Interactive Behavior Modeling for Large-Scale Crowd Simulations

Ming C. Lin
Dinesh Manocha
Department of Computer Science
University of North Carolina
Chapel Hill, NC 27599-3175
919-962-1974, 919-962-1749
lin@cs.unc.edu, dnu@cs.unc.edu

Latika Eifert
Angel Rodriguez
Army Research Laboratory HRED STTC
SFC Paul Ray Smith, Simulation and Training Technology Center
12423 Research Parkway, Orlando, FL 32826
407-384-5338, 407-208-3009
Latika.Eifert@us.army.mil, angel.luis.rodriguez@us.army.mil

Keywords:
multi-agent simulation, emergent behaviors, collision-avoidance, least-effort crowds

ABSTRACT: We survey some of our recent works on real-time simulation of large-scale crowd movement. Our primary focuses include techniques for automatically computing collision-free trajectories for each agent. We show how these methods can lead to generating emergent crowd behaviors, such as lane formation, edge effects, vortices, congestion avoidance, swirling and modeling varying crowd density.

1 Introduction

Crowds are ubiquitous in the real world and are an important component for evacuation planning and emergency response training systems. The problem of simulating a large number of virtual agents and human crowds has been studied in different fields including computer graphics, social and behavioral sciences, architecture, physics, psychology, civil and traffic engineering, and robotics. In particular, crowds are regarded as complex dynamical systems that exhibit distinct characteristics, such as emergent behaviors, self-organization, and pattern formation due to multi-scale interactions among individuals and groups.

Today’s first responders need a new generation of technology and the resources to prepare for and respond to terrorist attacks, natural disasters, and large-scale emergencies. The next-generation simulation and experiential technologies for the first responders can possibly help better prepare them for evacuation planning and disaster response, facilitate training experiences, and enable leaders and law-enforcement personnel to optimize their tactics, by using what-if simulations based on the real situations in which they might work and more effectively evaluating various alternatives. Crowd simulations with appropriate collective behavior models are needed for evacuation planning and training of first responders for unexpected events in urban environments. These technologies also have diverse applications in architecture design, emergency evacuation, urban planning, personnel training, education and entertainment.

One of the key challenges is to simulate large-scale crowds with tens or hundreds of thousands of agents. Such large crowds are becoming increasingly common in cities across the globe. Furthermore, some applications such as games or virtual environments need interactive simulation capabilities, i.e. simulating at 30 frame per second (fps) or more on current desktop systems. In addition to the overall performance, another major challenge is generating realistic crowd behaviors. Despite decades of observation and studies, collective behaviors are particularly not well understood for groups with non-uniform spatial distribution and heterogeneous behavior characteristics. Such scenarios include pedestrian movement, evacuation flows in complex structures, and coupled human-natural systems.

In this paper, we survey some of our recent works on addressing several key computational problems of simulating large-scale crowds in complex dynamic environments. These techniques are necessary building blocks to compute crowd movement, and eventually generate emergent collective behaviors. limit our coverage here to recent methods that are used to simulate the motion of large numbers of virtual agents at interactive rates. We give a brief overview of prior work on multi-agent and crowd simulation and present a broad classification of prior methods into various categories in Section 2. Next, we describe many recent approaches for local collision avoidance between multiple agents and global navigation that have been designed by our research team at the University of North Carolina at Chapel Hill and Army Research Laboratory HRED STTC. These include (a) a new local collision avoidance algorithm, ClearPath, between multiple agents [Guy et al. 2009] in Section 3; (b) a least-effort approach that can generate energy-efficient trajectories and emergent behaviors [Guy et al. 2010] in Section 4; (c) a novel formulation to model aggregate dynamics of dense crowds [Narain et al. 2009] in Section 5; (d) an efficient spatial conceptualization of avatar behaviors to effectively model personalities and social protocols [Yeh et al. 2008] in Section 6; (e) an effective approach to direct and control virtual crowds using navigational fields [Patil et al. May 2009] in Section 7.
These methods can be combined together to generate complex crowd behaviors that are frequently observed in urban environments. Furthermore, we use the computational capabilities of commodity multi-core and many-core processors to exploit parallelism on modern hardware to achieve interactive performance for tens of thousands of agents. We also demonstrate their application to simulating pedestrian crossings, motion of thousands of pilgrims at a mosque, trade shows, etc.

