Pacific Graphics 2014
J. Keyser, Y. J. Kim, and P. Wonka
(Guest Editors)

Volume 33 (2014), Number 7

Minimal Energy Based Hybrid Model For Multi-Agent Simulations

PaperID-1122

Abstract

The behavior of large human crowd is very complicated and subtle due to complex local collision strategies for varying crowd density distribution in the scene. We present a hybrid approach to simulate complex crowd behavior by combining continuous crowd algorithm and agent-based algorithm. For dense groups, we introduce a minimal-energy method to minimize conflict between dense groups and constrain the maximum density of each grid; For sparse groups, we adapt RVO algorithm to simulate agent-agent and agent-group interactions.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1. Introduction

The simulation of human crowds in realistic interactive scenes is a necessay and challenging task. Crowd simulation technologies have been widely applied in social psychology, transportation research and architecture. There're two major branches in crowd simulation research field, agent-based methods and continuous crowd methods. For agent-based methods, the total number of agents and density of the crowd are restrict due to the computational bottleneck. Continuous crowd methods which based on assumptions of medium to large crowd, suffer from poorer performance in low density regions.

In this paper, we focus on the problem of simulating crowds under varying densities. In high density regions, the movement of each agent are highly restricted and influenced by neighbor agents. From the macroscopic view, the movement of the crowd is driven by collision and cooperation between groups instead of individual will of each agent [NGCL09]. We define this kind of behavior as group-group interaction. In low density regions, agents have higher freedom of movement. The collision avoidance decision of each agent is based on its neighbor agents and goal. We define behaviors like this as agent-agent interaction, which is a very common assumption in agent-based methods [GCS*09, vdBSGM11, OPOD10]. Besides, there's a third type of behavior which we defined as agent-group interaction. We can see agent-group interactions happen in the boundary of dense crowds where individual agents outside the crowd may choose to join or avoid the crowd. Based on such obeservation, we develop a hybrid model to choose proper algorithm for each agent and integrate continuous crowd simulation result with agent-based simulation process to support all the agent-agent, agent-group and group-group interactions.

We proposed a novel minimal-energy based method to simulate interactions between dense groups. A dense group is a small set of agents sharing a common neighborhood. As for each group, the influence from neighbor groups is similar to label yielding process, we consider the collision and cooperation between groups as a labelling problem. With the formulation we proposed to adapt group-group interaction into labelling problem, Graph-Cut or Belief Propagation which has been successfully used in stereo matching, image restoration and optical flow [FH04, TF03, YWY*09], can be applied to solve the problem of dense crowd simulation.

The key contribution of this work can be summarized as follows:

- a hybrid model combining continuous crowd algorithm and agent-based algorithm for varying densities crowd to simulate interactions between agents and groups. §4
- a minimal-energy model which reduces the dense group collision energy to model interactions between groups. §3

We demonstrate the quality and performance of agent-group and group-group interactions simulated by our approach on two senarios. Our approach has a small computational cost and can simulate thousands of agents interactively on a single-core CPU. More analysis of our approach can be found in §5.

