This is the Pre-Published Version.

An Improved Dynamic Z* Algorithm for Rapid Replanning of Energy-Efficient Paths

Nuwan Ganganath†, Chi-Tsun Cheng, and Chi K. Tse
Department of Electronic and Information Engineering
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
†Email: nuwan.marasinghearachchige@connect.polyu.hk

***Abstract*—Recently proposed Dynamic Z* heuristic search algorithm enables mobile robots navigating on uneven terrains to replan energy-efficient paths using previous search results whenever an obstacle is encountered. This paper proposes some vital improvements to Dynamic Z* by optimizing the use of previous search results in replanning. Simulation results show that the proposed improvements can significantly reduce the computational cost of replanning in Dynamic Z* while producing equally energy-efficient paths.**

***Index Terms*—Mobile robot, path planning, replanning, energy-efficient, uneven terrain.**

I. Introduction

Mobile robots have been commonly utilized in outdoor environments where they have to navigate on uneven terrains to complete their operations [1]–[3]. Once their current and target locations are determined, efficient algorithms for path planning and motion control are needed to yield a successful navigation. Shortest paths have been popular in mobile robotics applications since they often enable robots to reach their targets quickly [4]–[6]. However, shortest paths on uneven terrains can be energy inefficient for mobile robots. Some of them can even be unrealizable due to physical constraints such as motion power limitations of the robots and their instability on steep slopes. Therefore, growing attentions have been devoted for finding physically feasible energy-efficient paths for mobile robots. One may refer to [7] for a recent review on such methods.

Using the energy-cost model introduced in [8], Ganganath et al. proposed a heuristic function to calculate the heuristic energy-cost between two given locations on an uneven terrain [9]. The proposed heuristic function uses zig-zag like virtual path segments to generate finite heuristic energy-costs on steep hills. With the proven admissibility and consistency of the proposed heuristic energy-cost function, they introduced Z* heuristic search algorithm [7] which can provide energy-optimal paths on gird-based elevation maps. Interestingly, the resultant paths of Z* on steep hills consist of zigzag-like path segments which can be realized under physical constraints of robots. Even though Z* is capable of finding physically feasible energy-efficient paths on uneven terrains, it needs to have complete information about the terrains including obstacle locations in order to provide such paths. Unfortunately, most real-world outdoor environments are highly unpredictable and obstacle locations can change with time. Initial paths generated by Z* cannot be realizable if obstacles coincide with those paths afterwards.

As a solution for energy-efficient path planning problems in dynamic outdoor environments, Dynamic Z* heuristic search algorithm was proposed in [10]. In contrast to Z*, Dynamic Z* plans its initial paths from goal nodes to start nodes. If the robot detects an obstacle on its current path, it replans its path using previous search results. In [10], it was shown that Dynamic Z* can be much faster in replanning in compared with Z*. We have recently discovered that the computational efficiency of Dynamic Z* algorithm can be further improved by optimizing the use of previous search results in replanning. Hence, this paper proposes an improved Dynamic Z* algorithm which is capable of rapid replanning of physically feasible energy-efficient paths for mobile robots navigating on uneven terrains.

The rest of the paper is organized as follows. Section II revisits some background materials on energy-efficient path planning using grid-based elevation maps. The proposed improvements to Dynamic Z* are explained in Section III. Simulation results of Dynamic Z* with the proposed improvements are presented and compared with its predecessor in Section IV. Some concluding remarks are given in Section V.

II. Background

Uneven terrain surfaces can be represented by grid-based elevation maps for computational purposes [7], [11]. In energy-efficient path planning, such maps are transformed into weighted graphs whose nodes represent physical locations on the terrain surfaces and edge costs represent energy-costs of traversing between nodes [7]. Let n_c and n_n be the nodes representing an arbitrary node and its neighboring node, respectively. In this work, we consider graphs with 8-connected neighborhoods. The energy-cost of a mobile robot with mass m traveling from n_c to n_n under gravitational force strength g and friction coefficient μ is given by

$$k(n_\mathrm{c}, n_\mathrm{n}) = \begin{cases} \infty, & \text{if } \phi(n_\mathrm{c}, n_\mathrm{n}) > \phi_\mathrm{m} \\ mgs(n_\mathrm{c}, n_\mathrm{n})(\mu \cos\phi(n_\mathrm{c}, n_\mathrm{n}) + \sin\phi(n_\mathrm{c}, n_\mathrm{n})), & \text{if } \phi_\mathrm{m} \geq \phi(n_\mathrm{c}, n_\mathrm{n}) > \phi_\mathrm{b} \\ 0, & \text{otherwise}, \end{cases} \tag{1}$$

where $s(n_c, n_n)$ and $\phi(n_c, n_n)$ are Euclidean distance and elevation angle between n_c and n_n, respectively [7]. Here, ϕ_m and ϕ_b denote critical impermissible and critical breaking angles, respectively.

