

Behavioral Animation in Crowd Simulation



Ying Wei

Goal

□ Realistic movement:

- Emergence of crowd behaviors consistent with real-observed crowds
- Collision avoidance and response
- Perception, navigation, learning, communicating, decision-making
- Full-body motion synthesis
- Adapt to dynamically changing virtual environment
- Real-time



□ Crowd Simulation methods:

- Social force model: [HFV00][HMFB01]
 - Apply repulsion and tangential forces to simulate interactions
 - Vibrate unnaturally in high-density crowds
- Rule-based agent model: [Rey87]
 - Used with cognitive models for more realism
- Cellular automata model: [KS02]
 - Fast and simple to implement
 - Tend to expose the underlying checkerboard pattern with high density population
- Flowing continuum model: [Hug03][TCP06]

-
- Group Behavior from Video: A Data-Driven Approach to Crowd Simulation
 - Controlling Individual Agents in High-Density Crowd Simulation

Group Behavior from Video: A Data-Driven Approach to Crowd Simulation



Kang Hoon Lee, Myung Geol
Choi, Qyoun Hong and Jehee
Lee

Seoul National University

Overview

- ❑ A data-driven method simulating crowd behavior that imitates real crowds
- ❑ Capture crowd videos
- ❑ Semi-automated tracker and keyframe-based interactive interface for 2D trajectory extraction
- ❑ Learn low-level action model from observed trajectories based on locally weighted linear regression [AMS97]
- ❑ Reproduce desired group behavior

Related Work

- Track multiple moving objects in videos
 - [CRM03], [ZN04]
 - [KBD05], [EBD*05]: MCMC-based tracker
- Reproducing observed movement patterns in a simulated environment
 - [LCF05]: dynamically update group formations from a set of formation patterns
 - [BJ03], [FWTK]: adapt steering models to walking data to improve path planning
 - [Sti00],[Tek02]: microscopic pedestrian model

Video Capture and Processing

- Uncontrolled environment

- Controlled environment:
 - 40 volunteers
 - Pedestrians crossing and streaming through a variety of environment features; lining up; watching performance, wandering, chatting, standing idly, etc.

Pre-Processing Video Data

- Identify environment features
 - Manually annotate environment features on a frame (assume only agents are moving dynamically)
- Track 2D trajectories of individuals
 - Keyframe-based tracking interface: [AHSS04]
 - Kernel-based tracking algorithm
 - User specify initial and final location of each individual: bidirectional tracking and linearly blending
 - Apply to refined time intervals if necessary
- Understanding high-level behaviors
 - Manually annotate trajectories with high-level behaviors associated

State-Action pair

- Actions are decided based on limited perception
- The behavior model is learned from state-action samples $\{(s_i, a_i)\}$
 - State s : environment feature, motion of nearby agents, own motion
 - Action a : 2D vector (speed)

State Vector

- Self speed:
 - (curPos-prevPos)/time interval
- Neighborhood formation:
 - Subdivide neighborhood into 8 radial regions
 - Include the current and previous frame

$$f = \begin{cases} (r-d)^2/r^2, & \text{if } d < r \\ 0, & \text{otherwise} \end{cases}$$

- Pivot
 - At most one pivot each behavior model
 - Local coordinates
- Intended moving direction
 - Average moving direction in a window of past and future frames (ten frames)
- Weighed empirically
- 21-D vector: PCA=>reparameterize to 8-12D for learning



Group Behavior Model

- Each low-level model is a primitive action
- Think of an action model as a function that takes the state of an agent as input and outputs a desired action at next frame
- High-level module controls transition between low-level action models

-
- State-action trajectories divided into groups
 - Each group is an action model
 - Locally weighted linear regression method
 - select a small set of samples similar to the query state (computation-demanding)
 - Divide possible outputs into clusters(k-means clusters)
 - Choose a cluster:
 - m_i : mean of i-th cluster
 - m_{prev} : mean of previous selected cluster
 - K : number of clusters
 - P_i : probability of selecting the i-th cluster

$$p_i = \frac{1 / \|m_i - m_{prev}\|}{\sum_{j=1}^K 1 / \|m_j - m_{prev}\|}$$

Locally Weighted Linear Regression

- Given samples $\{(s_i, \mathbf{a}_i)\}_{i=1}^k$ $\mathbf{a}_i = (x_i, y_i)$
 - regression model for x-coordinates of output vectors a matrix equation $S\beta = \mathbf{x}$, where S is a matrix whose i -th row is s_i^T , \mathbf{x} is a vector whose i -th element is x_i , and β is a vector of the model parameters. Locally weighted linear regression estimates the model parameters

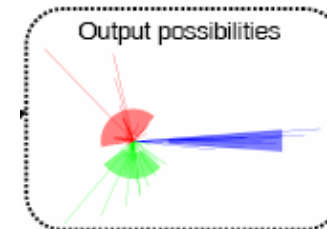
$$\beta = (S^T W S)^{-1} S^T W \mathbf{x}, \quad (3)$$

where W is a diagonal weight matrix with $W_{ii} = \exp\left(-\frac{1}{2\sigma}(s - s_i)^T (s - s_i)\right)$. This regression weighs near samples more than farther samples. Bandwidth σ determines how weights fall off with distance from s .

