

Motion Planning and Control for Mobile Robot with SOFM

Seok-Min Yun*, and Jin-Young Choi**

* School of Electrical Engineering and Computer Science, ASRI, Seoul National University, Seoul, Korea
(Tel : +82-2-872-7283; E-mail: smyoon@neuro.snu.ac.kr)

** School of Electrical Engineering and Computer Science, ASRI, Seoul National University, Seoul, Korea
(Tel : +82-2-872-7283; E-mail: jychoi@snu.ac.kr)

Abstract: Despite the many significant advances made in robot architecture, the basic approaches are deliberative and reactive methods. They are quite different in recognizing outer environment and inner operating mechanism. For this reason, they have almost opposite characteristics. Later, researchers integrate these two approaches into hybrid architecture. In such architecture, Reactive module also called low-level motion control module have advantage in real-time reacting and sensing outer environment; Deliberative module also called high-level task planning module is good at planning task using world knowledge, reasoning and intelligent computing. This paper presents a framework of the integrated planning and control for mobile robot navigation. Unlike the existing hybrid architecture, it learns topological map from the world map by using MST (Minimum Spanning Tree)-based SOFM (Self-Organizing Feature Map) algorithm. High-level planning module plans simple tasks to low-level control module and low-level control module feedbacks the environment information to high-level planning module. This method allows for a tight integration between high-level and low-level modules, which provide real-time performance and strong adaptability and reactivity to outer environment and its unforeseen changes. This proposed framework is verified by simulation.

Keywords: mobile robot, hybrid robot architecture, MST-based SOFM, topological map, path planning

1. INTRODUCTION

A fundamental problem in mobile robotics is the design of a suitable architecture for integration of the various components in a system. Two essential components in mobile robot are planning and control. So the integration of planning and control plays an important role in the whole system. Research in the planning and control of mobile robots has lasted two decades. Two basic approach emerged from these research efforts are deliberative [4, 5] and reactive [2, 3]. The two approaches can be distinguished by their different usage of data and global knowledge, speed of response, reasoning capability, and complexity of computation. Because of the complementarities of their strengths, their weaknesses can be mitigated by combining the two approaches in hybrid architecture [1, 7]. The integration of deliberative module and reactive module makes hybrid architecture easy to deal with the problem during the planning and control process. Among the existing hybrid frameworks, [9] emphasize high-level task planning. On the other hand, [10] focus on integrating the low-level reactive. The characteristic and intensity of integration between planning and control modules decides the whole hybrid framework's performance

This paper describe a novel integration method that it learns topological map from world map by using MST (Minimum Spanning Tree)-based SOFM (Self Organizing Feature Map) [6]and communicate between deliberative module and reactive module through it. Our hybrid architecture consists of deliberative module and reactive module. And each module contains several components.

The reactive module uses behavioral-based reactive [8] control model. It performs continuous response encoding [8] rather than discrete response encoding. Such a method can produce very low-level velocity to perform fine, smooth motion control.

The deliberative module comprises topological path learning and path searching components. Topological path learning component generates skeleton-like topological map that consists of nodes and links between two nodes. Any node is potential checkpoint and can be fired in path searching component. Path searching component compute the path between source and destination. The nodes on the selected

path will be fired and become checkpoints.

2. HYBRID ARCHITECTURE

2.1 Overview

Our hybrid framework is based the existing hybrid framework [1, 8] with some modification. The most significant modification is the application of topological map within the integration and bidirectional interaction between deliberative module and reactive module. As you see, there are two modules (deliberative vs. reactive) and four levels. At highest level, there are three components. Map building component can build world map in both of online and offline methods. In this paper, we deal with its result and use it directly. Topological map learning component is used for extracting topological map from world map by using MST-based SOFM learning method. Path searching component is in charge of providing an optimal path from source to destination in topological map. Topological map learning and path searching components comprise the deliberative module.

