

A Consistent Heuristic for Efficient Path Planning on Mobility Maps

Wanmai Yuan^{†‡}, Nuwan Ganganath[‡], Chi-Tsun Cheng^{†*}, Guo Qing[‡], and Francis C.M. Lau^{†‡}

[†]Dept. of Electronic and Information Eng., The Hong Kong Polytechnic University, Hung Hom, Hong Kong

[‡]Hong Kong Applied Science and Technology Research Institute, Hong Kong Science Park, Sha Tin, Hong Kong

[‡]School of Electronics and Information Eng., Harbin Institute of Technology, Harbin, Heilongjiang, P.R.China

[‡]The Hong Kong Polytechnic University Shenzhen Research Institute, Shenzhen, Guangdong, P.R.China

*Email: chi-tsun.cheng@polyu.edu.hk

Abstract—Efficient path planning has an utmost importance in the domain of autonomous navigation. Even though shortest path planning has been well discussed in the past, the same techniques might not always be used to find the fastest path in outdoor environments due to the inability of mobile agents to travel at their peak speed everywhere in irregular terrains. Mobility maps are an effective way for dealing with such irregularities. In this paper, we first introduce a grid-based mobility maps for representing speed limitations in outdoor terrains. Then, we propose a heuristic for finding the fastest path on such maps. The proposed heuristic is proven to be both admissible and consistent. Therefore, it can be used with A*-like heuristic search algorithms for obtaining fastest paths efficiently. Simulation results provided in this paper verify the optimality of paths that are found with the help of the proposed heuristic.

Index Terms—Heuristics, consistent, admissible, fastest path, mobility maps.

I. INTRODUCTION

Autonomous navigation has been a common interest of both the research community and the general public for many decades [1]–[3]. It has drawn even more attention in recent years with the emergence of self-driving cars [4], outdoor mobile robots [5], and planetary rovers [6], [7]. Path planning is one of the crucial components in autonomous navigation, which can be described as a process of finding a path between two given points such that one or more objectives are satisfied. In indoor environments which mostly consist of flat and uniform surfaces, the typical objective of path planning in literature was minimizing the path length [8]–[10]. This is a reasonable objective since the path length can be considered to be proportional to the travel time on such uniform surfaces. However, such an assumption can often be invalid in outdoor path planning. Outdoor terrains are usually uneven and their surface condition can vary both in space and time without any obvious pattern. Therefore, outdoor terrain path planning can be more challenging compared to its indoor counterpart.

Graph search algorithms have often been adapted for both indoor and outdoor path planning [11]–[15]. In these applications, an operational terrain of a mobile agent (e.g. a vehicle, a mobile robot, etc.) is represented using a graph that consists of a set of nodes and a set of edges. A node represents a spacial location on the terrain surface and an edge represents the connection between two nodes. An edge is associated

with one or more costs. For examples, the edge cost is a distance measurement when finding the shortest path and a time measurement when finding the fastest path. Early works on graph search algorithms are based on uniform cost search mechanisms. Dijkstra’s algorithm [12] is a prime example for such uniform cost search algorithms. It starts a search process from a source node and iteratively selects a node for expansion until it selects a target node for expansion. Here, node selection is based on the cost between the source node and a given node without considering its remaining cost to the target node. Thus, in finding the shortest path, it expands all the nodes that are closer to the source node compared to the target node. Obviously, this results in excessive expansion of nodes that do not lie on the optimal path, thus, degrading the efficiency.

Heuristic search algorithms alleviate this problem by considering the remaining distance from a particular node to the target node in their search processes. However, the efficiency of a search process and the optimality of solutions obtained from a heuristic search algorithm depend on the quality of its heuristic cost estimations. A*-like heuristic search algorithms [15]–[17] can be used to efficiently find shortest paths in outdoor environments when consistent heuristics are available. However, shortest paths in uneven terrains may not necessarily be the fastest as it is difficult to maintain a constant speed on an irregular terrain. A mobility map [18] can represent such irregularities by dividing the terrain into areal terrain patches according to the maximum achievable speeds of the agent in those patches. As the main contribution in this paper, we propose an admissible and consistent heuristic which can be used with A*-like heuristic search algorithms for finding a fastest path between two given points on grid-based mobility maps of outdoor terrains.

