The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Adaptive Node Sampling Method for Probabilistic Roadmap Planners

Byungjae Park and Wan Kyun Chung

Abstract— **This paper proposes an adaptive node sampling method for the probabilistic roadmap (PRM) planner. The proposed method substitutes the random sampling in the learning phase of the PRM planner and improves the configuration of the roadmap. This method uses two phase to determine nodes in order to construct the roadmap. First, the proposed method extracts initial nodes using the approximated cell decomposition and the Harris corner detector. Second, the positions of these nodes are optimized using a construction process of the centroidal voronoi tessellation (CVT). The proposed method determines the adequate number and positions of the nodes to represent the entire free space, and the PRM planner based on the proposed method finds out efficient paths even in narrow passages. These properties have been verified though experiments.**

I. INTRODUCTION

Path planning is becomming more important as the application field of mobile robots extends to outdoor environment such as rough terrain, air and underwater. So, it is important to search the optimal path in complex and extensive environments for a safe and efficient navigation.

The exact methods based on the grid representation of the environment are generally used to search the optimal path [1] [2]. These methods search the optimal path considering the travel cost and safety, and have the ability to deal with dynamic changes in the environment. However, the computational efficiency of these methods is deficient in complex and extensive environments because the number of grids increases rapidly as the size of the environment expends. To improve the computational efficiency, several researchers have proposed sampling based approaches. The probabilistic roadmap (PRM) planner is one of the typical sampling based approaches [3]. The PRM planner creates a roadmap that represents the connectivity of the free spaces in the environment. The PRM based planner searches for the collision free paths using two phases: a learning phase and a query phase [9]. In the learning phase, a roadmap is constructed by generating nodes and connecting them using the local planner, while considering a straight-line motion without any collision. After that, multiple queries can be answered to search the path without collision. Although the efficiency of the path from the PRM palnner is not better than that of the exact methods, the PRM planner has better computational efficiency than the exact methods because the entire free space in the environment is abstracted to a set of nodes and edges. However, the PRM based planner has a problem in the learning phase. Nodes are not distributed wide enough to connect the entire free space because nodes are generated by random sampling. This problem causes the following issues:

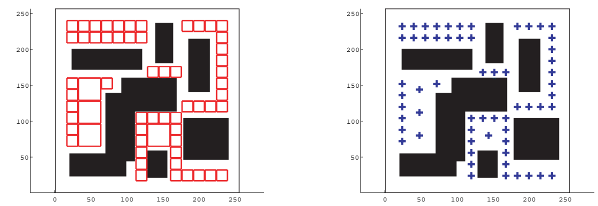
- Sufficient nodes are required to build the roadmap that connects the entire free space in the environment.
- The PRM planners have trouble when searching paths in narrow passages .
- The efficiency of the path searched by the PRM planners is not always guaranteed. The PRM planners always have a chance to find an inefficient path.

Several methods were proposed to sample adequate nodes to build the roadmap. Kabraki et al. proposed the sampling method that generated additional samplers in the neighborhood of nodes which were connected to a few nodes [3]. The approximated cell decomposition was used to determine the positions of nodes [4]. The other methods were proposed to search paths in narrow passages. The obstacle-based PRM(OBPRM) generated nodes in long and narrow passages based on the candidate points that were uniformly distributed on the surface of each obstacle [5]. Boor et al. proposed the gaussian random sampling method to substitute the uniform random sampling method [6]. This method generated nodes in narrow passages. The MAPRM generated node based on the medial axes of the free part of the environment to create nodes in narrow passages [7]. The non-uniform sampling method was proposed by van den Berg and Overmars [8]. This method assigned weights to labeled regions which were classified by the approximated cell decomposition and the watershed segmentation, then sampled the nodes based on assigned weights.

In this paper, we propose an adaptive node sampling method to construct the roadmap for the PRM planner. The proposed method substitutes the random sampling in the learning phase of the PRM planner and improves the configuration of the roadmap. This method uses two phase to determine nodes in order to construct the roadmap. First, the proposed method extracts initial nodes. Second, the positions of these nodes are optimized approximately. In the first phase, we use the approximated cell decomposition and the Harris corner detector to extract the initial nodes considering the geometric configuration of the environment. The approximated cell decomposition determines the initial nodes in broad regions of free spaces, and the Harris corner detection algorithm is applied to extract initial nodes in narrow passages. In the second phase, the positions of the initial nodes are optimized approximately by the construction