Below the deliberative module, three levels constitute reactive module. At the first level, the target reaching component determines the motion path between checkpoints. Every checkpoint on the selected path is the potential goal. When mobile robot reached a checkpoint, the next checkpoint relative to current checkpoint would be updated to be new goal.

The middle level in the lower three levels is obstacle avoidance component. It senses the local environment and detects unforeseen environment change. To avoid collision, it produces additional motor control command and makes itself to adapt local environment.

The lowest level is homeostatic control component. It perceives the internal state of mobile robot and maintains internal stability. But it is not necessary for our mobile robot's navigation system.

The three components in the lower level generate control command asynchronous. Command fusion components at the lowest level combines the control commands from each components of reactive module and a final command will be transmitted to actuators.

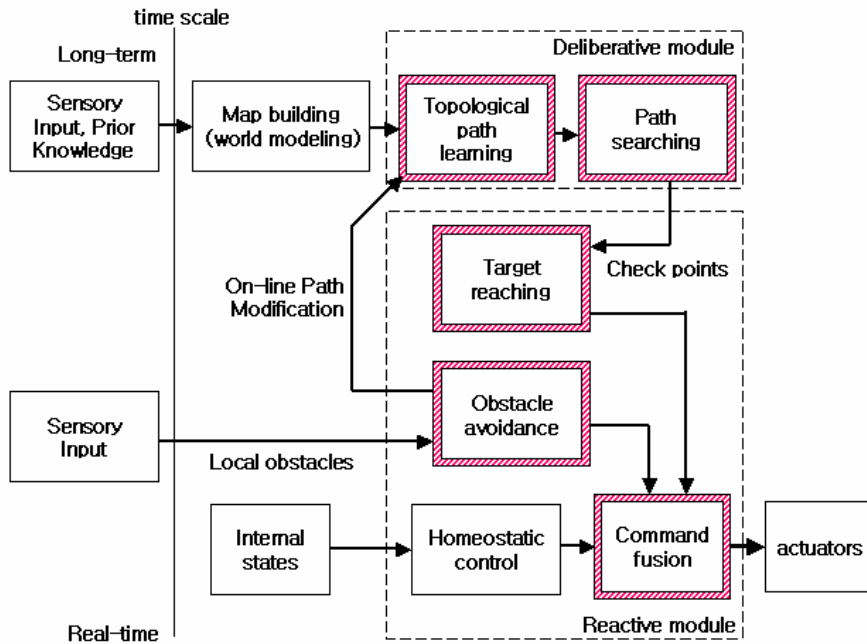


Fig. 1: Hybrid Architecture for mobile robot

The hybrid architecture proposed in this paper focus on the integration of deliberative module and reactive module. The topological map learned by MST-based SOFM is not only the essential to our deliberative module, but also crucial to reactive module. In deliberative module, it extracts path information from world map, and 2D map is reduced to linear topological map. It is more efficient and convenient for planning and reasoning in topological map rather than grid map. The path consists of checkpoints and link between them is sent to reactive module. In reactive module, checkpoints steers the mobile robot to destination. Also reactive module feedbacks the unforeseen environment changes to deliberative module by modifying the link between two relative nodes. Through such a bidirectional interaction on topological map, deliberative and reactive modules are integrated tightly in our hybrid architecture.

2.2 Deliberative Module

SOFM (Self-Organizing Feature Map) is also called Kohonen Network. It's based on competitive learning. The principal goal of the self-organizing feature map is to transform an incoming signal pattern or arbitrary dimension into a one- or two dimensional discrete map, and to perform this transformation adaptively with topology preserving.

Such a characteristic of SOFM is utilized in our Map learning process. It also can be used for some application like character recognition, image skeletonization. But the original SOFM is not suitable to extract the skeleton of image. In general, SOFM is a 2D grid map structure that unfits to express shape skeleton which is more similar to 1D linear topology structure. For example, it is impossible to extract cross shape skeleton correctly by using 2D grid map structure-SOFM in Fig. 2.