The rest of the paper is organized as follows. In Section II, concepts of mobility maps and mobility profiles are discussed, followed by an introduction to the graphical representation of mobility maps. Such section also includes a brief review on basic properties of heuristics. A time heuristic is proposed for finding the fastest path. Its admissibility and consistency are theoretically analyzed in Section III. A set of simulations were conducted using grid-based mobility maps. Simulation results are presented and discussed in Section IV. Concluding results are given in Section V.

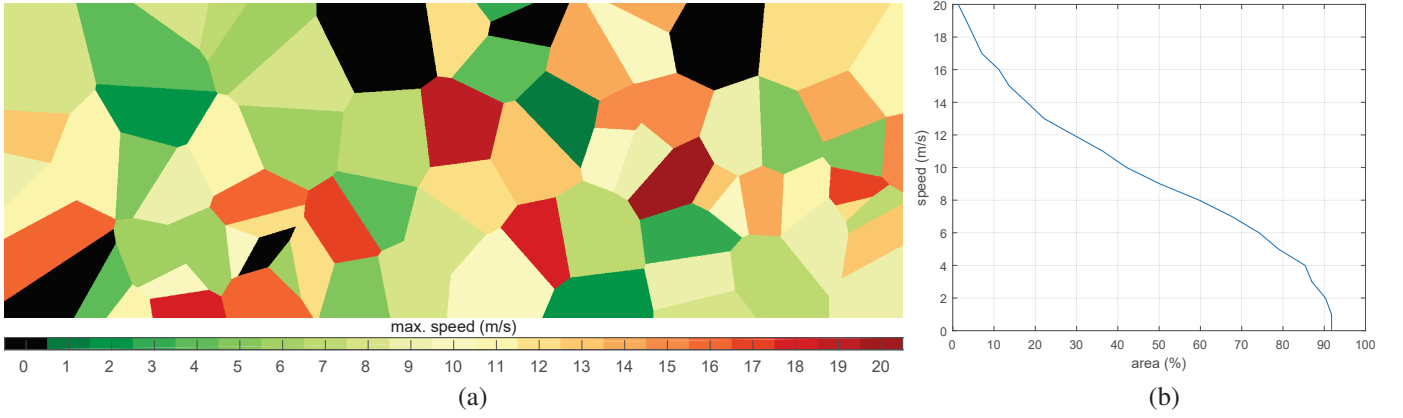


Fig. 1. (a) A mobility map of an arbitrary mobile agent and (b) its mobility profile.

II. PRELIMINARIES

A. Grid-based Mobility Maps

The maximum feasible speed between two given points on a given terrain has been used to characterize mobile agents [19]. The maximum feasible speed of a mobile agent within a particular operational environment is not only a result of the characteristics of the mobile agent and the environment, but also the interactions between them. The maximum speed on an areal terrain patch is constrained by one or more of following factors [18]:

- The availability of traction to overcome resistances due to terrain roughness, obstacles, vegetation, slope, etc.
- The maneuverability of the mobile agent in avoiding obstacles.
- The speed adjustments of the mobile agent to avoid obstacles.
- The operator's tolerance to the discomfort while driving on rough terrains.
- The operator's willingness to drive faster than a speed threshold when a limited visual distance is given.

Note that the last two factors may not be directly relevant to autonomous mobile agents, but they shall still be considered as their safety criteria.

Based on aforementioned factors, the maximum feasible speed within an areal terrain patch can be estimated. The estimated results for a mobile agent in an operational environment can be represented in form of a mobility map [18]. Fig. 1(a) shows such a mobility map for an arbitrary mobile agent. The total region of interest is characterized by 20 different speed limits. The zero-speed patches can be considered as the obstructed or prohibited regions in the operational environment. It is assumed that each areal terrain patch is sufficiently uniform to maintain the maximum speed in a straight line motion. A mobility map of an agent is very useful for finding fastest paths between two locations in its environment. The information available in a mobility map can be summarized using a mobility profile. Fig. 1(b) illustrates the mobility profile of the mobility map given in Fig. 1(a). According to the given mobility profile, nearly 60% of the

region of interest can be traversed at 8 m/s. The mobility profile of an agent is very useful in its operational planning [18].