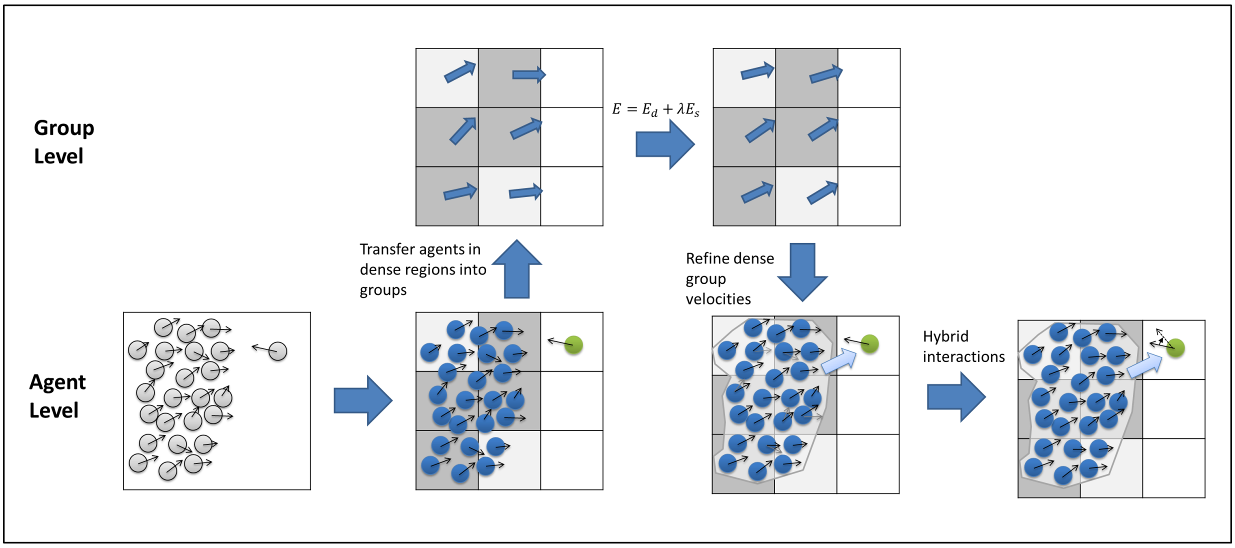

Figure 1: *Hybrid Model Pipeline.*

2. Related Work

In this section, we give a biref overview of prior work of crowd simulation and multi-agent system. There're two steps, navigation and collision avoidance in most crowd simulation systems. Navigation, also been called as global-planning, is the process to find a path from the current agent position to the goal that avoid collision with static obstacles in the scene. There're lots of work on this topic [FT-T99, BLA02, LD04, SGA*07, SAC*08]. We don't go for details of previous work in this aspect.

Collision avoidance is another hot topic with a large amount of works been proposed. One of the traditional ways to classify collision avoidance methods is based on the representation of crowds, either discrete or continuous. In discrete representation, also called as agent-based representation, the collision avoidance decision is made by each individual agent respect to the other agents and obstacles. A number of different models like social forces [HM95, H-BJW05, GLM09, SGA*07], socialogical factors [MTT97], psychological factors [PAB08] and synthetic-vision based steering [OPOD10] were used to formulate the collision avoidance behavior of agents. There're also a lot number of methods based on spatial and geometric relationships between agents [vdBLM08, vdBGLM09, GCS*09, vdBSG-M11]. As geometrically-based methods showed very good quality and became very popular these years, many other models were adapted into these methods for further quality improvement and more complicated agent behavior simulation, like least effort energy model [GCC*10], personality traits model [GKLM11] and stress model [KGML12]. Although agent-based methods works quite good in many situations, they suffer from the poor performance with dense crowds.

In continuous representation, agents are influenced by their dense neibourhood and have less freedom to make independant collision avoidance decisions. Continuum theory for pedestrains flow was first proposed by [Hug03] and extended by [TCP06]. The approach of [TCP06] combined global planning and local collision avoidance into a single pipeline and provided good performance for medium size crowd with a common goal. However, this approach is not suitable for tightly packed crowds and varying goals. The state-of-art work of continuous methods is [NGCL09], which combined a Lagrangian representation of individuals with a coarser Eulerian crowd model to capture discrete motion of individual agents and macroscopic flow of crowds. The work of [NG-CL09] can simulate great number of agents at interactive rates and have good performance on dense crowds. However, continuous methods fail to provide accurate results in low density regions as the assumptions of density are no longer hold.

To make good use of advantages of both methods and avoid the drawbacks, a hybridization of those two methods could be a good solution. [GNL13] presented a long range collision avoidance model which can work with both agent-based and continuous representations. Unlike their work, our approach focus on more complicated crowd behavior including agent-agent, agent-group and group-group interactions. We can simulate different crowd behaviors, e.g. whether a agent join or avoid a crowd in front, by simply changing parameters in our hybrid model system.

3. Dense Crowd Flow Energy Minimization

In this section, we introduce our approach of minimizing collision energy of dense crowds. In the observation of dense crowds behavior, groups of agents have inter-group collision, merging and negotiation while the freedom of movement of individual agents are reduced [TCP06, NGCL09, GNL13]. Based on the observation, we define two types of collision energy, inter-group and inter-agent collision energy. We use the continuous dense crowd model from [NGCL09] to represent and minimize inter-agent collision behavior(§3.1). We develop our inter-group energy model and employ Graph Cut to minimize inter-group collision energy(§3.2). Besides, we introduce a series of constraints to limit maximum density of each group(§3.3).