MST-based SOFM is an alternative plan to extract skeleton of image. It is tree shape self-organizing network that holds topology structure of network by spatial location of neurons. Owing to the characteristic of tree structure, MST-based SOFM is more adept in forming skeleton shape network than 2D grid map SOFM. In its learning process, the topology of neurons is adaptively transformed and trends toward show the

skeleton of input space.

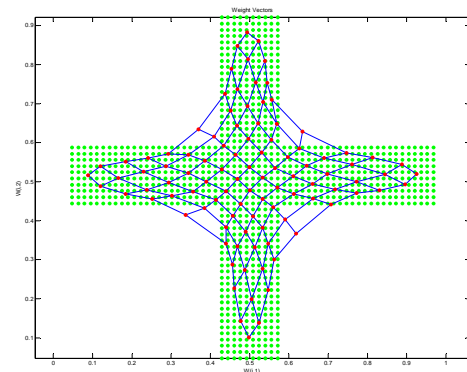


Fig. 2: Cross shape skeleton extracted using SOFM

The learning method of MST-based SOFM is not quite different than SOFM. The detailed learning algorithm is reported in [6]. Fig. 3 is the flow chart of this algorithm.

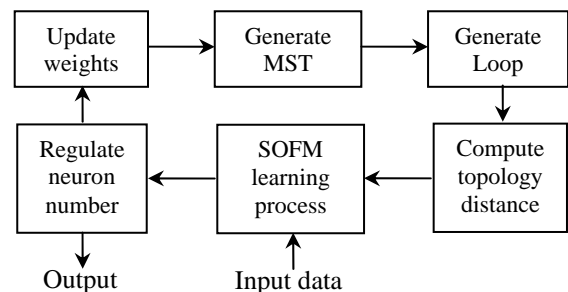


Fig.3: The flow chart of MST-based SOFM algorithm

Up to now, MST-based SOFM was introduced and the result image is shown in Fig. 4. As the figures show, the skeleton of image is extracted well. The result proved the fact that linear topology structure is more suitable for representing

shape skeleton than 2D grid map SOFM.

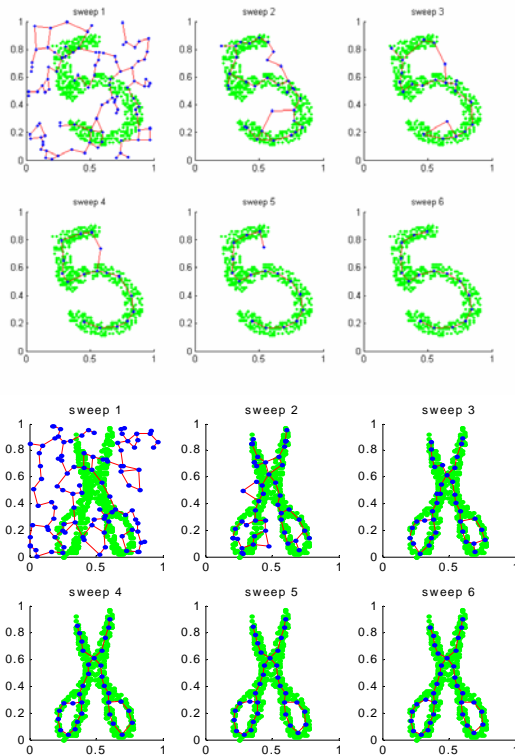


Fig. 4: Skeleton extraction with MST-based SOFM

We now apply MST-based SOFM to extract path from world map. Fig. 5 shows the resulting path with MST-based SOFM.

