Principal Investigator: **Lucian Vlad Lita**
Center for Automated Learning and Discovery
Carnegie Mellon University
Email: lita@cs.cmu.edu
Phone: (412) 268−1718

Planning Under Uncertainty with Abstraction
Proposal

Abstract

Large multi−agent systems face a number of intractable problems when planning in uncertain worlds. This intractability is partly due to the *single agent planning under uncertainty* problem – which is our focus. While we know that no exact algorithm for generating the optimal plan exists, this problem has been approached before heuristically with various degrees of success.

We propose to work on a planner under uncertainty based on the also well−researched technique called *abstraction*. To the author's knowledge, abstraction has never been used for planning purposes and under uncertainty in large graphs. We propose an abstraction model that reduces the complexity of the environment (graph representation). While the technique requires polynomial it has the potential to speed up the planning process and improve the planner's performance. Our proposed model looks at abstraction through graph compaction under uncertainty and attempts to simplify the graph representation of the environment while preserving the useful original information content.

1. Problem Description

1.1. Canadian Traveler Problem

We propose work on probabilistic algorithms for agent planning in the Canadian Traveler Problem (CTP). The CTP is the problem of finding a shortest path to a goal location in a graph, where individual edges of the graph might or might not be traversable [BarNoy91]. Whether or not an edge is traversable can only be found out by moving there. Hence, an optimal solution to a CTP is a contingency plan, which offers alternative routes if edges are not available. The CTP differs from other hard planning and optimization problems in that the state of the environment is only partially observable. It can be viewed as a partially observable Markov Decision Process (POMDP) with local and deterministic observations [Kaelbling97, Littman98, Blei99]. Finding an optimal contingency plan is known to be NP–hard [Blei99].

Our work will focus on an agent planner in a CTP setting, in a world, which involves multiple agents attempting to reach multiple target locations. In practice, due mostly to planning complexity, optimal solutions can only be found for very small graphs (e.g., 10 edges) and a small number of agents (e.g., 3), whereas many practical multi–agent CTP problems possess hundreds of edges and agents.

The CTP plays an important role in many practical problems. A classical example is packet routing in the Internet, where communication nodes might be up or down at random [Itai96]. Clearly, packet routing has to be performed with extremely limited computation, making extensive planning (as proposed here) infeasible. More related to the work described here is the multi–robot exploration problem [Burgard00, Simmons00, Yamauchi98] where teams of robots jointly explore an unknown environment. Here, the relative speed of the moving agents (robots) is much slower than in the packet routing problem, offering the opportunity for more extensive planning.

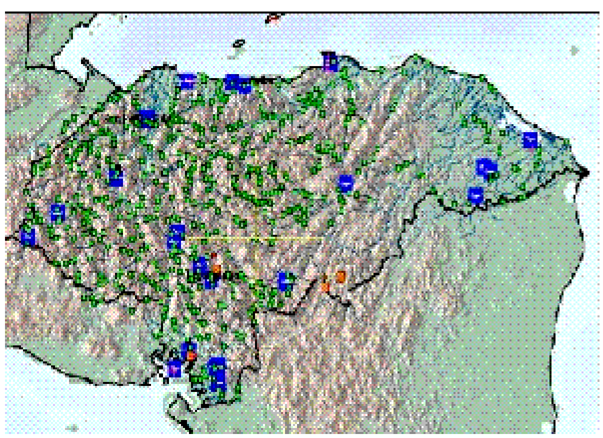

Fig. 1 Simulation of a multi–agent system during Hurricane Mitch in Honduras

The problem that motivates this research is the real life setting where agents (e.g., supply trucks) are tasked to bring emergency supplies to a number of population centers in a country devastated by a natural disaster. The road conditions are unknown. Hence, the optimal solution must offer backups from non−traversable roads. In particular, the natural disaster studied here takes place in October 1998 in Honduras, while and after Hurricane Mitch (Figure 1) destroyed up to 90% of the country's infrastructure in certain regions.

