

Utilizing Roadmaps in Evacuation Planning



Samuel Rodriguez and Nancy M. Amato

Parasol Lab., Dept. of Computer Science and Engineering, Texas A&M University, College Station, Texas, 77843-3112, USA

Abstract—In this paper we describe utilization of roadmaps in a general evacuation planning system for complex 3D environments. The problem consists of heterogeneous groups of agents whose behaviors and properties affect usage of the environment when creating evacuation plans. This planning system includes behaviors for those agents evacuating and directors that may be guiding the agents to improve evacuation. One aspect we focus on is modeling different forms of direction and communication between agents.

Index Terms—evacuation, agent guidance, building environments, cooperative agent interaction

I. INTRODUCTION

The evacuation simulation presented here consists of large numbers of agents performing complex tasks, which includes interaction between groups of agents. This is a difficult problem with applications in emergency training systems, robotics, computer graphics and animation. These simulation strategies can be used to study and train for emergency or disaster scenarios including civilian crowd control, evacuation of a building or city block, and many other immersive training situations. Behavioral-based simulations allow for a user to observe and study the effects of population type, director ability and agent characteristics on evacuation, without having to see this in practice. There has been a lot of work in simulating large numbers of agents evacuating an environment using a basic evacuation strategy. There is less work on many agents in an environment, performing evacuation plans, interacting with control agents, each with a wide range of agent properties, and adhering to environmental restrictions.

We present some novel techniques to this problem with our *main contributions* being:

- Evacuation in complex 3D environments using a roadmap-based approach
- Heterogeneous agent populations
- Incorporating different forms of communication to improve environmental usage

Other approaches to the evacuation planning problem are often restricted in their scope. We define the problem in a general

way to include evacuation and control behaviors that can be used, agent characteristics that define the evacuation, area types that should be considered, and interaction and cooperation that may occur. We are able to simulate these

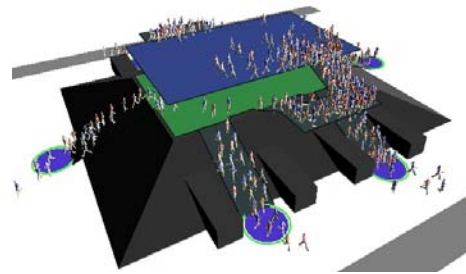


Fig. 1. A basic 3-level building example. (Color Plate 11)

evacuation scenarios in complex 3D environments by utilizing a general roadmap-based approach. The controlling behaviors we consider require some level of cooperation between the agents doing the controlling and reaction by the agents that are to be controlled.

This work is similar to [1] which proposed a way to integrate evacuation behaviors with roadmap-based techniques in simple 2D environments. This paper greatly extends [1] by considering 3D environments and how communication and heterogeneous parameters effect evacuation.

II. RELATED WORK

2.1 Roadmap-Based Behaviors

In [2, 3], the benefits of integrating roadmap-based path planning techniques with flocking techniques were explored. The global information provided by the roadmaps improved the behaviors of autonomous characters, and in particular, enabled more sophisticated group behavior than are possible using traditional flocking methods [4]. The roadmap provides a convenient representation of global information in complex environments. Modifying node and edge weights in a shared map also enables implicit communication between agents.

Shepherding behaviors are a type of flocking behavior in which outside agents guide or control members of a flock. In [5, 6], advanced techniques were presented for shepherds attempting to control a flock and simulated various herding behaviors. The agents in the flock were given basic flocking

*This research supported in part by NSF awards CRI-0551685, CCF- 0833199, CCF-0830753, IIS-096053, IIS-0917266 by THECB NHARP award 000512-0097-2009, by Chevron, IBM, Intel, Oracle/Sun and by Award KUS-C1-016-04, made by King Abdullah University of Science and Technology (KAUST).

Manuscript Received on February 1, 2011

E-mail: sor8786@neo.tamu.edu

techniques and were afraid of the shepherds, and so would react to the shepherds movement and try to avoid the shepherd. The shepherds were also equipped with a roadmap to find routes through the environment.