A mobility map is a combination of many areal terrain patches as shown in Fig. 1(a). For the ease of computations, such a map can be further discretized to obtain a grid-based mobility map in which each cell is a small square patch. The maximum feasible speeds within these square patches are inherited from the co-located patches in the original mobility map. If a square patch shares parts of more than one patch in the original mobility map, the maximum feasible speed of the patch with largest share is copied as that of the square patch. In case of a tie, such value is arbitrarily selected from one of the overlapping patches in the original mobility map. The final result of this procedure is a grid-based mobility map. The accuracy of such a representation increases with the grid resolution.

B. Graphical Representation

Once a grid-based mobility map is obtained, it is transformed into a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. Here, $\mathcal{V}$ is a set of nodes that represents the center points of cells in a grid-based mobility map. Each node $n \in \mathcal{V}$ connects upto 8 other nodes that represent the neighbouring cells in the grid-based mobility map. The set that contains all these connections is denoted by $\mathcal{E}$. Each connection $(n, n') \in \mathcal{E}$ is associated with some cost $c(n, n')$. Here, we consider two different costs: distance cost $c_d(n, n')$ and time cost $c_t(n, n')$. The distance cost can be obtained as

$$c_d(n, n') = d(n, n'),$$

where $d(n, n')$ is the Euclidean distance between n and n' . The time cost can be obtained as

$$c_t(n, n') = \frac{d(n, n')}{v_{\text{avg}}},$$

where v_{avg} is the average maximum feasible speed between n and n' . It is assumed that a mobile agent always travels at the maximum feasible speed of the areal terrain patch it is currently located and can adjust its speed accordingly as it enters a new patch.

C. Heuristics and Their Basic Properties

Given a source node $s \in \mathcal{V}$ and a target node $t \in \mathcal{V}$, here we consider the problem of finding a fastest feasible path from s to t using A*-like heuristic search algorithms. Such algorithms initiate from s and propagate along the graph until it reaches t if possible. During its traversal from s to t , A* creates a search tree of path segments, in which leaves are recorded on an OPEN list. It selects the next node to be expanded from OPEN such that

$$f(n) = g(n) + h(n),$$

is minimized. Here, $g(n)$ is the cost of a path on the search tree from the source node s to a given node n and $h(n)$ is a heuristic function that estimates the cost of a path from n to t . The behavior of a heuristic search process and the properties of its search results depend on the nature of its heuristics. They are briefly reviewed here, more details can be found in [14], [17], [20].

Definition 1: Let $h^*(n)$ be the cost of the optimal path from n to t . A heuristic function h is said to be admissible if

$$h(n) \leq h^*(n), \quad \forall n.$$

Definition 2: An algorithm is said to be admissible if it guarantees to find an optimal solution for any given problem when a solution exists.

In general, a heuristic function for A* is regarded as desirable based on two basic principles: avoid expanding any node which is surely not on an optimal path and avoid ignoring any node that might be on an optimal path. A* expands the least number of nodes to find an optimal path if $h(n) = h^*(n)$ for all n . When $h(n) = 0$, A* is equivalent to Dijkstra's algorithm which exhausts all possible solutions. Nevertheless, a practical approach would be to select something in between these two extremes. When A* is guided by admissible heuristics, it guarantees to return an optimal solution if a solution exists. Thus, A* is admissible if $h(n) \leq h^*(n)$ for all n . When $h(n) > h^*(n)$, A* overestimates the expected cost of a path through n . That can result in missing potential optimal solutions.

A selection of a heuristic function affects not only the quality of solutions found by A*, but also the efficiency of its search.

Definition 3: A heuristic function h is said to be *more informed* than another heuristic function h' if both are admissible and satisfy

$$h'(n) < h(n),$$

for every non-target node n .

A* can find an equally optimal solution by expanding lesser number of nodes when its search is guided by more informed heuristics.

