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

A Topological Approach of Path Planning for Autonomous Robot Navigation in Dynamic Environments

Aswin Thomas Abraham, Shuzhi Sam Ge† and Pey Yuen Tao

***Abstract*— This paper proposes a novel approach, Simultaneous Path Planning and Topological Mapping (SP^{2}ATM), to address the problem of path planning by registering the topology of the perceived dynamic environment as opposed to the conventional grid representation. The local topology is encoded, concurrent and incremental with path planning, by extracting only the admissible free space. The resulting Admissible Space Topological Map (ASTM) then serves as the minimum information to facilitate path planning in the 3D configuration space. Experimental results obtained from our mobile robot X1 in a complex planar environment, validates completeness and optimality of the algorithm.**

I. INTRODUCTION

Past decades have attracted a great deal of interest in the fundamental capabilities in mobile robotics namely, path planning and map building. This enables the robot to navigate from an initial configuration to a final one, in an environment unknown *a priori*. A recent and extensive survey of the subject can be found in [1]. The conventional problem of *goal oriented path planning* involves robot guidance from a start point to a known goal and is said to be *complete*, if it does so, in finite time. In addition to this, robots today must be capable of mapping a bounded environment through systematic exploration i.e. perform an *exploratory path planning*, a problem in which no predefined goal exist. The literature sees Simultaneous Localization and Mapping (SLAM) as a related problem. However, since SLAM focuses on producing an accurate map from perceived information for localizaton [2], a robot which is not provided with any predefined waypoints, must employ a path planning algorithm while mapping an unknown space. Several elegant solutions such as the bug algorithm [3] and potential functions have been proposed [4], [5] without producing a map, and are highly desirable for local planning. For efficient path generation globally, grid mapping based algorithms were proposed as a simple solution [6], but suffers from inefficiencies in time and space. Topological representation, therefore, becomes the essential approach to map building for a complete path planning solution under low memory requirements. The map which generally takes the form of a graph [7], [8], provides improved flexibility for dynamic path planning which is otherwise quite tedious and inefficient in pure metric approaches [9]. The *exploratory path planning* problem was introduced in [10] as the *sightseer strategy* and is different from *coverage path planning* [11], [12] employed for applications such as floor cleaning. Exploration strategies are proposed in [13], [14] and a reactive system is proposed in [15], where a topological map was maintained to aid path planning. A high level map known as the Enhanced Topological Map was proposed in [16] and a navigation graph in [17] to plan paths among polygonal obstacles. [18] proposes the solution by maintaining shortest path trees and hierarchical approximate cell decomposition. The well known contribution of the Generalized Voronoi Graph (GVG) [19] proposes incremental construction in [20]. The Gap Navigation Tree (GNT) was proposed in [21], by employing only a gap sensor and a Next Best View (NBV) algorithm in [2] by introducing the concept of a safe region. The globally optimal Simultaneous Path Planning and Topological Mapping (SP^{2}ATM) algorithm presented in this paper, only utilizes a Admissible Space Topological Map (ASTM), constructed online without any preprocessing or query stages. The algorithm also incorporates both local and global planning including obstacle avoidance at single layer, a property inherent only in metric maps. Unlike many map building algorithms which model the obstacles in the scene, our focus is restricted to extracting admissible spaces in the environment which is simultaneous with map building and considering the robot's maneuverability at the same time. Optimal motion is made possible by employing *extended range detectors* [22] to guide the robot through a set of instant goals such as in [23], but through continuous replanning which prevents the robot from always focussing on a single region. The main contributions of this paper are:

(i) A single layered, convergent and complete SP^{2}ATM algorithm which produces a globally optimal solution for *goal oriented path planning* and *exploratory path planning* while the ASTM of an unknown, unstructured environment is simultaneously constructed, is presented.
(ii) The admissible space in the present sensor view which is a part of the work space, but not the whole environment itself, is modelled by a set of quadrilaterals, which enables creation of a global Admissible Space Topological Map (ASTM).

In Section II, the concepts of topological mapping is presented. The SP^{2}ATM algorithm is presented in Section III, experimental results in Section IV, followed by conclusions in Section V.

†Corresponding author. Tel.: +65 6516 6821; Fax: +65 6779 1103; Email: samge@nus.edu.sg.

The authors are with the Social Robotics Lab, Interactive Digital Media Institute (IDMI), and the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 {samge, aswinthomas, pytao}@nus.edu.sg

TABLE I

NOMENCLATURE

$\mathbf{q}_r$, ϕ_z	position of the robot and orientation from the global frame y axis;
$\mathbf{q}_i$, $\mathbf{q}_t$	position of the instant goal and final goal in the global frame;
Ω_{ws}, $\mathcal{S}_{AF}$	the robot's workspace and admissible free space;
w_r, w_v	actual and virtual base width of the robot;
d_{max}, d_{min}	maximum and minimum range of the sensor;
$\mathcal{M}$, $\mathcal{K}$	the ASTM and the topology graph;
$\mathcal{C}$, w_c, D_c, θ_c	a corridor, its width, length and inclination from the sensor frame y axis;
$\mathcal{L}_{kj}$, $\mathcal{L}_k$	a leaf and set of all leaves in a range scan;
$\mathcal{T}_j$, $q_{\mathcal{T}_j}$, $N_{\mathcal{K}}$	the j-th node, its coordinates in the global frame and the total number of nodes in $\mathcal{K}$;
$\mathcal{P}_{kj}$, $\mathcal{P}_k$	the j-th possibility and set of all possibilities in $\mathcal{A}_k$;
$\mathcal{P}_{n\mathcal{T}_j}$, $N_{\mathcal{P}_{\mathcal{T}_j}}$, $\mathcal{P}_{\mathcal{T}_j}$	the n-th possibility, the total number and the set of all possibilities assigned to $\mathcal{T}_j$;
$d(a,b)$	the Euclidian Distance between any two points a and b in the global frame;
$\mathcal{P}r_{kj}$, $\mathcal{P}r_k$	the j-th possibility and the set of all possibilities in $\mathcal{A}_k$ after rejection;
t_d, σ_g	the tolerance period and tolerance in association between $\mathbf{q}_r$ and $\mathbf{q}_i$;
$R_{i,j}$	the shortest route from $\mathcal{T}_i$ to $\mathcal{T}_j$ in $\mathcal{K}$;
$d_V(i,j)$, $d_R(i,j)$	the Visibility Distance and the Route Distance between two nodes $\mathcal{T}_i$ and $\mathcal{T}_j$ in $\mathcal{K}$;
$\mathcal{T}_s$, $\mathcal{T}_v(i,j)$	starting node and the visibility node in $R_{i,j}$;
$\mathcal{I}_g$	the instant goal;