2.2 Evacuation Planning

There has been much work addressing different aspects of this problem. A survey of the main approaches that have been taken to study the evacuation problem is presented in [7]. Four main approaches were described: flow-based, cellular automata, agent-based and activity-based models. The focus of this work and the majority of related work presented here is on agent-based models. This allows the study of scenarios with varying agent capabilities and populations where we can look at the finer motion of the agents.

An early work, in agent-based evacuation is described in [8]. The goal of this system was to be able to handle thousands of individuals escaping in large, geometrically complex environments. The system allowed for automatic input of building plans and discretized the area into grid cells, assigned weights based on if a grid cell is a solid object, open space or a final destination. The underlying grid was used to compute a maximum travel distance. This work presented some interesting results but focused only on individual agent evacuation.

Some grid-based approaches have been proposed to model large scale evacuation in multi-level environments [9, 10]. A fluid-dynamics approach is presented in [9] to describe the trajectories of agents while moving through a building in the presence of hazards. Another grid-based approach, described in [10], handles complex multi-level buildings which includes stairwells with agents having the ability to have their own characteristics. The environmental usage of stairwells can be analyzed. Cellular automata approaches are often limited by having simple agent grid cell occupancy and often force oversimplification of the environments.

The way individual characteristics impact evacuation efficiency is studied in [11, 12]. This work is based on the social forces model developed in [13, 14] but includes other agent characteristics, which include group structure, dependence level of agent, altruism level and the desired speed of agents. Varying the individual agent properties impact the average flow of people.

Another survey, focused on virtual crowds, is presented in [15]. The work describes many approaches that have been proposed for crowd simulation such as the types of models that have been developed with an in-depth look at a few different models. Two of these are described next.

The idea of different levels of agent knowledge and planning ability is considered in [16]. This is in part due to psychology studies which show that building occupants usually decide to use familiar exits, such as where they entered the building. In some cases, emergency exits or exits not normally used for circulation are often ignored. Different agent types are considered including trained leaders, untrained leaders and followers. One interesting result they found is the optimal number of leader agents.

In [17], a system is developed for simulating the local motion

and global way finding behaviors of crowds moving in a natural manner within dynamically changing virtual environments. They are able to simulate patient and impatient agents and pushing between agents. Improvements on previous work, [13], were made by considering factors that reduce shaking and vibration caused by applying social forces in densely crowded areas. Other interesting factors of this work include avoiding fallen agents, wall avoidance, focusing on regions ahead of the agent, and pushing between agents. They also consider the problem of avoiding bottlenecks to pick better routes.

An approach to find the optimal evacuation time in simple 2D environments is described in [18] where k occupants with n possible exits use an evacuation function to select routes. This approach makes many assumptions about the environment such as assuming a uniform distribution of agents, the existence of a route to each exit and not considering the fact that valid paths may change as the environment changes. Similarly, we consider the idea that minimum evacuation time may not always lead to the best usage of the environment.

A benefit of agent-based systems is that heterogeneous agent populations can be created to study evacuation. This was done by varying the dependence and altruism level of agents [11], varying agent knowledge and training information [16], and including patient and impatient agents [17]. This is one reason we chose to develop an agent-based system.

Flexible frameworks need to be able to handle a number of situations. Many different approaches have been proposed to handle specialized environments including pedestrians moving through Penn Station [19], underground malls [20] and a passenger ship [21].

This all points to the need for a flexible system that can be used in many situations with the ability to simulate a number of different conditions. The system required will also need the ability to easily vary the agent population types, agent capabilities and cooperation level.

III. OVERVIEW OF BEHAVIORAL-BASED EVACUATION PLANNING

In this section, we describe the main components in our evacuation planning system. These include the problem definition, behaviors the agents are executing, usage of the roadmap, areas considered in our problem formulation and the overall simulation cycle.

3.1 Problem Definition

Studying the evacuation planning problem requires a clear definition of the environment, type of agents involved, the kind of evacuation that will take place and the interactions between agents that are considered.