Definition 4: Let $k(n, n')$ be the cost of an optimal path from a node n to a node n' on a given graph. A heuristic function h is said to be consistent if

$$h(n) \leq k(n, n') + h(n'), \quad \forall n, n'.$$

When A* is guided by consistent heuristics, it guarantees to find optimal paths to all expanded nodes. Time complexities of heuristic search algorithms can be measured in terms of the number of node expansion operations. It has been shown that no other algorithm, which is guided by equally informed heuristics, is capable of finding an optimal solution without expanding all the nodes that are explored by A* [20].

III. THE PROPOSED TIME HEURISTIC AND ITS THEORETICAL ANALYSIS

As the main contribution of this work, we propose a consistent heuristic function that estimates the time cost of a path from n to t . The proposed heuristic function is given as

$$h_t(n) = \frac{d(n, t)}{v_{\max}},$$

where $v_{\max}$ is the globally maximum speed in a mobility map of an operational environment.

Theorem 1: The heuristic function h_t is admissible.

Proof: Let $h_t^*(n)$ be the shortest possible travel time from n to t . Obviously, $h_t(n) \leq h_t^*(n)$ for all $n \in \mathcal{G}$. According to *Definition 1*, the theorem is proved. ■

Theorem 2: The heuristic function h_t is consistent.

Proof: Let $k_t(n, n')$ be the shortest possible travel time from n to n' . Hence, we have

$$\begin{aligned} k_t(n, n') &\geq \frac{d(n, n')}{v_{\max}}, \quad \forall n, n' \in \mathcal{G}, \\ k_t(n, n') + \frac{d(n', t)}{v_{\max}} &\geq \frac{d(n, n')}{v_{\max}} + \frac{d(n', t)}{v_{\max}} \end{aligned} \quad (1)$$

The triangular inequality can be stated as

$$d(n, n') + d(n', t) \geq d(n, t) \quad (2)$$

From (1) and (2), we have

$$\begin{aligned} \frac{d(n, t)}{v_{\max}} &\leq k_t(n, n') + \frac{d(n', t)}{v_{\max}} \\ h_t(n) &\leq k_t(n, n') + h_t(n'), \quad \forall n, n' \in \mathcal{G}. \end{aligned}$$

According to *Definition 4*, the theorem is proved. ■

With proven consistency, the proposed time heuristic can be used with heuristic search algorithms to find fastest paths on grid-based mobility maps.

IV. SIMULATIONS

A. Simulation Setup

In order to evaluate path planning performances of A* search algorithm with the proposed time heuristic, a set of simulations was performed using grid-based mobility maps introduced in Section II-A. All mobility maps used in the simulations were 200×200 m² squares with 200 cells in each side of the map. That accounts for a total of 40,000 cells, i.e. there are 40,000 nodes in a resultant graph. Each mobility map was randomly partitioned into 30 areal patches based on the concept of Voronoi diagrams. Some randomly selected patches were defined as obstructed regions. The rest of the