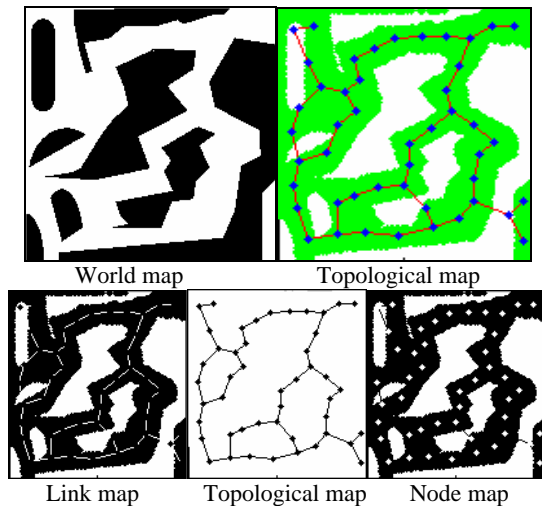


Fig. 5: Extract path from world map

Because the world map is much more complex than shape of scissors and character '5', there are some incorrect nodes and links that placed out of path. Even so, the topology of world map is also almost held in extracted topological path map. There are two approaches to eliminate the incorrect nodes and links; one is to degrade the complexity of world map by using map decomposition adaptively, and the other to detect and correct it on-line. We chose the latter instead of the

former, even though the former method can solve this problem fundamentally. In latter method, the topological map holds the incorrect nodes and links until on-line process. Despite the method cannot remove the incorrectness from the root, on the other hand, we can use it to test the fault tolerant property and adaptability of our hybrid architecture. As shown in Fig. 6, mobile robot chose optimal path (shown in white link in sub figure (2)); and he found incorrect link at the way, then removed the incorrect link and chose optimal path from current location to destination (shown in blue link in sub figure (3)); in the end, the topological map is changed correctly (shown in sub figure (4)).

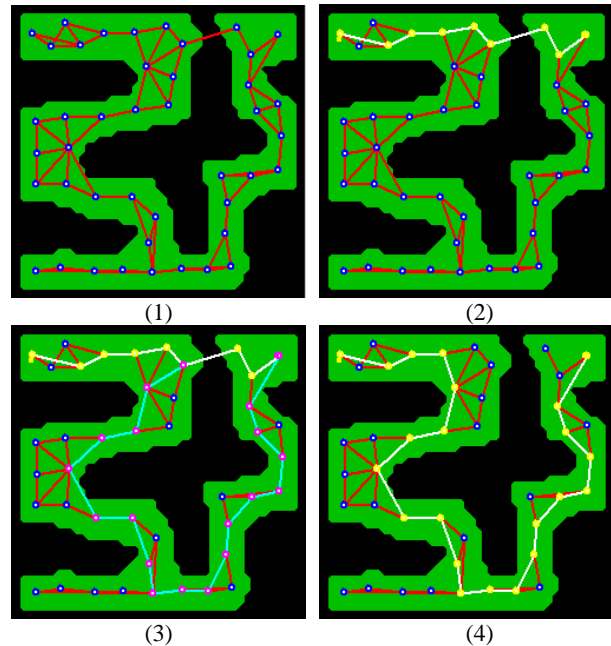


Fig. 6: Elimination of incorrect link

The optimal path is computed by dijkstra's shortest path algorithm [11]. The computation time of the algorithm is quite fast in this application due to the linear property of topological map.

2.3 Reactive Module

The mobile robot used in our simulation is shown as Fig. 7. There are seven sensors placed at intervals of 30 degrees and cover 180 degrees. We named the fourth sensor from the left side center sensor (shown as yellow color in Fig. 7).

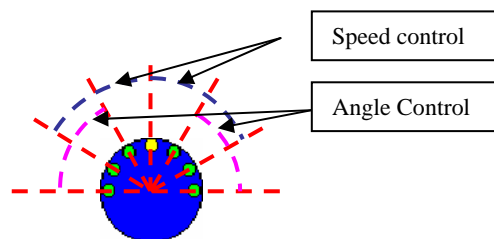


Fig. 7: Mobile robot used in our architecture

The center sensor and its two nearest neighbour sensors possess the information about front view and are used for determining the speed of robot. While the sensors without the center sensor have the information about lateral view and are used for determining the turning angle.

