

Multiple Mobile Robot Dispatch Strategy for Cooperative Applications in Confined Space

Doug Kim*

Computer Engineering Technology, Farmingdale State College (SUNY), Farmingdale, NY, USA

Abstract

In this paper, we propose a path planning and dispatch strategy of multiple mobile robots for cooperative applications in confined spaces. The proposed two level path planning method incorporates the segment reservation mechanism to maximize the utilization of robots. The two level path planning reduces the computational complexity while providing the flexibility for real-time adaptation. Any potential deadlock is completely eliminated and the starvation due to unpredictable service request patterns is avoided. Since the proposed methodology is used for confined areas, any possible path blockage due to idling robots is eliminated with idling robot relocation mechanism. The dynamic path planning is achieved by monitoring and controlling the speed of the robots. The dynamic re-planning is triggered with periodic monitoring of the service constraints. The method can be effectively used for low power multiple robot navigation applications and the performance of the proposed system is evaluated with large-scale simulations.

Introduction

There has been increasing interest in indoor service robot applications such as transporting specimens in the hospitals, mobilizing the wheelchairs, etc. [1-4]. Most of these applications require many robots to work simultaneously within the confined areas where many unexpected obstacles are presented. Multiple robot navigation and control in a large confined area is particularly interesting since the robots can be used to support various human activities [5,6]. One of the key objectives for the navigation system is to control the multiple robots simultaneously without interfering with human traffics. The robots are power constrained as well as time. The navigation system must guarantee to meet the power and timing requirements. Moreover, the robots must be able to handle static obstacles as well as unexpected dynamic obstacles. As the number of operating robots increases, the navigation system must be capable of minimizing performance degradation by using scalability of the system [7-9].

Numerous studies have been done in multiple robot path planning for reducing the energy consumption as well as minimizing navigation time [10-12]. Specifically, a hierarchical path planning of mobile robots in a complex in-door environment is discussed [13,14] where the work focuses on map representation of complex buildings. A path planning for sensor based energy constrained multiple robot navigation was proposed [15]. The path planning using randomized search method was also proposed a path planning for multiple robots to reduce the overall path length [16,17]. A path planning for collaborative multiple robot system is also proposed [18] where the communication protocol was used to avoid team collision. Most of these works assume that the robots are autonomously able to navigate in buildings [19]. While many works has addressed the multiple robot path planning for indoor applications, dynamic reconfiguration of the paths due to presence of uncertain obstacles were not properly addressed. Moreover, the path planning complexity in terms of computation is extremely large for practical size maps.

When robots navigate in a confined area with human traffic, the path planning becomes undoubtedly complicated. The paths generated may become unusable due to changes in the environment. In order to optimize the service complete time and utilization of robots, the dynamic path planning that can properly adapt with the environment changes being occurred in real-time is necessary. Most of the previous

work has attempted to optimize path planning schemes without considering the computational loads in real time applications. As the number of corridors and walls in the map increases, the computational complexity required to generate the paths should not increase proportionally [20].

In this paper, we propose a path planning and dispatch strategy of multiple mobile robots for cooperative applications in confined spaces. The proposed two-level path planning method incorporates the segment reservation mechanism to maximize the utilization of the services. The two-level path planning reduces the computational complexity while providing the flexibility for any adaptation. Any potential deadlock is completely eliminated and the starvation due to unpredictable service request patterns is avoided. Since the proposed methodology is used for confined areas, any possible path blockage due to idling robots is eliminated with the idling robot relocation mechanism. The dynamic path planning is employed by monitoring and controlling the speed of the robot navigation. The dynamic re-planning is triggered with periodic monitoring of the service constraints. The method can be effectively used for low power multiple robot navigation applications.

The remainder of this paper has 5 sections. In Section II, we present the overview of the application model and the proposed approach including the problem descriptions. Section III presents the reservation based path planning methodology as well as robot relocation strategy for maximizing the robot utilizations. Section IV extends the discussion to dynamic path planning under unpredictable environment changes. In Section V, we evaluate the proposed path planning methodology

***Corresponding author:** Doug Kim, Computer Engineering Technology, Farmingdale State College (SUNY), Farmingdale, NY, USA, Tel: 6314202589; E-mail: kimdc@farmingdale.edu

Received July 26, 2017; **Accepted** August 21, 2017; **Published** August 28, 2017

Citation: Kim D (2017) Multiple Mobile Robot Dispatch Strategy for Cooperative Applications in Confined Space. Adv Robot Autom 6: 173. doi: [10.4172/2168-9695.1000173](https://doi.org/10.4172/2168-9695.1000173)

Copyright: © 2017 Kim D. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

with large-scale scenarios. Finally, our contribution is summarized in Section VI along with future work.

Application Model and Problem Description

Application model and motivation

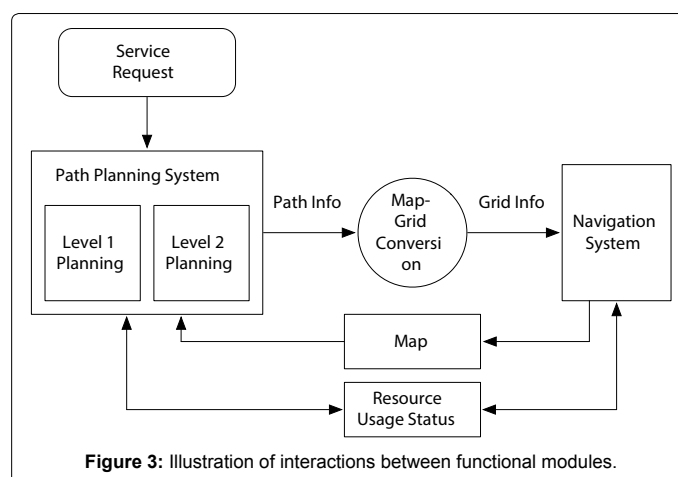
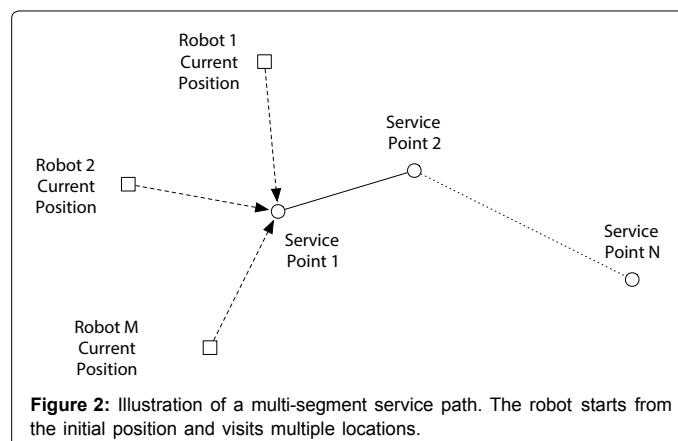
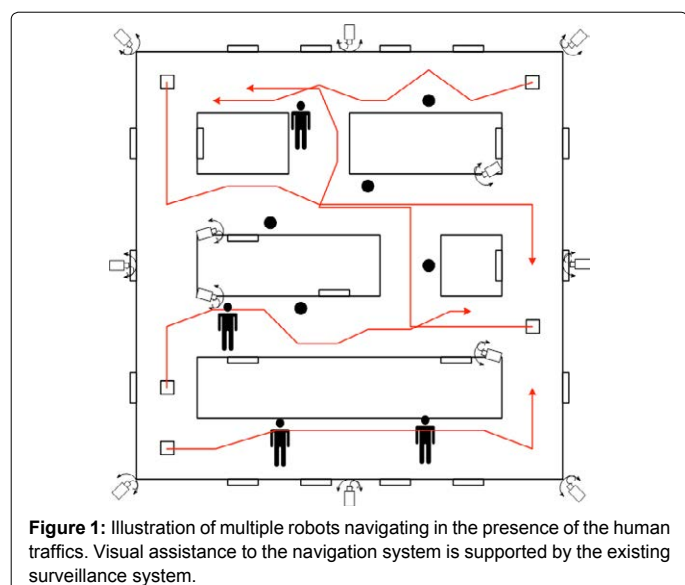
Figure 1 illustrates a large, confined and narrow environment where multiple robots navigate through human traffic. Both static and dynamic obstacles may be placed. The robots are controlled by the navigation system while various services are handled concurrently. Typically, the environment has some cameras for surveillance operations. The navigation system may interact with the surveillance systems for better management of the robot navigation.