The physical representation of our environment consists of obstacles in the environment, surfaces that the agent can move on, valid transitions between these surfaces and a bounding box restricting the available area. The problem also requires an input of certain areas in the environment that will dictate how evacuation proceeds. These areas are further labeled as being exits, safe areas and/or dangerous areas. The label of each of these areas (and the agents' knowledge of them) will affect the

evacuation plans that are generated. An example 2D environment with different area locations labeled is shown in Fig. 2(a). The simulation is initialized with different sets of agents, which are deployed at predefined locations and equipped with different behaviors (i.e., evacuation or

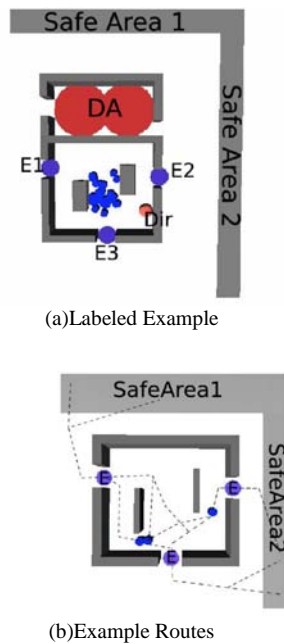


Fig. 2. (a) An example environment is shown with evacuating agents (shown in blue), Exits (E#) along walls of the rooms, safe areas and a dangerous area (DA). A director is also present at the corner of the room. (b) Example routes in a simple environment shown for each agent.

direction). In our simulations, we can define agents with varying capabilities and initial conditions. The agent capabilities and environmental knowledge will be further described in Section IV.

Evacuating agents respond to a situation in the environment by planning a route from their location to a known safe area in the environment. The evacuation concludes once all agents have reached a position within a valid safe area. This represents a behavioral-based system that results in evacuation plans that are determined by agent behaviors, capabilities and interaction. These different aspects are key in creating a general evacuation planning system.

3.2 Simulation Cycle

The outline of our simulation loop is given in Algorithm 1.

Algorithm 1 General Simulation Loop

```

Input: simulator sim, environment env
1: groupsall = sim.getAllGroups()
2: groupssel = sim.getNextSetOfGroups()
3: for g ∈ groupssel do
4:   g → applyBehaviorRule(env)
5: end for
6: for g ∈ groupsall do
7:   updateState(g)
8: end for
9: ResolveStateInEnvironment(groupsall,env)
10: Evaluation

```

We use a general concept of individual agent, calling them a

group. In this way the our framework is extendible, especially when creating behaviors, since a behavior created for individual agents can easily be applied to a clustering of agents, with additional logic used to ensure grouping restrictions are maintained.

The behavior that the agent is equipped with will determine how the agent reacts. These behaviors determine the actions that the agent or groups of agents take. At each time step, a subset of all the simulation groups execute a behavior rule. The behaviors developed need to be dynamic enough so they can be applied to a group at any point and the group will then start performing that behavior. At each time step, all groups update their state based on the last plan that was generated by the behavior rule. The state is then resolved with other agents and the environment (i.e. preventing collision).

3.3 Roadmaps and their Utilization

Using roadmaps in evacuation planning systems and crowd motion provide many advantages over traditional navigation techniques. A roadmap is a light weight representation of the environment. It allows an agent to find routes through the environment as it approximates the connectivity of the free space in the environment and can be quickly updated as the situation changes. Paths can be selected that avoid certain areas or are biased towards other areas.

Roadmaps are graph representations of an environment that encode feasible, collision-free paths through the environment. Agents navigating in environments consisting of multi-level surfaces, such as buildings, need the ability to map these spaces. The agents encode valid movements on the surfaces by first sampling collision-free nodes on each surface. Connections are allowed between nodes on the same surface in which the straight line along that surface, projected to a 2-dimensional plane, remains completely within the projected polygon and is not in collision with objects on the surface. Then connections are then between surfaces that are defined to be connected based on an input environmental configuration.

Sharing a roadmap among agents may seem to be an unrealistic capability, however, this may allow agents to select paths that more closely resemble paths that humans would select. This is because humans inherently have reasoning skills about where other agents may go and areas that may be congested. We investigate techniques for using a shared roadmap in our evacuation behavior that improve the evacuation routes selected. The shared roadmap may be adjusted by each agent resulting in the path selected being affected by other agents implicitly.