3.1. Dense Crowd Representation

Like [NGCL09], we first use Navigation Mesh as global planning algorithm for each agent(see for details). Then, the information of discrete agents is transfered to grids by the particle-in-cell method of fluid simulation [NGCL09]. The density ρ and velocity $\bar{\mathbf{v}}$ can be computed as,

$$\rho(\mathbf{p}) = \sum_i \omega\mathbf{p}(\mathbf{p}_i)m_i \tag{1}$$

$$\bar{\mathbf{v}}(\mathbf{p}) = \frac{\sum_i \omega\mathbf{p}(\mathbf{p}_i)\tilde{\mathbf{v}}_i}{\sum_i \omega\mathbf{p}(\mathbf{p}_i)} \tag{2}$$

where $\mathbf{x}_i$ is the position of the agent, $\tilde{\mathbf{v}}_i$ is the prefered velocity of the agent, m_i is the mass of each agent which are unity. $\omega\mathbf{p}(\mathbf{p}_i)$ is the bilinear interpolation weight associated with the agent position.

After the crowd is converted to continuous representation, we can refine the crowd information by minimizing the inter-group collision energy descripted in (§3.2 and §3.3).

Then, the density $\rho(\mathbf{p}_i)$ and velocity $\mathbf{v}(\mathbf{p}_i)$ at agent position can be interpolated from grid density and grid velocity with the same method mentioned in [TCP06]. We can get the final velocity of each agent by interpolation between the continuum velocity $\mathbf{v}(\mathbf{p}_i)$ and the agent's prefered velocity $\tilde{\mathbf{v}}_i$, depending on the crowd density $\rho(\mathbf{p}_i)$ at its location:

$$\mathbf{v}_i = \tilde{\mathbf{v}}_i + \frac{\rho(\mathbf{p}_i)}{\rho_{max}}(\mathbf{v}(\mathbf{p}_i) - \tilde{\mathbf{v}}_i) \tag{3}$$

3.2. Energy Minimization Model

According to our observation of dense crowd behavior, collisions, merging and negotiations happen between agent groups in dense crowds. However, not like local collision avoidance strategies for agents which can be considered as rigid-body, groups are highly influenced by neighbor groups and have the trend of sharing the same movement. Inspired by the observation, we develop a minimal energy model to describe and solve inter-group collision problem.

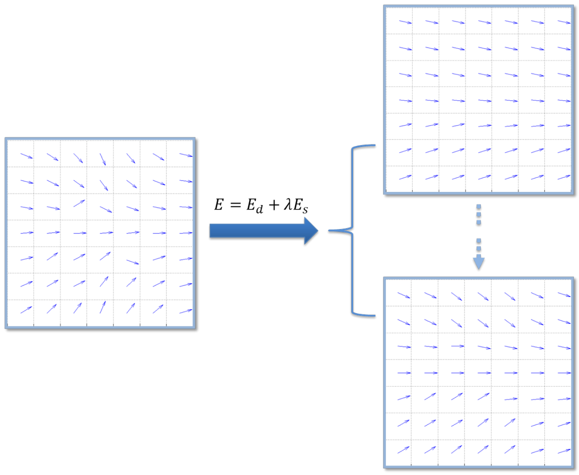

Figure 2: *Minimal-Engergy Process Illustration. The left image is group of dense grids with grid velocities. Images on the right side are grids with refined velocities. From right-top to right-bottom, the λ of smooth term decreased which means each grid is less influenced by neighbor grids.*

We define the inter-group collision problem as a labelling problem by assigning to each grid i a label l_i. Each grid maps to a group of agent and the number of grids are n. We call the energy minimization process twice to mininize collision energy of horizontal and vertical direction seperately. For each optimization pass, we can have m labels by dividing horizontal or vertical space into m pieces, each of them represent a segment x or y axis. The energy function E, composed of a data energy E_d and a smoothness energy E_s, is defined as:

$$E = E_d + \lambda E_s \tag{4}$$

The data energy E_d is the sum of data cost of each grid with a label $d_i(l_i)$. In our energy model, the data cost $d_i(l_i)$ is defined as the absolute distance between projection on x or y axis of the average grid velocity $\bar{\mathbf{v}}(\mathbf{p})$, and the value of selected label which means the effort of the group to change it's velocity on x or y axis to certain label.

