Alternative Pathfinding in Game Maps and Indoor Venues

Lingxiao Li and Muhammad Aamir Cheema
Faculty of Information Technology, Monash University, Melbourne, Australia
{lingxiao.li, aamir.cheema}@monash.edu

Abstract

Given a source s and a target t, alternative pathfinding aims to return a set of k alternative paths from s to t such that these paths are short, meaningful (e.g., no un-necessary detours), and significantly different from each other. While alternative pathfinding in road networks has received significant attention, to the best of our knowledge, it has not been studied in maps with obstacles such as game maps and indoor venues, e.g., airport, shopping center etc. Furthermore, it is not clear whether the techniques designed for road networks generate high-quality alternative paths for game maps and indoor venues. To this end, we present a web-based demonstration system that visualises the alternative paths in game maps and indoor venues generated by three of the most popular techniques originally designed for the road networks. This system will help evaluating these techniques and identifying potential limitations that must be addressed for better alternative pathfinding in game maps and indoor venues.

Introduction

Given a source s and a target t, a shortest path query returns the path from s to t with the minimum total cost (e.g., length, travel time etc.). The shortest path query has received huge research attention (see (Li et al. 2017) and references therein) due to its applications in various domains such as in road networks, social networks, game maps and indoor venues etc. In many cases, the shortest path may not meet a user's needs and it may be desirable to return several alternative paths so that the user has more options to choose their preferred path. For this reason, many modern map-based systems (e.g., Google Maps) present to the users a set of alternative paths from s to t instead of only the shortest path. In this paper, we consider the problem of alternative pathfinding which is to return k alternative paths (including the shortest path) from s to t. These alternative paths must be short, significantly different from each other and meaningful/natural (e.g., should not have un-necessary detours etc.). Note that returning the k shortest paths is not a good solution as these paths are likely to be very similar to each other (i.e., high overlap) and may contain needless detours.

Finding alternative paths in road networks has received huge research attention in the past few years, e.g., see (Chondrogiannis et al. 2020) and references therein. Given that there is no agreed definition of what constitutes a "good" set of alternative paths (Li et al. 2021) mainly because it is subjective to the users, many different techniques have been proposed that aim to compute high-quality alternative paths. A recent user study (Li et al. 2021) conducted on the road networks of Melbourne, Dhaka and Copenhagen compared some of the most notable approaches including Penalty (Akgün, Erkut, and Batta 2000), Disimilarity (Chondrogiannis et al. 2020) and Plateaus (Jones 2012) and reported that the alternative paths produced by these three approaches are perceived by the users to be of similar quality as those generated by Google Maps, arguably the most popular and widely used map-based service.

There is no existing work that aims to find alternative paths in the maps where movement in a Euclidean space is constrained by a set of obstacles (e.g., game maps or indoor venues). Also, it is not clear if the above mentioned techniques designed for road networks are suitable for such maps. Inspired by this, in this paper, we introduce a web-based demonstration system[1] that allows visualising alternative paths produced by Penalty, Dissimilarity and Plateaus on any map from a widely used set of benchmark maps.

System Description

Algorithms and Implementation Details

We implement Penalty, Plateaus and Dissimilarity which were originally designed for road network graphs but can be immediately applied on the visibility graphs used for pathfinding in game maps or indoor venues. Since a user may only be interested in alternative paths that are not significantly longer than the shortest path, we introduce a parameter $\alpha > 1$ and only report the alternative paths that have length at most $L \times \alpha$ where L is the length of the shortest path. Since the paths that are *non-taut* are not meaningful, we adopt existing techniques such that they discard the non-taut paths. Hereafter, we say that an alternative path is invalid if it is longer than $L \times \alpha$ or it is non-taut.

Penalty. This approach iteratively computes the shortest path from s to t and, after each iteration, it applies a penalty

[1]A video describing the algorithms and the demonstration system is available at https://www.bilibili.com/video/BV1P64y1C7zr and https://youtu.be/-Q74GYFZnBI

on each edge of the shortest path by increasing its edge weight (e.g., multiplying its current edge weight with a penalty factor greater than 1). As the edge weights on the shortest path found in the previous iteration are increased, it is likely that a significantly different shortest path will be found in the next iteration. The algorithm terminates: 1) when k valid paths are found; **or** 2) after n iterations where n is the number of vertices in the map.

Plateaus. This approach first creates two shortest path trees: T_s, rooted at the source s; T_t, rooted at the target t. Then, T_s and T_t are *joined* to obtain the common branches in the two trees. These common branches are called plateaus. Given a plateau with two end points u and v where u is the end closer to s and v is the end closer to t, an alternative path using this plateau is generated as $sp(s,u) \bigoplus sp(u,v) \bigoplus sp(v,t)$ where $sp(x,y)$ denotes the shortest path from x to y and $\bigoplus$ is the concatenation operation. Longer plateaus are expected to generate more meaningful alternative paths in road networks (Jones 2012). Therefore, the algorithm iteratively selects the plateaus in descending order of their lengths and generates an alternative path for each selected plateau. The algorithm terminates when the list of plateaus exhausts or when k valid paths have been generated.