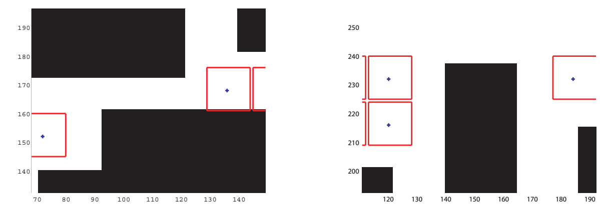
Byungjae Park is with Robotics Lab., Mechanical Engineering, Pohang University of Science and Technology (POSTECH), Pohang, KOREA; lain@postech.ac.kr

Wan Kyun Chung is with Faculty of Mechanical Engineering, Pohang University of Science and Technology (POSTECH), Pohang, KOREA; wkchung@postech.ac.kr



(a) (b)

Fig. 1. Initial nodes determined by the approximated cell decomposition: (a) Red boxes are decomposed cells, (b) Blue plus signs are initial nodes.



(a) (b)

Fig. 2. Limitation of extraction method to determine initial nodes using the approximated cell decomposition: There are no initial nodes in narrow passages.

process of the Centroidal Voronoi Tessellation (CVT) in order to improve the configuration and the connectivity of the roadmap.

The proposed adaptive node sampling method has the following properties.

- The proposed method determines the adequate number and positions of the nodes to represent entire free space in the environment.
- The PRM planner based on the proposed method finds out the path through narrow passages.
- The efficiency of path is improved by the optimization process.

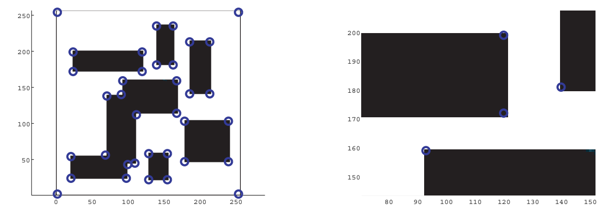
This paper is organized as follows. Section II and Section III describe the node extraction algorithm and the node optimization algorithm. An implementation detail and simulation results are presented in Section VI, and the conclusion follows.

II. NODE EXTRACTION

We present the initial node extraction that determines the number of initial nodes and their initial positions using geometric configuration of the environment. The approximated cell decomposition and the Harris corner detector are used to extract the initial nodes.

A. Approximated Cell Decomposition

Free space in the environment are divided into a set of sub-cells to determine the number of initial nodes and their positions. The number of nodes is the number of decomposed cells, and the position of each initial node is the center of each cell. The approximated cell decomposition uses the quadtree structure, which represents a partition of space in two dimensions by decomposing each cell into four equal quadrants, subquadrants, and so on. If there are obstacles in the region of interest of a cell, then the cell is divided into four sub-cells until the size of the cell reaches to the minimum cell size. As shown in Fig.1, if the minimum cell size is proper, decomposed cells cover most free space in environment. However, there is a limitation. If there are some narrow passages that are smaller than the minimum cell size, the approximated cell decomposition can not determine initial nodes in those regions as shown in Fig.2. To compliment this limitation, the proposed method uses additional algorithm to determine initial nodes in narrow passages.



(a) (b)

Fig. 3. Initial nodes in narrow passages: (a) Blue circles indicate Harris corners in the environment, (b) Harris corners are located near entrances of narrow passages.

B. Geometric Points Extraction

The approximated cell decomposition in the previous step can not extract initial nodes in narrow passages because subcells which have smaller cell size than threshold are ignored. The proposed method determines initial nodes in narrow passages by extracting salient geometric points since those points provide representative information of narrow passages. To extract salient geometric points, the Harris corner detector is applied [10]. The Harris corner detector is based on the local auto-correlation function of the intensity. The local auto-correlation function measures the local changes of the intensity with patches shifted by a small amount in different directions. Salient geometric points extracted by the Harris corner detector are depicted in Fig.3. These extracted points can be used as initial nodes since these points are located near entrances of narrow passages, convex and concave corners.