II. TOPOLOGICAL MAPPING

Scan readings obtained from a range sensor are analyzed from its minimum range d_{min} to maximum range d_{max} to extract admissible free spaces $\mathcal{S}_{AF}$ ($\mathcal{S}_{AF} = \Omega_{ws} \setminus \bigcup_{i=1}^{i=m} C_{\delta i}$) in the work space Ω_{ws}, which can be defined as portion of the environment that is traversable by the robot. $C_{\delta i}$ is the obstacle where $0 \le i \le m$. To ensure complete maneuverability of the robot, a virtual base width $w_v (w_v > w_r)$ is considered, where w_r is the actual base width.

Definition 1 (Corridor): A corridor denoted by $\mathcal{C} = (w_c, \theta_c, D_c)$ is a set of points of a rectangle, where w_c is the breadth, D_c the length and θ_c its orientation from the y axis in the sensor frame. It never encloses any point of the range scan in its area $w_c D_c$ and always rotates about the midpoint of its breadth $0.5 w_c$ placed at the origin of the sensor frame.

Definition 2 (Leaf): A leaf $\mathcal{L}_{kj} \in \mathcal{L}_k$ is a portion in the range scan for which a set of corridors of same D_c and width $w_c = w_v$ exist, such that it is enclosed by the same set of obstacles. $\mathcal{L}_k$ is the set of all leafs in all directions in a single range scan. The resulting width of the leaf is w_L and is considered only if $w_L \geqslant w_v$.

Analysis stops when the portion of the scan under consideration does not converge to a leaf. $\mathcal{L}_k$ takes the shape of a quadrilateral and Figure 1 shows leaves obtained in several scans. It can be clearly seen that a leaf $\mathcal{L}_{kj}$ does not cover the whole area between obstacles and the robot can now traverse through any point in the quadrilateral. Since the 3D world can be modelled by a combination of such 2D planes with several leaves; through analysis in a row wise manner, a complete representation in terms of $\mathcal{S}_{AF}$ is available. The resulting structure is made efficient by stitching similar segments in consecutive planes to form a single 3D quadrilateral.

Definition 3 (Possibility): A possibility $\mathcal{P}_k \in \mathcal{P}, \mathcal{P}_k \subseteq \mathcal{L}_k$ is a point in the global navigation frame. It can be described as a single point chosen out of all the points that describe $\mathcal{L}_{kj}$.

The Admissible Space Topological Map (ASTM) denoted by $\mathcal{M}$ is a two layer structure which illustrates the general topology of Ω_{ws} with $\mathcal{P}_k$ in its lower layer and an undirected graph $\mathcal{K}$ in the upper layer. $\mathcal{K}$ contains a set of nodes connected by edges expressed as

$$\mathcal{K} = \{\mathcal{T}, L_{\mathcal{K}}\} \tag{1}$$

where $\mathcal{T}_j \in \mathcal{T}$ is the j-th topology node and $L_{\mathcal{K}(i,j)} \in L_{\mathcal{K}}$ is the edge between $\mathcal{T}_j$ and another topology node $\mathcal{T}_i$. It depicts the overall topology of the environment with each node $\mathcal{T}_j$ describing $\mathcal{L}_k$ perceived from its location. This could result in partial overlap of leaves, but does not affect path planning.

Definition 4 (Topology Node): A topology node $\mathcal{T}_j$, is denoted as $\mathcal{T}_j = (q_{\mathcal{T}_j}, \mathcal{P}_{\mathcal{T}_j}, E_l, V_r)$; $l = 1, 2$. The four elements comprising a topology node are described as follows:

(i) $q_{\mathcal{T}_j}$ (Node Position): the node coordinates in the global navigation frame.

(ii) $\mathcal{P}_{\mathcal{T}_j}$ (Possibility): a set of all possibilities assigned to $\mathcal{T}_j$, $\mathcal{P}_{\mathcal{T}_j} = \{\mathcal{P}_{1\mathcal{T}_j}, \mathcal{P}_{2\mathcal{T}_j}, \ldots, \mathcal{P}_{n\mathcal{T}_j}\}$

(iii) E_l (Type): The node type $E_l \in E$ depicts the characteristic of the node.

1) E_1 (Explored): A node which has no possibilities i.e. $\mathcal{P}_{\mathcal{T}_j} = \emptyset$.
2) E_2 (Unexplored): A node having at least one possibility i.e. $\mathcal{P}_{\mathcal{T}_j} \neq \emptyset$.

(iv) V_r (Vicinity circle): an imaginary circle of radius r_t represents the region of coverage of a node in the map.