In the presence of the obstacles, the services must be handled with time and energy constraints. The services may be preemptive where the robot assigned to a specific service can be reassigned to the other service with a higher priority. Non-preemptive service support is also possible. The service can be either single or multiple segments where the multiple visiting locations are possible as illustrated in Figure 2. The path planning must be able to adapt to dynamically changing environments.

Problem description and approach

One of the key objectives for the path planning system is to support various service request patterns and to deploy multiple robots for maximizing the service fulfillment and the resource utilization. The path planning system is initiated whenever a service is requested. Then, a path from the initial position to the destination for the robot is created. In order to plan a path without incurring deadlock between multiple robots during the service navigation, the path planning system considers all paths simultaneously such that the robots and the navigation spaces are time-shared. To support the dynamic path planning according to the service patterns or the obstacles, the path generation must be computationally efficient so that real-time operation is possible.

Figure 3 illustrates the functional overview of the proposed system. To effectively search the feasible paths for all service requests, the graphical representation of the map is used such that the segments represent hallways while the nodes represent intersections or corners.



All possible segment to segment connections are precomputed to reduce the computational overheads for real-time dynamic path planning. The degree of concurrent navigation path is represented as capacity. When the capacity is one, only one robot can navigate through the corridor. Upon the service requests, the path sequences of each service are created and maintained within the segment reservation mechanism. The reservation mechanism can provide all possible path solutions under timing constraints. By using the reservation mechanism, the service utilization can be maximized since the current and future availability of the resources are known to the path planning system to assist in selection of the path. The two-level path planning approach is adopted where the first-level path planning searches for the initial solution with no obstacles considered. In the first-level, all paths are precomputed to minimize the computational overhead during the real-time operations. The second level path planning fine-tunes the paths when dynamic obstacles are present.

When there are idling robots in the environment, such robots may reduce the usage of paths since each path is limited by the finite capacity. The capacity is defined as a number of simultaneous navigation within the segment without any collision. In order to eliminate this issue, an idling robot relocation mechanism is incorporated. The path planning system treats an idling robot as another service type, where the service is internally generated with the current location as the service origin and the new idling location as the service destination.

Segment Path Planning

Graph representation and a path search

Multiple path search problem with an initial location of each robots, a path to the intermediate destinations and the final destination are illustrated in Figure 4. While the initial navigation segments of the robots are different, there are common segments to navigate through by all robots to reach the final service points. Once the paths are selected, a robot begins the service from the start point and completes it in the destination point.

In order to find a path, the map with complex corridors is represented as a graph where the nodes represent the intersections and the edges represent the corridors. Each segment has a capacity value indicating the number of robots that can navigate simultaneously without any possibility of collision. The map with its corresponding graph is illustrated in Figure 5.

In addition to the nodes created at the intersection, the nodes are created whenever the capacity of the segment changes due to placements of permanent obstacles. Each segment is associated with a physical dimension (i.e., length) so that the travel distance of the segment as well as travel time of the robots with a given speed can be determined.

Given a graph, dynamically searching for the best path for undirected graph is computationally costly. In order to reduce the search time, all possible paths from the segment-to-segment are precomputed. The

reason for considering the segment to segment paths is that all robots start from different locations in the segment and navigate to different locations in the other segment. Having the same segment as both start and final segment may be possible if the destination of the robot is in the same segment. Hence, whenever, the path search is necessary, the operation becomes a simple look up process.

To precompute the paths, the original graph is converted where segments and nodes of the original graph are converted where segments and nodes of the original graph are converted to nodes and segments in the new graph as illustrated in Figure 6. Note that while the edges in the original graph are transformed into the nodes in the converted graph so that there is one-to-one correspondence, the number of edges in the converted graph does not have one-to-one correspondence. This is due to the multiple connected segments in the original graph. As shown in the figure, there are three edges (i.e., for n5, n6, n7, n9). While there are multiple edges in the converted graph, they are actually the same nodes in the original graph. For example, if the path from e8 to e4 is considered, there are two possible paths e8-n5'-e4, e8-n5"-e5-n5-e4. If the path contains identical nodes, shortest hop is considered. The precomputed paths between the segments are maintained as illustrated in Table 1. Once the paths are precomputed, the effects of the multiple edges must be considered.

The types of services include preemptive or non-preemptive. In the preemptive case, the robot that is already assigned to the service can be interrupted and then allocated for another service. Once the service is interrupted, the interrupted service is placed back into the service queue. The service may or may not have a deadline so that service preemption depends on the remaining time to the deadline. For the non-preemptive service, the service may not be interrupted regardless of the deadline constraint.

Reservation based path planning strategy

Since multiple possible paths are available as illustrated in Figure 4, finding the best one is not trivial especially if some of the segments are already being used by other services. In order to improve the resource

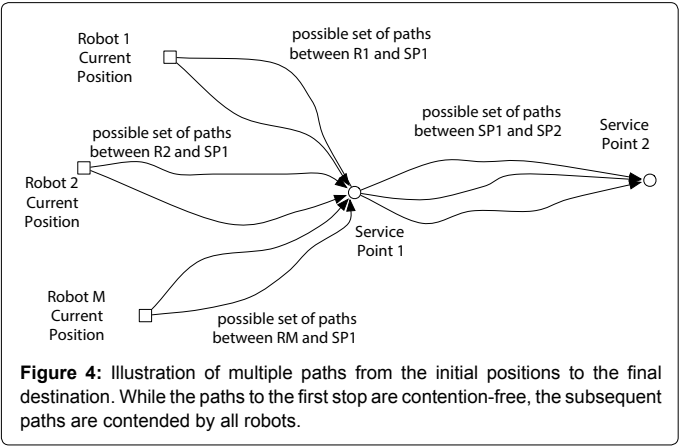


Figure 4: Illustration of multiple paths from the initial positions to the final destination. While the paths to the first stop are contention-free, the subsequent paths are contended by all robots.