1.2. CTP Issues

Although, there exist planners that work well on the CTP problem, they are very limited [Lita01] by their heuristics and by the amount of computation power available. Their solutions vary depending on the agent location and destination (goal) positions as well as on the size and structure of the graph representation of the environment. For instance if an agent and its goal are very far apart, the plan that is generated is not going to account for many contingencies – i.e. "if this bridge is down, I will try this road". Hence, shorter distances generate better plans while longer distances generate weaker plans – plans that usually do not work close to optimum.

We would like to remedy the shortcomings described above by simplifying (abstracting) the graph representation of the world such that even if the agent is far from the goal, the abstract, simplified graph will make the planning problem more tractable. This will allow the plan to be more robust and account for more potentially wrong decisions along the way. We anticipate that this technique is particular useful when we take into account the limited computation power.

2. Goals

Given the above noted shortcomings of existing planning strategies, we propose research that will focus on creating a type of abstraction under uncertainty that would enhance planners for the CTP problem. The abstraction technique, when used in conjunction with a planner could potentially provide:

- Fast planning time – achieve the same planning accuracy but use less computation/.
- Scalability – the abstraction would allow the planner to perform well in large size environment (high number of edges).
- High quality – using abstraction the agent would be able to produce a better plan than without abstraction.
- Statistic extraction – part of the abstraction process we need to extract statistics that will preserve the original information content.
- Multi−agent integration – agents using abstraction−based planners must work well within a multi−agent framework. Besides the individual benefits that better planning performance brings, agents might be able to share the abstraction process.

As a long–term goal, depending on the degree of success with meeting the goals above, we intend to investigate (the last goal): a multi–agent system where individual agents that share the abstraction task. This might yield a much faster, potentially continuously updating distributed planner.

3. Impact

We intend to enhance planning under uncertainty through the use of abstraction. This would yield fast algorithms that would perform extensive contingency planning indirectly in an abstract domain. All agent–based applications that exist in CTP type settings would immediately benefit from this research. These types of applications include emergency vehicle networks, robot navigation, military mobile unit groups, and personal digital assistant (PDA) navigation software.

In the Honduras scenario, trucks that provide supplies to emergency areas will be able to plan more accurately how to reach their goals, thus generating plans that are closer to optimum. Without abstraction, plans do not account for critical contingencies, thus being weaker and far from optimal.

From a theoretical perspective, this research would help develop algorithms for an area not widely studied. The developed algorithms could be further applied to broader POMDP settings with more general problem definitions.

4. Previous Work

The planning problems in deterministic settings have been widely studied. Techniques, heuristics, and planners have been developed and work quite well in polynomial time, generating solutions close to optimum [McDermott78, STRIPS].

Many researchers have also approached planning under uncertainty mostly from the more general framework of Markov Decision Processes [Littman98, Kaelbling97, Blythe99, Blai99]. There are many finite horizon solutions to planning under uncertainty problems, especially POMDP style problems. However, the CTP is different since once a road condition is observed, it doesn't change over time. Also, in realistic problems the search horizon can be quite far, and the graphs quite large for current solutions to yield near–optimal results.

We are attempting to apply a probabilistic abstraction to planning. However, most of the research done in abstraction is in deterministic scenarios. Abstraction in deterministic spaces has been studied in order to simplify huge deterministic graphs in the search for ways to generate fast plans [Valtorta87, Holte94,
Sacerdoti74]. Abstraction has also been slightly addressed in planning problems [Tash 1993]

5. Approach

5.1. Abstraction

Our model proposes a planner that will perform extensive path search. It will do so not by constructing larger contingency plans, but by compacting the graph representation of the world in an abstract fashion. The algorithm will perform in polynomial time while extracting at each step relevant statistics such as: probability to find any path to reach the goal, expected time it would take the agent to reach the goal, as well as the expected reward the agent would receive upon reaching the goal.