The ASTM can therefore be expressed as:

$$\mathcal{M} = \{\mathcal{K}, \mathcal{P}, L_{\mathcal{K}\mathcal{P}(i,k)}\} \tag{2}$$

where $L_{\mathcal{K}\mathcal{P}(i,k)} \in L_{\mathcal{K}\mathcal{P}}$ is the bridge between the i-th node $\mathcal{T}_i$ in $\mathcal{K}$ and the possibility $\mathcal{P}_k$.

A. Possibility Generation and Rejection

The ends of the leaf $\mathcal{L}_{kj}$ are transformed from the sensor frame to the global frame to form $\mathbf{q}_S$ and $\mathbf{q}_E$, the start and end position of the segment respectively in the global frame. For *exploratory path planning*, the center of the leaf is assigned as the j-th possibility using the midpoint formula. In *goal oriented path planning*, the ends of a segment are the most useful portions since it would give rise to efficient obstacle avoidance towards the goal. The segment end which is closer to the final goal $\mathbf{q}_t$ is chosen using the distance formula. $\mathcal{P}_{kj} = \min\{d(\mathbf{q}_S, \mathbf{q}_t), d(\mathbf{q}_E, \mathbf{q}_t)\}$ where $d(\mathbf{q}_S, \mathbf{q}_t)$ is the Euclidian distance between $\mathbf{q}_S$ and $\mathbf{q}_t$, $d(\mathbf{q}_E, \mathbf{q}_t)$ the Euclidian distance between $\mathbf{q}_E$ and $\mathbf{q}_t$.

The set of all possibilities $\mathcal{P}_k$ generated, are rejected by a possibility rejection criterion for easier decision process. The newly perceived set of possibilities $\mathcal{P}_k$ satisfying the criterion

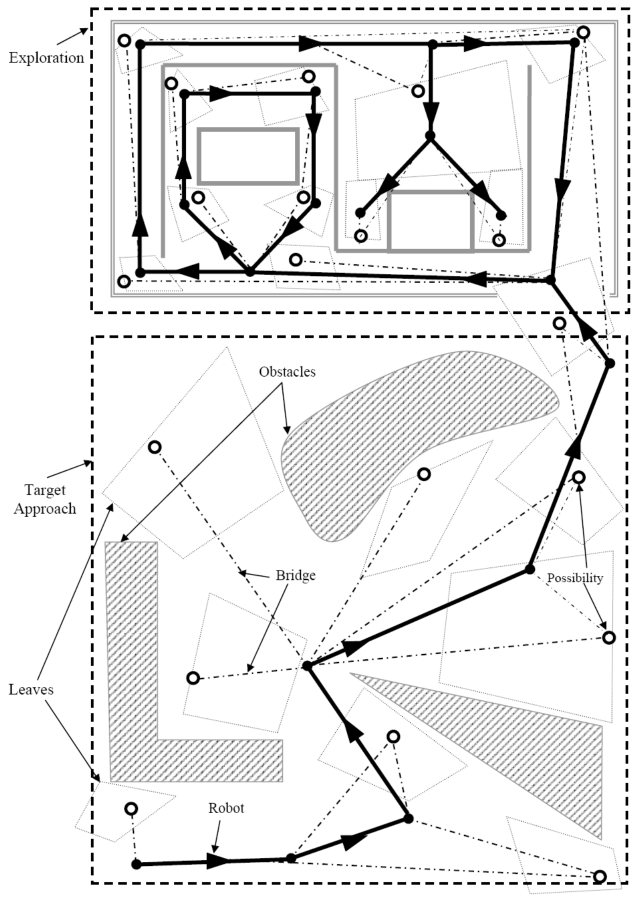


Fig. 1. Structure of the ASTM in a plane

are rejected, resulting in a set of possibilities $\mathcal{P}r_k$. During map building, it is also necessary to revise the existing set of possibilities in $\mathcal{M}$ for optimal coverage. The rejection criterion is described below where $\mathbf{q}_r$ denotes the position of the robot in the global frame.

if $d(\mathcal{P}_{kj}, \mathbf{q}_r) \leqslant d_{pr}(w_r)$ **then**
  Possibilities within the near vicinity of the robot
  $\mathcal{P}_k = \mathcal{P}_k \cap \mathcal{P}_{kj}$
if $d(\mathcal{P}_{kj}, q_{\mathcal{T}_j}) \leqslant r_t$ and $\mathcal{C}(w_r, \theta_{\mathcal{T}_j}, d(\mathbf{q}_r, q_{\mathcal{T}_j}))$ **then**
  Possibility $\mathcal{P}_{kj}$ within the vicinity circle of the completely visible $\mathcal{T}_j$
  $\mathcal{P}_k = \mathcal{P}_k \cap \mathcal{P}_{kj}$
if $d_\theta \geqslant d_{pt}; \theta_{\mathcal{T}_j} \leqslant \theta \leqslant \theta_{\mathcal{P}_{kj}}$ **then**
  Possibility $\mathcal{P}_{kj}$ within the vicinity circle of the completely visible $\mathcal{T}_j$ but not accessible by it.
  $\mathcal{P}_k = \mathcal{P}_k \cap \mathcal{P}_{kj}$
if $d(\mathcal{P}_{n\mathcal{T}_j}, \mathcal{P}_{kj}) \leqslant \sigma_p$ **then**
  Assigned Possibility $\mathcal{P}_{\mathcal{T}_j}$, $j \neq r$ very close to $\mathcal{P}_{kj}$
  if $d(\mathcal{P}_{kj}, \mathbf{q}_r) < d(\mathcal{P}_{kj}, q_{\mathcal{T}_j})$ **then**
    $\mathcal{P}_{\mathcal{T}_j} = \mathcal{P}_{\mathcal{T}_j} \cap \mathcal{P}_{n\mathcal{T}_j}$
  else
    $\mathcal{P}_k = \mathcal{P}_k \cap \mathcal{P}_{kj}$
if $d(\mathcal{P}_{n\mathcal{T}_j}, \mathbf{q}_r) < d(\mathcal{P}_{n\mathcal{T}_j}, q_{\mathcal{T}_j})$ **then**
  Assigned possibility $\mathcal{P}_{\mathcal{T}_j}$, $j \neq j_r$ closer to the node $\mathcal{T}_r$ than $\mathcal{T}_j$
  if $\mathcal{C}(w_r, \theta_{\mathcal{T}_j}, d(\mathbf{q}_r, q_{\mathcal{T}_j}))$ **then**
    $\mathcal{P}_{\mathcal{T}_j} = \mathcal{P}_{\mathcal{T}_j} \cap \mathcal{P}_{n\mathcal{T}_j}, \mathcal{P}_{\mathcal{T}_r} = \mathcal{P}_{\mathcal{T}_r} \cup \mathcal{P}_{n\mathcal{T}_j}$
if $d(\mathbf{q}_r, \mathcal{P}_{n\mathcal{T}_j}) < \lambda_{pr}$ **then**
  Assigned possibility $\mathcal{P}_{\mathcal{T}_j}$ they provide no more information when close to the robot.
  if $\mathcal{C}(w_r, \theta_{\mathcal{P}_{n\mathcal{T}_j}}, d(\mathbf{q}_r, \mathcal{P}_{n\mathcal{T}_j}))$ **then**
    $\mathcal{P}_{\mathcal{T}_j} = \mathcal{P}_{\mathcal{T}_j} \cap \mathcal{P}_{n\mathcal{T}_j}$