The target reaching component and obstacle avoidance component are introduced in Section 2.1. In here, we briefly describe the mechanism of command fusion component.

$$A_{ob} = 1 - dis_{min} / dis_{lmd} \tag{1}$$

Eq. (1) is to determine weight value of obstacle avoidance component. dis_{min} in the equation means the minimum distance to obstacle among the seven sensed data and dis_{lmd} means the farthest distance sensor can detect. This equation means the weight value is bigger as obstacle closer to robot

$$a = A_{ob} * a_{ob} + (1 - A_{ob}) * a_{goal} \tag{2}$$

$$v = A_{ob} * v_{ob} + (1 - A_{ob}) * v_{goal} \tag{3}$$

In Eq. (2) ~ (3), a , a_{ob} , a_{goal} , v , v_{ob} and v_{goal} are final turning angle, obstacle avoidance angle, goal directed angle, final velocity, obstacle avoidance velocity, goal directed velocity respectively. The two equations show that if the robot close to obstacle, obstacle avoidance component should take the precedence rather than target reaching component; if the robot far away from obstacle, target reaching should take charge of driving robot. Such a command fusion mechanism is verified in our simulation.

3. EXPERIMENTAL RESULT

In order to test the performance of each module, we test and analyze them separately. For reactive module, we test its obstacle avoidance performance; for deliberative module, we test the performance of planning and learning; in the end, we test the hybrid architecture with both performances above and adaptability to unforeseen environment changes.

3.1 Reactive Module

Reactive module consists of target reaching component and obstacle avoiding component. If reactive module runs independently, target reaching module can't give full play to its function. It no more than tells the robot the location of destination. But obstacle avoiding component can runs normally. As the Fig. 7 (1) shows, the robot has good performance in obstacle avoiding. The track is smooth and invariant to shape of obstacle.

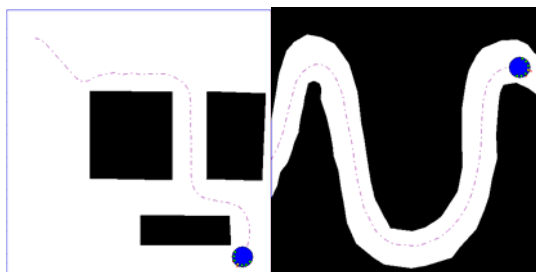


Fig. 7 (1)

But, with the increasing of complexity, reactive module can't reach the target properly. E.g. if there is local minimum in the world map, robot will fall into deadlock (shown as Fig. 7 (2)).

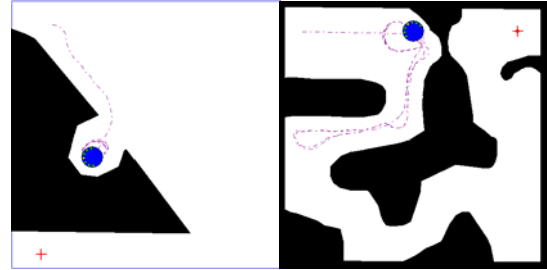


Fig. 7 (2)

Fig. 7: Test of reactive module

3.2 Deliberative Module

Deliberative module learns topological path and plans path for reaching target. As Fig. 8 (1) ~ (2) shows, the robot can follow the path indicated in topological map. But, whenever, it is ideal situation that it must ensure consistency of environment which any change in environment is intolerant during its running.

