Navigability and Reachability Index for Emergency Navigation Systems Using Wireless Sensor Networks

Lin Wang**, Wenyuan Liu

School of Information Science and Technology, Yanshan University, Qinhuangdao 066004, China

Abstract: Emergency navigation with a large number of sensors can serve as a safety service in emergencies. Recent studies have focused on navigation protocols to safely guide people to exits while helping them avoid hazardous areas. However, those approaches are not applicable in all circumstances. Both the dynamics of the environment and the mobility of users are key challenges for the efficiency and effectiveness of navigation protocols. The concepts of navigability and reachability are used to evaluate three typical navigation approaches. A large number of simulation results show that these two indicators effectively identify the performance levels of navigation protocols in changing environments.

Key words: reachability; navigability; emergency navigation; wireless sensor networks

Introduction

The navigation problem is often used for trajectory planning for planes and moving planning for robots[1-5]. Navigation is the process of monitoring and controlling the movement of a craft or vehicle from one place to another. In general, navigation is seen as a service which prevents the mobile user (e.g., robots, aircrafts or other people) from moving into undesirable areas (e.g., walls, obstacles or roadblocks) or dangerous areas (e.g., fire, or contaminated areas). Most existing movement planning algorithms based on combinatorial optimization[6], computational geometry[7], and graph theory[8] are centralized and assume the availability of global knowledge about the path topology. For example, widely used road navigation applications in Global Positioning Systems (GPS)[9] use centralized path planning algorithms with pre-loaded maps of the navigation environment.

A real-time variable safety net could be made of hundreds of small sensors with limited memory and multiple sensing capabilities that autonomously organize and reorganize themselves as ad-hoc networks in response to emergency tasks and to triggers from the environment. When an emergency happens, the navigation system deployed in advance is triggered to implement two guiding tasks. In the first, a trapped user needs to be guided out of the emergency area avoiding dangerous areas. In the second, rescuers need to be guided into the emergency region to help trapped people in danger.

In the last decade, a large number of works have proposed distributed navigation algorithms to address this problem using sensor networks[10-15]. These approaches can be divided into roadmap navigation methods based on directed graphs and random navigation systems based on stochastic process theory. Roadmap navigation guides a user to exits by a roadmap which is viewed as a subgraph of a communication graph formed of previously deployed nodes. However, data loss, asymmetric connections, network
congestion, and other unpredictable network conditions unaccepted overhead from communications and recomputing in energy limited sensor network systems. In random navigation systems, a node (nearby user or exit) uses its states to predict the states of its neighbors with users selecting a safe neighbor node as its movement direction and move towards the node until they arrive at an exit. This approach is more appropriate than a centralized system, since local oscillation often occurs in a dynamic environment.

Due to the distributed nature of wireless sensor networks and the dynamics of emergency environments, recent studies have proposed the distributed reactive guiding principle for responding to environmental dynamics. However, a topology-based system uses the roadmap as a subgraph of a communication graph abstracted from the sensor network, while the real physical environment may be unreachable when unreliable links exist. From the temporal and spacial point of view, the navigation system design should proactively react to spatial-temporal variations. This paper presents:

• A 3-D classification system for existing navigation protocols in terms of response method, path metrics, and updating techniques.
• Navigability and reachability metrics for navigating with wireless sensor networks and state-of-the-art simulations to evaluate the metrics.
• New principles for designing a common navigation system for the user’s escape or rescue.

1 Emergency Navigation Using Wireless Sensor Networks

An integral part of a navigation system is a path generation protocol for reliable guidance and robust adaptation to environmental dynamics. Sensor networks rely on a waypoint (also called as a fingerprint) state to self-organize into a roadmap structure. However, the network takes a substantial amount of time to decide a destination is unreachable and for the waypoint state to refresh timeouts to implicitly detect a non-navigatable waypoint in the roadmap. To improve the convergence in a real-time system, the sensors should trigger state updates to explicitly notify their neighbors.

Path generating approaches calculate the best path for each destination in a distance vector or a waypoint state basis. The distance vector approach computes the optimality incrementally along a path. Sensors calculate routes locally based on their current and past network states. They then iteratively send intermediate results to their neighbors until the routing tables stabilize. Once a navigation path emerges, it is used to guide a trapped user. In addition, every sensor contributes to establish a distributed state database using the waypoint state approach. Then, sensors run a safest path algorithm which depends on various navigation metrics using this topology database instance.