$$E_d = \sum_i d_i(l_i) \tag{5}$$

The smoothness energy E_s represents the velocity difference between neighbor grids. We use standard 4 neighbor system for dense grids, so that the smoothness energy is the sum of spacially varying horizontal and vertical neighbor smoothness costs $V_{ij}(l_i, l_j)$, where if $i = (p,q)$ and $j = (s,t)$ then $|p-s| + |q-t| = 1$. Let $\mathcal{N}$ denotes the set of all such neighboring grid pairs, the smoothness energy is

$$E_s = \sum_{\{i,j\}\in\mathcal{N}} V(|l_i, l_j|) \tag{6}$$

In equation 6, $V(|l_i, l_j|)$, also can be written as $V(\Delta l)$, is

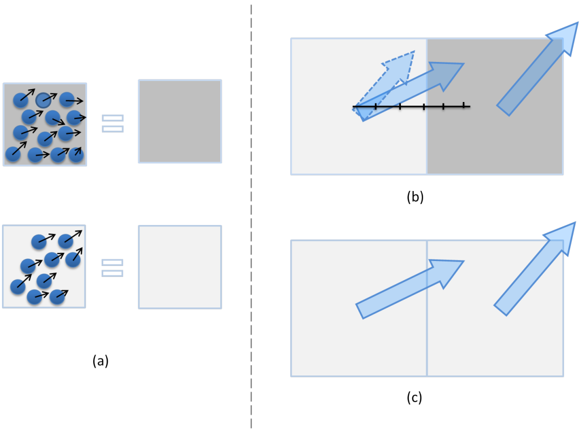

Figure 3: *Illustration of maximum grid density constraints. Image (a) is the transfering process from discrete agent information to continuous representation. Dark gray grids mean maximum density while light gray grids mean lower density. In image (b), the right grid is full of agents and becomes imcompressible. The velocity of left grid on x axis couldn't be larger than the right grid's. In image(c), both grids are not in maximum density means the velocity is not constrained.*

a non-decreasing function of the label difference, and can be directly computed as absolute differenct between two labels. The scale value λ in Equation 4 can be used to control the influence by neighbors. We can see the different optimization result by changing λ in Figure 2. In our model, $V(|l_i, l_j|)$ is adapted to restrict the maximum density of each group(§3.3).

With the definition of energy function E, we can use Graph Cut or Belief Propagation algorithms to compute labels for each grid to achieve the minimal energy cost. In our implementation, we use the Graph Cut implementation from [BVZ01, BK04, KZ04].

3.3. Constraints

While trying to reach the minimal energy cost of inter-group collision, we should also keep the density of each group below the maximum density ρ_{max} because of the incompressibility of human body. If the density of the grid i is below ρ_{max}, agents can be pushed into the grid from neighbor grids. When the density of the grid i is greater than ρ_{max}, the the grid is incompressible. The smooth term should be,

$$V(|l_i, l_j|) = \begin{cases} \alpha\,|l_i - l_j|\,, \rho_i \leq \rho_{max} \\ \Delta_{max}, \rho_i > \rho_{max} \end{cases} \quad (7)$$

where Δ_{max} is a constant value as a penalty of labels which could lead to potentially overcrowded (see in Figure 3).

4. Hybrid Crowd Simulation

Continuous crowd methods and agent-based methods are both based on some assumptions about agent number and crowd density. This point of view is also shared by other works like [GNL13]. If the assumption of large dense crowd can not hold, collisions between agents and oscillations caused by rearrange process often happen when applying continuous crowd algorithms in low density regions. On the other hand, high density is a nightmare of most agent-based algorithm. Large amount of agent could lead to high computational cost and low frame rate. Further more, for some geometric algorithms like RVO, whose collision avoidance process is based on information from neighbor agents, suffer from the risk failling to find a collision free velocity.

We addressed the problems by proposing a hybrid model to choose proper algorithm for each group of agents based on the density of the group. There're five steps in our hybrid model pipeline(see in Figure 1),

1. Seperate agents into groups, choose continuous or discrete algorithm for groups(see 4.1).
2. Transfer information of agents in dense regions into continuous representation(see 3.1).
3. Reduce inter-group collision and refine group velocities by minimal-energy method(see 3.2 and §3.3)
4. Update velocities of agents in dense region by group velocities(see 3.1) and extract contour and average velocities of dense groups(see 4.2).
5. Add dense groups into agent-based algorithm as moving obstacles and simulate agent-agent and agent-group interactions(see §4.3).

