A Graph-Search Based Navigation Algorithm for Traversing A Potentially Hazardous Area with Disambiguation

Xugang Ye, Johns Hopkins University, USA
Carey E. Priebe, Johns Hopkins University, USA

ABSTRACT

The authors consider the problem of navigating an agent to safely and swiftly traverse a two dimensional terrain populated with possible hazards. Each potential hazard is marked with a probabilistic estimate of whether it is indeed true. In proximity to any of these potential hazards, the agent is able to disambiguate, at a cost, whether it is indeed true or false. The method presented in this paper is to discretize the terrain using a two dimensional grid with 8-adjacency and approximately solve the problem by dynamically searching for shortest paths using the A* algorithm in the positively weighted grid graph with changing weight function.

Keywords: A* Algorithm, Disambiguation, Mark Information, Path Planning, Shortest Path

INTRODUCTION

The problem of navigating an agent with disambiguation capability to safely and swiftly traverse a potentially hazardous area is currently receiving considerable attention due to its application to minefield. The operational concepts of minefield reconnaissance via an airborne payload and minefield path planning given the detection data were first introduced in Witherspoon et al. (1995). Most of the early research efforts were put on the mine detection part. A thorough discussion of the mine detection can be found in Holmes et al. (1995). As the output of the mine detection, the detection map consists of a collection of objects with each marked with the estimate of probability of being a true hazard.

The focus of our work presented in this paper is the navigation part. That is to navigate the agent given the detection map a priori. In this problem, it is assumed that in proximity to any of the potential hazards the agent is able

DOI: 10.4018/joris.2010070102
to disambiguate, at a cost, its true-false status. Since the agent can disambiguate, the safety is not a concern. Therefore, the navigation objective is to minimize total traversal cost, which consists of the traveling cost and the disambiguation cost.

Due to the randomness sources like the locations, the true-false status, and the detection markers, the agent’s traversal is also random. If a disambiguation is performed, the traversal is named random disambiguation walk (RDW), which is a very novel object that appears in very recent probabilistic path planning literature.

The object random disambiguation walk first appeared in Priebe et al. (2005). However, in this original study, the object was named as random disambiguation path (RDP). We rename the object as random disambiguation walk because the agent may revisit a location. If the world is a (directed) graph so that the agent only travels along the arcs, then the agent may pass some node or arc more than once. In this discrete case, the agent’s final trajectory is precisely a walk, which is defined in the most popular graph theory books (e.g., West, 2000).

Despite the name of the object, an important analytical result in Priebe et al. (2005) is that there exists a positive probability that there will be a RDW that strictly reduces the expected traversal cost compared with any deterministic shortest path that circumvents any risk. This result suggests that any relevant navigation algorithm should be able to exploit the potential benefit of disambiguation(s). Both the work in Priebe et al. (2005) and the follow-up by Fishkind et al. (2007) explored the navigation methods that yield a disambiguation-included traversal with small expected cost. However, a naïve assumption in both works is that any marker is the probability of being true hazard and more realistically model the markers as the realizations of the underlying random variables.

The up-to-date framework for solving a continuous RDW problem was summarized in Fishkind et al. (2007) as three steps. The first step is to discretize the world by constructing a directed graph. In Fishkind et al. (2007), the tangent arc graph (TAG) was built for a RDW setting that contains m disk-shape regions with centers at the detection locations and radii as the effectiveness ranges. The TAG is a precise map representation. The downside is that constructing a TAG is computationally demanding.
Building Semantic Business Process Space for Agile and Efficient Business Processes Management: Ontology-Based Approach
www.igi-global.com/chapter/building-semantic-business-process-space/64138?camid=4v1a