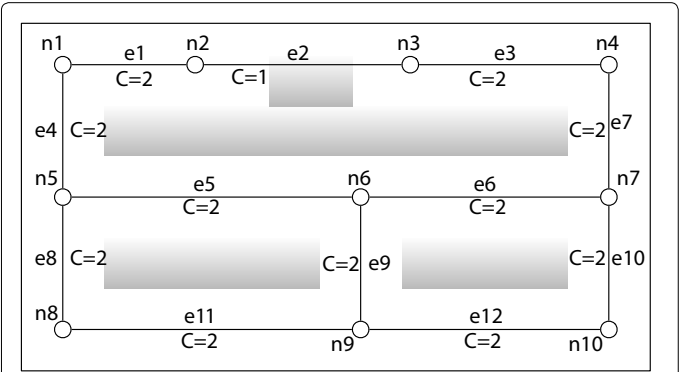


Figure 5: Graphical representation of the map. Each corridor is represented by an edge (segment) and each intersection is represented by a node. Each segment is marked with a capacity. Nodes are created if the capacity changes within the segment.

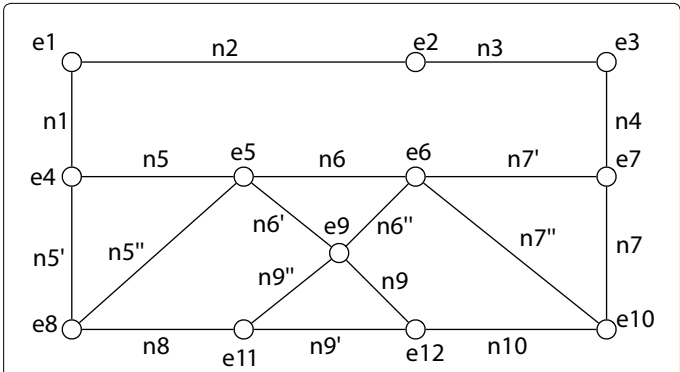


Figure 6: The original graph with 10 nodes and 12 edges are converted to a graph with 12 nodes and 18 edges.

	Destination Segment	Visiting Segment List	Visiting Node List
1	2	1,2	3
1	3	1,2,3	3,4
1	4	1,2,4	3,5
-	-	-	-

Table 1: Illustration of data structure containing all segment to segment paths for a given map. The data structure contains the segment list as well as the node list.

utilization, the reservation based path planning method is proposed. A basic idea is to reserve a segment to be used by annotating the usage time. Two data structures are used for the method. Table 2 illustrates the usage of the segment by the robots. For each segment, the index of the robot, beginning and finishing time of the usage, and direction are maintained. The other data structure which contains the information of the segment capacity is illustrated in Table 3.

When a service request is received, depending on the pre-emptive or non-preemptive status, we generate a set of possible paths. When possible paths are generated, we consider all robots that are idle or associated with preemptive services. The robots handling the non-preemptive services are not considered for incoming service requests. For each path from the initial location of the robot to the first destination, the segments are compared with the existing entries for the reservation status. For each segment in the path solution, the usage time must be reserved and annotated within the data structure illustrated in Table 2. The entry for the data structure is to specify which robot to use, when the service uses the segment and direction of the robot navigation. While annotating the reservation information, the segment capacity data structure is used as illustrated in Table 3. If the capacity is available, the segment can be used without any problem. However, if the limited capacity is available, the segment usage time needs to be shifted in time to allocate the segment resources so that the capacity becomes available when the segment is used unless it is in the same navigational direction. For simplicity for the discussion, first come first use approach is adopted.

A path of a robot is created up to service's destination point such that the segment including the starting point is reserved for the robot after the time that service is completed. When the segment includes a service point of another service, the replanning strategy reuses the robot. It is because another robot is not able to reserve the segment for bypassing or service during the time.

Figure 7 illustrates potential problem with the single capacity segment when the first come first use policy is used. This problem arises when the service does not have a deadline. If the first use first policy is used, and R1 and R2 are previously scheduled to use the segment as shown in the figure, the R3 is inserted between the R1 and R2 usage since R3 will use the segment before R2 even though the service associated with R2 is assigned before. Potential problem with this approach when the deadline is not provided, some of the service will be shifted as new service enters creating starvation situations. This

Segment 1	Segment 2	-	Segment N
R1,Te,Tl,Direction	R1,Te,Tl,Direction	-	R1,Te,Tl,Direction
R1,Te,Tl,Direction	R1,Te,Tl,Direction	-	R1,Te,Tl,Direction
R1,Te,Tl,Direction	R1,Te,Tl,Direction	-	R1,Te,Tl,Direction
-	-	-	-

Table 2: Illustration of the data structure maintaining the segment usage. This data structure indicates the robots on the segments. Each robot in the segment is associated with three timing factors.

	Segment 1	Segment 2	-	Segment N
Capacity	2	3		1
Usage	Direction Time	Direction Time		Direction Time
Usage	Direction Time	Direction Time		Direction Time
-	-	-	-	-

Table 3: Illustration of the data structure specifying the capacity of each segment. Multiple capacity values imply that multiple robots can navigate through the corridor simultaneously.

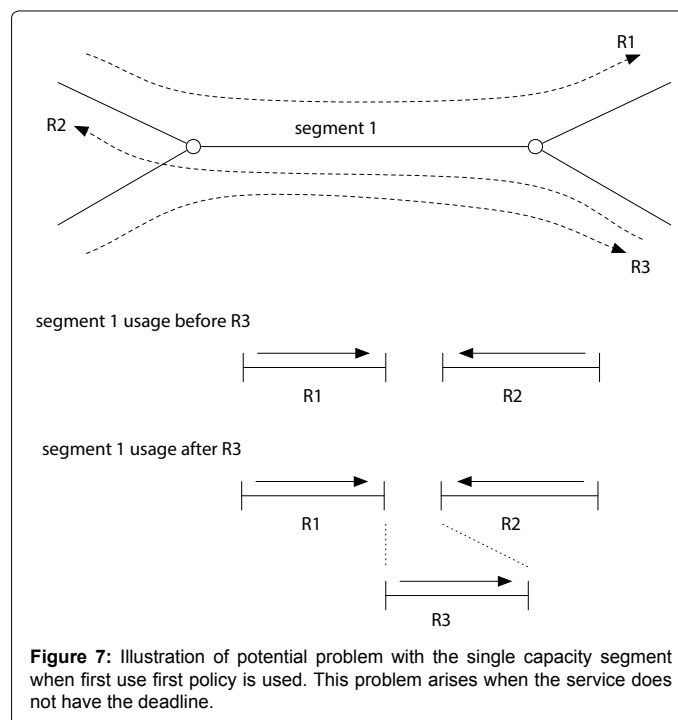


Figure 7: Illustration of potential problem with the single capacity segment when first use first policy is used. This problem arises when the service does not have the deadline.

problem can be resolved if a deadline is specified or internal priority is used since smallest remaining time to deadline will be allocated.