where $d_{pr}(w_r)$ is the radius of a robot vicinity circle, centered about the origin in the robot frame, $\theta_{\mathcal{T}_j}$ the angle $\mathcal{T}_j$ makes with the y axis in the sensor frame. $d_{pt} = \max\{d(\mathcal{P}_{kj}, q_{\mathcal{T}_j}), d(\mathcal{P}_{kj}, \mathbf{q}_r)\}$ and $\theta_{\mathcal{P}_{kj}}$ is the angle the j-th possibility $\mathcal{P}_{kj}$ makes with the y axis in the sensor frame. r is the identity of the node $\mathcal{T}_r$ to which the present robot position is associated and σ_p is the matching tolerance between any two possibilities which determines the degree of overlapping between quadrilateral areas, λ_{pr} the robot-possibility rejection distance and $\theta_{\mathcal{P}_{n\mathcal{T}_j}}$ the inclination of $\mathcal{P}_{n\mathcal{T}_j}$ with the y axis in the sensor frame.

B. Updating the ASTM

The topology node $\mathcal{T}_r$ to which $\mathbf{q}_r$ is associated must be known at any instant of time. Since the vicinity circle V_r of a set of topology nodes overlap with each other, the present robot position $\mathbf{q}_r$ could lie in more than one such circle, ensuring that the robot does not lose track of its position in $\mathcal{M}$. From the overlapping set of nodes, $\mathcal{T}_r$ is assigned to be the one closest to $\mathbf{q}_r$. A new node $\mathcal{T}_r$ is registered as explored i.e. E_1 in $\mathcal{M}$ and connected to $\mathcal{T}_r(t - t_s)$, the robot associated node known in the previous iteration, by an edge $L_{\mathcal{K}(r, r(t-t_s))}$, if either $\mathcal{T}_r$ is not obtained i.e. the robot has moved into a new region in Ω_{ws}, or a new set of $\mathcal{P}_k$ is obtained. t_s is the sampling time of the planner.

If the total number of nodes $N_{\mathcal{K}} = 0$, the node $\mathcal{T}_r$ is termed as the starting node $\mathcal{T}_s$ in Ω_{ws}. The new possibility set $\mathcal{P}r_k$ after rejection process are directly assigned to $\mathcal{T}_r$ if it is of type E_1 i.e. explored. If it is unexplored (E_2), the set $\mathcal{P}r_k$ is compared with already assigned possibilities $\mathcal{P}_{\mathcal{T}_r}$ in $\mathcal{T}_r$ to avoid any repetition.

$$d(\mathcal{P}r_j, \mathcal{P}_{n\mathcal{T}_r}) \geqslant \sigma_p; \quad 0 < j \leqslant \mathcal{P}r, \; 0 < k \leqslant N_{\mathcal{K}} \tag{3}$$

where $\mathcal{P}r_j$ is the total number of possibilities. All nodes $\mathcal{T}_j$ with possibility status $\mathcal{P}_{\mathcal{T}_j} = 0$ and $\mathcal{P}_{\mathcal{T}_j} > 0$ are updated to be of type E_1 and E_2 respectively. Nodes $\mathcal{T}$ are appended in $\mathcal{M}$ concurrently during robot motion. The robot traverses through an already mapped Ω_{ws} by following $\mathcal{T}$ in $\mathcal{K}$ along a designed route.

Definition 5 (Route): A route $R_{i,j} \in R$ denoted by $R_{i,j} = (S_{i,j}, N_{i,j}, d_R(i,j))$ is the shortest path in $\mathcal{K}$ from $\mathcal{T}_i$ to $\mathcal{T}_j$, where $S_{i,j}$ is the set of all nodes in the order $n = [i \dots j]$, both i and j inclusive. $N_{i,j}$ is the number of nodes in the route and $d_R(i,j)$ is the route distance.