1.1 Assumptions and problems

Figure 1 shows a sensor network consisting of a large number of static sensors with limited resources and capabilities. The assumptions are:
• The area is well covered by the network. That is, a dangerous event at any location in the area is detected by at least one sensor.
• Two sensors are neighbors if they are within communication range of each other.
• Each sensor knows its (2-D) geographic or virtual coordinates in the network and sensor locations are exploited as navigation landmarks to assist user movements.
• A node immediately broadcasts a received message and each packet received by a node has traveled through the shortest path.
• The user is able to track any targeted sensor node by measuring the strength and direction of the wireless signals. Thus, the navigating route can be interpreted as a sequence of nodes.
• The optimal navigation path is an extremely hard task, so an approximately optimal safe path is acceptable.
• The primary computations are performed distributively in the sensor network, thus, the sensors are used
to avoid dangers or obstacles that are undesired by the users.

Some of these assumptions are not tenable in a practical scenario. The first is that network stabilization takes a long time because of network congestion and link status transition. The second is that the environmental dynamics and the unreliable link states cause data loss and asymmetric connections. The third is that decision delays are greatly increased due to frequent recalculations and other unpredictable network conditions. These problems will severely reduce the navigation’s effectiveness; thus, a navigation system using sensor networks needs to address these problems as follows:

- The system must react to changes caused by dynamics in the dangerous area.
- The response must be real-time and concurrent for the users.
- The system must avoid oscillating user movements or trapping in local minimums.
- The system performance and efficiency must be verified in practical situations.

1.2 Proactive and reactive response modes

Environments are usually dynamic in many mission critical applications. For example, a fire disaster occurs in an underground coal mine. All personnel in the area are required to move quickly from their individual locations to specified safe locations (e.g., an exit or a safe isolation belt). However, the danger zone may quickly expand or shrink over time. A navigation system must provide real-time and adaptive responses for these dynamic conditions. The navigation protocols for the response mode can then be divided into proactive and reactive responses.

1.2.1 Reactive response

The classification of proactive and reactive protocols is orthogonal to the topology-centric and data-centric models shown in Fig. 2. For example, the Computational Geometry (CG) family of protocols\(^{[3,5,10-13,16-23]}\) uses a connected communication graph formed by sensors predeployed into the environment and is denoted by \(G = (V, E)\), where \(V\) represents a set of sensors and \(E\) represents a set of wireless links between adjacent neighbors. The navigation graph is a subgraph of the communication graph formed by the mapping between the network topology and the geographic topology. Existing CG-based approaches route users from a user node (path origin) to an exit node (path terminal) based on the navigation graph. The path is selected by a metric such as the maximum minimum safe distance, an artificial potential field, or the shortest distance and is updated reactively when the selected path is attacked owing to the dynamic nature of the environment. The path is mapped to a visual graph such as a Voronoi diagram, approximate fixed cell, exact cell decomposition, or potential function. Then a shortest path is generated by a well-known algorithm such as Dijkstra’s algorithm, the Bellman-Ford algorithm, the Floyd-Warshall algorithm, or Johnson’s algorithm. The data-centric navigation models are based on the spatial and temporal relationships between events. In contrast to topology-centric approaches, the users are directed according to data or attributes from the sensors, rather than being transmitted as a packet along a specified path. The navigated user is guided by a continual search process using various search algorithms such as the uniform cost search, greedy search, A* search, beam search, or hill climbing. The goal is to minimize transmissions by eliminating data redundancy, if present, during the collection process. In a similar manner, location-aware protocols utilize the relationships between sensors and the physical world to propagate packets according to absolute or relative geographic coordinate systems. In data-centric navigation, users equipped with sensors make local or global reversals once they arrive at a dangerous node which used to be a safe node. The reversal causes the users to fall into the reactive response mode which means that the user may return to the starting point.

When facing dynamic environments with the proliferation or reduction of dangerous conditions, a path
based on reactive response approaches is not required to generate a reachable path. These approaches passively triggering recalculations of the safe path are based on situation dynamics. The users are then guided to move back and forth in a local area, called oscillation, which often leads to inefficient navigation. Moreover, oscillations inevitably result in longer stays for the people in danger, aggravate people’s panic, and eventually cause unexpected navigation failures.