Let n_s and n_g be the nodes representing start and goal locations of a robot, respectively. The energy-cost of traversing from n_s to n_g via n_c is defined as $f(n_c) = g(n_s, n_c) + h(n_c, n_g)$. Here, $g(n_s, n_c)$ is the energy-cost of traveling from n_s to n_c which can be calculated using (1) for each intermediate path segments. The heuristic energy-cost is defined by

$$h(n_c, n_g) = \begin{cases} \frac{mg\Delta(n_c,n_g)}{\sin\phi_m}(\mu\cos\phi_m + \sin\phi_m), & \text{if } \phi(n_c, n_g) > \phi_m \\ mgs(n_c, n_g)(\mu\cos\phi(n_c, n_g) + \sin\phi(n_c, n_g)), & \text{if } \phi_m \geq \phi(n_c, n_g) > \phi_b \\ 0, & \text{otherwise,} \end{cases}$$

where $\Delta(n_c, n_g)$ is the elevation of n_g with respect to n_c [7]. Z* finds energy-efficient paths on elevation maps based on the search procedure introduced by Hart et al. [12]. It starts from n_s and tries to minimize $f(n_c)$ in each iteration on its way to n_g. The iterative procedure is terminated when $n_c = n_g$.

In contrast, the search procedure of Dynamic Z* is directed from n_g to the current robot location n_r. However, energy-costs are calculated in the same direction as Z* does. In the initial planning, Dynamic Z* assumes that all nodes with unknown traversability status are traversable. It manages two sets of nodes, namely OPEN set and CLOSED set, to keep track of its search history. Dynamic Z* starts by initializing OPEN set to n_g and CLOSED set to an empty set. In each iteration, a node in OPEN set with minimum f-cost, say n_c, is removed from OPEN set and added to CLOSED set. Then f-cost of all neighbors of n_c are calculated and added to OPEN set. The initial planning completes when $n_c = n_r$. After that the robot starts following that path to reach n_g. However, if the robot encounters an obstacle on its current path, Dynamic Z* replans another energy-efficient path to n_g using its previous search results.

While replanning, Dynamic Z* first removes all nodes from CLOSED set, whose f-costs are greater than or equal to the obstructed node. Then all the leaf nodes in the remaining search tree are moved to OPEN set and their f-costs are updated according to the current robot location. However, this method of redefining OPEN and CLOSED sets discards some useful search results from the previous planning which ultimately results in degrading the computational efficiency of Dynamic Z*.

III. Proposed Improvements to Dynamic Z*

Once a path is obstructed, the branch that is extending from the obstructed node of the previous search tree becomes invalid. Based on this hypothesis, we propose a novel method for rearranging OPEN and CLOSED sets for replanning of energy-efficient paths. The improved version is based on Dynamic Z* algorithm proposed in [10] with a modified REFRESH_SETS function to introduce a better management of previous search

Algorithm 1: Pseudocode of REFRESH_SETS function

1: **function** REFRESH_SETS()
2: CLOSED $\leftarrow$ CLOSED $\setminus \{previous[n_r]\}$
3: TEMP $\leftarrow \{previous[n_r]\}$
4: **while** TEMP $\neq \emptyset$ **do**
5: $n \in$ TEMP
6: **for** $\forall n_n \in \{neighbor[n]\}$ **do**
7: **if** $previous[n_n] == n$ **then**
8: **if** $n_n \in$ OPEN **then**
9: OPEN $\leftarrow$ OPEN $\setminus \{n_n\}$
10: **else if** $n_n \in$ CLOSED **then**
11: CLOSED $\leftarrow$ CLOSED $\setminus \{n_n\}$
12: **end if**
13: TEMP $\leftarrow$ TEMP $\cup \{n_n\}$
14: **else if** $previous[n_n] \neq n$ **and** $n_n \in$ CLOSED **then**
15: CLOSED $\leftarrow$ CLOSED $\setminus \{n_n\}$
16: OPEN $\leftarrow$ OPEN $\cup \{n_n\}$
17: **end if**
18: **end for**
19: TEMP $\leftarrow$ TEMP $\setminus \{n\}$
20: **end while**
21: **for** $\forall n \in$ OPEN **do**
22: $f[n] \leftarrow g[n] + h(n_r, n)$
23: **end for**
24: **end function**

results. Replanning techniques used in shortest path planning cannot be directly applied to energy-efficient path planning because the heuristics are not backward consistent [13].