3.4 Area Types

The areas we consider in our system deal with where an agent should be biased toward or avoid, if possible. They are defined by a range of area in the environment, for example near the boundary of the environment but can also be represented as position-radius areas.

These areas are considered environmental knowledge and affect the routes selected. A safe area is considered a goal region for an agent. These areas are used to generate goal locations. Exits are another area type considered. Agents must select routes that pass through an available exit in order to reach their final destination. Although an agent may know about

various safe areas or exits, they may be marked as either *available* or *toavoid*. Dangerous or hazardous areas may also exist in the environment. The agent avoidance properties of these areas will determine how potential routes that pass through these areas are weighted. An area type that is more dynamic and not explicitly defined is a congested area. These can arise from the over usage of certain areas (i.e. an exit) or if a faster agent is trying to navigate around slower agents. These congested areas are implicitly identified when finding a route through an area that has been weighted as overused.

IV. AGENT PARAMETERS

We attempt to model a range of agent characteristics, shown in Table I, each affecting some aspect of the evacuation behaviors. While this is not a complete set of agent characteristics, it is a substantial enough set to give the resulting simulation very dynamic characteristics. Agent capabilities such as speed and sensory characteristics affect how an agent moves and what is sensed. The environmental knowledge of each agent will affect the evacuation routes selected. The areas known to an agent potentially alter the routes as will the roadmap used by the agent.

There are many behavioral variables that affect an agent. Some of these include the amount of cooperation allowed and weighting of dangerous areas by agents.

A number of navigation abilities are given to the agents. The agents use the roadmap as a guide through the environment, however they do not need to follow the paths exactly. Reactive avoidance is employed as a low level force rule to avoid the nearest agent. This is similar to the basic flocking rule [4], considering only separation to the nearest agent. The level of path optimization done by each agent, reducing the potential length of the route, is a parameter we allow to be variable between agents. This also prevents agents that are following a similar path from appearing to follow the *same* path.

V. EVACUATION BEHAVIOR

Agents planning a path through the environment to a safe location have to take into account different environmental aspects. This includes considering available areas such as safe exits, potential dangerous regions, and other locations that are considered safe to create an evacuation route. The roadmap is well suited for finding these kinds of paths. We achieve this by using path evaluation and roadmap re-weighting such that paths found adhere to restrictions that may be known to an agent.

The evacuation behavior first updates an agent's information about known areas. An agent can update it's information by either first hand observation or communication with directing agents. The paths that will be selected are by the lowest edge weight. In this way, areas are avoided by assigning higher edge weights to areas that have been marked as areas to avoid or known dangerous areas. This will prevent routes being selected that pass through these areas. The agent will then evaluate each route from it's location to each available exit and the safe area

nearest the exit. The route with the best score is then selected as the evacuation route. An example of how a safe route can be selected using environmental information is shown in Algorithm 2.

Algorithm 2 Safe Route Selection

Input: Agent g_i , known exits E , known safe areas SA , known dangerous areas DA , environment env

- 1: $g_i.updateDangerousAreas(env, DA)$
- 2: $g_i.updateRdmpWithKnownAreas()$
- 3: **for all** $e \in E$ **do**
- 4: **if** $s.hasMarkedExitAsAvoid(e)$ **then**
- 5: **continue**
- 6: **end if**
- 7: $P1 = findPath(g_i.getPos(), Near(e))$
- 8: $nSA = nearestAvailableSafeArea(e, g_i)$
- 9: $P2 = findPath(e, nSA.posInArea())$
- 10: $P3 = P1 + P2$
- 11: $score = evaluateExitRoute(P3)$
- 12: **if** $score$ of $P3$ is best **then**
- 13: Save $P3, score$
- 14: **end if**
- 15: **end for**

VI. DIRECTION BEHAVIOR

The goal of our direction behaviors is to simulate a range of objects and forms of direction that may exist in a real evacuation. These forms of direction can provide evacuating agents either local or global information, depending on the capability of the director and the kind of direction being modeled.