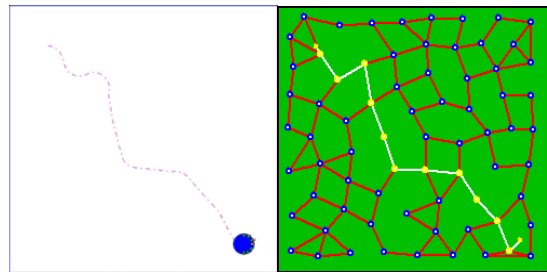


Fig. 8 (1)

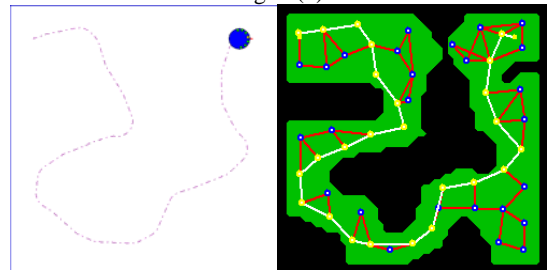


Fig. 8 (2)

Fig. 8: Test of deliberative module

3.3 Hybrid architecture

So far, both of reactive and deliberative modules have tested and analyzed individually. Now, we test hybrid architecture in two sides. At first, we compare the path generated by hybrid architecture with that by deliberative module. Fig. 9 (1) shows the track of deliberative module.

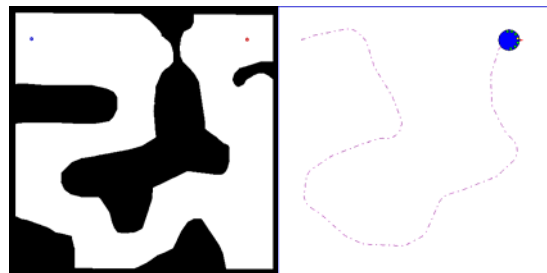


Fig. 9 (1)

In Fig. 9 (2), left figure shows that track of hybrid architecture, right figure is the overlapped result of two tracks. In it, blue track is generated by deliberative module, and the other is generated by hybrid architecture. From the track of

hybrid architecture, we found it shows the effort to avoid obstacle while it close to obstacle. But the other one have no such effort and may result collision in reality. This result proved the merit of hybrid architecture.

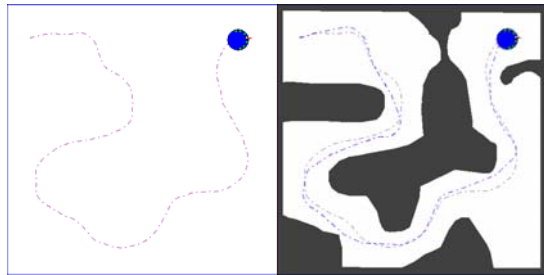


Fig. 9 (2)
Fig. 9: Test of hybrid architecture

Secondly, we test the adaptability of hybrid architecture in unforeseen environment change. There are two unforeseen change in Fig. 10. One is caused by internal incorrect link of topological map (shown in sub figure (2) by dot circle). And the other is external environment change-add an obstacle on environment (shown in sub figure (4) by dot circle). In the sub figure (3), the track showed robot succeeded in reaching the destination. Through the bidirectional interaction between reactive module and deliberative module, robot learned unforeseen environment changes and adapted itself to new environment.