III. NODE OPTIMIZATION

The efficiency of paths is not always guaranteed when the PRM planner uses the roadmap constructed base on the initial nodes that determined by the approximated cell decomposition and the Harris corner detector because the nodes in this roadmap are not distributed regularly to connect entire free space in the environment. To improve the efficiency of the paths, more regularly distributed configuration of nodes is required. The proposed method distributes the initial nodes in the environment with considerations of a geometric configuration using the CVT construction process. Fig.4 shows a configuration of nodes in the CVT. In this figure, the CVT has almost regularly distributed nodes and

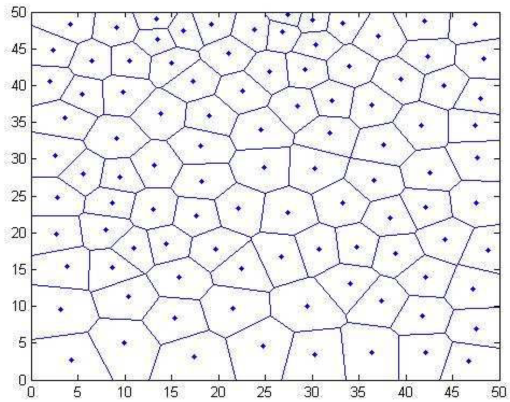


Fig. 4. Example of the CVT

voronoi cells because the CVT construction method updates the positions of its nodes.

In the construction process of the CVT, positions of nodes are updated iteratively until each position of the nodes is equivalent to each centroid of voronoi cells [11]. The centroid of a voronoi cell is defined as follows:

$$z_i^* = \frac{\int_{V_i} x\rho(x)dx}{\int_{V_i} \rho(x)dx}, \quad (1)$$

z_i^* is the mass centroid of a voronoi cell V_i, z_i is a voronoi node, and $\rho(x)$ is a density function. The CVT is only defined when $z_i^* = z_i$ for all of nodes in a voronoi tessellation.

The McQueen's algorithm [12] and the Lloyd's algorithm [13] are well known processes to construct the CVT using the initial nodes. The McQueen's algorithm does not require a heavy computational burden. However, it can not construct the CVT precisely since it is an approximation algorithm. The Lloyd's algorithm divides the environment into CVT precisely. However, it has a very heavy computational burden because the area of each voronoi cells should be calculated in every iterations. For these reasons, the proposed method uses the probabilistic Lloyd's algorithm [14]. This algorithm is an intermediate strategy based on the McQueen's algorithm, and Lloyds algorithm. This algorithm approximately estimates the centroid of nodes using non-parametric samples instead of using the area of each voronoi cell. The probabilistic Lloyd's algorithm guarantees a fast convergence speed with less computational burden than that of the Lloyd's algorithm, and a better precision than that of the McQueen's algorithm and.

As shown in Fig.5, the probabilistic Lloyd's algorithm uses random samples to update the position of each node. Z_1, Z_2 are current nodes and Z_1^*, Z_2^* are updated nodes. W_1, W_2 are groups of nearest neighbors of Z_1, Z_2, and the black dashed line is the voronoi edge. The position of the node are shifted toward to the centroid of the group of the nearest neighbors of random samples. Algorithm 1 describes the probabilistic Lloyd's algorithm in detail.

The probabilistic Lloyd's algorithm uses the energy value

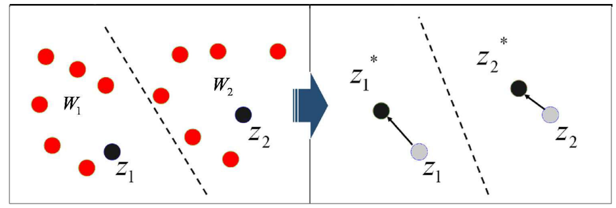


Fig. 5. The positions of nodes are updated using random samples in the probabilistic Lloyd's algorithm

Algorithm 1 Probabilistic Lloyd's Algorithm.

1. Choose the random sample number q and constraints $\{\alpha_i, \beta_i\}_{i=1}^{2}$ and choose an initial set of k nodes $\{z_i\}_{i=1}^{k}$ and set $j_i = 1$ for $i = 1, \ldots, k$.
2. Generate the random samples $\{y_r\}_{i=1}^{q}$ according to the probability density function $\rho(x)$.
3. For $i = 1, \ldots, k.$, gather together in the set W_i all the random samples y_r, closest to z_i among $\{z_i\}_{i=1}^{k}$. and set $z_i \leftarrow \frac{(\alpha_1 j_i + \beta_1)z_i + (\alpha_2 j_i + \beta_2)u_i}{j_i + 1}$ and $j_i \leftarrow j_i + 1$. u_i is the centroid of W_i.
4. If the positions of the nodes meet some convergence criterion, terminate; otherwise, return to step 2.

as a criterion to decide convergence of the positions of nodes and to finish the iterative construction process. The energy value of the original Lloyd's algorithm is defined as follows:

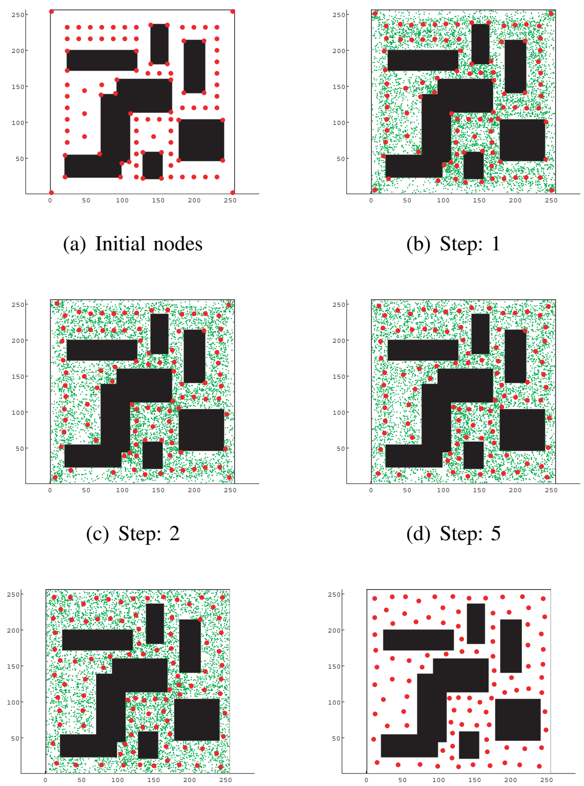
$$K\left(\{z_i\}_{i=1}^{k}, \{V_i\}_{i=1}^{k}\right) = \sum_{i=1}^{k} \int_{V_i} \rho(x)|x - z_i|^2 dx. \quad (2)$$

The original energy value is modified to calculate the criterion based on random samples as

$$K\left(\{z_i\}_{i=1}^{k}, \{W_i\}_{i=1}^{k}\right) = \sum_{i=1}^{k} \sum_{y_r \in W_i} \rho(y_r)|y_r - z_i|^2. \quad (3)$$

If nodes are more uniformly distributed in free regions of environment, this modified energy value becomes smaller. The proposed method finishes the CVT construction process when this energy value is bounded. Fig.6 shows an example of the optimization process. In this figure, the positions of nodes and random samples are respectively represented by red dots and green dots. The shift of each nodes is depicted in Fig.7.

The trajectories of nodes are indicated by black lines; blue dots represent initial nodes; and red dots mean updated nodes. If there are many nodes in the same region, some of those nodes are moved to other regions that do not have enough nodes. Eventually, nodes are regularly distributed in free space in the environment with considerations of the geometric shapes. Fig. 8 shows the change of the energy value of the probabilistic Lloyd's algorithm in this example. The energy value has a declining tendency. Aforementioned example shows that if nodes have a regular distribution, the energy value is bounded.



(a) Initial nodes (b) Step: 1

(c) Step: 2 (d) Step: 5

(e) Step: 10 (f) Update finished

Fig. 6. The positions of nodes are optimized by the probabilistic Lloyd's algorithm. Red dots represent nodes and green dots represent random samples: (a) Initial nodes determined by the approximated cell decomposition and the Harris corner detector, (b,c,d,e) Construction steps, (f) The positions of nodes are updated.

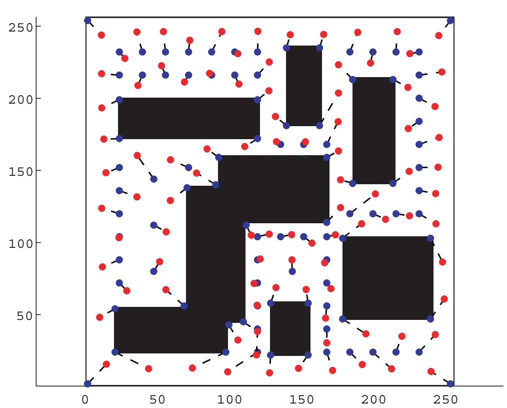


Fig. 7. Initial nodes are shifted by the probabilistic Lloyd's algorithm. Blue dots indicate initial nodes, red dots mean updated nodes, and the shift of those nodes is represented by black lines.