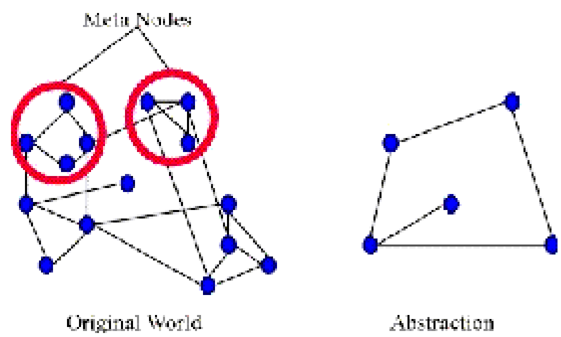

Fig. 2 The performance increases for a multi–agent system using BA*, as the number of agents grows. (performance is measured in relative time, so lower is better)

In Figure 2 a very simple graph representation of the world is presented. Since optimal planning in the original world presents computational difficulties, abstracting the graph into a clustered representation could potentially generate plans in less time. Our research will explore the different ways of abstraction in order to determine the appropriate methods for graph compaction. Initial abstraction methods will incorporate:

- distance/time information – a set of nodes with the property that all nodes in the set reach each other fast enough (e.g. short distances between cities).
- probability information – two nodes connected directly or indirectly through edges with low probability will not be likely to be removed in consecutive steps
- expected reward – combining probability and distance into an approximate expected reward (since the exact reward cannot be computed in polynomial time) will yield scores (rewards) for pairs of nodes.
- connectivity degree – nodes with high connectivity are likely not to be removed during the abstraction process.

5.2. Graph Compaction

The type of abstraction we propose is takes into account distances between nodes of the graph, as well as probabilities assigned to arcs in the graph. We are simplifying the graph according to the following algorithm:

> ***while*** *abstraction level not reached*
> *{*
> *a) remove node from graph*
> *b) reconnect and combine edges*
> *c) keep track of edge combinations through a Binary Decision Network*
> *}*

a) We choose the node to remove from the graph using on a heuristic based on arc weights, node connectivity, and probability. This is going to ensure that nodes removed are either close to a high degree node, or that they don't influence very much the high probability paths.

b) We reconnect the edges using two simple base cases, where we have two edges with probability a and b and with weights (time) A and B.

Sequential(a,b):	$P = ab$	(probability)
	$T(\text{path} \mid a\wedge b) = A+B$	(time)
Parallel(a,b):	$P = a+b-ab$	(probability)
	$T(\text{path} \mid a\|b) = aA+(1-a)b(A+B)$	(time)

Using these base cases, we merge pairs of arcs in the graph into virtual arcs in the abstract graph, thus reducing the complexity.

c) Using a directed a–cyclic structure (BDD), we maintain a list of dependencies between arcs in the graph. This way given any virtual edge, we can track its ancestors and we can recreate the formula (a series of sequential and parallel merges) that was used to compose it.

We intend to experiment with different abstraction levels in order to determine if deeper abstraction yields qualitatively good solutions. Abstraction levels are chained abstraction starting with the original world. With each abstraction, the planning complexity is reduced.

In the event that deeper abstraction is successful we also want to experiment with extreme abstraction where we use abstraction to compute statistics such as expected reward between two nodes, expected time, and probability to reach the goal from the agent location.

5.3. Evaluation

We will evaluate the performance abstraction−based planning when using different heuristics for choosing nodes and merging arcs. The simulations are going to be performed in various graphs ranging from small artificially generated graphs to large−scale graphs such as West Honduras. We are also going to compare our results to our baseline no−abstraction planning. We expect to see an increase in performance for non−trivial agent−goal cases (e.g. agent and goal are far apart). Performance will be measured through the time the agent takes to reach the goal. Abstraction heuristics to the baseline (without abstraction) will also be combined with abstraction levels (how many nodes are removed from the graph) in order to find which heuristics work best on what degrees of abstraction.

