60	Scan,0.70 Brag,0.64 Get-ctrl,0.64 Ctrl-romote,0.64	Scan,0.76 Brag,0.72 Get-ctrl,0.70 Ctrl-romote,0.70	Unobserved Action: Ps,0.70 Cons,0.64	Unobserved Action: Ps,0.76 Cons,0.70
zp,ips,r2l,pod	Scan,0.7	Scan,0.91	Scan,0.76	Ps,0.91	Misleading action：r2l

three algorithms are identical, and the possibility of various attack intentions is approximate. The unobserved actions containing Ps and Cons can be inferred by PFA algorithm and MAP algorithm. Since Ps and Cons act as the part of the sub-actions of Scan and Ctrl-romote respectively, the part of the child node probably appears due to the possible emergence of the two plannings. However, plan recognition algorithm based on PKG lacks the capability to deal with the unobserved actions. The MAP algorithm and PFA algorithm possess the ability to infer the unobserved actions, which can be more efficient than PKG algorithm.

Among the action sequence of (zp,ips,r2l,pod), (zp,ips,pod) is the incomplete action sequence of plan Dos. So there has the unobserved actions. r21 is misleading action because it is neither related to target Dos, nor supporting other targets adequately. In this case, the attack intention recognition results obtained by three algorithms are nearly the same. PFA algorithm and MAP algorithm can infer the unobserved action like Ps quite well. The MAP algorithm succeed in identifying that r21 is misleading action. By contrast, PFA algorithm lack the capability to identify the misleading action. Therefore, the MAP algorithm has more functions than the PFA algorithm.

As is shown in the above three examples, the PFA algorithm can infer the actions that is either unobserved or future happen, and it is much better than PKG algorithm. However, MAP algorithm can distinguish the misleading actions, which has more advantages compared with PKG algorithm.

4.2. Comparison with Existing Algorithms

As an improvement to the typical plan recognition algorithm presented by Kautz, PKG algorithm overcame the shortage of Kautz' s method, simplified the recognition algorithm and introduced Bayesian theory, which could deal with probability theory. PFA algorithm further improved the plan PKG algorithm by the adding the of function of predicting future actions. MAP algorithm goes on to make a great development to PFA algorithm. In this paper, the concept of reliability is introduced into the PKG. The unobserved actions can be identified and supplemented by the calculations of values of reliability. In addition, misleading actions can be processed to make the results more reasonable. In the four algorithms mentioned above, all of them use graph structure to represent domain knowledge. There are some similarities and differences among the algorithms. Table **2** demonstrates the comparison among the four methods. It can be inferred

Table 2. Comparisons among Kautz method, PKG Algorithm, PFA algorithm and MAP algorithm.

	Kautz Method	PKG Algorithm	PFA Algorithm	MAP Algorithm
Structure of knowledge	Two special plans are introduced such as END and ANY. It uses planning hierarchical graph to store knowledge.	It uses AND/OR graph to store knowledge. The PKG is more simple and clearer compared with hierarchy plan graph.	It is similar to PKG. But it adds the support of parent node to children.	It is similar to PKG. It adds reliability to every node.
Search method	It produces one sub-graph according to each observation. Then, it checks restriction with an attempt to merge into solution graph.	It uses width-First-Search once to construct the solution graph.	It uses Bottom-up and Top-down to search PKG, for constructing solution graph.	It Bottom-up to search PKG and Top-down to process misleading action.
Conflict solution	It may occur timing constraints conflict. This situation can only be found until the final merger.	It checks the restriction on the observed results in searching procedure and can cut the plan that can't satisfy the conditions.	It checks restriction on the observed results in the first searching. In the second searching, it solves the conflict that PKG can't solve and predict future actions	It supplements either the unobserved actions or the future actions in the first searching. In the second searching, it identifies and deletes the misleading action.
Results	were solution set.	It the plan.	It, it can predict future actions.	Ite results are more reasonable.

that MAP algorithm has the simplest structure as well as most comprehensive function of plan recognition.

