Position-Based Multi-Agent Dynamics for Real-Time Crowd Simulation

Extended Abstract

Tomer Weiss
University of California, Los Angeles
tweiss@cs.ucla.edu

Alan Litteneker
University of California, Los Angeles
alitteneker@cs.ucla.edu

Chenfanfu Jiang
University of Pennsylvania
cffjiang@seas.upenn.edu

Demetri Terzopoulos
University of California, Los Angeles
dt@cs.ucla.edu

ABSTRACT

Exploiting the efficiency and stability of Position-Based Dynamics (PBD), we introduce a novel crowd simulation method that runs at interactive rates for hundreds of thousands of agents. Our method enables the detailed modeling of per-agent behavior in a Lagrangian formulation. We model short-range and long-range collision avoidance constraints to simulate both sparse and dense crowds. The local short-range interaction is represented with collision and frictional contact between agents, as in the discrete simulation of granular materials. We incorporate a cohesion model for modeling collective behaviors and propose a new constraint for dealing with potential future collisions. Our new real-time crowd simulation method is suitable for use in interactive games.

CCS CONCEPTS

• **Computing methodologies → Animation**; *Real-time simulation*;

KEYWORDS

position-based dynamics, crowd simulation, collision avoidance

ACM Reference format:

Tomer Weiss, Alan Litteneker, Chenfanfu Jiang, and Demetri Terzopoulos. 2017. Position-Based Multi-Agent Dynamics for Real-Time Crowd Simulation. In *Proceedings of SCA '17, Los Angeles, CA, USA, July 28-30, 2017,* 2 pages.
DOI: 10.1145/3099564.3108160

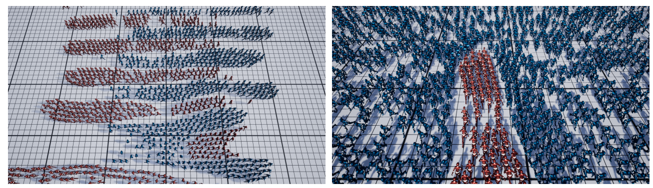

Figure 1: Our fast, robust, stable, and easily implemented method, ideally for use in games, simulates both sparse and dense groups of agents at interactive rates.

1 INTRODUCTION

Crowd simulation is useful in visual effects, animations, and games. Efficiently simulating the motions of numerous agents with realistic interactions among them has been a major focus of research in recent decades [Thalmann 2007]. Among various modeling considerations, collision avoidance remains challenging and time consuming. Collision avoidance algorithms can be classified into discrete and continuum approaches [2013]. Continuum approaches, such as the technique proposed by Narain et al. [2009], have proven efficient for large-scale dense crowds, but are less suitable for sparse crowds. Force-based discrete approaches, such as the recently proposed power-law model [2014], are well suited for sparse crowds, but can be computational expensive and may require smaller time steps due to explicit time integration.

We employ Position-Based Dynamics (PBD) [Müller et al. 2007; Stam 2009], as an alternative discrete algorithm for simulating both dense and sparse crowds. While more carefully designed models, such as the social force model [Helbing and Molnar 1995] and the power law model [Karamouzas et al. 2014], can yield realistic crowd behaviors, they occasionally require elaborate numerical treatments to remain stable and robust. Given the success of PBD in simulating various solid and fluid materials in real-time physics, our work further extends the idea to crowd simulation.

We adopt the PBD framework since it is a real-time, unconditionally stable, implicit scheme. To deal with anticipatory agent contact, we introduce novel long-range collision avoidance constraints. Additionally, to approximate collective group behavior, we adopt PBD constraints used in modeling granular material and fluids. Due to the flexibility of PBD in defining positional constraints among particles, our proposed framework provides a new platform for artistic design and control of agent behaviors in crowd modeling and animation.