The modified REFRESH_SETS function is given in Algorithm 1. It starts by removing the obstructed node from the CLOSED set {2}. (Numbers in the curly braces refer to line numbers in Algorithm 1.) That node is then used to initialize a new set called TEMP set {3}. The objective of using TEMP set is to track all nodes who have the obstructed node as an ancestor in the search tree. REFRESH_SETS function iteratively processes all nodes in TEMP set {4-20}. The iterative procedure starts by arbitrarily selecting a node n in TEMP set {5}. Then it processes all the neighboring nodes n_n of n, maximum 8 nodes in this implementation, and check whether n is the parent of n_n {6-18}. If that condition is satisfied and n_n belongs to OPEN or CLOSED sets, it will be removed from the corresponding set {8-12} and added to TEMP set {13}. If n_n is in CLOSED set but not a child node of n, then it is removed from CLOSED set and added to OPEN set {14-16}. At the end of this iterative procedure, it is guaranteed that the obsolete branch started from the obstructed node no longer exists in the search tree. The updated OPEN set represents new leaf nodes of the search tree.

The heuristic energy-costs (h-costs) of the nodes in OPEN set are outdated since the robot has moved to a new location. Therefore, h-costs of all nodes in the OPEN set need to be updated according to the current robot location {21-23}. As a consequence, f-costs need to be updated as well. Note that g-costs remain same since they represent the energy-cost of reaching n_g from a given node, which is independent of n_r. Nevertheless, energy-costs of nodes in CLOSED set need not to be updated because they do not contribute in the replanning

TABLE I: SIMULATION RESULTS.

Simulation	n_s (m)	n_g (m)	Total energy-cost (J)		# of nodes visited during initial planning		# of nodes visited during replanning	
			Dynamic Z*	Dynamic Z* with RR	Dynamic Z*	Dynamic Z* with RR	Dynamic Z*	Dynamic Z* with RR
I	(64,13)	(40,92)	447.5753	447.5753	1249	1249	2625	44
II	(30,14)	(71,88)	258.7391	258.7391	6172	6172	1688	219
III	(92,44)	(5,45)	6097.5644	6097.5644	2262	2262	2649	77
IV	(90,14)	(27,69)	5343.3769	5343.3769	7156	7156	1185	142

process. The hierarchical structure of nodes in CLOSED set remains same as they are locally consistent despite of the new robot location.

IV. SIMULATIONS

In [10], it has been shown that Dynamic Z* can provide physically feasible energy-efficient paths which associate with same energy-cost as the paths obtained by repeatedly applying Z*. Furthermore, Dynamic Z* has managed to visit a lower number of nodes in replanning by using its previous search results. In order to evaluate the modified Dynamic Z* with rapid replanning (RR) against its predecessor in [10], we conducted a set of computer simulations using a terrain model which can be given by

$$z(x,y) = 4.726\left[\sin\left(\frac{y}{3\pi}\right) - \cos\left(\frac{x}{3\pi}\right) - 0.3\sin\left(3\sqrt{\left(\frac{x}{3\pi}\right)^2 + \left(\frac{y}{3\pi}\right)^2}\right)\right]^2.$$

The base of the terrain is a square shaped grid map with an area of 100×100 m^2 and 100 grids each side. The mass, velocity, and maximum motion power of the simulated model of the robot utilized in this work are $m = 25$ kg, $v = 0.5$ ms^{-1}, and $P_{\text{max}} = 100$ W, respectively. Throughout all the simulations, following parameters remained fixed: $\mu = 0.01$, $\mu_s = 1.0$, and $g = 9.81$ ms^{-2}.

In each simulation, n_s and n_g are carefully selected such that they represent different elevations on the terrain surface for a fair evaluation. In all the simulations, obstacles are generated uniformly at random such that obstacles cover 10% of the total area of terrain base. Initially, both path planners under test only have information about the terrain elevation, but not the obstacle distribution. Therefore, the initial paths are planned assuming that the environment is obstacle-free. As the robot follows the generated paths, it can sense the surrounding environment and detect obstacles if there is any. Simulation results of the path planners under test are given in TABLE I and paths generated are shown on the terrain maps in Fig. 1.