2 Related Work

In this section, we give a brief overview of prior work related to multi-agent navigation and interactive crowd simulation. There is also extensive work on generating emergent behaviors in computer animation, virtual environments and social sciences.

Several techniques have been proposed to animate or simulate large groups of autonomous agents or crowds. Most of these methods use a rather simple representation for each agent - this could be a circular shape in a 2D plane or a cylindrical object in the 3D space, and compute a collision-free trajectory for each agent. After computing the trajectory using a simple representation, these techniques use either footstep planning or walking synthesis methods to compute a human-like motion for each agent along the given trajectory.

At a broad level, prior methods for computing crowd movements can be classified into the following six categories [Guy et al. 2010]:

- **Potential-based methods**: These algorithms focus on modeling agents as particles with potentials and forces [Helbing and Molnar 1995].
- **Boid-like methods**: These approaches are based on the seminal work of Reynolds which create simple rules for computing the velocities [Reynolds 1987; Reynolds 1999].
- **Geometric methods**: These algorithms compute collision-free paths using sampling in the velocity space [van den Berg et al. 2009a] or by using optimization methods [Guy et al. 2009; van den Berg et al. 2009b].
- **Field based methods**: These algorithms tend to compute fields for agents to follow [Yersin et al. 2005; Pettré et al. 2009], or generate navigation fields for different agents based on continuum theories of flows [Treuille et al. 2006] or fluid models [Narain et al. 2009].
- **Least effort crowds**: Based on the classic principle of Least Effort proposed by Zipf (1949), many researchers have used that formulation to model the paths of agents [Still 2000; Sarmady et al. 2009]. Recently, it has been combined with multi-agent collision avoidance algorithms [van den Berg et al. 2009b] and used to efficiently and automatically generate emergent behaviors for a large number of agents [Guy et al. 2010].

In addition to these broad classifications, there are many other specific approaches designed to generate crowd behavior based on cognitive modeling and behavior [Shao and Terzopoulos 2005; Yu and Terzopoulos 2007] or sociological or psychological factors [Pelechano et al. 2007]. Other techniques include directing crowd behaviors using guidance field specified by the user or extracted from videos and computing smooth navigation fields [Patil et al. May 2009].

3 ClearPath – Collision Avoidance for Multi-Agent Simulations

In our work presented here, we assume that the scene consists of multiple heterogeneous agents, each of which is moving toward an independent goal. The goal position can change or some agents may only have intermediate goals. Furthermore, the scene consists of static and dynamic obstacles and it is important that each agent avoids collisions with other agents and all the obstacles. The behavior of each agent is governed by some extrinsic and intrinsic parameters and computed in a distributed manner for each agent independently. The overall simulation proceeds in discrete time steps and we update the state of each agent, including its position and velocity during each time step based on its goal position and the other agents or obstacles in the scene.

Our overall framework assumes that the agents are moving on a 2D plane, though our approach can be extended to handle agents moving in 3D space. At any time instance, each agent has the information about the position and velocity of nearby agents. The proximity information in terms of nearby agents can be computed using a kD-tree. Moreover, the simplest representation of each agent corresponds to a circle or a convex polygon.

Our approach is decomposed into two levels. The higher-level deals with the global path planning towards the goal using a precomputed roadmap [LaValle 2006], and the lower-level addresses local collision avoidance and navigation using ClearPath [Guy et al. 2009]. We assume that the other agents are dynamic obstacles whose future motions are predicted as linear extrapolations of their current velocities. ClearPath provides a principle to select a velocity for agent \( A_i \) and implicitly assumes that the other agents \( A_j \) use similar collision avoidance reasoning. Essentially we pose the local collision avoidance problem for \( N \) agents as a combinatorial optimization problem, subject to motion constraints we impose on the system.