1.2.2 Proactive response
Reactive response protocols attempt to solve the problem of navigating a single person out of dangerous areas through the shortest and safest path with the sensors serving as navigation infrastructure. However, people may find no way out because of dangers or obstacles in real changing environments. Consequently, they have to wait for rescuers to remove obstacles. Navigating every individual with the same strategy may lead to pedestrian congestion (i.e., even though they may know the right way to evacuate, some may spend much time waiting or being obstructed by congestion in the path), which is very dangerous in dynamic emergency environments.

Proactive response protocols are designed to deal with these problems. More and more methods take both the tendencies of the emergency dynamics and crowd congestion into account. Most approaches predict where dynamic obstacles might occur in the future by extrapolating their current velocities and directing the entity to avoid collisions accordingly. However, those approaches do not suffice when other agents are encountered. The other entities are treated as dynamic obstacles to the user, on which the user is a dynamic obstacle to others. Hence, estimating the future trajectories of other entities by extrapolating their current velocities may cause undesirable oscillations in their motion.

2 Navigation Approaches for Wireless Sensor Networks

This section describes the functional characteristics of three canonical navigation protocols based on the proactive and reactive classifications. These protocols are a distributed navigation protocol based on an artificial Potential Field (PF)\cite{10}, a location-free navigation protocol that generates the Medial Axis (MA) of the safe region\cite{13}, and a distributed in-network path planning protocol based on a Partial Reversal (PR) method\cite{15}.

2.1 PF-based distributed navigation
Lengyel et al.\cite{1} presented a robot motion planner that rasterizes configuration space obstacles into a series of bitmap slices and then uses dynamic programming to compute the distance from each point to the goal and the paths in this space. This method guarantees that the robot finds the best path to the goal. Rimon and Koditschek\cite{2} discussed an artificial potential field for robot motion planning. A robot moving according to the potential will never hit obstacles, but may get stuck in local minima. The PF-based distributed navigation protocol combines these two methods to find the best path to the goal, which is both safe and short, and modify the path to exploit the distributed nature of the sensor network.

Usually, the goal node, an exit or a rescuer, generates an attractive potential which pulls the user toward the goal. The event node, a danger zone or an obstacle, generates a repulsive potential which pushes the user away from the goal. The negated gradient of the total potential is the artificial force acting on the object as shown in Fig. 3. The direction of this force is the current best motion direction. The protocol navigates the user through three stages.

2.1.1 Computing potential field stage
The potential field is computed in the following way. When any node in the network is triggered by danger, it will diffuse a message to its neighbors carrying the danger information which includes its source node id, the potential value, and the number of hops from the

![Fig. 3 Example to demonstrate the PF-based method](image)
source of the message to the current node. When a node receives multiple messages from the same source node, it keeps only the message with the smallest number of hops. The message with the least hops is kept because that message most likely traveled along the shortest path. The current node computes a new potential from this source node. The node then broadcasts a message with its potentials and number of hops to its neighbors. After this configuration procedure, nodes may have several potentials from multiple sources. Each node computes its current danger level by adding all the potentials. The potential field information stored at each node can be used to guide an object equipped with a sensor that can talk to the network in an on-line fashion. The safest path to the goal can be computed using a dynamic programming algorithm.

2.1.2 Dynamic programming path stage
The goal node broadcasts a message with the danger degree of the path, which is zero for the goal. When a sensor node receives a message, it adds its own potential to the potential provided in the message and broadcasts a message updated with this new potential to its neighbors. If the node receives multiple messages, it selects the message with the smallest potential (corresponding to the least danger) and remembers the sender of the message.

2.1.3 Navigating user stage
A user of the sensor network can rely on the information provided by the previous two stages to get continuous feedback from the network on how to traverse the area. The user asks the network for the next path. Neighboring nodes reply with their current values and the user chooses the best possibility from the returned values.

2.2 MA-based location-free navigation
Li et al. [13] considered the navigation with limited location sensing when sensors do not give accurate location information or interactions between the users and the sensor network are limited. Location information may not always be available in many realistic situations where emergency guidance is needed, such as underground tunnels, coal mines, or complex indoor areas.

The main idea of MA navigation is to embed a distributed road map in the sensor network. This road map is built based on the distribution of dangerous areas; therefore, it can characterize safe areas in the field. The navigation system maintains the road map as a public infrastructure across the network and guides users across the field using the same road map, which reduces the overhead of planning individual routes for different users. The road map is updated in an event-driven manner as dangers vary. The navigation protocol builds the road map, guides navigation using the road map, and reacts to emergency dynamics.