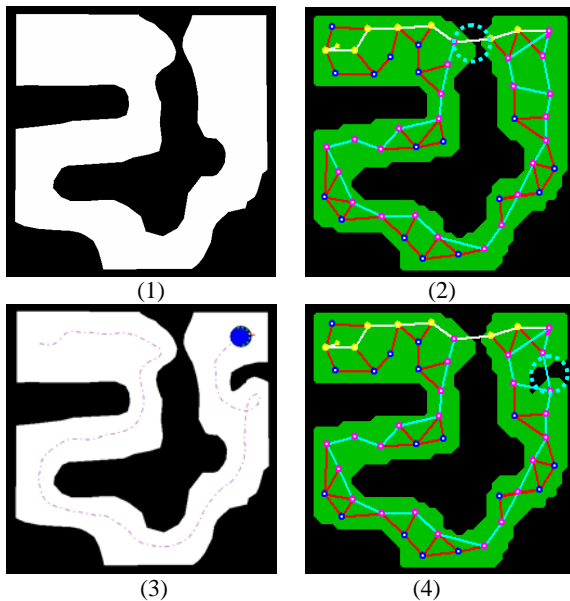


Fig. 10: topological map adapt to unforeseen environment change

The integration of deliberative module and reactive module makes our hybrid architecture possess higher performance with complementarities of their strengths. The bidirectional interaction between two modules coupled them up tightly. And it ensures the performance of our hybrid architecture.

4. CONCLUSION

We described a hybrid architecture that integrate planning and control by using topological map for mobile robot navigation. Unlike the existing hybrid architecture, it learns topological map from the world map by using MST

(Minimum Spanning Tree)-based SOFM (Self-Organizing Feature Map) algorithm. In this hybrid architecture, the relationship between high-level module and low-level module is not simply delivering commands but bidirectional interaction. The reactive module in this hybrid architecture perform continuous response encoding, as the result, it generates smoother track. In deliberative module, it uses topological map to search path and generate checkpoints instead of plotting the whole detailed path. It makes the control of mobile robot more easily and more naturally.

MST-based SOFM learning method is an effective way to generate topological map from world map. Research in path planning methods is separated into offline and online situation. But the topological map learned using MST-based SOFM can be used in both situations. At the Experiment stage, we found the topological map can not be learned accurately, when the world map is complex and large. So an important direction of future research would be to develop techniques for map decomposition and topological map integration. We suppose that multi-level SOFM will be the answer.

ACKNOWLEDGEMENT

The work described in this paper was supported by a grant from the project of Developing SIC (Super Intelligent Chip) and its Applications under the program of Next generation technologies of Korea Ministry of Commerce, Industry and Energy.

REFERENCES

- [1] R. C. Arkin, *Behavior-Based Robotics*, MIT Press, Cambridge, MA, 1998
- [2] R. Brooks, "A robust layered control system for a mobile robot," *IEEE J. Robot. Automat.*, 2(1): 14-23, 1986
- [3] J. H. Connell, *Minimalistic Mobile Robotics: A Colony Architecture for an Artificial Creature*, Academic Press, 1990
- [4] R. Chatila and J.-P. Laumond, "Position referencing and consistent world modeling for mobile robots," *In Proc. ICRA*, pages 138-145, 1985
- [5] G. Giralt, R. Chatila and M. Vaisset, "An integrated navigation and motion control system for autonomous multi-sensory mobile robots," *In Proc. 1st International Symposium on Robotics Research*, pages 191-214, 1984
- [6] Rahul Singh, Vladimir Cherkassky and Nikolaos Papanikolopoulos, "Self-Organizing Maps for Skeletonization of Sparse Shapes," *IEEE Transactions On Neural Networks*, Vol. 11, No. 1 January 2000
- [7] B. H. Krogh and C. E. Thorpe, "Integrated path planning and dynamic steering control for autonomous vehicles," *In Proc. ICRA* pages 1664-1669, 1986
- [8] Kian Hsiang Low, Wee Kheng Leow and Marcelo H. Ang, Jr, "A Hybrid Mobile Robot Architecture with Integrated Planning and Control," *In Proc. 1st International Joint Conference on Autonomous Agents & MultiAgent Systems (AAMAS-02)*, pages 219-226, 2002.
- [9] J. H. Connell, "SSS: A hybrid architecture applied to robot navigation," *In Proc. ICRA*, pages 2719-2724, 1992.
- [10] W. Choi and J. - C. Latombe, "A reactive architecture for planning and executing robot motions with incomplete knowledge," *In Proc. IROS*, volume 1, pages 24-29, 1991.
- [11] Mark Allen Weiss, *Data Structures and algorithm analysis in C++*, Addison Wesley Longman, 1999.