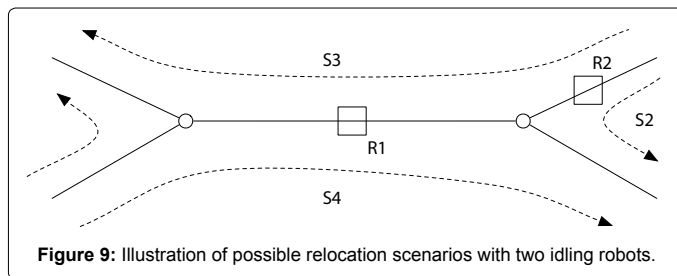
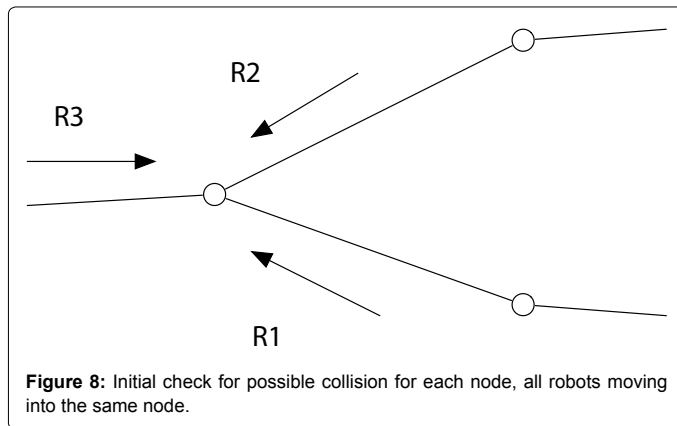
Once the possible paths are computed based on the first come first use policy, the path solution must be checked for any deadlock. The deadlocks may create collisions. In order to check for any deadlock, each node is sequentially observed. If there are more than two incoming robots at any node and they are scheduled to use the segment currently used by the other robots, the deadlock will occur. So when selecting the path solution, this process must be performed before finalizing the path plan. This is illustrated in Figure 8.

Robot relocation to increase utilization

Due to the limited capacity of each segment, the idling robots may affect the path searching by reducing the available capacity. Thus, during the path search, the idling robots need to be relocated to release the segment resources. Figure 9 illustrates why relocation is necessary. Assume R1 and R2 are idling. Four service paths are defined. While S1 can be navigated, S2, S3, and S4 cannot use the segment because of idling robots. S2 can be initiated after relocating the R2 to the segment where the R1 is idling. Hence, the idling robots need to be relocated to the segment that will not be used soon by the other services. When the robots are to be relocated, the idling robot in the single capacity segment should be considered first. The robot relocation method is summarized in Algorithm 1.

Planning for minimum service distance vs. minimum service time

There are two possible path planning constraints: planning for the minimum service distance and planning for the minimum service time. While it is possible to obtain a solution satisfying the both constraints, usually it is not easy to obtain such path planning solution. It depends on the traffic loads within the area, the number of navigating and idling robots, and the service request patterns. These are not known accurately before the actual planning and the start of navigation.



However, it is possible to impose these two such constraints during the path planning stage. The minimum service distance solution is usually preferable for minimizing the power dissipation of services while the minimum service time is usually preferable for maximizing the resource utilization. Overall reservation based path planning method is summarized in Algorithm 2.

Algorithm 1: Idle Robots Relocation Strategy

- 1:
- 2: Relocation of the idling robots process;
- 3:
- 4: for (the segment which the idle robot is located is schedule to be used) do
- 5: if (idle robot is in single capacity segment) then
- 6: Determine how soon to move away, T_{min} ;
- 7: for (each segment connected to the current segment) do
- 8: Determine the time for the robot to move away, $T_{move,l}$, $T_{move,r}$;
- 9: Determine the usage time of the adjacent segment, $T_{use,i}$;
- 10: Maintain the segment with the maximum $T_{use,i}$;
- 11: end for
- 12: Select the segment with the maximum $T_{use,i}$;
- 13: Generate an internal service for this robot;
- 14: end if

- 15: if (idle robot is in the multiple capacity segment) then
- 16: if (both capacity is to be used) then
- 17: repeat above steps if the robot is on the segment to be used sooner;
- 18: else
- 19: move the robot to the unused segment
- 20: end if
- 21: end if
- 22: end for
- 23:

Dynamic Path Planning for Navigation with Uncertainty

External factors influencing path replanning

There are many factors influencing the paths to be re-planned. Depending on the available resources and the service constraints, the existing paths for the services are revised to accommodate a new service. It is possible that without the replanning, the new services may not be able to utilize the resources with the maximum efficiency. Thus, the key reason for replanning is due to the services patterns and resource availability.

The transit time of the existing services also influences replanning even without the new services when there are obstacles in the area. The robots themselves whether navigating or idling also affects the transit times especially if the dynamic obstacles are presented in the paths. During the robot navigation, the navigation delay by one of the robots influences the other robots to slow down and as a result, the service time constraints may not be satisfied. Hence, the alternate paths may need to be computed under such conditions. From the path planning perspective, any change in the transit time modifies the effective distance of each edge as obstacle placement creates additional nodes. Such changes are highly dynamic that replanning must be performed in real-time to adapt to the changes.

Algorithm 2 Reservation Based Path Planning

- 1:
- 2: Parameters: (Preemptive, Non-preemptive);
- 3: Parameters: (Shortest distance, Shortest time);
- 4: Process for multiple segments;
- 5:
- 6: for (each segment) do
- 7: if (first segment) then
- 8: Process for the first service segment;
- 9: if (Preemptive) then
- 10: Process for preemptive, consider all robots;
- 11: Select either shortest distance or shortest time.
- Save this solution;
- 12: This solution (robot) will be considered for the subsequent service segment;

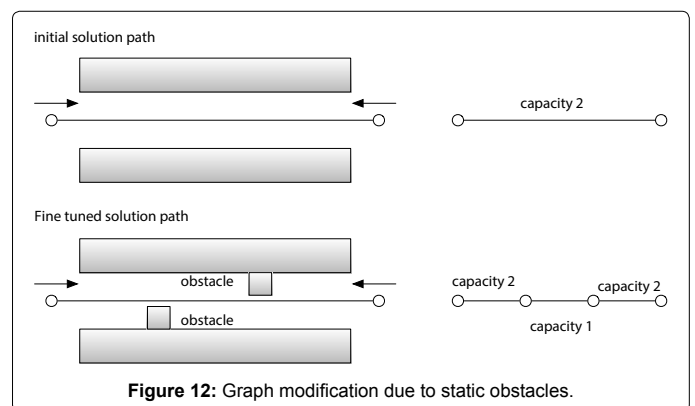
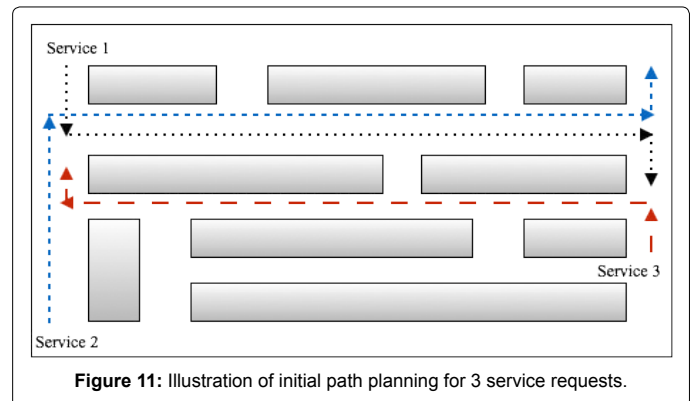
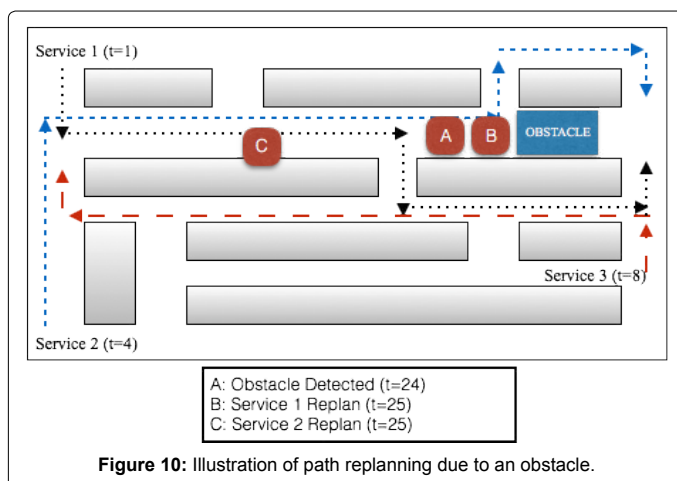
- 13: else
- 14: Process for non-preemptive, consider all robots in idle and all other robots from the final destination;
- 15: Select either shortest distance or shortest time.
- Save this solution;
- 16: This solution (robot) will be considered for the subsequent service segment;
- 17: end if
- 18: else
- 19: Process for the subsequent service segment;
- 20: Consider the robot selection from the first service segment process;
- 21: Select either shortest distance or shortest time. Save this solution;
- 22: This solution (robot) will be considered for the subsequent service segment;
- 23: end if
- 24: end for
- 25:
- 26: Select the solution for dispatch and update resource data structure;
- 27:

Figure 10 illustrates a scenario where two of the initial path plans in Figure 11 are modified due to an obstacle appearing in the path. The path plan for the third service request, Service Request 3 is not affected because all of the segments in the path plan will be available by the time the robot for the service request 3 enters the each segment.

Second level path planning

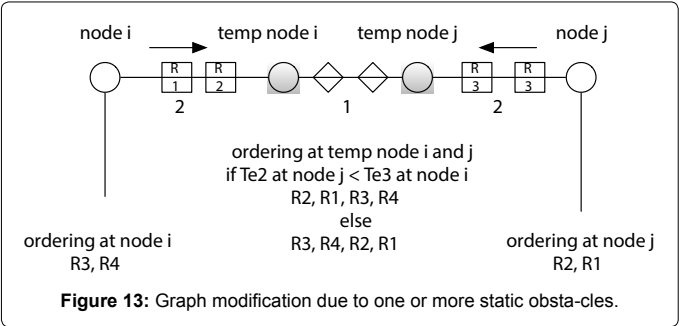
The path planning methodology discussed in the previous section assumed that the original graph would not change.

All possible paths were precomputed under this assumption. While all possible paths are precomputed at the first level path planning to reduce the computational overhead during the real time operation, the



precomputed paths do not consider unexpected obstacles. Often, the path changes due to such unexpected obstacles are locally confined. In order to handle this dynamic modification of the graph in real-time, the second level path planning is incorporated. Regardless of any obstacle, the initial path planning generates a possible solution. Once the solution is generated, the actual path and the timing are adjusted as required per dynamic events during the navigation. Figure 12 illustrates a situation where the static obstacles are discovered. In the original graph, the corridor has the capacity of 2. However, the discovery of static obstacles modifies the values of the segment capacity. To cope with the change, the new graph needs to be used. The second level path planning is not initiated for the dynamic obstacles such as human traffics. In case of the dynamic obstacles, the robot navigation is temporarily stopped until the obstacles are cleared. The second-level path planning is only initiated for appearance or disappearance of the static or very slowly moving obstacles. Whenever a new static obstacle is presented in the area, the paths for the existing services that are using the segment with the obstacles are revised.

Figure 13 illustrates the modification of the original graph due to the presence of one or more static obstacles. Two temporary nodes are inserted between the original two nodes, node i and node j. The temporary nodes enclose the static obstacles. The capacity between the two temporary nodes is reduced to 1 such that bidirectional navigation is not possible. If $T_{e,2}$ at node j is less than $T_{e,3}$ at node i, then the ordering of temp node i will have R2, R1, R3, R4. Here regardless of the value of $T_{e,1}$, the robot 1 will follow the robot 2. The ordering at temporary node j will have R2, R1, R3, R4 as well. In this case, both robot 1 and robot 2 will pass through the segment between the two temporary nodes first. The ordering process is reversed if $T_{e,2}$ at node j is larger than $T_{e,3}$ at node i. When the first sets of robots are determined, the maximum speed is used to pass through the segment



Node 1	Node 2	-	Node N
R1:(Te, TI, Ta)	R6:(Te, TI, Ta)	-	R7:(Te, TI, Ta)
R2:(Te, TI, Ta)		-	R9:(Te, TI, Ta)
R4:(Te, TI, Ta)		-	
-	-	-	-

Te: Earliest Arrival Time
TI: Latest Arrival Time
Ta: Expected Arrival Time

Table 4: Illustration of the data structure maintaining the ordering of robot arrivals. For each robot, there are three timing factors.

between two temporary nodes. If the capacity changes after the obstacles are more than 1, only the sequence of the grids is modified and the ordering will be strictly based on the original Te. The new set of values at the temporary nodes is computed based on the current positions of the robots. If there are more than one static obstacle on the segment between the two temporary nodes, the path sequence in the segment between the temporary nodes are correctly generated.

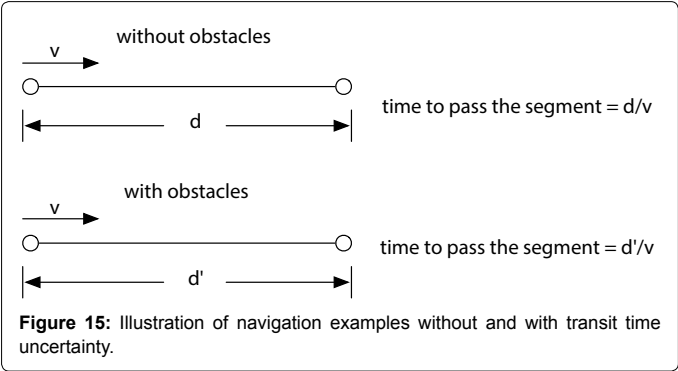
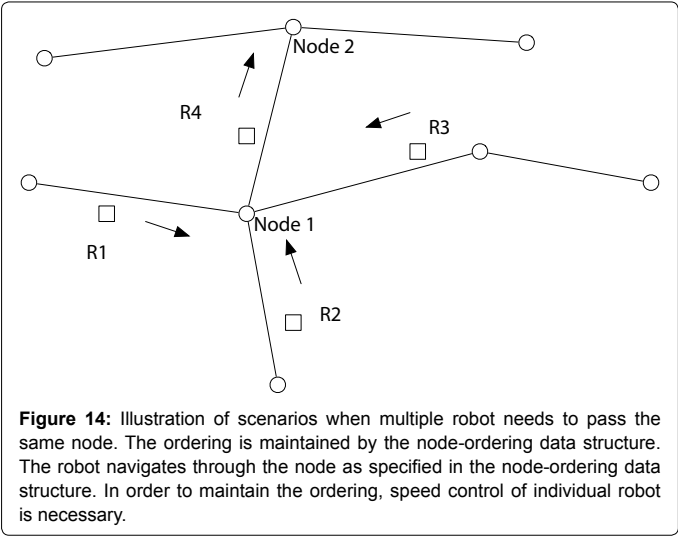
If the capacity of the segment is 1 before the static obstacles, the segment is temporarily removed, and all existing services that use the segment must be revised. If the robots are already in the segment when the static obstacles are presented, the robots move out of the segment while the service path is being revised.