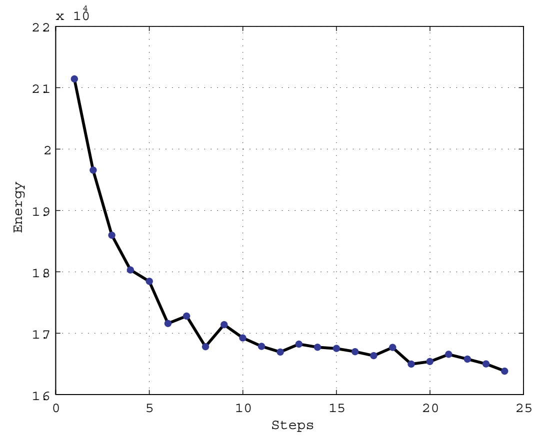


Fig. 8. The change of the energy value in the construction process of the CVT using the probabilistic Lloyd's algorithm.

IV. EXPERIMENTAL RESULTS

A. Experimental Setup

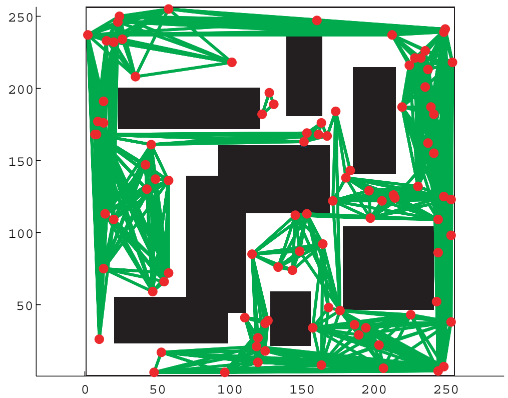
The proposed method was integrated in the PRM planner to verify the improvement of the efficiency of planned paths. All of our experiments were run with MATLAB on a Core II Duo 3.16GHz with 2GB of internal memory. Algorithm 2 describes the procedure of the PRM planner based on the proposed method. It was implemented to the node sampling step of the learning phase of the PRM planner to determine nodes for the construction of the roadmap. Dijkstra's algorithm was applied to the query phase to search the shortest path to connect a starting position and a goal [15].

Algorithm 2 PRM Planner Based On Adaptive Sampling.

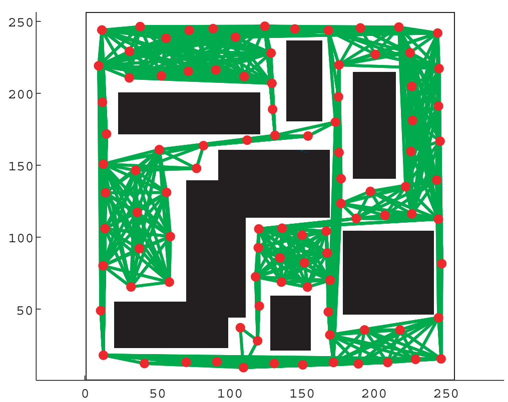
1. Learning Phase
   a. Node Sampling
      - Extracting nodes V_f in free regions using the approximated cell decomposition.
      - Extracting nodes V_n in narrow passages using the Harris corner detector.
      - Optimize the positions of the initial nodes $V(V = V_f \cup V_n)$ using the probabilistic Lloyd's algorithm for building the CVT.
   b. Edge Searching
      - Try to connect each pair of updated nodes V^*.
      - Successful connections without a collision become an edge of the roadmap.
2. Query Phase
   a. Connect a starting point and a goal to the roadmap.
   b. Search the shortest path to connect start and goals using the graph searching algorithm.

B. Results

The difference between two roadmaps based on the random sampling and the proposed sampling in shown in Fig.9.



(a) Random sampling



(b) Adaptive sampling

Fig. 9. Comparison of the roadmaps. Green lines represents edges of the roadmaps: (a)The roadmap based on the random sampling, (b)The roadmap based on the adaptive node sampling

Each roadmap has 96 nodes. The size of the environment is 256×256 units. The clearance of a path is 10 units to avoid collision with obstacles in the environment. The configuration of the roadmap in Fig.9(a) is irregular because of an irregular distribution of nodes. Some nodes in this roadmap does not connected to other nodes in free space. The PRM planner based on the random sampling is not able to search a path that passes a narrow regions in the environment. On the other side, the roadmap based on the adaptive sampling method, that is shown in Fig. 9(b), have much better configuration than that based on the random sampling. There were some edges that pass through narrow passages in the environment because of the nodes in narrow passages. These differences increase the efficiency of the path and the robustness of the path planning.