The Bellman-Ford algorithm [24] is applied to find the shortest route $R_{i,j}$. The Bellman-Ford equation is given as:

$$d_R(i,j) = \min_{k \in neighbors} \{d(i,k) + d_R(k,j)\} \tag{4}$$

Definition 6 (Visibility distance): The visibility distance denoted by $d_V(i,j)$ is the Euclidian distance from $\mathbf{q}_r$ to the farthest visible node $\mathcal{T}_v(i,j)$ in the route $R_{i,j}$. $\mathcal{T}_v(i,j)$ is termed as the visibility node in the route and satisfies

$$\max_{r\leqslant k\leqslant j}\{d(\mathbf{q}_r, q_{\mathcal{T}_k})\};\quad \mathcal{C}(w_r, \theta_{\mathcal{T}_v(i,j)}, d(\mathbf{q}_r, q_{\mathcal{T}_v(i,j)})) \tag{5}$$

A new edge is created in between the present node $\mathcal{T}_r$ and any other node $\mathcal{T}_j$ if it satisfies the following:

(i) $j \neq r$.
(ii) $j \neq jn$ where $\mathcal{T}_{jn}$ is an existing immediate neighbor of $\mathcal{T}_r$ i.e. sharing a common edge.
(iii) There exists a corridor $\mathcal{C}(w_r, \theta_{\mathcal{T}_j}, d(\mathbf{q}_r, q_{\mathcal{T}_j}))$ to the node $\mathcal{T}_j$ from $\mathbf{q}_r$.
(iv) The cost of travel from $\mathcal{T}_j$ to $\mathcal{T}_r$ is large. $d(\mathbf{q}_r, q_{\mathcal{T}_j}) < \frac{d_R(j,r)}{\lambda_l}$ where λ_l is the route cost scale down factor.

III. The SP^{2}ATM Algorithm

The algorithm generates a new instant goal $\mathcal{I}_g \in \bigcup_{k=1}^{k=A_{tot}} \mathcal{P}_k$ or $\bigcup_{n=1}^{n=N_\mathcal{T}} \mathcal{P}_{\mathcal{T}_n}$, which is an intermediate point $\mathbf{q}_i$ in Ω_{ws} towards which the robot steers, for every sampling time t_s. The planner follows a state machine framework $\mathcal{G} = \{\mathcal{G}_1, \ldots, \mathcal{G}_5\}$, existing in a single state $\mathcal{G}_r$ at any instant of time. The different modes are Goal Approaching mode ($\mathcal{G}_1$), Braked mode ($\mathcal{G}_2$), Exploration mode ($\mathcal{G}_3$), Mapping mode ($\mathcal{G}_4$) or Trace Back mode ($\mathcal{G}_5$). The algorithm initially starts in the Braked mode, applies possibility rejection and redirects to $\mathcal{G}_1$, $\mathcal{G}_3$ or $\mathcal{G}_5$ as required. In $\mathcal{G}_4$, the algorithm continues map building with no $\mathcal{I}_g$ generation. For the goal approaching mode the path planner provides instant goals $\mathcal{I}_g$ enabling the robot to move along a path which leads to the goal at $\mathbf{q}_t$.

1: **if** $d(\mathbf{q}_r, \mathbf{q}_t) \leqslant \sigma_g$ **then**
2: Goal Reached, $\mathcal{G}_r = \mathcal{G}_2$, GOTO step 9
3: **if** $\mathcal{C}(w_r, \theta_t, d(\mathbf{q}_r, \mathbf{q}_t))$ exist **then**
4: $\mathbf{q}_i = \mathbf{q}_t$, GOTO Step 9
5: Search $\mathcal{P}_{n\mathcal{T}_j}$ satisfying the minimum path length equation

$$\min_{0<j\leqslant N_\mathcal{T}}\{\min_{0<k\leqslant N_{\mathcal{P}_{\mathcal{T}_j}}}\{d(q_{\mathcal{T}_v(i,j)}, \mathbf{q}_r) + d_R(v,j) + d(\mathcal{P}_{n\mathcal{T}_j}, q_{\mathcal{T}_j}) + d(\mathcal{P}_{n\mathcal{T}_j}, \mathbf{q}_t)\}\} \tag{6}$$

6: **if** $\mathcal{C}(w_r, \theta_{\mathcal{P}_{n\mathcal{T}_j}}, d(\mathbf{q}_r, \mathcal{P}_{n\mathcal{T}_j}))$ exist **then**
7: $\mathbf{q}_i = \mathcal{P}_{n\mathcal{T}_j}$, GOTO Step 9
8: Move to node $\mathcal{T}_j$, $\mathcal{G}_r = \mathcal{G}_5$
9: EXIT

where θ_t is the angle the goal makes with the y axis in the sensor frame, $d_R(v,j)$ the route distance of the route $R_{v,j}$ and $d(\mathcal{P}_{n\mathcal{T}_j}, \mathbf{q}_t)$ is the Euclidian distance between $\mathcal{P}_{n\mathcal{T}_j}$ and $\mathbf{q}_t$. The algorithm also produces $\mathbf{q}_i$ to explore the entire bounded space Ω_{ws}.