Navigation transit time and service time variations

Once the service path is decided by the path planning algorithm, the assigned robot starts navigation by following the given path sequence. During the robot navigation, the flow of the robots is constantly monitored. Since there may be obstacles that interfere the flow of the robots, the actual transit time may vary. The speed of the robots must be carefully controlled to avoid any collision among the robots. The speed of the robots is controlled with the node-ordering data structure as illustrated in Table 4. The node-ordering is simply an ordered list in time of all incoming robots to a specific node as illustrated in Figure 14. By maintaining the order, any potential deadlock can be avoided. Moreover, based on the ordering, the speeds of all robots are adjusted.

As long as the node ordering is maintained, the deadlock free navigation is possible. Hence the path planning must guarantee correct node ordering. The estimated arrival time needs to be estimated periodically using the speed maintained by the navigation system. The estimated arrival (or transit) time is estimated as illustrated in Figure 15. If any of the deadlines is likely to be violated, the new path needs to be determined.

Based on these current navigation speeds of the robots, the transit times of all segments are estimated. The actual transit times are most likely different from the ideal transit time of the segment calculated using the segment length and the speed of the robot. Hence, when the



Service Index	Robot Assigned	Completion	Deadline	Preemptive
1	2	1.34	No	Yes
3	1	2:00	2:13	No
2	4	2:11	2:15	No
-	-	-	-	-

Table 5: Illustration of the data structure maintaining the outstanding services. If the service is preemptive, the robot can be reassigned for other service. If non-preemptive, the service and the robot pair is fixed until the end of the service.

path is planned for a new service, the estimated transit time for the segment is used to generate a solution. As shown in Table 5, each service and the associated robot status is maintained in real-time. The table maintains the service and the robot assignment as well as the required deadline and estimated completion time. When the completion time is estimated, the service time is also profiled and estimated since the total time for the service included both the transit time as well as the service time at each destination.

Whenever the path planning handles the service, the path planning decides the strategy for generating the path planning. There are two types of path planning. The first type is based on the minimum waiting time and the second type is the minimum distance. The minimum waiting time strategy is usually used for services with tight deadlines. These types of services are dispatched as soon as a robot is available. If there are enough time remaining for satisfying the deadline, the minimum distance strategy is used. The robots navigate within the timing constraints according to the node-ordering information specified by the path generation.

However, when the navigation control system estimates that the expected time navigation time will not satisfy the time constraint, it is necessary to replan the paths of the services. Algorithm 3 summarizes the service handling algorithm.

Dynamic path replanning strategy

The idling robots as well as the placement of any obstacles are considered in the proposed path planning system for redistribution to minimize congestion and to avoid any potential disconnection of the paths. These two essentially modify the capacity of the segment dynamically. When the segment capacity changes after the path planning, the potential deadlock may occur even though the paths are generated initially free of deadlocks. In order to avoid the deadlock, the existing path solution is constantly monitored for any potential violation of timing constraints. Algorithm 4 summarizes overall dynamic path planning algorithm with replanning triggers.

Algorithm 3: Algorithm for Handling the Service Patterns

```

1:
2: Wait for (service event);
3: if (service event) then
4:   put the event into event queue;
5: end if
6:
7: First come first serve but preemptive service first;
8: while (event queue is not empty) do
9:   Select the fist non-preemptive service;
10:  Obtain the solution using the reservation based path planning;
11: end while
12:

```

Algorithm 4: Overall Dynamic Path Planning Algorithm

```

1:
2: Overall dynamic path planning process;
3: Wait for (Service events) or (Navigation events) or (Timer events);
4:
5: Process if new service is requested or existing service is terminated;
6: if (Service event) then
7:   if (new service) then
8:     Execute for handling the service patthterns;
9:   Update service status data structures;
10:  else if (completed service) then
11:    Update the robot status;
12:  end if
13:  Execute for handling the service patterns;
14: end if

```

```

15:
16: Periodically check the internal data structures for reschduling the paths;
17: if (Timer event) then
18:   Check for potential deadlock;
19:   Check the data structures for reschedule trigger;
20: end if
21:
22: Process if the Navigation server requests;
23: if (Navigation event) then
24:   Update the status data structures;
25: end if
26:

```

Evaluation

Simulation setup

A large-scale map with complex confined corridors is considered in the simulation. The capacity of each corridor is used as a parameter. The graph used in the simulation is illustrated in Figure 16. Before the evaluation, all possible paths are precomputed and stored in the system. Various services types and patterns are generated. The services are multi-leg with the specified service times in each destination. Both preemptive and non-preemptive services are considered. In addition, the low power paths are generated whenever possible.

Effects of service request pattern

We evaluate the performance of path planning strategy for different service request patterns. Three main types of the request patterns are used in the simulation. Figure 17 illustrates the service request patterns considered. The first type of service patterns has series of bursty requests separated by a specified amount of time. If the number of bursty requests are smaller than the available number of robots and the separation is large enough, the resource utilization will be an issue. Hence we consider two cases as illustrated in the figure. The second type is sparse service request and the simplest configuration. The third type is the bursty requests that are more than the number of robots available. Each service consists of intermediate service points and the final service point. Both preemptive and non-preemptive services are considered.

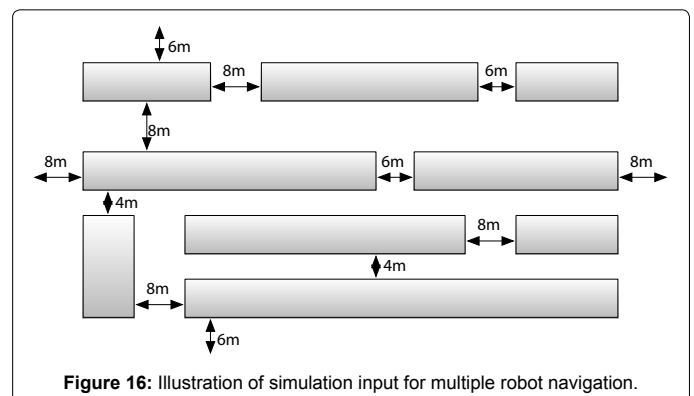


Figure 16: Illustration of simulation input for multiple robot navigation.

The performance is measured with the service complete time. The paths are generated for the minimum service time as well as the minimum distance traveled. The path planning for the shortest distance may be suitable for low power consumption while for the shortest time may be suitable for maximizing robot utilization. The comparison of the service time as a function the robots are illustrated in Figure 18. Three service patterns are considered with the first 10 services plotted for illustration. First two plots considered the non-preemptive service and the next two plots considered preemptive. For each set of plots, 4 robots and 10 robots are used. As shown in Figure 18, as the number of robots increases, the service time increases since the service may have to wait for the resources (segments) to be available.