4.1. Crowd Seperation

Our crowd seperation method are based on group densities from §3.1. There're two cases for varying densities. For the ith grid, if $\rho_i \in [\rho_{min}, \rho_{max}]$, it should be marked as a dense grid and apply continuous algorithm described in §3. Otherwise, if $\rho_i \in (0, \rho_{min})$, it should be marked as a sparse grid and apply agent-based algorithm like RVO.

4.2. Dense Crowd Contour Extraction

After crowd seperation, we have a set of dense grids for continuous crowd method and a set of sparse girds for agent-based method. Before we move to the inter-group collision energy minimization step, we should first merge dense grids into several groups by detecting connected components. Then we extract the contour $\mathcal{C}_i$ of each group $\mathcal{G}_i$. As we only need rough contours to support agent-group interactions, we use a simple contour extraction algorithm described below,

- For each group $\mathcal{G}_i$ DO
  1. Sort agents in group $\mathcal{G}_i$ by its vertical position in ascending order, positions of agents on the top and bottom of the list are marked as $\mathbf{p}_{min}$ and $\mathbf{p}_{max}$

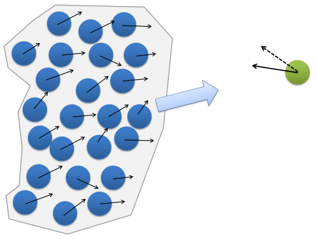

Figure 4: *Illustration of hybrid crowd interactions. The blue agents are in high density region, while the green agents are in low density region. As the contour and average of the dense group is extracted and the group is set as obstacle, the green agent tries to avoid collide with the dense group.*

2. Compute the vertical range $\Delta = \mathbf{p}_{max,y} - \mathbf{p}_{min,y}$ of the group
3. Divide Δ into N buckets, $N = \Delta/r$. Put agents into buckets according to its vertical position
4. For each bucket j, sort agents in this bucket by horizontal position in ascending order, and mark the position of agents on the top and bottom of the list are marked as $\mathbf{p}_{j,left}$ and $\mathbf{p}_{j,right}$.
5. Connect points in such an order: $\mathbf{p}_{min}$, $\mathbf{p}_{1,left}$, $\mathbf{p}_{2,left}$, $\cdots$, $\mathbf{p}_{N-1,left}$, $\mathbf{p}_{N,left}$, $\mathbf{p}_{max}$, $\mathbf{p}_{N,right}$, $\mathbf{p}_{N-1,right}$, $\cdots$, $\mathbf{p}_{2,right}$, $\mathbf{p}_{1,right}$ and $\mathbf{p}_{min}$ to create a contour polygon $\mathcal{C}_i$.

- END for loop

4.3. Hybrid Crowds Interaction

With agents in low density regions and contours of dense groups, we can simulate agent-agent and agent-group collision avoidance by applying a agent-based algorithm. Here, we use reciprocal velocity obstacle (RVO) algorithm [vd-BLM08] implemented in the RVO2 library. Before calling RVO to simulate, we add contours of dense groups with a offset $\Delta x = \Delta t \cdot \sum_i \mathbf{v}_i$, and then put the updated contours into RVO as obstacles to support interactions between agents and groups. We define a scale parameter $\mathcal{S}$ as the discomfort distance to control whether the agent prefer to avoid or join a dense group. The obstacle will be a Minkowski sum of the contour with a offset and a circle with radius of $\mathcal{S}$. The simulation of hybrid crowd interactions is illustrated in Figure 4.

5. Results

Our approach was implemented in C/C++. We provide our run-time performance on a Intel Core i7 at 2.9GHz with 4 cases in Table 1. We measured the per frame performance of both agent-based module and continuous module, and the entire hybrid pipeline. From Table 1, we can see that agent-based is the performance bottleneck when there're large