According the the results given in TABLE I, both path planners under test have found equally energy-efficient paths in all simulations. Therefore, we can confirm that the proposed modifications to Dynamic Z* do not degrade the energy-efficiency of its paths. During initial plannings, both algorithms have visited same number of nodes. This is as expected as initial planning steps are identical in both algorithms. Interestingly, the proposed modifications to Dynamic Z* are able to significantly reduce the number of node visits in replanning. This has been possible because the proposed modifications enable Dynamic Z* to make maximum use of the previous search results by carefully removing only the obsolete branches from the search tree. Therefore, the modified Dynamic Z* has to expand a minimum number of nodes to find an energy-efficient path between n_r and a leaf node in the updated search tree. Such a considerable reduction in the number of nodes visited during replanning can definitely help to reduce the computational time of path replanning. Hence, the proposed modifications can ultimately help mobile robots to navigate seamlessly on uneven terrains with incomplete information.

V. CONCLUSION

Dynamic Z* is originally proposed as an extension to Z* algorithm for fast replanning of energy-efficient paths on uneven terrains if previously planned path are inaccessible due to changes in the environment. Even though Dynamic Z* partially uses its search history to speed up the replanning process, due to its deficient node removal method, some useful information are discarded without being used in replanning. In this paper, we proposed a novel method to update the search tree once obstacles are detected on the previously planned paths. With the proposed improvements, Dynamic Z* is able to considerably minimize the number of node visits during replanning while producing the same energy-efficient paths as its predecessor does. With its rapid replanning, Dynamic Z* with the proposed improvements can be useful for mobile robots operating in dynamic outdoor environments.

ACKNOWLEDGMENT

This work is supported by the Department of Electronic and Information Engineering, the Hong Kong Polytechnic University (Project RTKL) and the Hong Kong PhD Fellowship Scheme.

REFERENCES

[1] X. Ning and L. Liu, “A two-mode ins/cns navigation method for lunar rovers,” *Instrumentation and Measurement, IEEE Transactions on*, vol. 63, no. 9, pp. 2170–2179, 2014.

[2] G. Hegde, C. Ye, C. Robinson, A. Stroupe, and E. Tunstel, “Computer-vision-based wheel sinkage estimation for robot navigation on lunar terrain,” *Mechatronics, IEEE/ASME Transactions on*, vol. 18, no. 4, pp. 1346–1356, 2013.

[3] T. Setterfield and A. Ellery, “Terrain response estimation using an instrumented rocker-bogie mobility system,” *Robotics, IEEE Transactions on*, vol. 29, no. 1, pp. 172–188, 2013.