1: **if** $\mathcal{P}_{\mathcal{T}_j} = \emptyset;\ 0 < j \leqslant N_\mathcal{T}$ **then**
2: **if** $d(\mathbf{q}_{\mathcal{T}_s}, \mathbf{q}_r) \leqslant \sigma_g$ **then**
3: Exploration complete, $\mathcal{G}_r = \mathcal{G}_2$, GOTO step 16
4: **else**
5: Move back to start node $\mathcal{T}_s$, $\mathcal{G}_r = \mathcal{G}_5$
6: GOTO step 16
7: Compute $\mathcal{P}_{n\mathcal{T}_r}$ satisfying the nearest active possibility equation

$$\min_{0<n<N_{\mathcal{P}_{\mathcal{T}_j}}}\{d(q_{\mathcal{T}_r}, \mathcal{P}_{n\mathcal{T}_r})\} \tag{7}$$

8: $\mathbf{q}_i = \mathcal{P}_{n\mathcal{T}_r}$, GOTO step 16
9: **if** $E_{\mathcal{T}_r} = E_2$ **then**
10: **if** $\mathbf{q}_i = \mathcal{P}_{n\mathcal{T}_r}$ **then**
11: **if** $d(\mathbf{q}_r, \mathbf{q}_i) \leqslant \sigma_g$ **then**
12: Update a node $\mathcal{T}_r$ in $\mathcal{K}$
13: **if** $E_{\mathcal{T}_r} = E_1$ **then**
14: Search and move to $\mathcal{T}_j$ satisfying

$$\min_{0<j\leqslant N_\mathcal{T}}\{d(\mathbf{q}_r, q_{\mathcal{T}_j})\};\ E_{\mathcal{T}_j} = E_2 \tag{8}$$

15: $\mathcal{G}_r = \mathcal{G}_2$
16: EXIT

Step 12 in the algorithm is a map updation in addition to the node registration procedure mentioned in Section II-B. When the robot is moving towards $\mathcal{I}_g$ which is a possibility $\mathcal{P}_{n\mathcal{T}_r}$ of the present unexplored node $\mathcal{T}_r$, a new node $\mathcal{T}_j$ is updated having an edge $L_K(r,j)$ connected to $\mathcal{T}_r$. In both path planning tasks discussed, situations arise where it is necessary to move back to a destination node $\mathcal{T}_d$ in $\mathcal{K}$. Instant goals $\mathcal{I}_g$ are generated to enable the robot to move along $\mathcal{K}$ by computing the route to $\mathcal{T}_d$, $R_{r,d} = [\mathcal{T}_r \ldots \mathcal{T}_d]$.

1: **if** $d(\mathbf{q}_r, q_{\mathcal{T}_d}) \leqslant r_t$ **then**
2: Switch to previous state $\mathcal{G}_1$ or $\mathcal{G}_3$
3: **if** $\mathcal{T}_v(r,d)$ exist; $\mathcal{T}_v(r,d) \neq \mathcal{T}_r$ **then**
4: $\mathbf{q}_i = \mathbf{q}_{\mathcal{T}_v(r,d)}$, GOTO step 8
5: **else**
6: **if** $r < d$ **then**
7: $\mathbf{q}_i = \mathbf{q}_{r+1}$
8: EXIT

where $\mathcal{T}_v(r,d)$ is the visibility node in the route $R_{r,d}$. The SP^{2}ATM algorithm produces a complete solution in most dynamic cases through continuous replanning since the AST only registers $\mathcal{S}_{AF}$. However, if an assigned possibility $\mathcal{P}_{\mathcal{T}_j}$ or registered node $\mathcal{T}_j$ is later inaccessible, the planner does not converge. This means that the robot cannot reach the vicinity circle V_r of $\mathcal{T}_j$ or $\mathcal{P}_{\mathcal{T}_j}$ within σ_g. The following additional steps are appended to the existing algorithm.

1: **if** $\mathcal{C}(w_r, \theta_{\mathcal{P}_{n\mathcal{T}_j}}, d(\mathbf{q}_r, \mathcal{P}_{n\mathcal{T}_j}))$ does not exist **then**
2: **if** $|t - t_p| > t_d$ **then**
3: Inaccessible possibility, $\mathcal{K} = \mathcal{K} \cap \mathcal{P}_{n\mathcal{T}_j}$
4: **else**
5: GOTO step 13
6: $t_p = t$
7: **if** $\mathcal{G}_r = \mathcal{G}_5$ **then**
8: **if** $\mathcal{C}(w_r, \theta_{\mathcal{T}_j}, d(\mathbf{q}_r, q_{\mathcal{T}_j}))$ does not exist **then**
9: **if** $|t - t_n| > t_d$ **then**
10: Inaccessible node, $\mathcal{K} = \mathcal{K} \cap (\mathcal{T}_j, L_\mathcal{T}(r,j))$
11: **else**
12: GOTO step 14
13: $t_n = t$
14: EXIT

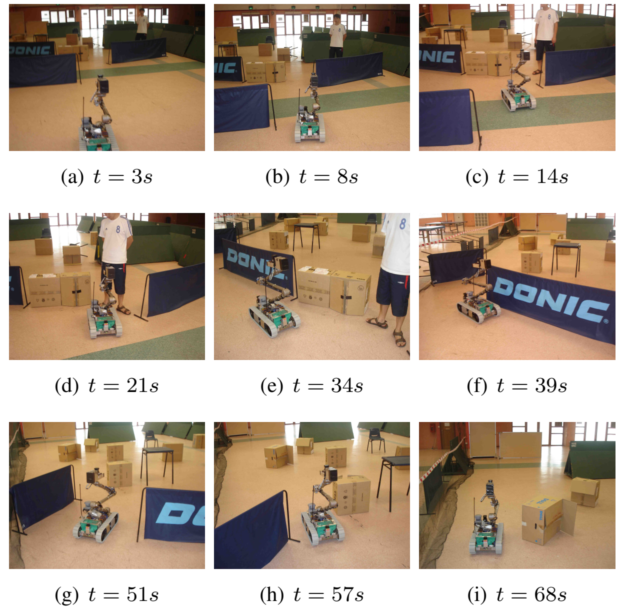
where t_p and t_n are the time at which an inaccessible possibility and node was found. t_d is termed as the tolerance period for which the planner waits to reconfirm the inaccessibility of a node or possibility.