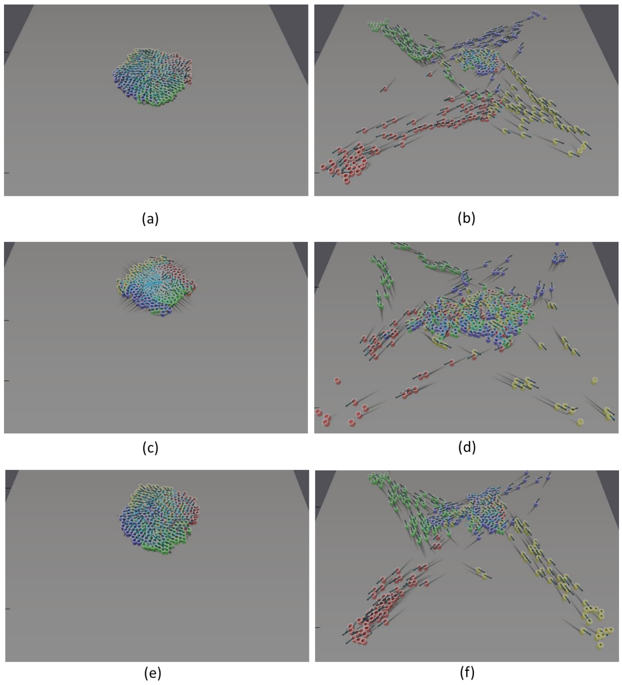

Figure 5: *4 groups of agents crossing the center. The top row is generated by our hybrid pipeline. The middle row is simulated only by RVO algorithm. The bottom row is simulated only by continuous method implemented in our pipeline.*

amount of agents in the scene. While the agent number is below 10K, our method can reach real-time performance for most cases.

We tested our approach on a number of scenes. The first case shows different performances of our hybrid model, pure agent-based algorithm and pure continuous crowd algorithm, see in Figure 5. In this case, four groups with same number of people on the four corners of a square, heading towads each other and crossing the center. In the simulation by RVO algorithm, the four groups of people stuck at the center, and only agents on the edge of the crowd can keep moving, while the agents in the hybrid model and continuous crowd algorithm begin to spin as a whole group until each group find path to their goal. The groups in hybrid model and continuous crowd algorithm share similar movement till most agents stop stucking and spinning in the center. Then, the rest agents in the center of hybrid model can get faster to their goal because some of the agents are applied agent-based algorithm and able to avoid the dense groups due to the decrease of the density of the center region.

Our second case shows different behaviors of agent-group interaction with different discomfort distance $\mathcal{S}$ of group contour. From the experiment, we can see that, with greater discomfort distance $\mathcal{S}$ which means a larger obstacle, indi-

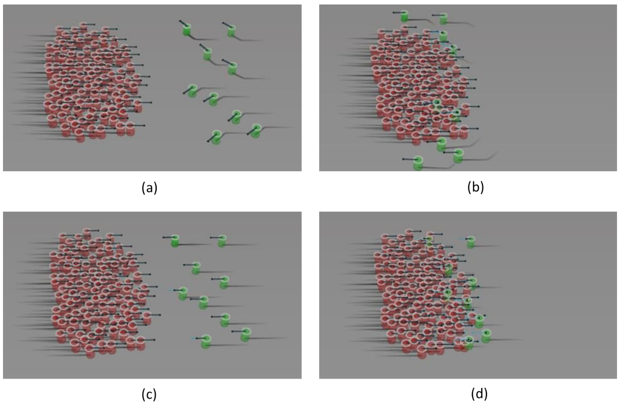

Figure 6: *Interaction between agents and groups. There're two groups of agents crossing the center. The left(red) group is a dense group. The right(green) group is a low density group. The two rows of the image are set by different discomfort distance $\mathcal{S}$ of group contour. In the top row, the dense group is considered as a rigid object, the individual agents in green try to avoid collision with the group. In the bottom row, the dense group is considered as a much smaller obstacle than its real size, the individual agents choose to join the group.*

Scene	# Agent	Modules	Time(ms)
Crossing	500	Agent-based	0.7 ~3.8
		Continuous	0.2 ~2.7
		Total Hybrid	1.1 ~6.9
4 Groups	4000	Agent-based	6.6 ~22.1
		Continuous	24.2 ~42.5
		Total Hybrid	37.4 ~49.0
4 Groups L	10000	Agent-based	60.7 ~373.4
		Continuous	36.6 ~61.5
		Total Hybrid	107.3 ~379.5
4 Groups XL	25000	Agent-based	0.7 ~436.1
		Continuous	25.7 ~170
		Total Hybrid	143.7 ~480.3

Table 1: *Performance of our method*

vidual agents are more likely to avoid collision with dense groups. And individual agents are more likely to join dense groups with smaller $\mathcal{S}$.

