A METHODOLOGY FOR THE GENERATION OF THE 2-D MAP FROM UNKNOWN NAVIGATION ENVIRONMENT BY TRAVELING A SHORT DISTANCE

N.Bourbakis and D.Sarkar
Binghamton University
T.J.Watson School
AAAI Lab
Binghamton, NY 13902

ABSTRACT

In this paper a methodology is presented for the generation of a 2-D map from an unknown navigation environment by traveling a short distance. The methodology proposed here is based on the synthesis of the knowledge extracted from consecutive free navigation spaces, during the movement of an autonomous mobile robot. The generation of the 2-D map of the space is classified into three cases: (a) space without obstacles; (b) space with standing obstacles; and (c) space with moving obstacles.

KEYWORDS: 2-D Map Generation; Navigation in Unknown Space; Synthesis of Space Segments; Knowledge Extraction from
Unknown Space.

This work is a part of RFG grant 1992-93

1. INTRODUCTION

Knowledge acquisition from an unknown navigation space is one the challenging problem facing in the modern robotics (intelligent robots) and AI today [1-8]. In realistic situations, only the boundary and the outer shape of the current free navigation space is known and the interior structure and/or formation of the space are totally unknown. The space that we are talking about, could be the surface of an unknown planet, destroyed battlefield, or demolished landscape. Depending on the situation, traveling through the navigation space by human may be hazandrous, expensive, time consuming, inefficient and most of all may be life threatening.

A technique for the generation of the 2-D space map was proposed by [3]. This method scans the entire area. When stationary obstacles exist inside the navigation space, this method goes around of each obstacle and at the end generates the complete 2-D sapce map. In case that the obstacles are moving in the navigation space this methodology does not work. Moreover, it is a time consuming approach, especially when the number of stationary objects is great.

In this paper we present a formal methodology, which extracts knowledge by traveling through an unknown navigation space and generates a complete 2-D map of the navigation environment. Little or no knowledge is assumed regarding the navigation space and knowledge is accumulated solely by traveling in it. The methodology of generation of the 2-D map has been studied under certain space conditions:

a. *Space without any obstacles*
b. *Space with stationary obstacles*
c. *Space with moving objects*

The methodology of generating the 2-D map is based on the graph modeling of the navigation space and the synthesis of the successive free navigation spaces. Moreover, the proposed methodology generates the 2-D map by traveling a short distance in the total space.

This paper is organized into six section. Section 2 provides some definitions and notations. Section 3 deals with the representation of the knowledge extracted from the navigation space. Section 4 presents the synthesis of the shape-graphs. Section 5 discusses the generation of the space map and section 6 concluded the overall presentation.

2. NOTATIONS AND DEFINITIONS

In this section we provide a number of notations and definitions in order to accommodate the understanding of the following sections.

Notation 1: GNS represents the global navigation space.

Definition 1: A 2-D obstacle, b(j), jeZ.is defined as the two dimensional surface, Sb(j) $\subset$ GNS, where a moving object (robot R) cannot go through it, and the visual and laser rays stop or reflect on it.

Definition 2: A robot, R(n), neZ, is a moving "obstacle" in GNS by using its own power to do so.

Notation 2: BS represents the set of all the obstacles in GNS,
BS C GNS.

Notation 3: RS represents the set of all the robots in GNS,
RS $\subset$ GNS.

Notation 4: Sr(n) represents the 2-D surface covered by a robot R(n) in GNS.

Notation 5: $E(b) = \bigcup \{Sb(j)\}$ and $E(r) = \{Sr(n)\}$ represent the total surfaces covered by the total number of obstacles and the total number of robots in GNS respectively.

Definition 3: The total free navigation space inside GNS, is defined as FNS = (GNS - [E(b) + E(r)]).

Notation 7: t(i), $i \in Z$, represents the current time.

Notation 8: NS(t(i)) represents the navigation space at the time t(i).

Definition 4: The current free navigation space $FNS^n(t(i))$ is defined as the free space perceived by the robot R(n) at the time t(i), $FNS^n(t(i)) \subset FNS$.