In the experiments, the PRM planner based on the proposed method searched the shortest paths that connect 10 different goals and the same starting point. We compare the efficiency of these paths to the paths searched by the PRM planner based on the random sampling. We used the length of the path as the travel cost to compare the efficiency of

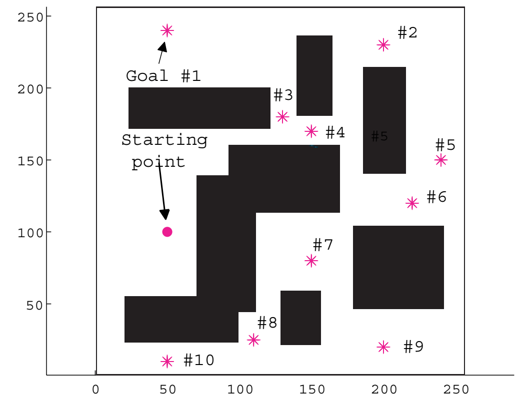
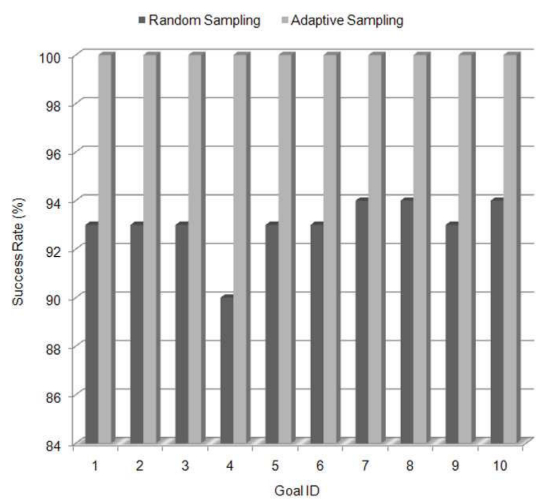
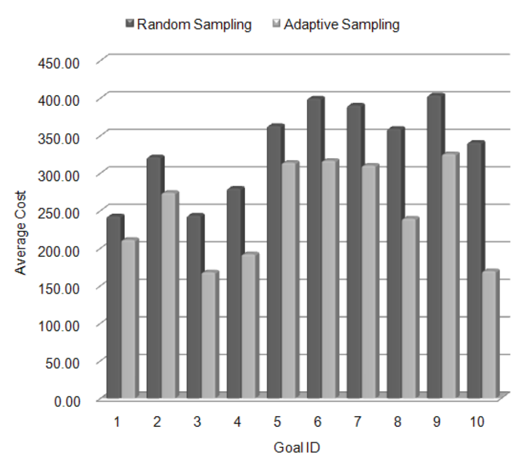


Fig. 10. The starting point and 10 different goals in the experiments.



(a) Success Rate



(b) Average Cost

Fig. 11. The efficiently of the paths to 10 different goals: (a) The success rate of the path planning is increased using the PRM planner based on the adaptive node sampling method, (b) The PRM planner based on the adaptive node sampling method searches more efficient paths than the paths searched by the PRM planner based on the random node sampling. The length of the path means the travel cost.

the path. Fig.10 shows the common starting point and 10 different goals. The PRM planner based on the proposed method was run 100 times to construct the roadmaps.

The experimental results are shown in Fig. 11. The success rates of the path planning with the PRM planner based on the adaptive sampling are higher than ones based on the random sampling. The PRM planner based on the random sampling failed to search the paths because the roadmap did not cover entire free space in the environment when the configuration of the roadmap was irregular. The average travel costs of paths from the starting point to 10 different goals planned by the PRM planner based on the adaptive sampling method are smaller than ones based on the random sampling method because the adaptive sampling method based roadmap connects the entire free space in the environment and the nodes of this roadmap were almost regularly distributed. Fig.12 is examples to compare the efficiency of the paths planned by the PRM planner based on the random sampling and the adaptive sampling. In this figure, the paths planned by the PRM planner based on the adaptive sampling method are shorter than the pathes based on the random sampling method.

V. CONCLUSION

An adaptive node sampling method for the PRM planner has been proposed. This method improves the efficiency of the paths by determination and optimization of the nodes in the roadmap. The experiential result shows the following properties of the proposed method.

- The proposed method determines nodes to connect entire free space in environment. The roadmap has a balanced configurations because the nodes are distributed regularly in the environment.
- The PRM planner based on the proposed method improves the efficiency of the path.
- The PRM planner based on the proposed method searches paths robustly even in narrow passages.