Local direction is either a barrier or other locally perceived information provided by an agent. *Global direction* is modeled by allowing the directors to provide the evacuating agents with whom they communicate with global information (outside of sensory range) about specified exits, safe areas or known hazardous areas that should be avoided or that they should be aware of. The directors are equipped with information about the areas that they are providing. In an immersive training system, the placement and effectiveness of directors can be examined. This kind of direction could be a human or robot directing the evacuating agents, or an announcement on a speaker system.

TABLE 1: HETEROGENEOUS AGENT PROPERTIES CONSIDERED IN THIS WORK.

Agent capabilities:
- speed capabilities: velocity, acceleration
- sensory capabilities: view radius, angle
Behavioral variables:
- cooperation – between/with directors
- roadmap sharing
- dangerous areas weighting (bravery parameter)
Environmental Knowledge:
- Known areas (exits, safe areas, dangerous areas)
- Roadmap information: mapping, quality, etc
Navigation abilities:
- agent reactive avoidance - near agent avoid
- path optimization – urgency

We attempt to model a realistic form of cooperation by

allowing a certain probability for direction to be accepted. This enables us to simulate agents that may not trust or accept certain forms of direction. This can include ignoring a barrier or sign that exists in the environment or even ignoring a police agent. To handle this situation, a directing agent may have potentially useful information to an evacuating agent, g_i but a probability, P_d is assigned which describes the chance that g_i will accept this information. This value, P_d is determined from the effectiveness of the directing agent and the cooperation value from g_i . Depending on the type of interaction that has taken place (accepting the information or not), g_i remembers the encountered director for a predefined number of time steps. The interaction is either restricted by the evacuating agent's sensory range or by the directional agent's alert distance.

We also model two forms of director communication. An *alert* involves providing information to the evacuating agent and allowing the agent to use the information as needed (update plan if a better one exists). *Assignment* allows for a director to guide the evacuating agent as desired by the director. This is a more active form of direction and allows for the directing agents to cooperate and coordinate in order to improve evacuation and better utilize environmental resources (i.e. exits).

An example of cooperative weighted assignment of exits is shown in Algorithm 3.

Algorithm 3 Cooperative Weighted Exit Assignment

Input: Director $_j$, Agent g_j to direct

- 1: $U_{total} = \sum_{i \in Ex} usage(e_i)$
- 2: **for** $e_i \in Director_j \rightarrow getExits()$ **do**
- 3: $ue_i = Usage_{assigned}(e_i, U_{total})$
- 4: $Dist_i[e_i] = restrictedPathToExit(e_i, g_j)$
- 5: $Dist_{total} += Dist_i[e_i]$
- 6: **end for**
- 7: compute weights, W_i for $i \in EX$, where

$$W_i = ue_i + \left(\frac{Dist_{total} - Dist_i[e_i]}{Dist_{total}} \right)$$
- 8: select exit probabilistically based on weight
- 9: re-weight exit usage based on exit selection

In this algorithm, a director attempts to select a good exit for an evacuating agent. The director considers usage of each exit and computes paths to the exits that the director is biasing evacuation toward. A weight is then computed for each exit which considers both the usage of the exits and distance it would take to reach the exit (with the directors knowledge of the environment). The director then probabilistically selects the assigned exit based on the weights generated. Coordination between directors comes in the form of communicating exit information about the assigned exit usage. The coordination can also be predefined by the user of the evacuation planning system by specifying which set of exits each director is biasing evacuation toward.

VII. BETTER EVACUATION ROUTES

Roadmap Re-weighting. We utilize roadmap re-weighting in a number of ways. This can be used to bias agents away from certain areas or prevent passage through other areas. The modification to the roadmap edges can be by either a

predefined value or some scaling factor of the edges previous weight. The agents use this re-weighting scheme to avoid nodes in the roadmap near dangerous areas or exits marked as *toavoid* that are known to the agent.