Any feasible solution to all of constraints, which are separately formulated for each agent, will guarantee collision avoidance. We solve this problem as a quadratic optimization function with non-convex linear constraints for each agent. It can be shown to be NP-Hard [Guy et al. 2009] for non-constant dimensions via reduction to quadratic integer programming. It has a polynomial time solution when the dimensionality of the constraints is constant – two in our case. In practice, ClearPath is more than one order of magnitude faster than prior velocity-obstacle based methods [van den Berg et al. 2008]. Please see the video demonstration at:

http://gamma.cs.unc.edu/CA/ClearPath.mov.

4 The Least-Effort Crowd

Many researchers in different fields have noticed that the human and crowd motion in real-world scenarios is governed by the principle of least effort (PLE). One of the earliest works in this area is Zipf’s classic book on human behavior [Zipf 1949]. The basic essence of this principle is that humans tend to move through the environment to their goals using the least amount of effort, by minimizing the time,
Figure 1: The least-effort crowd simulation algorithm can automatically generate many emergent crowd behaviors at interactive rates in the simulation of Shibuya Crossing (left, middle) that models a busy crossing at the Shibuya Station in Tokyo, Japan. (right). The trajectory for each agent is computed based on minimizing an effort function. There is a high correlation between the trajectories computed by the least-effort crowd simulation algorithm and the real trajectories captured by a video.

5 Hybrid Crowds

Dense crowds exhibit a low interpersonal distance and a corresponding loss of individual freedom of motion. This observation suggests that the behavior of such crowds may be modeled efficiently at a coarse level, treating its motion as the flow of a single aggregate system. Based on such an abstraction, we develop a novel inter-agent avoidance model that decouples the computational cost of local planning from the number of agents, allowing very large-scale crowds consisting of hundreds of thousands of agents to be simulated at interactive rates [Narain et al. 2009]. A key characteristic of large-scale crowds is not only the number of agents, but also the density of the agents in terms of number of agents per square meter. Whenever the density goes beyond 4 agents per square-meter, which is considered very high, the crowd is treated as a whole. This includes modeling the crowd as a fluid and a continuum that responds to local influences by assuming that the individuals in the continuum move so as to optimize their behavior to reach non-local objectives [Hughes 2002].

Our overall method for modeling very large-scale crowds combines a Lagrangian representation [Helbing and Molnar 1995; Helbing et al. 2005] of individuals with a coarser Eulerian crowd model [Hughes 2002; Hughes 2003; Treuille et al. 2006], thus capturing both the discrete motion of each agent and the macroscopic flow of the crowd. In dense crowds, the finite spatial extent occupied by humans becomes a significant factor. This effect introduces new challenges, as the flow varies from freely compressible when the density is
low to incompressible when the agents are close together. This characteristic is shared by many other dynamical systems consisting of numerous objects of finite size, including granular materials, hair, and dense traffic. The new mathematical formulation to model the dynamics of such aggregate systems in a principled manner can be found in [Narain et al. 2009].

**Results:** Fig. 2 shows some of the results for dense crowds of up to 100,000 agents closely packed in complex scenes simulated at interactive rates. The authors measured the performance of their algorithm on an Intel Core i7-965 machine at 3.2 GHz with 6 GB of RAM. Even with very large numbers of agents, they can achieve close to interactive performance. This method supports general scenarios with independent, heterogeneous agents and the number of unique goals has no effect on the performance of the system. Please see the video demonstration at:

**http://gamma.cs.unc.edu/DenseCrowds/dense-crowd.mov.**

**6 Composite Agents**

We introduce the notion of “composite agents” to effectively model different avatar behaviors for agent-based crowd simulation. Each composite agent consists of a basic agent that is associated with one or more proxy agents [Yeh et al. 2008]. A proxy agent possesses the same kind of external properties as an agent, i.e. if every agent’s external property consists of the velocity, then the proxy agent must have its own velocity as well. The values of these external properties, can be different, e.g. the proxy can possess a velocity that is different from anyone else. The external properties of a proxy \( P_j \) is denoted as \( \varepsilon_j \). The internal state \( \iota_j \) of a proxy agent, however, need not be the same set of properties of the basic agent. We also define that \( P_j \) has access to the internal state \( \iota_i \) of its parent \( A_i \). We denote the set of all proxy agents being simulated as \( \text{Proxies} = \bigcup_i \text{proxy}(A_i) \)

This formulation allows an agent with given physical properties to exercise its influence over other agents and the environment. We can consider such an influence in terms of spatial extent and/or personal space of each agent, as well as ‘aura’ that some agents can impose on others nearby. Composite agents can be introduced to most agent-based simulation systems and used to model emergent behaviors among individuals. These behaviors include aggression, impatience, intimidation, leadership, trailblazing and approach-avoidance conflict, etc. in complex scenes. In practice, there is negligible overhead of introducing composite agents in crowd simulation.