Relative to the artificial life approach [Shao and Terzopoulos 2007] in the broader context of multi-human simulation, our approach is posited toward the opposite end of the complexity/fidelity spectrum. The benefit of our work is that it offers a numerical framework for crowd simulation ideally for use in interactive games, which is fast, robust, stable, and easy to implement.

DOI: 10.1145/3099564.3108160

2 METHOD

Our position-based formulation includes several modifications to the standard PBD scheme as well as additional constraints for short-range and long-range collision avoidance between agents. Orthogonal to our constraint-based scheme, higher-level agent behaviors result from roadmap velocity planning at the agent level.

Desired Velocity. In agent locomotion, it is desirable to include the inertia effect before predicting an agent's desired velocity. Denoting the preferred velocity given the planner with $\boldsymbol{v}_i^p$, we calculate the agent velocity $\boldsymbol{v}_i^b$ as a linear blending between $\boldsymbol{v}_i^p$ and the current velocity $\boldsymbol{v}_i^n$, as follows:

$$\boldsymbol{v}_i^b = (1-\alpha)\boldsymbol{v}_i^n + \alpha\boldsymbol{v}_i^p, \tag{1}$$

where $\alpha \in [0,1]$. We set $\alpha = 0.0385$ in all our simulations.

Frictional Contact and Cohesion. We model local particle contacts with an inequality distance constraint as in standard position-based methods:

$$C(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\| - (r_i + r_j) \geq 0, \tag{2}$$

where r_i and r_j are the radii of agents i and j. To model frictional behavior between neighboring agents, we further adopt kinematic frictions as described in [Macklin et al. 2014].

Coherence. To encourage coherent agent motions, we add the artificial XSPH viscosity [Macklin and Müller 2013] to the updated agent velocities. For our simulations, with particles with radius 1, we use $h = 7$ and $c = 217$.

Long Range Collision. Karamouzas et al. [2014] describe an explicit force-based scheme for modeling crowds. Similarly to their power law setting, the leading term is the time to collision τ, defined as the time when two discs representing particles i and j touch each other in the future (see [Karamouzas et al. 2014] for more details). We estimate a future collision state between i and j using τ, with $\hat{\tau} = \Delta t * \lfloor \tau/\Delta t \rfloor$, where $\lfloor \cdot \rfloor$ denotes the floor operator. This is simply clamping τ to find a discrete time spot slightly before the predicted contact. With $\hat{\tau}$, we have

$$\hat{\boldsymbol{x}}_{i,j} = \boldsymbol{x}_{i,j}^n + \hat{\tau}\boldsymbol{v}_{i,j}. \tag{3}$$

We define the colliding positions with

$$\boldsymbol{x}_{i,j}^{\tilde{}} = \boldsymbol{x}_{i,j}^n + \tilde{\tau}\boldsymbol{v}_{i,j}, \tag{4}$$

where $\tilde{\tau} = \Delta t + \hat{\tau}$. We enforce a collision free constraint on $\tilde{\boldsymbol{x}_i}$ and $\tilde{\boldsymbol{x}_j}$. To prevent over-stiff behaviors, we define the stiffness to be $ke^{-\hat{\tau}^2/\tau_0}$, where k is a user-specified constant.

Sliding Model. The total relative displacement is

$$\boldsymbol{d} = (\tilde{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_i) - (\tilde{\boldsymbol{x}}_j - \hat{\boldsymbol{x}}_j), \tag{5}$$

which can be decomposed into contact normal and tangential components as follows:

$$\boldsymbol{d_n} = (\boldsymbol{d} \cdot \boldsymbol{n})\boldsymbol{n}, \qquad \boldsymbol{d}_{\mathrm{t}} = \boldsymbol{d} - \boldsymbol{d_n}, \tag{6}$$

where $\boldsymbol{n} = \frac{\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{x}}_j}{\|\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{x}}_j\|}$ is the contact normal. The long-range collision model will cause agents to slow down due to motion along the contact normal from the collision resolve, which is often undesirable in dense scenarios (Fig. 1). Hence, we preserve only the tangential component in the positional correction to $\boldsymbol{x}_{i,j}^*$. This results in a sliding behavior in response to the predicted collision, which prevents agents from being pushed back into a dense flow.