2.2.1 Building the road map
The entire monitoring region is referred to as $R$ and the dangerous regions (or obstacles) are referred to as $E$. The road map is built on the region in $R$ and outside of $E$. The medial axis algorithm is used by the nodes to obtain locations relative to the dangerous regions. The medial axis is a set of points (the line with the arrow in Fig. 4), which are midway between the two closest points on boundaries of dangerous areas.

2.2.2 Guiding navigation on the road map
The road map divides region $R$ into different cells. First, a dangerous area is within a cell which is separated by road segments from other cells. Second, the distances $d_r$ and $d_c$ on the road backbone are used by each node to denote the distances to the exit and to the dangerous region. The algorithm first compares $d_c$ for each point with the largest $d_c$ and maintains all the points. If there is some $d_c$ with the same value along a path, the current node selects the paths with the shortest path with the smallest $d_r$. Each user is then guided from the inside of the cell to the road backbone. Users from different cells will be navigated towards the gateway along the road backbone. The last-mile navigation is guided along the route that connects the

![Fig. 4](example.png) Example to demonstrate the MA-based method
2.2.3 Reacting to emergency dynamics

The emergency dynamics may cause the original medial axis to no longer provide a safe route due to expanding dangerous areas. In this approach, each point in the field maintains a record of the set of the closest dangerous points and the distances between them. Once a node is switched into a dangerous area, only those points having it as their closest dangerous node need to be updated. Similarly, if a node is switched out of a dangerous area, only those points having it as their previous closest dangerous neighbor need to be updated.

2.3 PR-based distributed in-network navigation

The PF and MA methods need frequent or periodic updates to keep every sensor aware of the global and local information that may frequently change in dynamic hazardous environments. In the PR approach, each sensor exploits locally available geographic information to complete the global path planning. The dead end removal is used by sensors without outgoing edges in the communication graph and link direction readjustment is used by sensors to keep the path lengths from increasing too much at some sensors.

Hence, the first step in that path planning is to set up directed navigation links between adjacent sensors in the network. A directed graph is formed with local information for navigation in both the safe zone and dangerous zones. Each source sensor, $s_i$, is associated with a quadruple $(a_i, v_{si}, i, p_{si})$ in the safe zone or with another quadruple $(b_i, v_{ei}, i, p_{ei})$ in the dangerous zone. Initially $a_i = b_i = 0$ for all sensors. Both $v_{si}$ and $p_{si}$ are assigned the Euclidean distance (which means that the node locations must be accurate). If the Euclidean distance is the shortest distance to the nearest destination sensor, it is regarded as the safe distance. Similarly, both $v_{ei}$ and $p_{ei}$ denote the distances to the nearest dangerous sensors. For two sensors in a safe zone, if $(a_i, v_{si}, i) > (a_j, v_{sj}, j)$, i.e., $a_i > a_j$ or $(a_i = a_j$ and $v_{si} > v_{sj})$ or $(a_i = a_j$ and $v_{si} = v_{sj}$ and $i > j)$, the link is directed from $s_i$ to $s_j$.

The partial reversal method is used to prevent the user from falling into a local minimum which is common in incremental path planning. This method uses the distance information between adjacent sensors, including multiple destination sensors, and separately handles dead ends within the safe zone and nodes within dangerous zones. The partial reversal strategy is demonstrated in Fig. 5.

Finally, removal of dead ends may cause path length inefficiency at some sensors. In order to improve the path length, the reversal of dead ends may cause path length inefficiency at some sensors. Path length efficiency is handled using the virtual distance to determine the link direction instead of the reversal counter. A link which connects two sensors is adjustable if the sensors have different values of the reversal counter or an absolute difference in the virtual distances larger than the physical distances.

2.4 Analysis

Both PF and MA navigations treat the hop count as the metric for the distance between any two nodes. In real-time service, each predeployed sensor floods its state to all its neighbor nodes. When a hole occurs due to sensor failure in the network, the hop count no longer correctly indicates the danger level. At the same time, frequent flooding causes network congestion, and the end-to-end delay is increased, which leads to intolerable wait time. Although the user may find a new navigation path, rebuilding the road map or recomputing the path may result in severe power overhead and communication cost. Therefore, these approaches are impractical due to space and power limits in sensor networks.