Dissimilarity. This approach uses a dissimilarity function $dis(p_1, p_2)$ which measures the dissimilarity between two paths p_1 and p_2. We use the Jaccard distance as a measure of dissimilarity, i.e., $dis(p_1, p_2) = 1 - |p_1 \cap p_2| / |p_1 \cup p_2|$ where $|p_1 \cap p_2|$ (resp. $|p_1 \cup p_2|$) is the total weight of the edges common in (resp. in the union of) the two paths. This approach aims to iteratively access candidate paths in ascending order of their lengths and each candidate path p to the result set P if its dissimilarity to all paths in the current result set is at least equal to a user-defined threshold θ.

In our implementation, the penalty factor in Penalty is 1.4 and the dissimilarity threshold θ is 0.5 (these values were found to be the most reasonable among the values we tried). For a more detailed description of the above techniques, please see (Li et al. 2021).

User Interface

Our web-based system is publicly available on the internet.[2] A user can select any map from the widely used repository[3] of game and indoor maps, a value of k from 1 to 5, one of the algorithms, and a value of α ranging from 1.05 to 5. The user can click anywhere on the map to select a source s and a target t. When both s and t are selected, the system displays up to k alternative paths generated by the selected algorithm (see Fig. 1). Using the drop down menus, the user can zoom in/out, choose a different algorithm, and change the values of k and α to obtain the updated alternative paths for the same s and t. The alternative paths are shown in different colors along with their lengths (in ascending order).

Conclusion and Future Work

In this paper, we present our web-based demonstration system that displays up to k alternative paths in game and indoor maps. The system can be used to compare the quality of alternative paths generated by three of the most popular alternative pathfinding techniques designed for road networks. An important direction for future work is to conduct a comparative study to evaluate not only the path quality of these techniques but also their running time and memory requirements etc. Also, efficient algorithms need to be designed to quickly compute high-quality alternative paths. Finally, it is important to investigate whether there are other possibly better techniques for alternative pathfinding in such maps.

[2] http://www.aamircheema.com/alternative-demo/

[3] https://movingai.com/benchmarks/

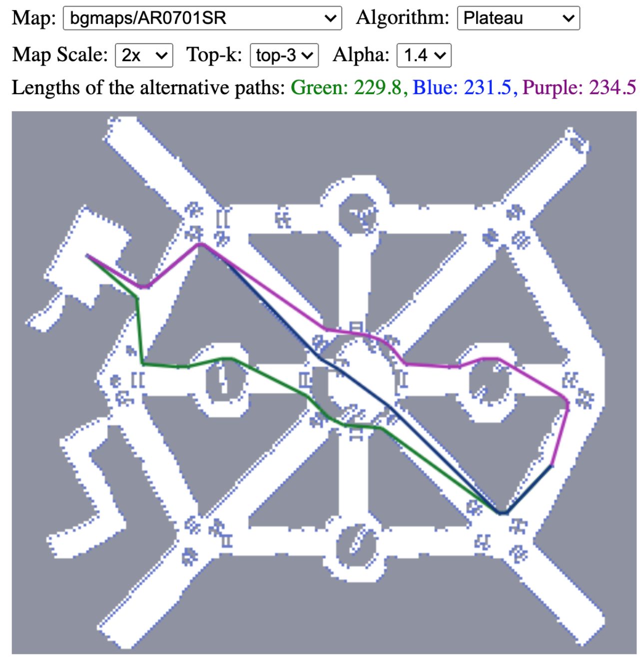

Figure 1: Three alternative paths reported by Plateaus

Acknowledgements

This work was supported by the Australian Research Council (ARC) FT180100140.

References

Akgün, V.; Erkut, E.; and Batta, R. 2000. On finding dissimilar paths. *European Journal of Operational Research* 121(2): 232–246.

Chondrogiannis, T.; Bouros, P.; Gamper, J.; Leser, U.; and Blumenthal, D. B. 2020. Finding k-shortest paths with limited overlap. *The VLDB Journal* 1–25.

Jones, A. H. 2012. Method of and apparatus for generating routes. US Patent 8,249,810.

Li, L.; Cheema, M. A.; Lu, H.; Ali, M. E.; and Toosi, A. N. 2021. Comparing alternative route planning techniques: A comparative user study on Melbourne, Dhaka and Copenhagen road networks. *IEEE TKDE* .

Li, Y.; U, L. H.; Yiu, M. L.; and Kou, N. M. 2017. An Experimental Study on Hub Labeling based Shortest Path Algorithms. *PVLDB* 11(4): 445–457.