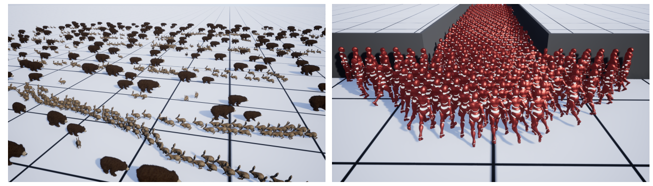

Figure 2: Bears and rabbits (left) and Bottleneck (right) scenarios.

Acceleration Limiting. After the constraint solve, we further clamp the maximum speed of the agents for a more smooth motion.

3 RESULTS

We implemented our framework in CUDA on an NVIDIA GeForce GT 750M, with $\Delta t = 1/48$ sec for all experiments (2 substeps per frame). We solve constraints in parallel, employing a Jacobi solver with a delta averaging coefficient of 1.2 (see [Macklin et al. 2014] for additional details). For all simulations, we use 1 stability iteration to resolve possible remaining contact constraints from the previous time step, and 6 iterations for the constraint solve loop.

We demonstrate the robustness of our position-based framework in a variety of scenarios (Figs. 1, 2). Sparse and dense passing scenarios demonstrate two groups of agents locomoting in opposite directions, passing each other. In Fig. 2, the bears and rabbits demonstration showcases how a Lagrangian PBD scheme may be employed to model agents of different sizes, whereas in the bottleneck demonstration, a multitude of agents must pass through a narrow corridor to reach their goal.

4 CONCLUSION

We adapted Position-Based Dynamics (PBD) as an alternative discrete algorithm for simulating multi-agent dynamics. Our novel method enabled demonstrations of interesting group interactions, such as groups passing each other seamlessly, as well as the formation of traffic lanes and subgroups with minimal interference. We also demonstrated our method on groups of agents of various sizes, densities, and target locomotion goals.

REFERENCES

Abhinav Golas, Rahul Narain, and Ming Lin. 2013. Hybrid long-range collision avoidance for crowd simulation. In *Sym Inter 3D Graph Games (I3D '13)*. 29–36.

Dirk Helbing and Peter Molnar. 1995. Social force model for pedestrian dynamics. *Phys Rev E* 51, 5 (1995), 4282.

Ioannis Karamouzas, Brian Skinner, and Stephen J. Guy. 2014. Universal power law governing pedestrian interactions. *Phys Rev Lett* 113 (2014), 238701.

Miles Macklin and Matthias Müller. 2013. Position based fluids. *ACM Trans Graph* 32, 4 (2013), 104:1–104:12.

M. Macklin, M. Müller, N. Chentanez, and T. Kim. 2014. Unified particle physics for real-time applications. *ACM Trans Graph* 33, 4 (2014), 153:1–153:12.

Matthias Müller, Bruno Heidelberger, Marcus Hennix, and John Ratcliff. 2007. Position based dynamics. *J Vis Comm Imag Repre* 18, 2 (2007), 109–118.

Rahul Narain, Abhinav Golas, Sean Curtis, and Ming C. Lin. 2009. Aggregate dynamics for dense crowd simulation. *ACM Trans Graph* 28, 5 (2009), 122:1–122:8.

Wei Shao and Demetri Terzopoulos. 2007. Autonomous pedestrians. *Graphical Models* 69, 5-6 (2007), 246–274.

Jos Stam. 2009. Nucleus: Towards a unified dynamics solver for computer graphics. In *IEEE Conf Comp-Aided Design and Comp Graph*. 1–11.

Daniel Thalmann. 2007. *Crowd Simulation.* Wiley Online Library.