The PR navigation system addresses the power and traffic constraints. However, the method regards a reliable link as a reachable segment of the navigation path and guides a user across the covered region just as

![Fig. 5 Example to demonstrate the PR-based method](image)
it transmits data packets from a source node to a terminal node, which is not the case. Safety is always the most important metric in emergency navigation systems. However, the user’s safety becomes even more challenging in a dynamic environment.

These three navigation approaches react passively to the environmental dynamics. A sensor navigation system with a proactive response will more effectively ensure that all users are guided out of the emergency area. Some key insights for practical systems are that if there is a path along which the user can reach the destination, but the individual navigation time \( t \) exceeds the time bound \( T \) \((t > T)\), then the emergency region is completely filled by the dangerous region \((R - E = \emptyset)\), which is a “navigable but unreachable system”. However, if there is enough time for the users to reach the destination, but they are not navigable due to obstacles or congestion, this is a “reachable but unnavigable system”. These two cases are overlooked in existing works; so they are described in detail in Section 3 and simulated to illustrate the utility of the navigability and the reachability, N-R index, in Section 4.

### 3 Navigability and Reachability

This section gives mathematical definitions for the navigability and reachability. Both the navigability and reachability are used to measure the navigation system’s efficiency and effectiveness in terms of the temporal and spatial relationships.

#### 3.1 Navigability index

Navigability is used in the field of water transport. A body of water, such as a river, canal or lake, is navigable if it is deep, wide, and slow enough for a vessel to pass. Preferably, there are few obstructions such as rocks or trees to avoid. Bridges must have sufficient clearance and high water speed may make a channel unnavigable. Water may be unnavigable because of ice, particularly in winter. Navigability also depends on the context since a small river may be navigable by smaller craft, such as a motor boat or a kayak, but unnavigable by a cruise ship. Shallow rivers may be made navigable by the installation of locks that increase and regulate water depth, or by dredging.

The movement of multiple users in an emergency environment can be analysed as a flow with the real physical environment and the available path going through a passage network. Thus, navigability means that the physical space is sufficient for navigating all trapped users and rescuers to a goal zone or point without any pain. The physical space refers specifically to a Cyber-Physical Space (CPS) featuring a combination of computational and physical elements.

The entire navigation time can be divided into \( \lambda t \) slots with the navigability index formulated as \( \Omega(t) \), constrained by the conditions:

\[
\Omega(t) = \max \sum \left( \frac{N_i(t)}{C_i(t)} \right) \\
\text{s.t. } C_i(t) \leq \max \min D_i(t), \\
N_i(t) = \max \sum n_i, \\
t \leq T,
\]

where \( C_i(t) \) denotes the capacity of the \( i \)-th path at time \( t \), which is the number of users passing a place at one time. \( N_i(t) \) indicates the number of users who need to navigate out of the emergency region along the path in time \( t \). \( n_i \) is the number of users of one place on the path at time \( t \). \( D_i(t) \) is the length of the \( i \)-th path to the nearest dangerous area at time \( t \).

The wireless sensor network application needs to include the sensing layer (physical layer) and the transmission layer, as well as the MAC layer for the navigation system. The sensor must be designed to sense the environmental dynamics including the number of users and the path capacity.

Several key problems need be settled in measuring the system navigability:

- The navigation time is divided into \( \lambda t \) slots with a reasonable \( \lambda \).
- The real-time capacity of the navigation path must be predicted using a data-centric distributed algorithm.
- The navigation strategy needs to change according to the dynamic system navigability.

#### 3.2 Reachability index

In graph theory, the reachability is the ability to get from one vertex in a directed graph to another vertex. Thus, define reachability as the ability to guide users from one place in a road map to an other place covered by the sensor network.

The formal definition of a reachable path is a navigation path consisting of a series of waypoints. For a user navigating along this path, every waypoint on this path corresponds to a time point at which the user
arrives. A navigation path is considered to be a reachable path, if and only if at any waypoint and for the corresponding time point, the user safety is guaranteed. A formalized definition of the path reachability index is

$$R(t) = \frac{\min \max u_{it}}{\max \min d_{it}} = \mathbb{R}$$

(2)

where $t_{it}$ denotes the slot at which a user reaches a node and $t_{it}$ indicates the slot when the border of the dangerous region intersects the guiding path. In terms of individual safety, the path is reachable if and only if the ratio is less than 1. That means the navigating user has passed through the region (or segment) before the danger covers it. Otherwise, the path is “swallowed” by a dangerous area.