6. Conclusion

In this paper, we present a hybrid model to simulate varying densities crowd. We define three types of interactions in crowd behaviors, agent-agent interaction, agent-group interaction and group-group interaction. Our hybrid model enables the simulation of more realistic crowd movement with combinations of the three interactions, especially the agent-group interaction which is first introduced in crowd simulation. Finally, we formulate and develop our minimal-energy model to reduce collision energy between dense groups.

There're some limitation in our approach. First, because the formulation of energy model are based on Graph-Cut algorithm, we're using discrete value to represent a continuous interval. We're looking for interpolation methods to provide continuous value for energy model. Secondly, our pipeline is not parallelize implemented which limits our performance. GPU based implementation are needed in our future work.

References

[BK04] Boykov Y., Kolmogorov V.: An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence 26* (2004), 1124–1137. doi:10.1109/TPAMI.2004.60. 4

[BLA02] Bayazit O. B., Lien J.-M., Amato N. M.: Better Group Behaviors in Complex Environments using Global Roadmaps. *Artificial Life* (2002). 2

[BVZ01] Boykov Y., Veksler O., Zabih R.: Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 23*, 11 (Nov 2001), 1222–1239. doi:10.1109/34.969114. 4

[FH04] Felzenszwalb P. F., Huttenlocher D. P.: Efficient Belief Propagation for Early Vision. In *Computer Vision and Pattern Recognition* (2004), vol. 1, pp. 261–268. doi:10.1109/CVPR.2004.1315041. 1

[FTT99] Funge J., Tu X., Terzopoulos D.: Cognitive modeling: Knowledge, reasoning and planning for intelligent characters. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1999), SIGGRAPH '99, ACM Press/Addison-Wesley Publishing Co., pp. 29–38. URL: http://dx.doi.org/10.1145/311535.311538, doi:10.1145/311535.311538. 2

[GCC*10] Guy S. J., Chhugani J., Curtis S., Dubey P., Lin M., Manocha D.: Pledestrians: A least-effort approach to crowd simulation. In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2010), SCA '10, Eurographics Association, pp. 119–128. URL: http://dl.acm.org/citation.cfm?id=1921427.1921446. 2

[GCS*09] Guy S. J., Chhugani J., Satish N., Lin M. C., Manocha D.: ClearPath: Highly Parallel Collision Avoidance for Multi-Agent Simulation. In *Symposium on Computer Animation* (2009), pp. 177–187. doi:10.1145/1599470.1599494. 1, 2

[GKLM11] Guy S. J., Kim S., Lin M. C., Manocha D.: Simulating heterogeneous crowd behaviors using personality trait theory. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2011), SCA '11, ACM, pp. 43–52. URL: http://doi.acm.org/10.1145/2019406.2019413, doi:10.1145/2019406.2019413. 2

[GLM09] Gayle R., Lin M. C., Manocha D.: Multi-Robot Coordination using Generalized Social Potential Fields. In *International Conference on Robotics and Automation* (2009), pp. 106–113. doi:10.1109/ROBOT.2009.5152765. 2

[GNL13] Golas A., Narain R., Lin M.: Hybrid long-range collision avoidance for crowd simulation. In *Proceedings of the*

ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (New York, NY, USA, 2013), I3D '13, ACM, pp. 29–36. URL: http://doi.acm.org/10.1145/2448196.2448200, doi:10.1145/2448196.2448200. 2, 3, 4

[HBJW05] HELBING D., BUZNA L., JOHANSSON A., WERNER T.: Self-Organized Pedestrian Crowd Dynamics: Experiments, Simulations, and Design Solutions. *Transportation Science 39* (2005), 1–24. doi:10.1287/trsc.1040.0108. 2

[HC04] HONG L., CHEN G.: Segment-Based Stereo Matching Using Graph Cuts. In *Computer Vision and Pattern Recognition* (2004), vol. 1, pp. 74–81. doi:10.1109/CVPR.2004.1315016.