IV. Experimental Result

The experiments were conducted on our robot X1 equipped with a Hokuyo UTM-30LX laser scanner ($d_{max} = 30m$) interfaced to a Pentium M, 1.6GHz (1Gb RAM) computer. For the localization of the robot, we used a simple system consisting of the onboard odometry with a single axis fiber optic gyroscope based on [25]. The SP^{2}ATM algorithm steered X1 while participating for the TechX Challenge competition, Singapore, 2008.

For testing the performance of the goal approaching mode, a goal was assigned 18m away from the robot at an angle 5° from the global frame y axis. The series of events and map building process are shown in Figures 2 and 3 respectively. While mapping, crosses in magenta are the explored nodes, orange crosses are unexplored and orange lines are connections between them. The line extending towards a possibility is in green, obstacles as red and the instant goal, a green cross. Each grid in grey represent $1m^2$ area and the goal is shown as a white cross. In Figure 3(a) and 2(b), it can be seen that a possibility very close to the target is visible to the robot. It chooses to move towards it in Figure 2(c) and 3(b). However in Figure 2(d) and 3(c) it can be seen that the path is blocked by a person and when the visibility to the instant goal fails, the robot waits for $t_d = 5$ seconds, reconfirms it and deletes the possibility from memory. It then moves towards the next best possibility as shown in Figure 3(d) and 2(f). In Figure 3(f) the target is directly visible to the robot and it moves towards it, completing the path planning in 1 min 8 sec. Nodes were registered when new possibilities are found while focussing only on the goal. This is the reason why a few possibilities can be seen left behind on either side of the path travelled. It deals well with close obstacles without relying on any collision avoidance algorithm.

In order to examine the exploration behavior, the robot was put to test in a small area of $150m^2$ with a person moving around the bounded region. A portion of the environment was also opened up for the robot to place a new possibility outside the bounded area and was closed later to observe the behavior. The portion of the experiment which adapts to the situation is shown with the map building process in Figure 4. The robot initially observes the room in the front and the corridor towards the right in Figure 4(a,b)). It finds two possibilities in the room and several of them in the corridor. These are associated with nodes and as the robot moves, the instant goal shifts and the robot moves further. It can be seen that the assigned possibilities are removed and shifted to forward nodes. In Figure 4(a), a portion of the environment is opened up and the laser rays fall out of the boundary. This enables the planner to place a possibility outside the barrier. In Figure 4(c), this area is closed up. However the possibility remains in memory until Figure 4(f) in which, the registered

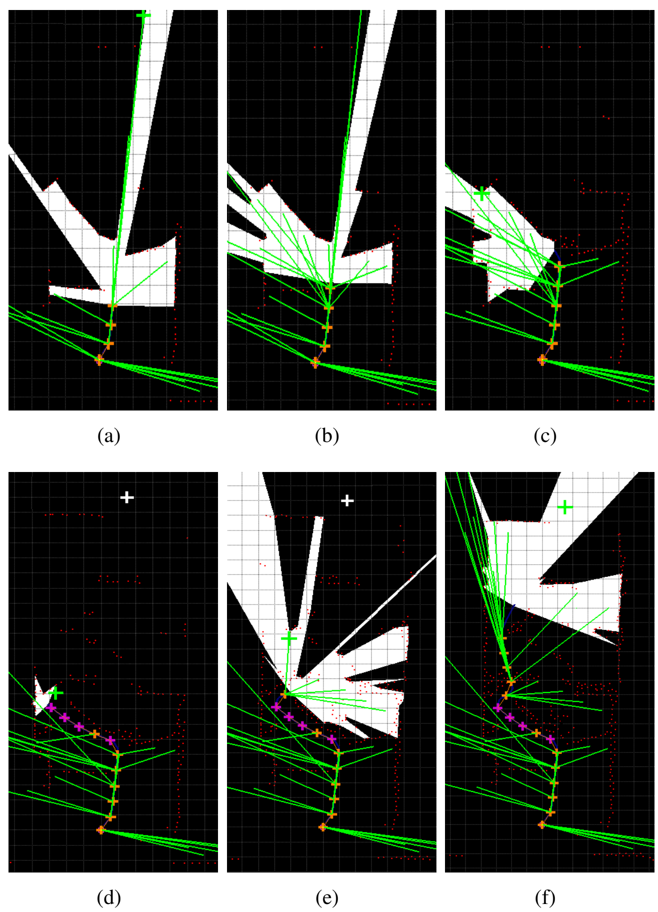


(a) $t = 3s$ (b) $t = 8s$ (c) $t = 14s$

(d) $t = 21s$ (e) $t = 34s$ (f) $t = 39s$

(g) $t = 51s$ (h) $t = 57s$ (i) $t = 68s$

Fig. 2. Snapshots showing dynamic nature of the goal approaching experiment in a $150m^2$ area



(a) (b) (c)

(d) (e) (f)

Fig. 3. Map built during goal oriented path planning in a static and dynamic environment

possibility outside the boundary is chosen and removed from the memory after t_d seconds. The exploration of the area was complete in 4 min 42 sec.