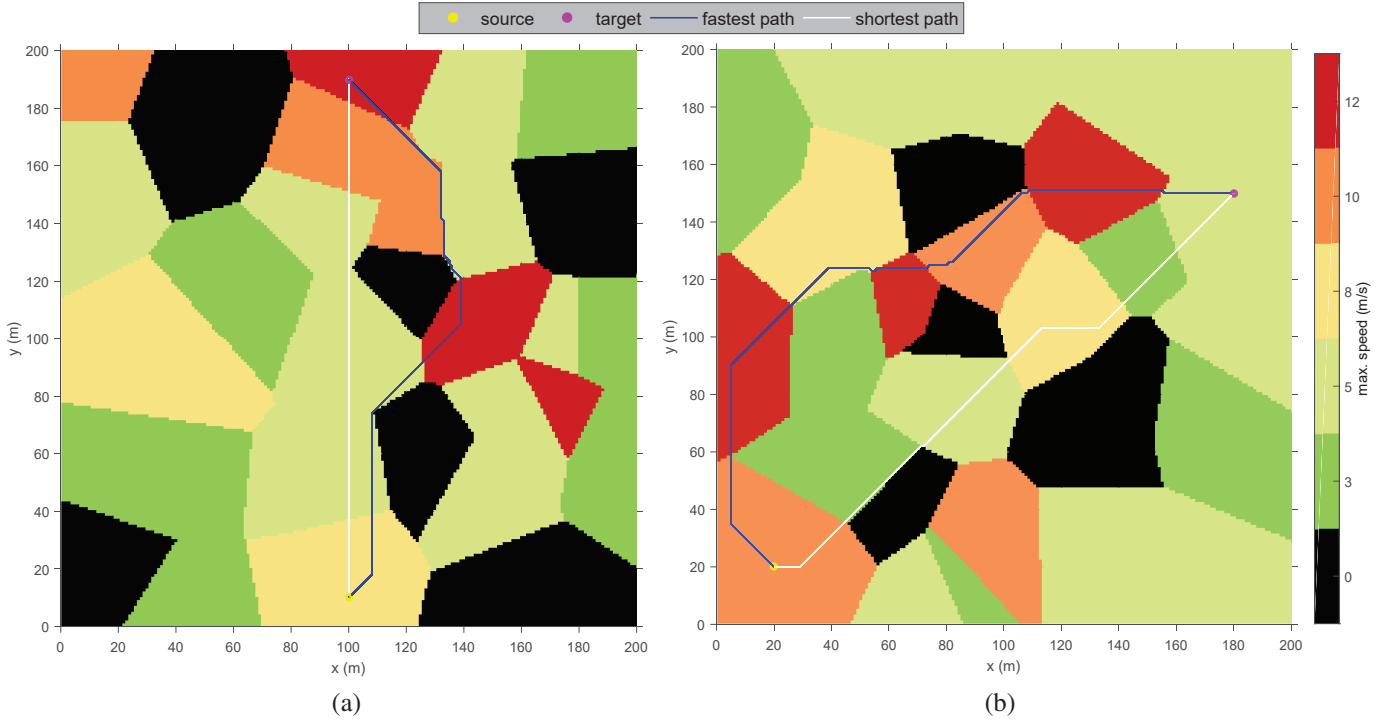


Fig. 2. Shortest and fastest paths obtained in (a) Simulation I and (b) Simulation II, respectively.

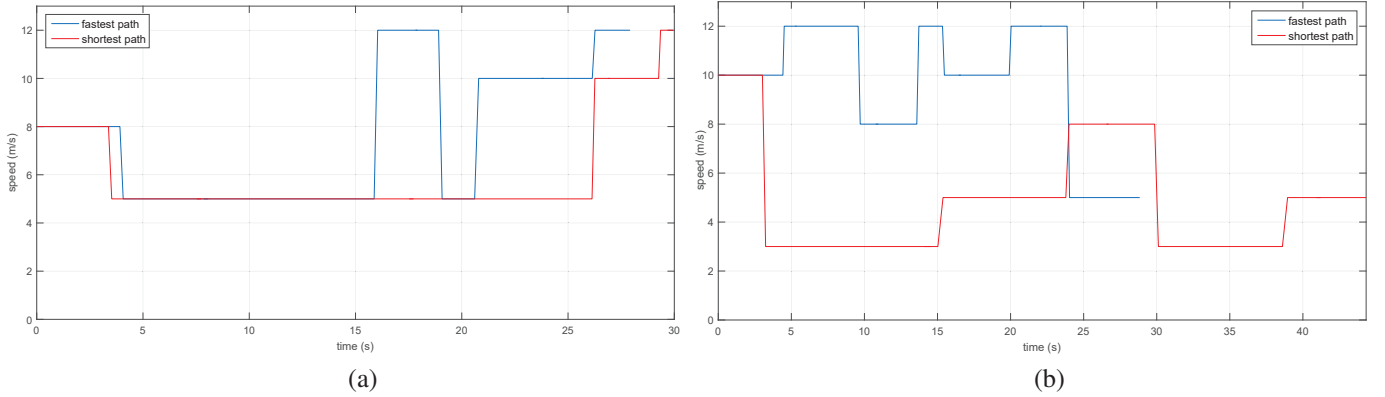


Fig. 3. Speed profiles of mobile agents that follow the shortest and fastest paths in (a) Simulation I and (b) Simulation II, respectively.

patches were randomly assigned with 5 different maximum feasible speed values selected from a range of $[3, 12]$ m/s. Dijkstra's algorithm was utilized to generate reference paths as it guarantees to find optimal solutions if they exist. All simulations were performed using MATLAB on a computer with Intel Core i5-6200U processor, 16GB memory, and Windows 10 operating system.