Fig. 3 demonstrates a variety of behaviors generated using our novel concept of Composite Agents on top of any agent-based simulation. Please see the video demonstration at:

**http://gamma.cs.unc.edu/CompAgent/CompAgent.avi**
7 Directing Crowds Using Navigation Fields

Most existing agent-based systems assume that each agent is an independent decision making entity. Some of the methods also focus on group-level behaviors and complex rules for decision making. The problem with these approaches is that interactions of an agent with other agents or with the environment are often performed at a local level and can sometimes result in undesirable macroscopic behaviors. Due to the complex inter-agent interactions and multi-agent collision avoidance, it is often difficult to generate desired crowd movements or motion patterns that follow the local rules.

In a recent work, we address the problem of directing the flow of agents in a simulation and interactively control the simulation at run time [Patil et al. May 2009]. Their approach is mainly designed for goal-directed multi-agent systems, where each agent has knowledge of the environment and a desired goal position at each step of the simulation. The goal position for each agent can be computed from a higher-level objective and can also dynamically change during the simulation.

This approach for directing the crowds uses discretized guidance fields to direct the agents. These guidance fields correspond to a vector field, which are used to specify the motion direction of the agents. Based on these inputs, they compute a unified, goal-directed, smooth navigation field that avoids collisions with the obstacles in the environment. The guidance fields can be edited by a user to interactively control the trajectories of the agents in an ongoing simulation, while guaranteeing that their individual objectives are attained. The microscopic behaviors, such as local collision avoidance, personal space and communication between individual agents, are governed by the underlying agent-based simulation algorithm.

Results: This approach is general and applicable to a variety of existing agent-based methods. The usefulness of this approach is illustrated in the context of several simulation scenarios shown in Fig. 4. The user edits the simulation by specifying guidance fields, that are either drawn by the user or extracted from a video sequence. The overall approach can be useful from both artistic and data-driven perspectives, as it allows the user to interactively model some macroscopic phenomena and group dynamics. Please see the video demonstration at:

http://gamma.cs.unc.edu/DCrowd/DCrowd.avi.

8 Application to An Evacuation Scenario

Our crowd simulation system synthesizes these methods described above and finds biomechanically realistic paths for each individual agent in the simulation. To navigate in the virtual environment, agents walks towards their desired goal or exit while avoiding others nearby. To validate our approach, we compare the results predicted by our simulation techniques to those collected in real-world evacuation experiments. Figure 5(a) shows a diagram of an overhead view of one such scenario. Figure 5(b) shows an still from one of the runs in progress, where a participant had just passed the exit. We analyzed data from four different exit studies. While there are minor variations between studies as shown in Figure 6, there are clear consistent trends that hold among them. We varied the width of the simulated exit and cal-
calculated the flow rate of the simulated individuals past the exit. The results of this study shown in Figure 6 indicates that as with the human data [Kretz et al. 2006], increasing the exit width increases the flow through the exit. More importantly, if we compare the flow rates at any given width, we see that the predicted flow consistently falls within the range observed in the human studies for a wide variety of exit widths.

Figure 5: Evacuation Experiment (a) Overhead diagram of the evacuation scenario. Participants (circles) are asked to move down the hall (vertical lines) through the narrow passage. (b) Still image of participants taking part of study. The walls can be adjusted to widen or narrow the exit.

Figure 6: Flow Rate Comparison A comparison of the effect of exit width (horizontal axis) on the flow rate (vertical axis) for real humans (dashed lines) and simulated individuals (solid line). Simulated individuals have similar exit rates as the real humans for a variety of exit widths.