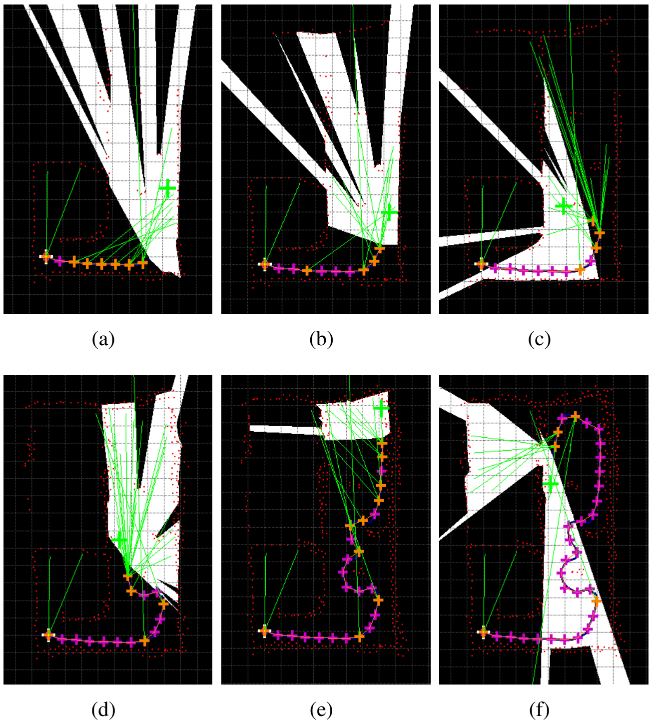


Fig. 4. Map building process showing dynamic nature of the exploration experiment in a $150m^2$ area

V. Conclusion

This paper formulates a new approach, Simultaneous Path Planning and Topological Mapping (SP^{2}ATM), to solve the problem of path planning, assisted by an incrementally and concurrently built topological map. First, a representation of the environment which demands minimal storage requirements was made possible through a topological approach. The traversable regions in the environment, as perceived by a range sensor directs the robot to construct the global topology of the environment represented by an Admissible Space Topological Map (ASTM). The map-based path planner was shown to guide the robot along an optimal path to move towards a known goal for *goal oriented path planning* and visit all spaces in an environment for *exploratory path planning*. Even though the experiments have been conducted without a local planner, the authors do not believe that SP^{2}ATM is a completely robust solution by itself. The robot motion could be very slow in constrained situations and would require the help of a simple collision avoidance module like a local potential function for faster convergence.

References

[1] S. S. Ge and F. Lewis, *Autonomous Mobile Robots: Sensing, Control, Decision Making and Applications*. Boca Raton, FL: CRC Press,Taylor and Francis Group, 2006.

[2] H. H. Gonzlez-Banos and J.-C. Latombe, "Navigation strategies for exploring indoor environments," *The International Journal of Robotics Research*, vol. 21, no. 1011, pp. 829 – 848, 2002.

[3] V. Lumelsky and A. A. Stepanov, "Path-planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape," *Algorithmica*, no. 2, pp. 403–430, 1987.

[4] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 278 –288, 1991.

[5] S. S. Ge and Y. J. Cui, "New potential functions for mobile robot path planning," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 5, pp. 615–620, 2000.

[6] A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE Journal of Robotics and Automation*, vol. 3, no. 3, pp. 249–265, 1987.

[7] B. Kuipers and Y.-T. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Robotics and Autonomous Systems*, vol. 8, no. 1-2, pp. 47–63, 1991.

[8] E. Remolina and B. Kuipers, "Towards a general theory of topological maps," *Artificial Intelligence*, vol. 152, p. 47 104, 2004.

[9] S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.

[10] V. J. Lumelsky, S. Mukhopadhyay, and K. Sun, "Dynamic path planning in sensor-based terrain acquisition," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 4, pp. 462–472, 1990.

[11] H. Choset, "Coverage for robotics - a survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, pp. 113–126, 2001.

[12] S. S. Ge and C. H. Fua, "Complete multi-robot coverage of unknown environments with minimum repeated coverage," *Proceedings of 2005 IEEE International Conference on Robotics and Automation*, pp. 715–720, April 2005.

[13] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes, "Robotic exploration as graph construction," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 6, pp. 859–865, 1991.

[14] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes, "On map exploration, validation, and self-location in a graph-like world," *Robotics and Autonomous Systems*, vol. 22, no. 2, pp. 159–178, 1997.

[15] M. J. Matarić, "Integration of representation into goal-driven behavior-based robots," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 304–312, 1992.

[16] B. S. Ryu and H. S. Yang, "Integration of reactive behaviors and enhanced topological map for robust mobile robot navigation," *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, vol. 29, no. 5, pp. 474–485, 1999.

[17] G. Foux, M. Heymann, and A. Bruckstein, "Two-dimensional robot navigation among unknown stationary polygonal obstacles," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 1, pp. 96 – 102, 1993.

[18] M. Barbehenn and S. Hutchinson, "Efficient search and hierarchical motion planning by dyanmically maintaining single-source shortest path trees," *IEEE Transactions on Robotics and Automation*, vol. 11, no. 2, pp. 198 – 214, 1995.

[19] C. Ó'Dúnlaing and C. K. Yap, "A "retraction" method for planning the motion of a disc," *Algorithmica*, vol. 6, pp. 104–111, 1985.

[20] H. Choset, S. Walker, K. Eiamsa-Ard, and J. Burdick, "Sensor-based exploration: Incremental construction of the hierarchical generalized voronoi graph," *The International Journal of Robotics Research*, vol. 19, no. 2, pp. 126–148, 2000.

[21] B. Tovar, R. Murrieta-Cid, and S. M. LaValle, "Distance-optimal navigation in an unknown environment without sensing distances," *IEEE Transactions on Robotics and Automation*, vol. 23, no. 3, pp. 506–518, 2007.

[22] E. U. Acar, H. Choset, and J. Y. Lee, "Sensor-based coverage with extended range detectors," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 189–198, 2006.

[23] S. S. Ge, X. C. Lai, and A. A. Mamun, "Boundary following and globally convergent path planning using instant goals," *IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics*, vol. 35, no. 2, pp. 240–254, 2005.

[24] R. E. Bellman, *Dynamic Programming*. Princeton, NJ: Princeton Univ. Press, 1957.

[25] H. Chung, L. Ojeda, and J. Borenstein, "Accurate mobile robot dead-reckoning with a precision-calibrated fiber-optic gyroscope," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 1, pp. 80–84, 2001.