6. Results

Previous work on the CTP problem has yielded good results but our recent study [Lita01] also suggests that there is room for improvement, particularly when abstraction is concerned since it is a new way of looking at the planning problem.

6.1. Previous Results

Traditionally, planning problems in deterministic spaces involve finding a path from an agent to its goal. A commonly used algorithm is A* which uses an optimistic and thus admissible measure of goal distance to efficiently find the optimal path [Nilsson82]. In our previous work, we proposed Belief A* BA*, an algorithm based on A* which efficiently, handles planning under uncertainty. BA* searches in the space of all possible plans and produces the best contingency plan over the belief space found under time restrictions. The algorithm generates a contingency plan represented by a tree and not a by a path as in the deterministic case. For practical purposes the BA* implementation uses memoization to cut down on the planning complexity in a similar fashion to value iteration.

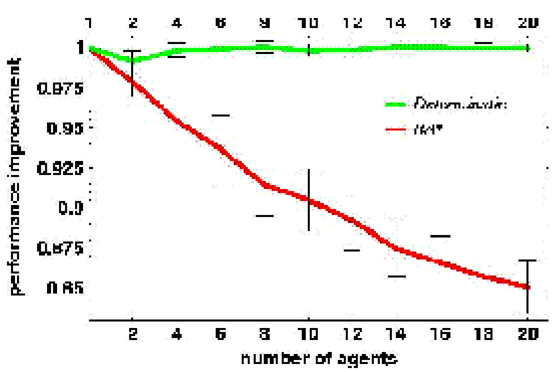

Fig. 3 The performance increases for a multi−agent system using BA*, as the number of agents grows. (performance is measured in relative time, so lower is better)

6.2. Initial Results

We have performed an initial set of evaluations and simulations, which show that very basic abstraction does not reach the performance of a system with no abstraction, but offers a tradeoff in terms of computation time. This is particularly useful when there are external computation or time limits imposed on planning. The initial results are interesting and suggest that there is much room for improvement.

7. Research Issues

There are several problems with currently developed systems. The fully observable planners eliminate from the problem complexity by assuming that the agent can obtain precise information from the environment. However we are addressing the problem where the agent deals with probabilistic statements as well as with the topology of the world. When applied to partially observable worlds it is easy to observe the pathology of the plans generated.

The existing agent−based systems, including the BA* presented earlier, can handle a CTP type setting but have major limitations. The most important limitation is scalability. These planners work well with small worlds represented by small graphs (e.g. 10 edges) but when applied to large−scale worlds, they yield inefficient solutions. Since the nature of the problem is non−polynomial, the contingency plans generated are exponential in the number of nodes and edges. Constructing such plans is time consuming, and evaluating these plans in order to generate the best one considered involves extracting and comparing statistics. We propose to solve this problem through abstraction, compacting the graph representation of the world. This proposed solution would avoid the structural complexity while preserving the relevant semantic information.

Another limitation directly related to the size of the environment is the quality of the plans. Current probabilistic planners produce contingency plans that are evaluated based on reward obtained after the plan was executed, or based on an expected reward. However, since these plans are exponential in the number of edges, the size of the generated plans grows quickly and is limited by the computation power of the engine running the planning. This implies that even with a very good heuristic, the plans cannot consider most of the good paths tot he goal, and must rely on the paths already discovered − which might be representing of the average path or not. Further, assuming that the planner produces several paths to the goal, the plans would generate long but narrow paths that don’t branch out and don’t interconnect. In a realistic world, paths would intersect and would be dependent on each other. Thus the quality of the plan varies greatly from one world and situation to another.

Our research will address that limitation but will also have to choose among abstraction and compaction methods. These abstraction methods for uncertain worlds have not been yet established, and our work will explore a wide range of graph compaction techniques based on connectivity, probability, time, and expected reward.