[HM95] HELBING D., MOLNÃĄR P.: Social force model for pedestrian dynamics. *Physical Review E 51* (1995), 4282–4286. doi:10.1103/PhysRevE.51.4282. 2

[Hug03] HUGHES R. L.: THE FLOW OF HUMAN CROWDS. *Annual Review of Fluid Mechanics 35* (2003), 169–182. doi:10.1146/annurev.fluid.35.101101.161136. 2

[KGML12] KIM S., GUY S. J., MANOCHA D., LIN M. C.: Interactive simulation of dynamic crowd behaviors using general adaptation syndrome theory. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, 2012), I3D '12, ACM, pp. 55–62. URL: http://doi.acm.org/10.1145/2159616.2159626, doi:10.1145/2159616.2159626. 2

[KZ04] KOLMOGOROV V., ZABIH R.: What energy functions can be minimized via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 26*, 2 (Feb 2004), 147–159. 4

[LD04] LAMARCHE F., DONIKIAN S.: Crowd of Virtual Humans: a New Approach for Real Time Navigation in Complex and Structured Environments. *Computer Graphics Forum 23* (2004), 509–518. doi:10.1111/j.1467-8659.2004.00782.x. 2

[MTT97] MUSSE S. R., THALMANN D., TERZOPOULOS D.: A model of human crowd behavior: Group interrelationship and collision detection analysis. *Computer Animation and Simulation* (1997), 39–51. 2

[NGCL09] NARAIN R., GOLAS A., CURTIS S., LIN M. C.: Aggregate dynamics for dense crowd simulation. *ACM Trans. Graph. 28*, 5 (Dec. 2009), 122:1–122:8. URL: http://doi.acm.org/10.1145/1618452.1618468, doi:10.1145/1618452.1618468. 1, 2, 3

[OPOD10] ONDREJ J., PETTRÃL' J., OLIVIER A.-H., DONIKIAN S.: A synthetic-vision based steering approach for crowd simulation. *ACM Transactions on Graphics 29* (2010). doi:10.1145/1833351.1778860. 1, 2

[PAB08] PELECHANO N., ALLBECK J. M., BADLER N. I.: *Virtual Crowds: Methods, Simulation, and Control*, vol. 3. 2008. doi:10.2200/S00123ED1V01Y200808CGR008. 2

[SAC*08] SUD A., ANDERSEN E., CURTIS S., LIN M., MANOCHA D.: Real-Time Path Planning in Dynamic Virtual Environments Using Multi-Agent Navigation Graphs. *IEEE Transactions on Visualization and Computer Graphics* (2008). 2

[SGA*07] SUD A., GAYLE R., ANDERSEN E., GUY S., LIN M., MANOCHA D.: Real-time navigation of independent agents using adaptive roadmaps. In *Proceedings of the 2007 ACM Symposium on Virtual Reality Software and Technology* (New York, NY, USA, 2007), VRST '07, ACM, pp. 99–106. URL: http://doi.acm.org/10.1145/1315184.1315201, doi:10.1145/1315184.1315201. 2

[TCP06] TREUILLE A., COOPER S., POPOVIĆ Z.: Continuum crowds. *ACM Trans. Graph. 25*, 3 (July 2006), 1160–1168. URL: http://doi.acm.org/10.1145/1141911.1142008, doi:10.1145/1141911.1142008. 2, 3

[TF03] TAPPEN M. F., FREEMAN W. T.: Comparison of Graph Cuts with Belief Propagation for Stereo, using Identical MRF Parameters. In *International Conference on Computer Vision* (2003), pp. 900–907. doi:10.1109/ICCV.2003.1238444. 1

[vdBGLM09] VAN DEN BERG J. P., GUY S. J., LIN M. C., MANOCHA D.: *Reciprocal N-body Collision Avoidance*. 2009. doi:10.1007/978-3-642-19457-3_1. 2

[vdBLM08] VAN DEN BERG J., LIN M., MANOCHA D.: Reciprocal velocity obstacles for real-time multi-agent navigation. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on* (May 2008), pp. 1928–1935. doi:10.1109/ROBOT.2008.4543489. 2, 5

[vdBSGM11] VAN DEN BERG J. P., SNAPE J., GUY S. J., MANOCHA D.: Reciprocal collision avoidance with acceleration-velocity obstacles. In *International Conference on Robotics and Automation* (2011), pp. 3475–3482. doi:10.1109/ICRA.2011.5980408. 1, 2

[YWY*09] YANG Q., WANG L., YANG R., STEWÃL'NIUS H., NISTÃL'R D.: Stereo Matching with Color-Weighted Correlation, Hierarchical Belief Propagation, and Occlusion Handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence 31* (2009), 492–504. doi:10.1109/TPAMI.2008.99. 1