Weighted Path Parameterization. Another re-weighting parameter we look at is the cost of passage. Whereas the normal weighting of dangerous areas is to make the weights in the roadmap so high that passage is infeasible, there are situations where passage through a dangerous area is needed. In this situation, portions of the roadmap that pass through the dangerous areas are scaled by a factor (unique to each agent). A low scaling factor will make passage through these areas slightly less desirable where as a high scaling factor makes passage unlikely.

Path Sharing. A shared roadmap between evacuating agents is beneficial to better utilize the environment. This can be used to help the agents make intelligent decisions which human agents may make with little effort. These can be about areas of the environment that are overused or potential exits that may lead to quick evacuation. In using roadmap re-weighting to select better paths, an agent that has selected an evacuation route will then scale that route in the roadmap by some pre-defined value. This will make the edges along that route seem somewhat more or less desirable (depending on the scaling factor) to the next agent selecting a route.

Path Optimization. After an agent extracts an evacuation route, the path can be smoothed to make it more natural and reduce the distance the agent will have to travel through the environment. We define the amount that the agents attempt to smooth the paths through agent parameters. In this way, agents can be given a high sense of urgency by heavily smoothing paths whereas agents with less urgency may do less path smoothing.

VIII. EXPERIMENTAL RESULTS AND DISCUSSION

We present results in two example environments under a number of simulation conditions. The first environment, shown in Fig. 1, is a symmetrical three level building. The second environment, Fig. 3, also consists of 3 levels but is much more complex. All examples show 500 agents evacuating and considering environmental and interaction conditions dictated by the scenario. Evacuation times reported are in number of simulated time steps averaged over ten trial runs. Agents equipped with heterogeneous parameters have values centered around specifications of the homogeneous agent parameters. Additional examples and animations are provided on our webpage:

<http://parasol.tamu.edu/groups/amatogroup/research/flock/>

8.1 Basic Three Level Building

The third level of this environment has two stairways to the second level which has six available exits and three safe areas. Due to the nature of the environment, agents are biased toward exits 1–4 because safe areas are nearby and the stairwells lead to areas near the exit. A safe area is also located near exit 5. Exit 6 is somewhat out of the way with no safe area located near by.

The scenarios considered in this example are:

- (A) homogeneous agents,
- (B) heterogeneous agents,
- (C) shared roadmap between evacuating agents,
- (D) heterogeneous agents with differing knowledge,
- (E) 2 blocking agents,
- (F) 2 cooperating directors, and
- (G) 2 cooperating directors with cooperation parameter of 0.95.

In all the scenarios considered, each agent is equipped without full environmental knowledge except in scenario (D). The scenarios with directors are blocking exits 4 and 5.

It is interesting to note that in scenario (A), with homogeneous agents selecting the shortest possible route, the evacuation time is minimized. This could be considered ideal conditions, however the agents almost completely ignore an available exit. Just equipping the agents with heterogeneous parameters, (B), causes the evacuation time to increase greatly. While agents sharing a roadmap, (C), causes the evacuation time to increase, it also results in a more even usage of the environment, shown by the exit usage. Similar usage occurs when the agents have differing knowledge, representing conditions where agents in an environment may not always know a building's floor plan but rather have knowledge of areas they passed through while in the building. In (E), agents that encounter the blockade then proceed to their nearest available exit. Another interesting aspect is that when cooperating directing agents are included, a more even usage of the environment is seen.

TABLE 2: RESULTS SHOWING EVACUATION UNDER DIFFERENT SCENARIOS (FROM ENVIRONMENT IN FIG. 1)

Scenario	Evac. Time	Exit Usage					
		1	2	3	4	5	6
(A)	2040	115	70	52	152	112	0.1
(B)	2814	107	71	54	148	112	0.1
(C)	3605	97	96	76	122	72	37
(D)	3260	91	84	72	113	102	39
(E)	3382	163	77	235	0	0	26
(F)	3961	175	143	118	0	0	64
(G)	3943	171	140	114	8	6	61