To measure the reachability of the navigation system online:

- The velocities of the user and danger must be predicted.
- The navigated nodes must have a low-power, real-time way to update their paths.
- The independence of data transmitted to multiple mobile nodes must be guaranteed.

These measurement needs are satisfied by the three navigation approaches introduced in Section 2.

4 Simulations

The evacuation algorithms were evaluated using simulations instead of real tests which are impossible with real fires or other hazards with reasonable changes in the hazards. Thus, the protocols were implemented in Matlab and TOSSIM (on TinyOS 2.0.2) with extensive simulations to study the performance of these strategies.

The simulations mainly contained two parts. The first part was a GUI which is implemented in Visual C++ 6.0 to display the groups’ movement, the current hazard situation, and the predicted hazard development. The second part was the wireless sensor network to monitor the environment, collect data, and send packets to the sink node for further processing. The evaluations simulated a wireless sensor network in TOSSIM, such as gas leak. The area was rectangular and about 10 units by 10 units with a unit being about 30 cm. A path map with all path segments, crossing points, and exit points was partitioned into small cells with side being 0.5 units. The maximum person speed was set to 0.01 units per unit time slot (here a time slot is 1 s). In the results reported here, each crossing point and each path segment covers a number of cells. The length and width (capacity) of a path segment are determined by the number of cells it covers. In addition, the maximum allowed moving speed by people using each path segment at any time slot depends on whether its covered cells are exposed to the gas leak and on the path situation as well. For example, different path segments had different maximum allowed speeds even with no hazards to simulate flat corridors or stairs. Some sensor nodes were put at crossing points, along path segments, and inside rooms. A sensor node reporting a value larger than some threshold indicates that the cell with the sensor is hazardous.

4.1 Simulation configuration

The nodes were randomly deployed in a 10×10 area with 9, 16, 25, and 36 nodes as shown in the communication graph. The navigation path can be based on various navigation approaches. For example, a navigation path using the PF algorithm is shown in Fig. 6. The packet transmission delay is set equal to two-thirds of the speed of light. Link delay jitter was randomly set to 1 to 3 ms. The packet loss was randomly set to 0 to 0.01. The packet loss was set ∞ if there was no link between the two nodes. A quadruple <$cost, delay, delay jitter, packetloss$> is used to show the link state as shown in Fig. 7.

4.2 Navigability for a dynamic environment

The navigability is used to measure the global influence of the dynamic navigation environment. Figure 8 shows the Cumulative Distribution Function (CDF)
Navigability and Reachability Index for Emergency Navigation …

Fig. 7  A road map with the link state

Fig. 8  CDF of global navigability in different environments

"14-users/3" indicates that 14 users need to navigate with an average capacity of 3. The results in Fig. 8 show that the global navigability is greater than 1 for more than 25 users and the environment is not navigable. The navigation system is then not efficient for all users. Increasing of path capacity will improve the global navigability\(^{[24]}\).

As the emergency expands, nodes in the dangerous area become invalid which sharply increases the average navigation time as shown in Fig. 9. The average navigability improves as the number of nodes and the navigation time increase. The navigability is also improved by more time slots; however, a greater number of time slots increase the computational complexity and the communication cost.

4.3 Reachability for the navigation roadmap

The reachability is used to measure the ratio of the global path navigation time with the time covered. The three methods are used to calculate average reachability. The system is assumed to have deployed in the area, including 4 dangerous speeds of 1.4. Figure 10 shows the navigation efficiency for the three methods. If the average reachability is greater than 1, the path is unreachable. Therefore, only 60% of the paths are reachable for PF navigation. 98% of the paths are reachable with PR and 100% with MA navigation.

5 Related Work

Besides the classical methods presented in Section 2, practical, robust navigation systems have been developed for mobile robot navigation, mobile sensor networks, flow control, and link scheduling.