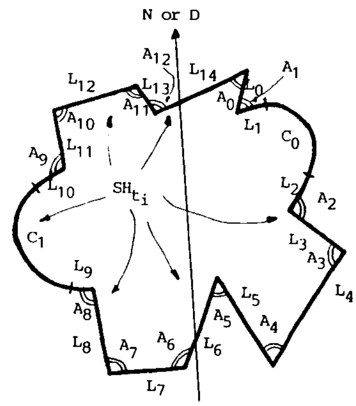


Figure 1: b) Extraction of the FNS shape

3. KNOWLEDGE EXTRACTION AND REPRESENTATION FROM THE FREE NAVIGATION SPACE

In this section the extraction and represention of knowledge from the geometric form of the current free navigation space $FNS^n(t(i))$ perceived by a moving robot R(n) are presented. In particular, the extraction is related with the construction of the shape SH $(FNS^n(t(i))$ of the current free space. Figure 1 shows graphically the generation of the shape. Then, the relationships among the straight line and curve line segments of the shape $SH^n(t(i))$ are defined and the representation of the shape (knowledge) with the use of syntactic and semantic information is obtained by using directed graph with attributes [6].

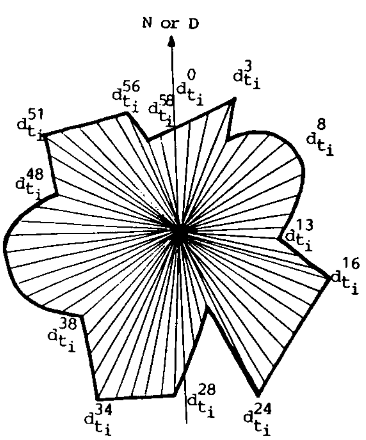


Figure 1: a) Construction of the FNS shape

4. SYNTHESIS OF CONSECUTIVE SHAPE-GRAPHS

In this section, we describe the synthesis of successive shapes expressed in graph forms [5].

Two shapes $SH^n(t(i))$, $SH^n(t(j))$, with i#j can be considered for synthesis if the graph form of these shapes satisfy the following basic proposition:

Proposition : Two shapes $SH^n(t(i))$, $SH^n(t(j))$, i#j, extracted by the same robot R(n) at two consecutive time intervals in the same navigation space, can be considered as candidates for composition into a new shape $SH^n(t(ij))$ if and only if their graph forms have at least one common node, $G^n(t(i)) \ni N(k) = N(m) \in G^n(t(j))$ with the same properties Pr and the similar relationships Rs with the other nodes, where Pr={size,color,length,curvature,etc}, and Rs={connectivity, parallelism, symmetry, relative-distance, relative-magnitude,etc}.

Starting the synthesis process, we have to search initially the graph form of each shape for graph-nodes that satisfy the proposition above. If we detect such a node in the first graph, then we save its characteristics and we proceed with the nodes of the second candidate graph. If there is at least such a node in the second graph, then we attempt to match its characteristics with the corresponding ones of the node, which belongs in the first graph. If there is a successful matching, then these particular nodes will be the starting point for the synthesis of the two graphs. More specifically, the synthesis process of two consecutive shapes (using their graph-forms) is based especially on the connectivity relationship (angle) of the nodes with the same properties. Figure 2 shows graphically, the synhesis of straight line segments where the segment with the maximum clockwise angle is eliminated. The synthesis process of the rest nodes for these two candidate graphs is based on the detection of closed subgraphs and determination of their "extended" new subgraph forms.

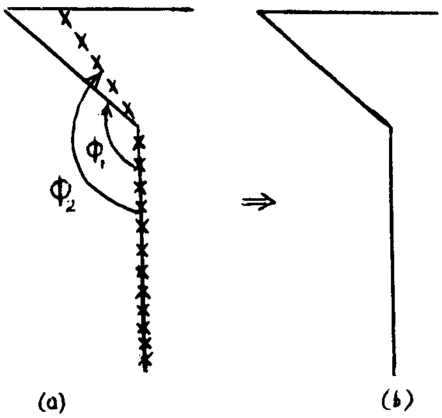


Figure 2: Synthesis of staight line segments taken from two consecutive shapes.
a) Matching of the segments
b) The new segments after synthesis

At the end of the synthesis process, the two shapes generated a new shape which represents the map of two consecitive free navigation spaces. By repiting this synthesis process, finally the map of the navigation space will be produced. Noe there is a critical question. For how long does a robot have to travel in order to generate a complete 2-D map of an unknwon space? The next section attepts to give some answers to the question above.