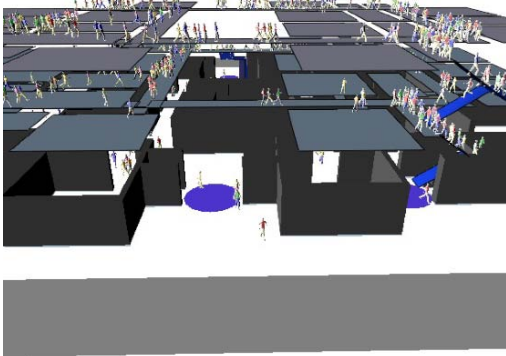


Fig. 3. A complex 3-level building example. (Color Plate 12)

8.2 Three Level Building

This environment, shown in Fig. 3 without walls, consists of many rooms, passages and stairwells between levels. There are also five exits to two available safe areas. Exit 2 and 4 are the most heavily used as these are the main passages to an exit for agents in the upper levels in the environment. There is a stairwell near Exit 1 between the first and second floor.

The scenarios considered here are:

- (A) homogeneous agents,
- (B) heterogeneous agents,
- (C) shared roadmap, heterogeneous agents, and
- (D) 3 cooperating directing agents blocking off one exit, heterogeneous agents.

All agents are equipped with full environmental knowledge. In (D), the directing agents (one per floor) are all guiding agents away from an exit near a corner stair well. Again, we can show that a heterogeneous set of agents evacuates slower than a group of agents with the same capabilities. This represents scenarios where some agents are in a hurry and some unaffected by the situation. Agents sharing information (C) are able to make better usage of the environment, also a common situation seen in the real world where some people are willing to take a longer route in order to avoid congestion. In fact, many agents select routes that avoid the more heavily used exits (2 and 4). Including directors (D), allows the agents to avoid the exit in question (Exit 4) however some agents do get near enough that it is considered reached in our usage metric.

TABLE 3: RESULTS SHOWING EVACUATION UNDER DIFFERENT SCENARIOS (FROM ENVIRONMENT IN FIG. 3)

Scenario	Evac. Time	Exit Usage				
		1	2	3	4	5
(A)	3577	25	176	92	205	0
(B)	4375	22	209	77	185	6
(C)	6653	138	158	99	68	35
(D)	9850	107	297	86	1.4	6

IX. CONCLUSION

We presented a framework for evacuation systems which can be applied in complex environments. The main features of this system include our general roadmap-based approach and interacting, heterogeneous sets of agents for more realistic evacuation scenarios. We also consider communication both within a set of evacuating agents and between controlling and evacuating agents. The direction aspect of this work alone, is often not considered in many evacuation systems. These are some of the more complex evacuation scenarios considered in the literature, both in the type of environments tested and interactions occurring.

REFERENCES

- [1] S. Rodriguez and N. M. Amato, "Behavior-based evacuation planning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2010, pp. 350-355.