B. Simulation Results

In the first set of simulations, the source and target locations of the mobile agent were respectively selected as $(100, 10)$ m and $(100, 190)$ m. The shortest and fastest paths obtained using Dijkstra's algorithm are shown in Fig. 2(a). Note that the shortest and the fastest paths are clearly different from each other. Speed profiles of the mobile agents that follow those paths are shown in Fig. 3(a). The mobile agent that followed

the fastest path has reached the target in 27.9185 s whereas the agent that followed the shortest path has taken 29.9364 s. Next, A* with the proposed time heuristic was utilized to search for a fastest path. The obtained path coincides with the path obtained using Dijkstra's algorithm. This confirms that the A* is capable of finding fastest paths on grid-based mobility maps when guided by the proposed time heuristic h_t . According to the results given in Table I, Dijkstra's algorithm has expanded 26406 nodes in its search for the fastest path whereas A* with h_t has found the exact same path by just expanding 14751 nodes.

In the next set of simulations, the source and target locations of the mobile agent were respectively selected as $(20, 20)$ m and $(180, 150)$ m. The shortest and fastest paths obtained using Dijkstra's algorithm are shown in Fig. 2(b). Again, the

TABLE I
ANALYSES ON THE NUMBER OF NODES EXPANDED WHILE
SEARCHING FOR THE FASTEST PATH.

Simulation	Travel time (s)		Number of nodes expanded	
	Dijkstra's	A* with h_t	Dijkstra's	A* with h_t
I	27.9185	27.9185	26406	14751
II	28.8348	28.8348	29394	20356

shortest and the fastest paths are different from each other, which confirms with our earlier statement that the shortest path may not necessarily be the fastest. Speed profiles of the mobile agents that follow those paths are shown in Fig. 3(b). The mobile agent that followed the fastest path has reached the target in 28.8348 s whereas the agent that followed the shortest path has taken 44.3263 s. Similar to the previous case, A* with h_t found the exact fastest path as Dijkstra's algorithm did. Nevertheless, Dijkstra's algorithm has expanded 29394 nodes in its search for the fastest path whereas A* with h_t has expanded just 20356 nodes for finding the same path.

C. Discussion

According to the results explained in the previous section, it is obvious that A* is capable of finding the fastest path efficiently when its search is guided by the proposed time heuristic. However, h_t may not be the most informed time heuristic for finding the fastest path on grid-based mobility maps. If one can find a time heuristic which is more informed than h_t , that would guide A*-like search algorithms to find the fastest path even more efficiently.

Even though paths found using both A* and Dijkstra's algorithm are optimal on grid-based mobility maps, they are not necessarily optimal on the original mobility maps. This is a result of limiting navigational directions to only 8 options by overlaying the grid layout. Such a drawback can be alleviated by using finer grids or any angle path planning algorithms [21].

V. CONCLUSION

Mobility maps are commonly used for indicating maximum feasible speeds of mobile agents traversing in different regions of an operational environment. In this paper, we introduced a grid-based mobility map which paves the way to use graph search algorithms for path planning on mobility maps. Furthermore, as the main contribution of this paper, we proposed a time heuristic for finding the fastest path on mobility maps efficiently. The proposed heuristic is proven to be both admissible and consistent. Therefore, when an A*-like search algorithm is guided by the proposed heuristic, it guarantees to find an optimal solution if such a solution exists. Simulation results provided in this paper verify this claim. The proposed time heuristic can be used for mobile agent path planning in many outdoor applications such as search and rescue, firefighting, military, and exploration.