5. GENERATION OF THE SPACE MAP BY TRAVELING A SHORT DISTANCE

The generation of the space map requires the classification of the unknown space into three main categories:

. *Space without Obstacles*
. *Space with stationary obstacles*
. *Space with moving obstacles*

5.1. SPACE WITHOUT OBSTACLES

Here the problem is to generate the 2-D map of the navigation space by traveling a short distance, under the condition that no obstacles exist inside the space. The solution is rather trivial if the shape of the space is regular geometric one, as shown in figure 3. In this case, the shape of the space is simple and the center of gravity (or geometric center) Cgis ease to be located. Thus, if a robot detects such a shape then it calculates the center of gravity. This means that the robot has to travel a distance d[p(x,y),Cg] from its current location p(x,y) to Cg. When the robot will reach the Cg then its confidence function takes its maximum value, thus the 2-D map can be generated.

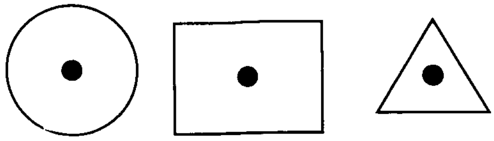


Figure 3: Some primitive geometric shapes

In case however where the shape of the navigation space is not a regular geometric shape, the problem becomes more complex, see examples in figure 4. For these cases, we partition the shape of the navigation space in order to obtain a set of simple (primitive) geometric regular shapes, figure 5. Thus, for each primitive we define a center of gravity. At this point, all the centers of gravity are connected by straight line segments (if possible) and the robot has to reach the nearest center of gravity. From that center of gravity the robot will travel on the line segments (gravity-line), which connect the centers of gravity, by minimizing the traveling distance for the generation of the 2-D space map, see figure 6. Note that the confidence function takes its maximum value by traveling on the gravity-line. It is also important to be noticed that if the

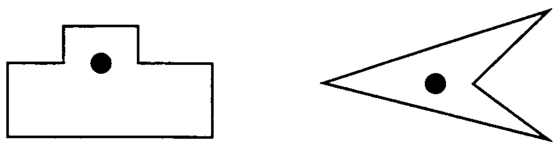


Figure 4: Complex geometric shapes

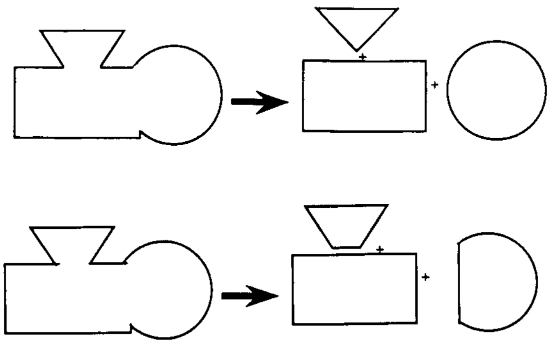


Figure 5: Partitioning of a geometric shape into a set of primitive shapes.

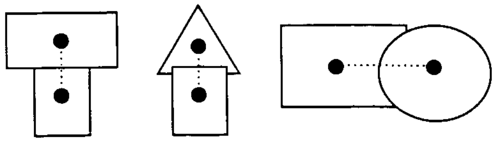


Figure 6: Connection of the centers of gravity.

navigation space is partitioned into "n" regular geometric shapes and if all the centers of gravity are connected within the space, then visiting these Cg points in an optimal way is equivalent to finding a permutation g1,g2,g3,...gn, which minimizes the total traveling distance (traveling salesman problem). A heuristic solution is proposed here in order to avoid such an NP complete problem. The solution is coming by generating the "skeleton" (thinning) of the navigation space [9], see figure 7. Thus, the complexity of the problem is reduced by eliminating the partitioning process and the complexity of the connection of the centers of gravity. In this case, the robot has to travel on the skeleton line in order to generate the space map.

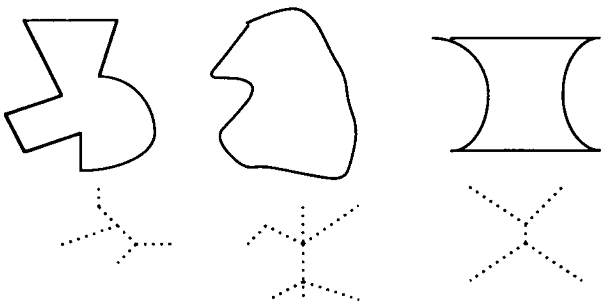


Figure 7: Irregular shapes and their skeletons

5.2. SPACE WITH STATIONARY OBSTACLES

The solution to this particular problem is similar to the generation of the skeleton line. Then the skeleton line is converted into an equivalent graph, see figure 8. The generation of the complete graph is based on the synthesis of the current graphs generated by the movements of the robot. Firtsly, the robot uses its starting point as the "root" of the current graph G1 (see a1 in figure 8). Thus each segment of the current skeleton represents a branch of the graph. If the robot will be moved to a3 point, then a new graph G2 is generated and the new graph is synthesized with the pervious one for the production of a graph G3 which represents two consecutive free navigation spaces.