9 Discussion

In this paper, we have presented a brief survey of algorithms for real-time behavior modeling and simulation of large-scale crowds in dynamic scenes. We also highlighted their applications on interactive crowd simulation in urban simulations and virtual environments.

Among these methods, ClearPath is designed to take advantage of upcoming many-core architectures to achieve real-time collision avoidance of hundreds of thousands of discrete agents at interactive rates. In contrast, the 'hybrid crowd' approach for large-scale crowds is better suited for highly dense scenarios, capable of simulating up to millions of individual agents in nearly real time by exploiting their constrained movement. The approach to direct and control crowds using navigation fields is general and versatile. It can use any low-level collision avoidance algorithm and is complimentary to most current methods that compute trajectories for the agents. The resulting system allows the user to specify the navigation fields by either sketching paths directly in the scene via an intuitive authoring interface or by importing motion flow fields extracted automatically from crowd video footage, in order to generate the desired crowd behaviors. This technique is complementary to other methods and most suitable for interactive editing and hypothesis testing for 'what-if' scenarios.

Finally, the algorithms to simulate least-effort crowds and composite agents can be combined and used in any multi-agent simulation systems. In practice, these algorithms generate natural-looking trajectories and crowd behaviors, as well as individual agents that we observe. They are also relatively easy to parallelize on multi-core architectures.

These techniques are promising and effective for modeling and simulation of many virtual agents in urban settings for interactive applications, such as training of first responders. Based on the techniques surveyed in this paper, we are continuing to develop application systems and interfaces to enable more efficient evacuation planning and more effective crowd control for emergency response. For example, we can extract the crowd flows from a live recording of crowds in a panic situation due to an unexpected event using computer vision techniques. Then we use the extracted crowd flows to drive a crowd simulation through navigation fields (Section 7) to “replay” the past and simulate the alternative ‘what-if’ scenarios in the future with a combination of methods from Sections 3-6.

Acknowledgments: We would like to thank all our students and collaborators, whose work have been described in this survey. These include Jur van den Berg, Jatin Chuggani, Sean Curtis, Pradeep Dubey, Abhinav Golas, Stephen Guy, Rahul Narain, and Sachin Patil. The research presented here is supported in part by RDECOM.

References


HELBRING, D., BUZNA, L., JOHANSSON, A., AND WERNER, T. 2005. Self-organized pedestrian crowd dynamics: Ex-
experiments, simulations, and design solutions. *Transportation Science* 39, 1–24.


**Author Biographies**

MING C. LIN is currently John R. and Louise S. Parker Distinguished Professor of Computer Science at the University of North Carolina at Chapel Hill. Her research interests include physically-based modeling, robotics and simulated environments. She has published more than 220 papers in these areas and worked closely with industry and DoD agencies on modeling and simulation.

DINESH MANOCHA is currently Phi Delta Theta/Matthew Mason Distinguished Professor of Computer Science at the University of North Carolina at Chapel Hill. His research interests include virtual environments, robotics and parallel computing. He has published more than 300 papers in these areas and worked closely with industry and Army Labs in terms of transitioning these technologies into DoD systems.

LATIKA EIFERT is currently the Lead Principle Investigator and Science and Technology Manager in Embedded Training for the Ground Simulation Environment Technology Branch of the ARL-HRED-STTC. She has over 20 years of engineering experience including Lockheed Martin, consulting and now here at STTC. She has published several papers in training and simulation technology.

ANGEL RODRIGUEZ is currently the Branch Chief for Ground Simulation Environment Technology Division at the ARL-HRED-STTC. He has over 28 years in research and acquisition in government simulation and training field being employed with General Electric Aerospace, Naval Aviation Training Systems Division, U.S. Army Simulation Training and Instrumentation Command and now with STTC. He has published over thirteen research and acquisition papers in the area of training and simulation.
Figure 4: (a, b, c) Motion patterns detected in a video of a crosswalk in Hong Kong with 3 frames from the original video; (d) Motion flow field; (e) Detected motion patterns; (f) Video-based motion patterns from (e) used as guidance fields; (g) Agent motion generated by the navigation fields. (h) Sketch-based guidance fields to specify lane formation in the simulation; (i) Lane formation generated by goal-directed navigation fields.