5.1 Mobile robots navigation

There have been many studies on path planning for multiple agents for robotic motion in real and virtual environments\(^{[4]}\). Those methods can be classified into global (or centralized) and local (or distributed) methods. The global paths represent the connectivity of the collision-free space as a graph or a roadmap, with search algorithms to compute a path for each agent\(^{[25-27]}\). Recently, some algorithms have been proposed to extend the roadmap-based methods to dynamic environments, multiple agents, and deformable models\(^{[26,28-31]}\).
However, they have only been applied to relatively simple environments composed of a few robots and restricted obstacles. These approaches may not scale well to environments with a large number of independent agents. Sud et al.\cite{19} presented an algorithm for navigating a large number of independent agents in complex and dynamic environments. Tahirovic and Magnani\cite{32} presented a navigation and motion planning algorithm for mobile vehicles in rough terrain. The passivity concept was used to enhance the Model Predictive Control (MPC) to stabilize the goal position and guarantee task completion. The framework can use any vehicle model taking into account the vehicle dynamics and terrain structure, as well as the wheel-terrain interaction. Posada et al.\cite{33} described an approach for purely vision-based mobile robot navigation. The visual obstacle avoidance and corridor following behavior rely on segmentation of the traversable floor region in the omnidirectional robocentric view. A naive Bayes classifier was used to predict the occupancy of the robots local environment by fusing evidence provided by different segmentations and models. The classification error was analyzed based on ground truth data generated by a Photonic Mixer Device (PMD) camera and manually segmented scenes.

A large number of recent studies in robotics navigated mobile robots in crowded areas (e.g. supermarkets, subways, airports and entertainment establishments) taking real-time person-robot interactions into account.

### 5.2 Mobile sensor networks

Routing research has shown that mobility and disruption-tolerance are significant factors for emergency navigation and that data independence (or broadcast authentication) is a key challenge in Cooperative Collision Warning (CCW) and Congested Road Notification (CRN) applications.

Protocols have been evaluated on mobile networks with various connectivities, packet losses, and transmission delays as in real-time navigation problems. Recently, routing using wireless mobile networks has been of great interest with diverse target environments with mobile or vehicular ad hoc networks (MANETs) and disruption-tolerant networks (DTNs). Traditional link-state or distance vector routing breaks down in DTNs where a contemporaneous end-to-end path is unavailable. Most proactive as well as on-demand MANET protocols also assume the availability of a contemporaneous end-to-end path. Also, the node mobility pattern is not known a priori so a geographic (forwarding-based) routing protocol is not suitable.

R3 described a protocol that was robust to various connectivity characteristics\cite{34}. Their simple model ignored the effects of load and interference and showed that replication yields significant delay gains if and only if the path delays have high unpredictability without making any other assumptions about the underlying delay distribution.

Fast Authentication (FastAuth)\cite{35} enables real-time, lightweight authentication by exploiting the predictability of future beacon messages, while Selective Authentication (SelAuth)\cite{35} provides fast isolation of malicious senders. As a result, invalid signatures are contained to a small area without impacting the rest of the network. These abilities are necessary for navigation with sensor networks.

### 5.3 Flow control and link scheduling

Although the throughput is often the most important performance metric in flows with congestion control, packet delay is also very important because practical congestion control protocols need to set retransmission timeouts based on the packet delay. The delay significantly impacts the recovery speed when packet loss occurs. Huang et al.\cite{36} simultaneously maximized the total utility and achieved low end-to-end delay.

Dynamic back-pressure scheduling is important for greater responsiveness to link variations, queue hot-spots, and node mobility. Backpressure routing refers to techniques grounded in stochastic network optimization\cite{37} with Utility Optimal Lyapunov Networking algorithms.

Emergence Rescue Navigation (ERN)\cite{14} uses the wireless network as a navigation infrastructure and takes into account both pedestrian congestion and rescue force flexibility. Human movements are regarded as network flows on the graph. The maximum flow and minimum cut on the graph are used to provide fireman with rescue commands that eliminate key dangerous areas, which may significantly reduce congestion and save more people.

Flow control and link scheduling are used to predict congestion caused by users or obstacles. However,
most prediction approaches are centralized algorithms which are not suitable for navigation in wireless sensor networks.

6 Conclusions and Future Work

Measurements in a distributed navigation system play an important role in the maintenance of the navigation system during emergencies, but testing and evaluation of navigation systems is not possible in practical scenarios. Therefore, measurement indexes are important in navigation systems using sensor networks. Since the users are more concerned about safety, the indexes should affect the safety and effectiveness of the system.

This paper describes two measurement indexes to evaluate navigation approaches. These navigation methods are then evaluated using a quantitative method to evaluate their navigability and reachability. Since predictions of dangerous areas are crucial for global safety, future work will consider a predicatable distributed navigation protocol.

References