VI. ACKNOWLEDGMENTS

This work was supported in part by the Acceleration Research Program of the Ministry of Education, Science and Technology of the Republic of Korea and the Korea Science and Engineering Foundation [R17-2008-021-01000-0], in part by the Korea Health 21 R&D Project, Ministry of Health & Welfare, Republic of Korea under Grant A020603, in part by the Agency for Defence Development and by Unmanned Technology Research Center(UTRC), Korea Advanced Institute of Science and Technology, and in part by the Korea Science and Engineering Foundation(KOSEF) under Korea Government Grant MOST R0A-2003-000-10308-0, in part by the IT R&D program of MKE/IITA [2008-F-038-1, Development of Context Adaptive Cognition Technology].

REFERENCES

[1] D. Ferguson and A. Stentz, "The Delayed D* Algorithm for Efficient Path Replanning," in Proc. of IEEE International Conference on Robotics and Automation, pp. 2045-2050, 2005.

[2] M. Likhachev, D. Ferguson, G. Gordon, A. Stentz and S. Thrun, "Anytime Dynamic A*: An Anytime, Replanning Algorithm," in Proc. of the International Conference on Automated Planning and Scheduling, 2005.

[3] L. E. Kavraki, P. Švestka, J. Latombe and M. H. Overmars,"Probabilistic Roadmaps for Path Planning in high-dimensional configuration spaces," in IEEE Transacions on Robotics and Automation, vol. 12, no. 4, pp. 566-580, 1996.

[4] N. I. Katevas, S. G. Tzafestas and C. G. Pnevmatikatos, "The Approximate Cell Decomposition with Local Node Refinement Global Path Planning Method: Path Nodes Refinement and Curve Parametric Interpolation," in Journal of Intelligent and Robotic Systems, vol. 22, no. 3-4, pp. 289-314, 1998.

[5] N. M. Amato and Y. Wu, "A Randomized Roadmap Method for Path and Manipulation Planning," in proc. IEEE International Conference on Robotics and Automation, pp. 113-120, 1996.

[6] V. Boor, M. H. Overmars and A. F. van der Stappen, "The Gaussian Sampling Strategy for Probabilistic Roadmap Planners," in proc. IEEE International Conference on Robotics and Automation, pp. 1018-1023, 1999.

[7] S. A. Wilmarth, N. M. Amato and P. F. Stiller, "MAPRM: A Probabilistic Roadmap Planner with Sampling on the Medial Axis of the Free Space," in proc. IEEE International Conference on Robotics and Automation, pp. 1024-1031, 1999.

[8] J. P. van den Berg and M. H. Overmars, "Using Workspace Information as a Guide to Non-uniform Sampling in Probabilistic Roadmap Planners," in International Journal of Robotics Research, vol. 24, no. 12, pp. 1055-1071, 2005.

[9] H. Choset, Mevin. M. Lynch, S. Hutchinson, G. A. Kantor, W. Burgard, K. E. Kavraki and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations*, MIT Press, 2005.

[10] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," in proc. of The Fourth Alvey Vision Conference, pp. 147-151, 1988.

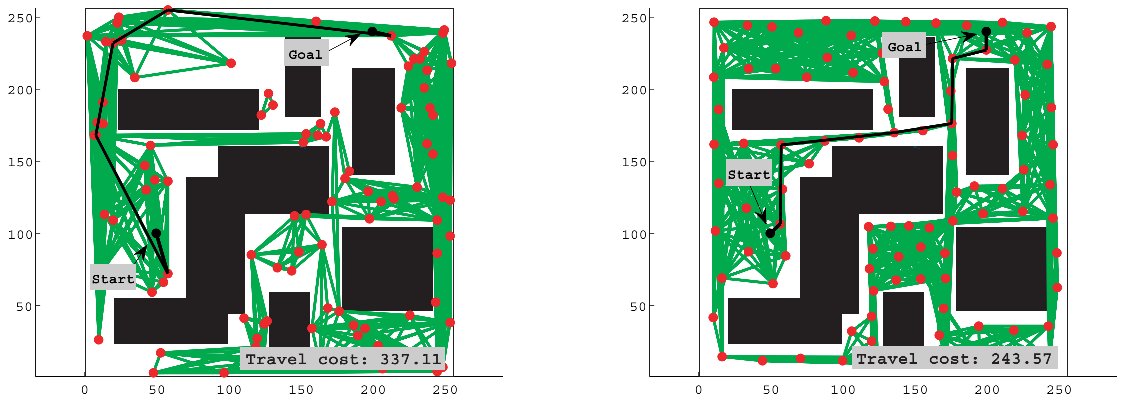
[11] Q. Du, V. Faber and M. Gunzburger, "Centroidal Voronoi Tessellations: Applications and Algorithms," in SIAM Review, vol. 41, no. 4, pp. 637-676, 1999.