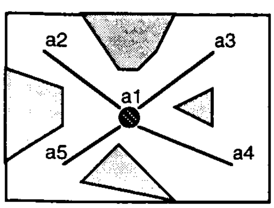
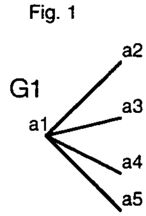


Fig. 1



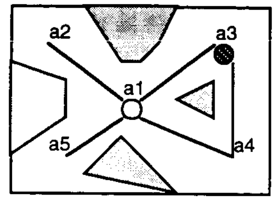


Fig. 3

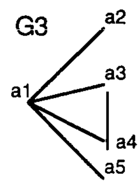
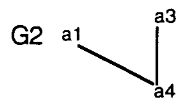


Fig. 4



G3 = G1 ⊕ G2

Figure 8: Graphical representation of a space with stationary objects, and the graph representation of the free space.

The important feature of the G3 graph is that it has the ability to detect some graph nodes, which were perceived from the previous position. Such a case is the point a4. Thus this important observation plays a very significant role for the shorter distance to be traveled. More specifically, the robot has information related with the point a1, a3 and a4, thus there is no need for it to travel on the segment (a1a4), since it "knows" some information about the shape and the path related to a1a4 brach of the graph. Another interesting point is the generation of intersected nodes. This means that, as the robot is moving and the new graph is generated, there are some locations from which the graph generates new braches, which intersect other braches constrcuted previously. The reason is that the robot can see areas viewed before from different angle. Thus it combines that knowledge for the generation of the new graph.

5.3. SPACE WITH MOVING OBSTACLES

In this case, which is also the most complex, the methodology followed is similar with the skeletonization of the navigation space with the only difference that the moving objects inside the space request one extra processing step. This step is the real-time detection of the moving objects in the each current free navigation space [6] and the appropriate reconstruction of the new graph.
The complexity of this case increases significantly when the number of moving objects in the navigation area increases too.

6. CONCLUSIONS

In this paper, a methodology for the generation of the 2-D space map by traveling a short distance was presented. More specifically, the study of the problem was partitioned into three subcases, 1) space without obstacles; 2) space with stationary obstacles; and 3) space with moving obstacles. The advantage of this methodology is the ability to minimize the redundancy during the traveling and maximize the confidence function for the generation of the 2-D map. The main disadvantage of the methodology proposed in this paper is the risk of generating a 2-D space map with some lost of information.

REFERENCES

[1] T.Lozano-Perez,Spatial Planning: A configuration space approach, IEEE Trans. on Computers, 32,2,1983, pp.108-120.

[2] R.Brooks,Solving the find-path problem by good representation of the free space,IEEE Trans. on SMC, 13,3,1983,pp.190-197.

[3] J.Barraquand and J.Latombe,Robot Motion Planning:A distributed representation approach,IEEE J. on Automation & Robotics, 1991

[4] N.Bourbakis, Real-time path planning autonomous robots in a 2-D unkonwon space,IJ.on Intelligence & Robotic Systems 4,1991,pp.333-362.

[5] N.Bourbakis, Knowledge-based acquisition during real-time path planning in unknown space, in Learning and Planning Methods and Applications, WSP,311-331,1991

[6] N.Bourbakis, An knowledge acquisition scheme using attributed graphs,Technical Report 1989.

[7] L.Gonzenes,Collision avoidance for robots in an environmental flexible assembly cell,Proc. IEEE Conf. on Robotics,CA,1984.

[8] M.Maas, A.Mogzadeh and N.Bourbakis, Fusion of image and laser sensory data for 3-D modeling of the navigation space, Int. AIAA Conf. on Intelligent Robots, March 1994, TX

[9] N.Bourbakis N.Stefenssen and B.Saha, A parallel skeletonization methodology and its VLSI implementation, TR 1993, sub. to Int. Journal.