Similar comparison is evaluated and illustrated for the service distance in Figure 19 as the number of robots increases from 4 to 10, the distances travelled for the first 4 service requests does not change much. There is an increase in distance travelled for service request 5 and

6 for the service patterns 5 and 6. This is due to additional congestion caused by the 5th and 6th robots that were not available previously.

Effects of uncertain navigation time

The performance of the path planning algorithm is evaluated under dynamic navigation uncertainty as a function of the planning not including the original path planning without any obstacles. Preemptive service type is used in the evaluation. Only the transit time variation due to the node ordering and interference among robots are considered. Both preemptive and non-preemptive services are evaluated.

Figure 20 illustrates the number of replanning of the paths for the robots with dynamic obstacles. As more random obstacles are presented during the navigation, the number of replanning increases. This is compared with cases with no obstacles. For all 4 cases illustrated in Figure 20, as more robots put into service, more frequent replanning is observed.

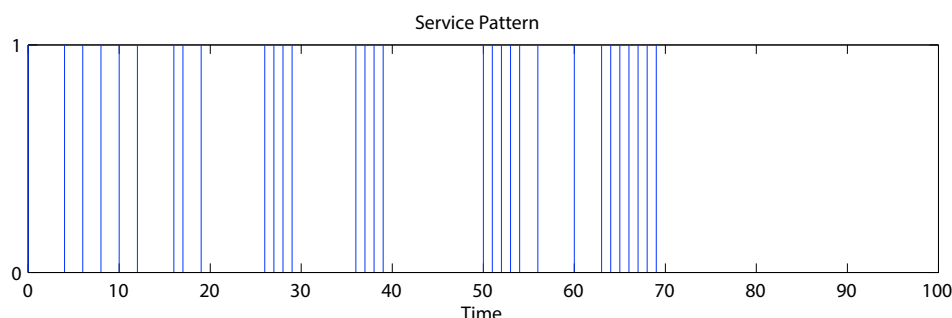


Figure 17: Illustration of the service patterns used in the simulation.

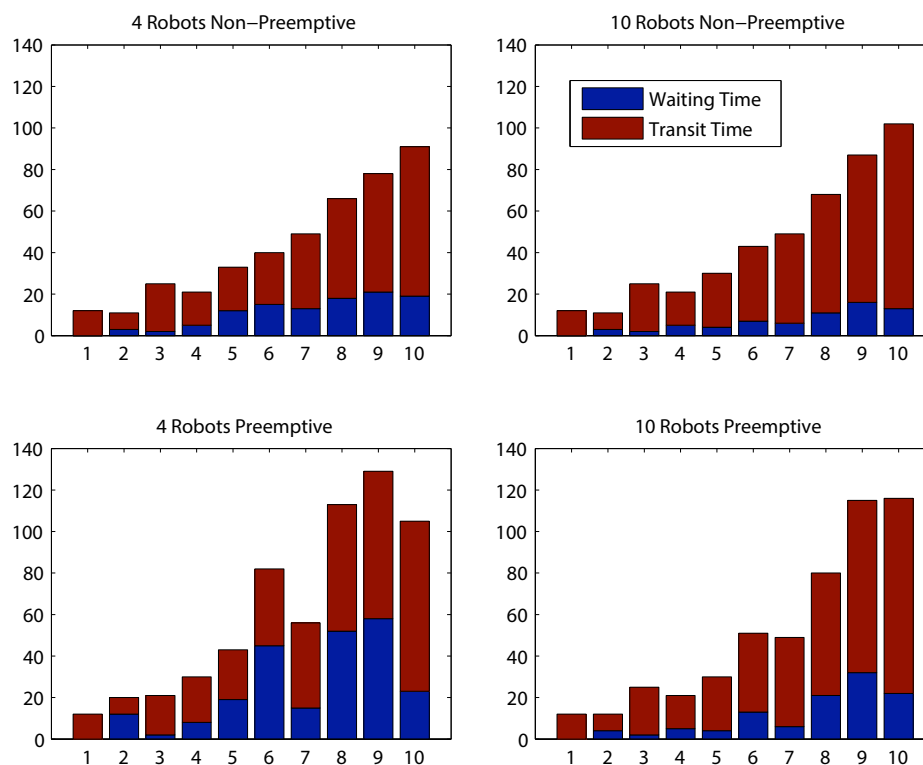


Figure 18: Illustration of the service times for the first 10 service requests.

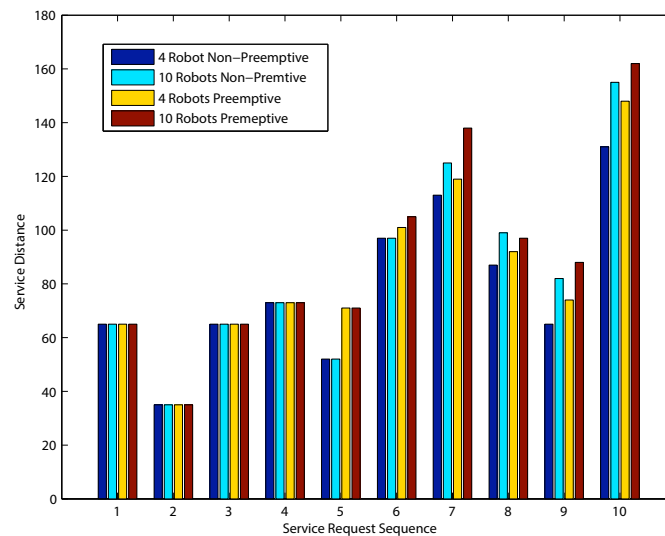


Figure 19: Illustration of the service distances for the first 10 service requests.

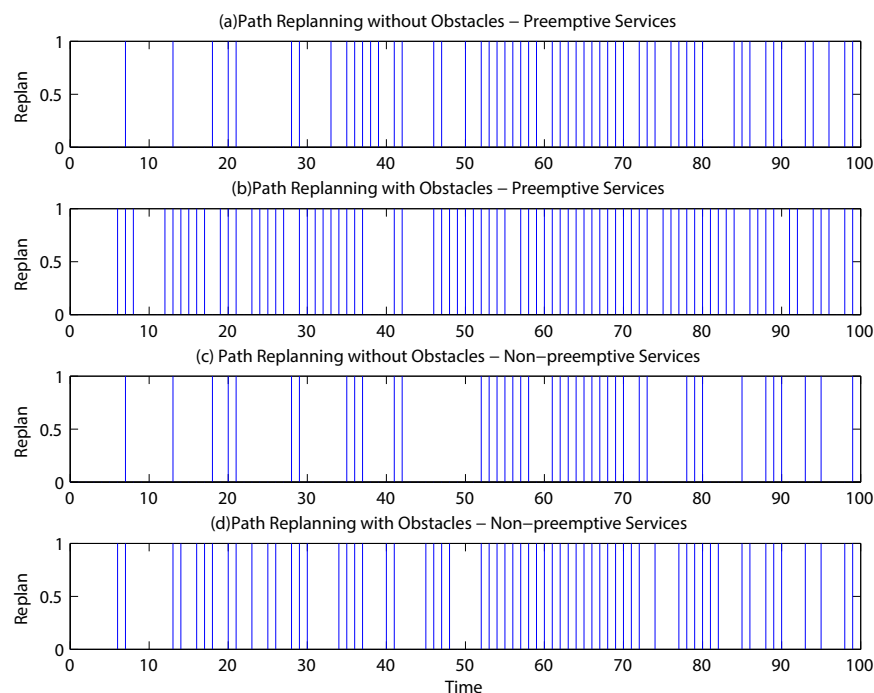


Figure 20: Illustration of the dynamic replanning frequency with and without obstacles.