- [2] O. Bayazit, J. Lien, and N. M. Amato, "Roadmap-based flocking for complex environments," in *Proc. Pacific Graphics*, Oct. 2002, pp. 104–113.
- [3] O. Bayazit, J. M. Lien, and N. M. Amato, "Better group behaviors in complex environments using global roadmaps," in *Artif. Life*, Dec. 2002, pp. 362–370.
- [4] C. W. Reynolds, "Flocks, herds, and schools: A distributed behavioral model," in *Computer Graphics*, 1987, pp. 25–34.
- [5] J.-M. Lien, O. B. Bayazit, R.-T. Sowell, S. Rodriguez, and N. M. Amato, "Shepherding behaviors," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, April 2004, pp. 4159–4164.
- [6] J.-M. Lien, S. Rodriguez, J.-P. Malric, and N. M. Amato, "Shepherding behaviors with multiple shepherds," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, April 2005, pp. 3413–3418.
- [7] G. Santos and B. E. Aguirre, "A critical review of emergency evacuation simulation models," in *Workshop on Building Occupant Movement During Fire Emergencies*, 2005, pp. 27–52.
- [8] P. Thompson and E. Marchant, "A computer model for the evacuation of large building populations," *Fire Safety Journal*, vol. 24, pp. 131–148, 1995.
- [9] E. Galea, J. Perez-Galparsoro, and J. Pearce, "A brief description of the exodus evacuation model," in *Proceedings of the International Conference on Fire Safety*, 1993, pp. 149–162.
- [10] Mott-MacDonald, et. al., *STEPS Simulation of Transient Evacuation and Pedestrian Movements User Manual*. Mott-MacDonald, 2003.
- [11] A. Braun, S. R. Musse, L. P. L. D. Oliveira, and B. E. J. Bodmann, "Modeling individual behaviors in crowd simulation," in *Proceedings of the 16th International Conference on Computer Animation and Social Agents (CASA-03)*, 2003, pp. 143–148.
- [12] A. Braun, B. E. J. Bodmann, and S. R. Musse, "Simulating virtual crowds in emergency situations," in *Proceedings of the ACM symposium on Virtual reality software and technology*, 2005, pp. 244–252.
- [13] D. Helbing, I. Farkas, and T. Vicsek, "Simulating dynamical features of escape panic," in *NATURE*, 2000, pp. 487–490.
- [14] D. Helbing, L. Buzna, A. Johansson, and T. Werner, "Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions," in *TRANSPORTATION SCIENCE*, 2005, pp. 1–24.
- [15] N. Pelechano, J. Allbeck, and N. Badler, *Virtual Crowds: Methods, Simulation, and Control*. Synthesis Lectures on Computer Graphics and Animation, Morgan & Claypool, 2008.
- [16] N. Pelechano and N. Badler, "Modeling crowd and trained leader behavior during building evacuation," in *IEEE Computer Graphics and Applications*, 26(6), 2006, pp. 80–86.
- [17] N. Pelechano, J. Allbeck, and N. Badler, "Controlling individual agents in high-density crowd simulation," in *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2007.
- [18] S. C. Pursals and F. G. Garzon, "Optimal building evacuation time considering evacuation routes," *European Journal of Operational Research*, vol. 192, pp. 692–699, 2009.
- [19] W. Shao and D. Terzopoulos, "Autonomous pedestrians," in *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. New York, NY, USA: ACM Press, 2005, pp. 19–28.
- [20] H. Furuta and M. Yasui, "Evacuation simulation in underground mall by artificial life technology," in *Proceedings of the Fourth International Symposium on Uncertainty Modeling and Analysis*, 2003.
- [21] C. W. Johnson and L. Nilsen-Nygaard, "Extending the use of evacuation simulators to support counter terrorism," in *International Systems Safety Conference*, 2008.



Nancy M. Amato is a professor of computer science and engineering at Texas A&M University where she co-directs the Parasol Lab, is Deputy Director of the Institute for Applied Math and Computational Science, is past chair of the Council of Principal Investigators and chair of the Alliance for Bioinformatics, Computational Biology, and Systems Biology. She received undergraduate degrees in Mathematical Sciences and Economics from Stanford University, and masters and Ph.D. degrees in Computer Science from UC Berkeley and UIUC. She received an NSF CAREER Award, is an IEEE Fellow, is an ACM Distinguished Speaker, and was a Distinguished Lecturer for the IEEE Robotics and Automation Society. She was an Associate Editor of the IEEE Transactions on Robotics and Automation and the IEEE Transactions on Parallel and Distributed Systems. She is a member of the Computing Research Association's Committee on the Status of Women in Computing Research (CRA-W) and of the Coalition to Diversity Computing (CDC); she co-directs the CDC/CRA-W Distributed Research Experiences for Undergraduates (DREU) program and the CDC/CRA-W Distinguished Lecture Series. She is Co-Chair of the NCWIT Academic Alliance. Her main areas of research focus are motion planning and robotics, computational biology and geometry, and parallel and distributed computing.



Samuel O. Rodriguez received his Bachelor of Science degree in Computer Engineering in 2002 from Texas A&M University, College Station, TX. Since 2003 he has been working on his Ph.D. in Computer Science, also from Texas A&M University. He received a National Physical Science Consortium (NPSC) Fellowship, LSAMP Bridge to the Doctorate Fellowship, and University Graduate Merit Fellowship. His research focus is on robotics, graphics and multi-agent systems.

During his time in graduate school he has been a research and teaching assistant working with Dr. Nancy Amato.