ACKNOWLEDGMENT

This work is supported by the Department of Electronic and Information Engineering, the Hong Kong Polytechnic University (Projects RUWM and G-YBKH) and NSF of China (Grant No. 61372095).

REFERENCES

- [1] M. A. Turk, D. G. Morgenthaler, K. D. Greban, and M. Marra, "VITS-A vision system for autonomous land vehicle navigation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 3, pp. 342–361, 1988.
- [2] J. Guivant, E. Nebot, and S. Baiker, "Autonomous navigation and map building using laser range sensors in outdoor applications," *Journal of robotic systems*, vol. 17, no. 10, pp. 565–583, 2000.
- [3] N. Ganganath, M. Walker, and H. Leung, "Fuzzy cognitive map based situation assessment framework for navigation goal detection," in *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2013, pp. 1444–1449.
- [4] L. Gomes, "When will Google's self-driving car really be ready? It depends on where you live and what you mean by 'ready'," *IEEE Spectrum*, vol. 53, no. 5, pp. 13–14, 2016.
- [5] K. Tanaka, H. Ishii, D. Kuroiwa, Y. Okamoto, E. Mossor, H. Sugita, Q. Shi, S. Okabayashi, Y. Sugahara, and A. Takanishi, "A novel approach to increase the locomotion performance of mobile robots in fields with tall grasses," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 122–129, 2016.
- [6] Z. Sun, Y. Jia, and H. Zhang, "Technological advancements and promotion roles of Chang'e-3 lunar probe mission," *Science China Technological Sciences*, vol. 56, no. 11, pp. 2702–2708, 2013.
- [7] G. V. Levin, "The curiousness of Curiosity," *Astrobiology*, vol. 15, no. 2, pp. 101–103, 2015.
- [8] L. J. Guibas and J. Hershberger, "Optimal shortest path queries in a simple polygon," *Journal of Computer and System Sciences*, vol. 39, no. 2, pp. 126–152, 1989.
- [9] Y. Li, J.-H. Park, and B.-S. Shin, "A shortest path planning algorithm for cloud computing environment based on multi-access point topology analysis for complex indoor spaces," *The Journal of Supercomputing*, pp. 1–14, 2016.
- [10] Z. Shao, J. B. Lee, D. Taniar, and Y. Bo, "A real time system for indoor shortest path query with indexed indoor datasets," in *Australasian Database Conference*. Springer, 2016, pp. 440–443.
- [11] N. Ganganath, C.-T. Cheng, and C. K. Tse, "Rapidly replanning A*," in *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*. IEEE, 2016, pp. 386–389.
- [12] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
- [13] M. Likhachev, D. Ferguson, G. Gordon, A. Stentz, and S. Thrun, "Anytime search in dynamic graphs," *Artificial Intelligence*, vol. 172, no. 14, pp. 1613 – 1643, 2008.
- [14] N. G. Marasinghe Arachchige, "Heuristic search algorithms with applications to path planning on uneven terrains," Ph.D. dissertation, The Hong Kong Polytechnic University, 2016.
- [15] N. Ganganath, C.-T. Cheng, and C. K. Tse, "A constraint-aware heuristic path planner for finding energy-efficient paths on uneven terrains," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 601–611, 2015.
- [16] S. Koenig and M. Likhachev, "Fast replanning for navigation in unknown terrain," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 354–363, 2005.
- [17] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *Systems Science and Cybernetics, IEEE Transactions on*, vol. 4, no. 2, pp. 100–107, 1968.
- [18] J. Y. Wong, *Theory of ground vehicles*, 3rd ed. John Wiley & Sons, 2001.
- [19] C. J. Nuttall, A. A. Rula, and H. Dugoff, "Computer model for comprehensive evaluation of cross-country vehicle mobility," SAE, Tech. Rep. 740426, 1974.
- [20] J. Pearl, *Heuristics: intelligent search strategies for computer problem solving*. Addison-Wesley Pub. Co., Inc., Reading, MA, 1984, ch. Formal properties of heuristic methods, pp. 73–85.
- [21] A. Nash and S. Koenig, "Any-angle path planning," *AI Magazine*, vol. 34, no. 4, pp. 85–107, 2013.