Resource utilization

Figure 21 illustrates the service time as the number of robots. The different curves correspond to the different service types and modes. The system tries to utilize as many robots as possible. While service time goes down as more robots are utilized in the system, the service time goes up again after a critical number indicating that the segments are becoming the limiting factor instead of the number of robots. Having a higher number of robots than this critical number does not yield better average service time. The actual value of the critical number depends on the capacity of segments as well as the size of the service area.

Conclusion

In this paper, we propose a multiple mobile robots path planning and dispatch strategy for cooperative applications in confined space. The proposed two-level path planning method incorporates the segment reservation mechanism to maximize the utilization of the services. The two-level path planning reduces the computational complexity while providing the flexibility for any adaptation. Any potential deadlock is completely eliminated and the starvation due to unpredictable service request patterns is avoided. Since the proposed methodology is used for confined area, any possible path blockage due to idling robots is eliminated with idling robot relocation mechanism.

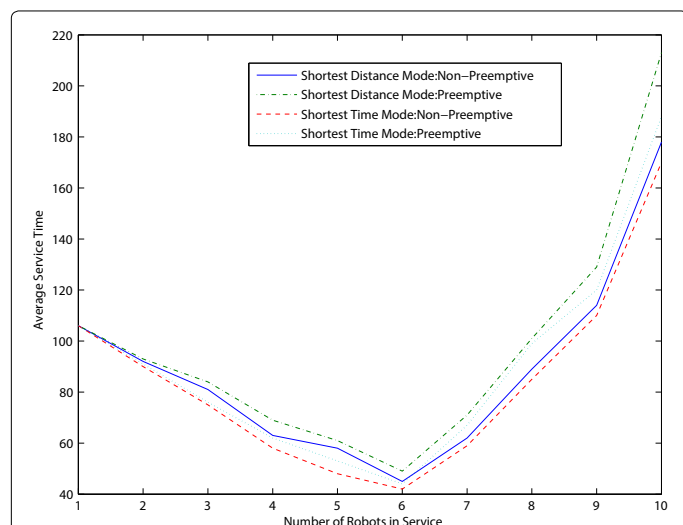


Figure 21: Illustration of the service time as a function of number of robots. There is a critical number of robots that minimizes the average service time. Beyond this critical number of robots, the service time increases due to limited segments availability.

The dynamic path planning is possible by monitoring and controlling the speed of the robot navigation. The dynamic re-planning is triggered with periodic monitoring of the service constraints. The method can be effectively used for low power multiple robot navigation applications and the performance of the proposed system are evaluated with a large-scale simulation.

References

- Ahn HS, Lee MH, MacDonald BA (2015) Healthcare robot systems for a hospital environment: CareBot and ReceptionBot. *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication* 571-576.
- Chen GT, Huang LK, Tsai CW, Chiang MC (2015) An intelligent robot for home healthcare. *Proceedings of International Conference on Consumer Electronics* 94-95.
- SolisJ, Teshome TS, Rosa JP (2015) Towards developing a multipurpose assistive vehicle robot capable of providing assistance to caregivers and support to elderly people. *Proceedings of IEEE International Conference on Automation Science and Engineering* 1145-1150.
- Breazeal C (2011) Social robots for health applications. *Annual International Conference of Engineering in Medicine and Biology* 5368-5371.
- Chinnaiah MC, Savitri TS, Kumar PR (2015) A novel approach in navigation of FPGA robots in robust indoor environment. *Proceedings of the IEEE International Conference on Advanced Robotics & Intelligent Systems* 1-6.
- Ko HJ, Kim JH (2013) Map building and path generation for Indoor mobile robot by using global and local views. *Proceedings of the IEEE International Conference on Control, Automation & Systems* 1243-1246.
- Portugal D, Rocha RP (2011) On the performance and scalability of multi-robot patrolling algorithms. *Proceedings of the International Symposium on Safety, Security & Rescue Robotics* 50-55.
- Julian BJ, Angermann M, Schwager M, Rus D (2011) A scalable information theoretic approach to distributed robot coordination. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots & Systems*, 5187-5194.
- Chwa HS, Shyshkalov A, Lee K, Shin I (2014) Scalable Path and Time Coordination for Robot Formation. *Proceedings of the IEEE International Conference on Cyber-Physical Systems, Networks & Applications* 65-70.
- Hansen C, Itjen J, Meike D, Ortmaier T (2012) Enhanced approach for energy-efficient trajectory generation of industrial robots. *Proceedings of the IEEE International Conference on Automation Science & Engineering* 1-7.
- Dogru S, Marques L (2015) Energy Efficient Coverage Path Planning for Autonomous Mobile Robots on 3D Terrain. *Proceedings of the IEEE International Conference on Autonomous Robot Systems and Competitions* 118-123.
- Bjorkenstam S, Gleeson D, Bohlin R, Carlson JS, Lennartson B (2013) Energy efficient and collision free motion of industrial robots using optimal control. *Proceedings of the IEEE International Conference on Automation Science & Engineering* 510-515.
- Seder M, Mostarac P, Petrovic I (2015) Hierarchical path planning of mobile robots in complex indoor environments. *Transactions of the Institute of Measurement and Control* 33: 1-3.
- Park B, Choi J, Chung WK (2012) An efficient mobile robot path planning using hierarchical roadmap representation in indoor environment. *Proceedings of the IEEE International Conference on Robotics & Automation* 180-186.
- Sipahioglu A, Kirli G, Parlaktuna O, Yazici A (2010) Energy constrained multi-robot sensor-based coverage path planning using capacitated arc routing approach. *Robotics and Autonomous Systems* 58: 529-538.
- Bennewitz M, Burgard W, Thrun S (2001) Optimizing Schedules for Prioritized Path Planning of Multi-Robot Systems. *Proceedings of the IEEE International Conference on Robotics & Automation*, 271-276.
- Rao M, Dudek G, Whitesides S (2007) Randomized algorithms for minimum distance localization. *Transactions of the International Journal of Robotics* 26: 917-933.
- Palanisamy P (2005) Path Planning Protocol for Collaborative Multi-Robot Systems, Path Planning Protocol for Collaborative Multi-Robot Systems. *Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation* 721-726.
- Pol RS, Murugan M (2015) A review on indoor human aware autonomous mobile robot navigation through a dynamic environment survey of different path planning algorithm and methods. *International Conference on Industrial Instrumentation and Control* 1339-1343.
- Aaron E, Kranakis E, Krizanc D (2011) On the Complexity of the Multi-Robot, Multi-Depot Map Visitation Problem. *The IEEE International Conference on Mobile Adhoc and Sensor Systems* 795- 800.