2015

Dynamic Hierarchical Search on the GPU

Francisco M. Garcia

Norman I. Badler
University of Pennsylvania, badler@seas.upenn.edu

Mubbasir Kapadia

Presentation at Motion in Games, Paris, 2015.

Dynamic Hierarchical Search on the GPU

Dynamic Hierarchical Search on the GPU

Francisco M. Garcia[1] Norman I. Badler[2] Mubbasir Kapadia[3]
[1]University of Massachusetts - Amherst [2]University of Pennsylvania [3]Rutgers University

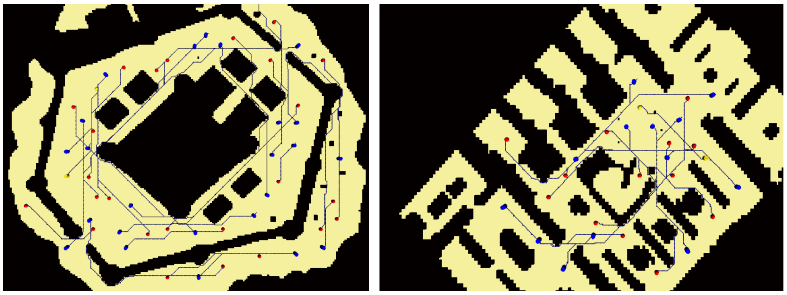

Figure 1: *Plans computed for several agents and goals on challenging benchmarks.*

1 Introduction

Although very well researched, path planning continues to be one of the most crucial aspects for robotic applications and real time simulations. To achieve a high degree of automation when exploring vast environments, it is necessary to plan in large, complex worlds. In addition, if we add the requirements of meeting real time constraints the problem becomes increasingly difficult. A common way of addressing the computation complexity of such problems is to create hierarchies or clusters, where several states are collapsed into one. By sacrificing optimality in the final plan, it is possible to search fewer states and still get a sound solution. However, such planners generally assume a static environment. When considering dynamic environments, in which the configuration of the state space may change at any time, it is necessary to efficiently reuse previous search efforts while repairing plans to accommodate dynamic changes; otherwise, real-time planning becomes unattainable.

In this work we present a new planning algorithm that combines the properties of HPA* and D* Lite, to efficiently plan in complex, dynamic environments using hierarchical abstractions of the search space, while reusing previous plans to accommodate dynamic changes in the environments. We further improve on our method by utilizing the GPU to repair plans and still meet real-time constraints in complex environments composed of tens of thousands of state. Finally, we demonstrate the benefits of our method in a multi-agent setting, in several different benchmarks.

2 Our Method

The purpose of our framework is dealing with planning problems with real-time constraints in the face of dynamic world changes and goal changes. We exploit the benefits of D* Lite [**?**], to handle changes in the environment, and HPA* [**?**], to obtain a new plan when a goal change occurs. Our algorithm creates different levels of abstraction in which smaller sub-problems, referred to as intra-edge paths, are solved and their solutions cached. When changes are observed, the intra-edge paths of the affected abstract states are first repaired, before repairing the resulting plan by using a similar approach to D* Lite. If, however, there is a change in some agent's goal, then the intra-edge paths are used to quickly generate a new solution. This approach assumes a backwards search, from goal to start state.

Several abstract states might be affected with obstacle movements, making it difficult to quickly repair intra-edge plans even by reusing previous plans. We can further improve our solution if we take advantage of the GPU. Previously, [**?**] presented a GPU approach for dynamic planning with multiple agents with independent goals. In a nutshell, the algorithm updates every state in the state space that has already been updated as well as the frontier in parallel. If we consider each entrance in an abstract state as a goal and every other entrance as an agent, we can exploit this GPU method for the preprocessing and repairing by solving several smaller sub-problems in parallel. By following this strategy, we are able to substantially improve the running time of our search method.

References

BOTEA, A., MLLER, M., AND SCHAEFFER, J. 2004. Near optimal hierarchical path-finding. *Journal of Game Development 1*, 7–28.

KAPADIA, M., GARCIA, F., BOATRIGHT, C., AND BADLER, N. 2013. Dynamic search on the GPU. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, 3332–3337.

KOENIG, S., AND LIKHACHEV, M. 2002. D*lite. In *Eighteenth National Conference on Artificial Intelligence*, American Association for Artificial Intelligence, Menlo Park, CA, USA, 476–483.