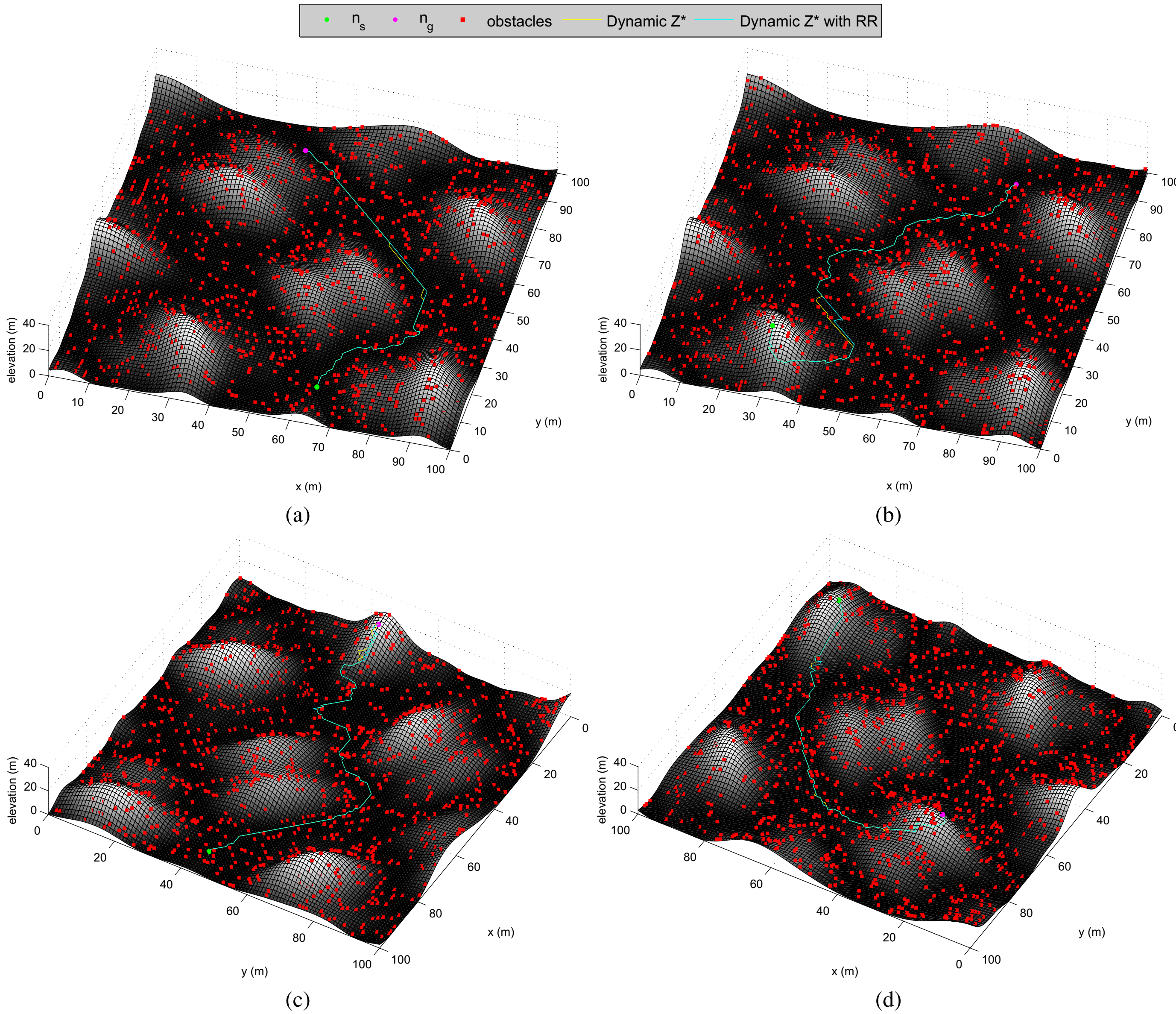

Fig. 1: Energy-efficient paths generated by the algorithms under test in (a) Simulation I, (b) Simulation II, (c) Simulation III, and (d) Simulation IV.

[4] G. E. Jan, C.-C. Sun, W. C. Tsai, and T.-H. Lin, "An $o(n \log n)$ shortest path algorithm based on delaunay triangulation," *Mechatronics, IEEE/ASME Transactions on*, vol. 19, no. 2, pp. 660–666, 2014.

[5] D. W. Hodo, D. M. Bevly, J. Y. Hung, S. Millhouse, and B. Selfridge, "Optimal path planning with obstacle avoidance for autonomous surveying," in *Proc. IEEE 36th Annu. Conf. Ind. Electron. Soc. (IECON)*, 2010, pp. 1577–1593.

[6] A. Srivastava, D. Kartikey, U. Srivastava, V. Srivastava, and S. Rajesh, "Non holonomic shortest robot path planning in a dynamic environment using polygonal obstacles," in *Industrial Technology (ICIT), 2010 IEEE International Conference on*. IEEE, 2010, pp. 553–558.

[7] N. Ganganath, C.-T. Cheng, and C. K. Tse, "A constraint-aware heuristic path planner for finding energy-efficient paths on uneven terrains," *Industrial Informatics, IEEE Transactions on*, vol. 11, no. 3, pp. 601–611, June 2015.

[8] N. C. Rowe and R. S. Ross, "Optimal grid-free path planning across arbitrarily contoured terrain with anisotropic friction and gravity effects," *Robotics and Automation, IEEE Transactions on*, vol. 6, no. 5, pp. 540–553, 1990.

[9] N. Ganganath, C.-T. Cheng, and C. K. Tse, "Finding energy-efficient paths on uneven terrains," in *Mechatronics (MECATRONICS), 2014 10th France-Japan/8th Europe-Asia Congress on*. IEEE, 2014, pp. 383–388.

[10] N. Ganganath, C.-T. Cheng, and C. K. Tse, "Rapid replanning of energy-efficient paths for navigation on uneven terrains," in *Industrial Informatics (INDIN), 2015 International Conference on*. IEEE, 2015, pp. 408–413.

[11] E. Berthier, E. Schiefer, G. K. Clarke, B. Menounos, and F. Rémy, "Contribution of alaskan glaciers to sea-level rise derived from satellite imagery," *Nature Geoscience*, vol. 3, no. 2, pp. 92–95, 2010.

[12] 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.

[13] S. Koenig and M. Likhachev, "Fast replanning for navigation in unknown terrain," *Robotics, IEEE Transactions on*, vol. 21, no. 3, pp. 354–363, 2005.