8. Work Plan

The time frame consists of a sequential set of steps that we need to take in order to develop the abstraction algorithms for uncertain environments:

- During the first two weeks we will implement the basic graph compaction technique.
- During the next two weeks we will experiment with different heuristics for choosing nodes to remove and ways to combine arcs based on probabilistic quantities. Also, for each technique, if it applies we will calculate the time/space complexity and evaluate the feasibility of such technique in practice.
- Another week will be spent on evaluating the new abstraction techniques on small graphs, comparing them with optimal solutions (that can be found exhaustively on these graphs).
- We will use the remainder of the time to evaluate how well planning using abstraction performs on large−scale, realistic graphs (parts of Honduras) by comparing different types heuristics. Simulating plan execution is also required to test the performance of the plans generated by abstraction.
- If time permits we will also look into the multi−agent aspect of planning with abstraction. Then we will explore a distributed abstraction/compaction strategy using a variable number of agents.

9. Summary

Deterministic planning has been studied extensively with various degrees of success. Abstraction methods have also been used on deterministic graphs in order to speed up the planning process. Planning under uncertainty has also been studied but is not feasible under certain settings (e.g. CTP), especially when the graphs are large. We intend to investigate the effects of abstraction when applied to planning in uncertain environments. We plan to do this via graph compaction and statistics extraction from the graph representation of the world.

If successful, this research would contribute to the field of planning under uncertainty with a new set of algorithms, and potentially promising results. This research could extend to systems that use general probabilistic planning and systems that plan in more general uncertain environments.

References

- Blei, D. M. and Kaelbling, L. P. (1999). *Shortest Paths in a Dynamic Uncertain Domain,* IJCAI Workshop on Adaptive Spatial Representations of Dynamic Environments.
- Burgard, W. and Fox, D. and Moors, M. and Simmons, R. and Thrun, S. (2000). *Collaborative Multi−Robot Exploration*, Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)
- Yamauchi, B. and Langley, P. and Schultz, A.C. and Grefenstette, J. and Adams, W. (1998). *Magellan: An Integrated Adaptive Architecture for Mobile Robots*, Institute for the Study of Learning and Expertise (ISLE)
- Simmons, R. and Apfelbaum, D. and Burgard, W. and Fox, D. and Moors, M. and Thrun, S. and Younes, H. (2000). *Coordination for Multi−Robot Exploration and Mapping*, Proceedings of the AAAI National Conference on Artificial Intelligence
- Matari, M. J. (1997). Reinforcement Learning in the Multi−Robot Domain, Autonomous Robots, 4, pg 73−83
- Nilsson, N. J. (1982). *Principles of Artificial Intelligence*, Springer Publisher, NY
- Bar−Noy, A. and Schieber, B. (1991). *The Canadian Traveler Problem,* Proceedings of the 2nd ACM−SIAM Symposium on Discrete Algorithms.
- Itai, A. and Shachnai, H. (1996). Adaptive Source Routing in High−Speed Networks, Journal of Algorithms, 20, pg 218−243
- Kaelbling, L.P. and Littman, M.L. and Cassandra, A.R. (1998). *Planning and Acting in Partially Observable Stochastic Domains*, Artificial Intelligence, 101:1−2, pg 99−134
- Blythe J., (1999). *An Overview of Planning Under Uncertainty*
- Tash J. K. (1993). *A Framework for Planning under Uncertainty*. AAAI Spring Symposium pp. 137−141
- McDermott D. (1978). *Planning and Acting*. Cognitive Science, 2, 1978.
- Holte R. (1994). *Searching with abstractions: A unifying framework and new high−performance algorithm.* In Proceedings of the Canadian Conference on Artificial Intelligence, 263−270.

- Lita L. V., Schulte J., and Thrun S. (2001). *A Multi–Agent System for Agent Coordination in Uncertain Environments*. Poster to appear at the 5th International Conference on Autonomous Agents.