[12] J. McQueen, " Some methods for classification and analysis of multivariate observations," in Proceedings, Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, pp. 281-297, 1967.

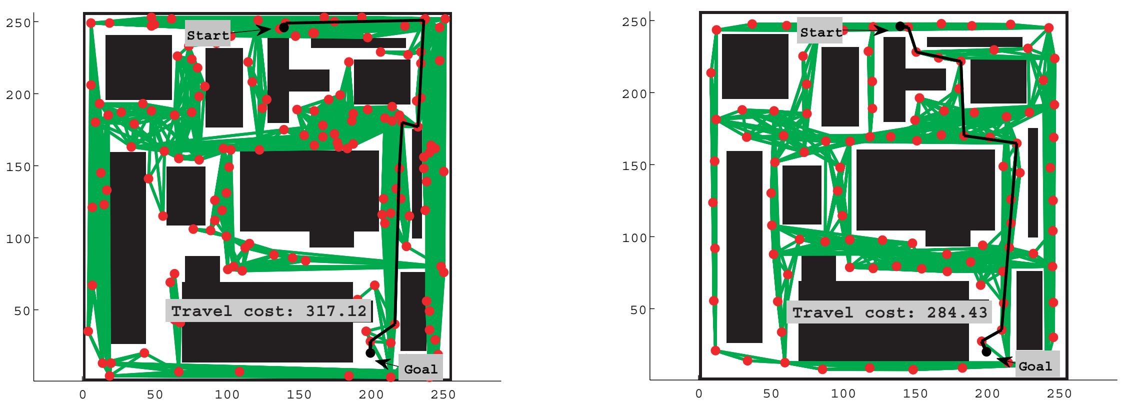
[13] S. Lloyd, "Least square quantization in PCM," in IEEE Transactions on Information Theory, vol. 28, pp. 129-137, 1982.

[14] L. Ju, Q. Du and M. Gunzburger, "Probabilistic methods for centroidal Voronoi tessellations and their parallel implementations ," in Parallel Computing, vol. 28, issue. 4, pp. 1447-1500, 2002.

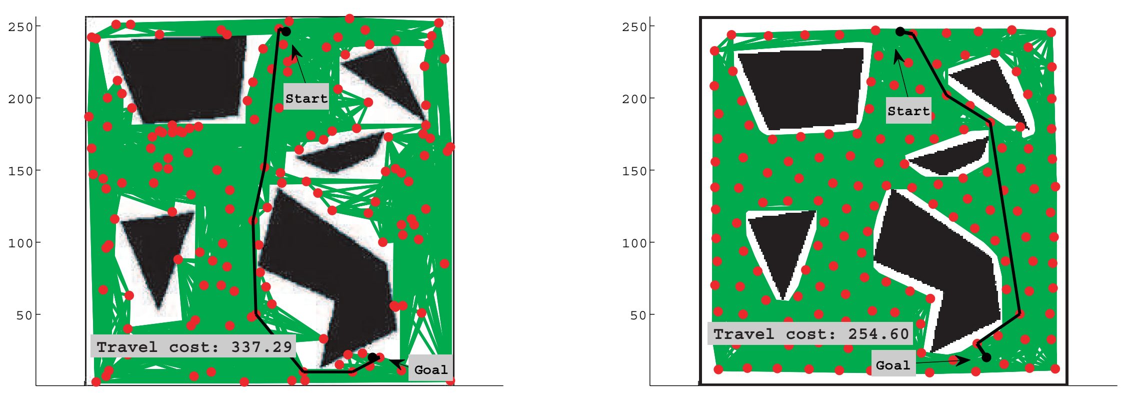
[15] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, *Introduction to Algorithms*, Second Edition, MIT Press and McGraw-Hill, 2001.



(a) Simulation map 1



(b) Simulation map 2



(c) Simulation map 3

Fig. 12. Examples to compare the efficiency of the pathes planned using the random sampling method based roadmap and the adaptive sampling method based roadmap. Left column: The planned pathes using the random sampling method based roadmap. Right column: The planned pathes using the adaptive sampling method based roadmap. Black lines represent the path to connect the starting point to the goal. Red dots mean updated nodes, and green lines represents edges of the roadmap.