CONCLUSION

In this paper, plan recognition research based on MAP is presented. By introducing the concept of original reliability into PKG, the problem that PKG algorithm cannot infer the unobserved action has been solved; by introducing misleading action recognition process, the problem that PKG algorithm cannot process the misleading action of hostile agent and mistaken operation problem have been also solved. Compared to PFA algorithm, the algorithm of this paper adds the misleading action process, and can filter the misleading actions of hostile agent, which is more suitable for hostile plan recognition. The experiment result shows that this algorithm has good performance, and the MAP has greatly enhanced the accuracy of plan recognition. This algorithm is suitable for adversarial planning environment such as intrusion detection, tactical planning, intelligence games, etc. And it has a good application prospect. In this algorithm, the recognition process adopted the basic width search in searching knowledge graph. And choosing the root node in the largest scope ensured the addition of action unobserved. Combining the optimization techniques, such as heuristic and machine learning to improve the efficiency of plan recognition is the unsolved problems of this paper. Applying the results of this paper to hostile environment like intrusion detection and building recognizer that is applicable to specific areas are also the direction of our research.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGMENTS

This paper is supported by the National Science Foundation of China (61340059) and science and technology key projects of He'nan province (142102210081).

REFERENCES

[1] E. Charniak and R.P. Goldman, "A Bayesian model of plan recognition", *Artificial Intelligence*, vol. 64, no. 1, pp. 53-79, 1993.

[2] C.F. Schmidt and N.S. Sridharan, J.L. Goodson, "The plan recognition problem: an intersection of psychology and artificial intelligence" , *Artificial Intelligence*, vol. 11, no. 1, pp. 45-83, 1978.

[3] D. Litman and J. Allen, "A plan recognition model for subdialogues in conversation", *Cognitive Science*, vol. 11, no. 1, pp. 163-200, 1987.

[4] H. Karen and L. Victor, "Knowledge-Based command understanding: an example for the software development environment", *Technical Report 82-6*, Department of Computer and Information Science, University of Massachusetts, Amherst, 1982.

[5] A. Henry and A. Kautz, "*Formal theory of plan recognition*". Rochester: University of Rochester, 1987.

[6] E. Charniak and R.P. Goldman, "A Bayesian model of plan recognition", *Artificial Intelligence*, vol. 64, no. 1, pp. 53-79, 1993.

[7] Y.F. Jiang and N. Ma, "A plan recognition algorithm based on plan knowledge graph," *Journal of Software*, vol. 4, pp. 686-692, 2002.

[8] S. Braynov, "Adversarial planning and plan recognition two sides of the same coin", In: *The Second Secure Knowledge Management Workshop (SKM),* New York, 2006.

[9] R. Ramanujan, A. Sabharwal, and B. Selman, "On Adversarial Search Spaces and Sampling-Based Planning", In: *ICAPS,* 2010 vol. 10, pp. 242-245.

[10] E. Raboin, U. Kuter and D. Nau, "Adversarial planning for multi-agent pursuit-evasion games in partially observable euclidean space", In: *Eighth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2012.

[11] B. King, A. Fern, and J. Hostetler, "On Adversarial Policy Switching with Experiments in Real-Time Strategy Games", In: *ICAPS* 2013.

[12] L. Li, D. Wang, Y. Wang and W. Gu, "An approach to the misleading action solving in plan recognition", In *International Conference on Machine Learning and Cybernetics (ICMLC),* 2012, vol. 4, pp. 1285-1289.

[13] D. McDermott and J. Handler, "Planning: what is, what it could be, an introduction to the special issue on planning and scheduling," *Artificial Intelligence*, vol. 76, pp. 1-16, 1995.

[14] J. Zhang, Z. Cai, Y. Gan, B. Zhang and L. He, "A new algorithm for predicting future actions in plan recognition, "In *International Conference on Computational Intelligence and Security Workshops (CISW)*, 2007, pp. 140-143.

[15] T. Kichkaylo, T. Ryutov, M.D. Orosz and R. Neches, "Planning to discover and counteract attacks", *Informatica (Slovenia)* vol. 34, no. 2, pp. 159-168, 2010

[16] C.W. Geib and R.P. Goldman, "Plan Recognition in Intrusion Detection Systems", In *Proceedings of the Second DARPA Information Survivability Conference and Exposition (DISCEX II),* 2001, pp. 329-342.

Received: September 22, 2014 Revised: November 03, 2014 Accepted: November 06, 2014