- Similarly build the regression model for y-coordinate

-
- If a state is significantly different from any samples in the training data

- Perturb the regression output



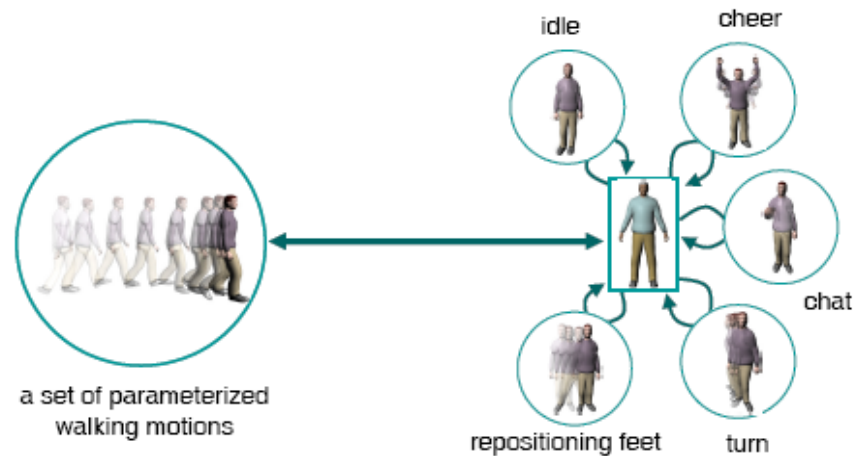
- Radial lines: possible outputs
- Radial windows: allowable perturbations
- Perturbation limited by a constant multiple of standard deviation from mean direction and distance
- Choose the perturbation that minimize the distance to the nearest sample

High-Level Behavior: Crowd Simulation

- High-level module controls transition between low-level action models
 - Finite state machine: each action model corresponds to a state
 - Encode transition and control parameters such as average duration of each action
 - Example: repeated transition between locomotion and group interaction
 - Measure average size of groups, average duration of staying, average interpersonal distance in groups
 - User creates environment layout and annotates appropriate behavior model in each part of the environment

Full-body Motion Synthesis: moCap

- ❑ **Locomotion:** walk in various speeds and turning angles
- ❑ **in-place:** Chat, stand idly, cheer, turn, reposition feet
- ❑ **Synthesize motion along given trajectory**
 - Lookahead (0.66 sec)




Experiment Results


- An hour of video in a 10mx10m region
- About forty volunteers walked in a variety of environment setups
- about 30 min to postprocess 200 frames
- about 20 people each frame
- For each query, find 100 nearest samples and sort into three groups

Discussion

- ❑ Capability of reproducing realistic group behaviors in simulated environments
- ❑ Hard to accomplish by rule-based agent models
- ❑ Need quantitative evaluation of results beside visual comparison

- 
-
- Group Behavior from Video: A Data-Driven Approach to Crowd Simulation
 - Controlling Individual Agents in High-Density Crowd Simulation

Controlling Individual Agents in High-Density Crowd Simulation



N. Pelechano, J.M. Allbeck and
N.I. Badler

University of Pennsylvania, USA



Realistic
Crowd
Animation

Rule-based
Models

HDAC

Unrealistic
Crowd
Animation

Cellular
Automata
Models

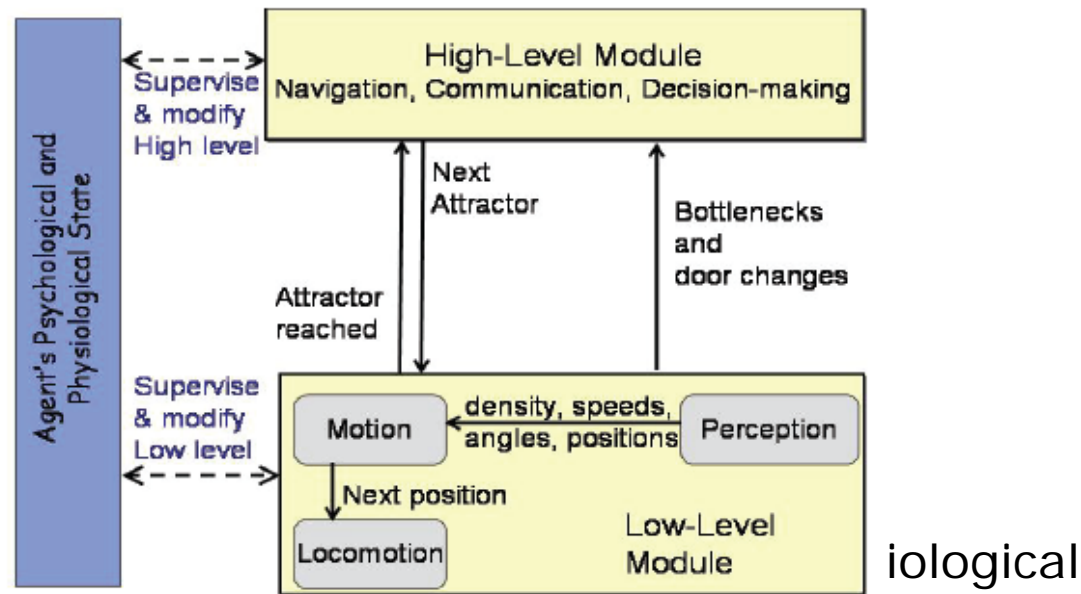
Social
Forces
Models

Handles Low-Med
densities

Handles Low-High
densities

HiDAC Architecture Overview

- Multi-agent system without centralized controller
- Agent behaviors are computed at two levels:



- Both levels are biological
- High-level module determines attractor point. [PB06][POS05]

The HiDAC Model

- A parameterized social force model that depends on psychological and geometrical rules
- The movement of agent i depends on the desired attractor, while avoiding walls w , obstacles k and other agents j and trying to keep its previous direction of movement to avoid abrupt changes in its trajectory

$$\mathbf{F}_i^{To}[n] = \mathbf{F}_i^{To}[n-1] + \mathbf{F}_i^{At}[n]w_i^{At} + \sum_w \mathbf{F}_{wi}^{Wa}[n]w_i^{Wa} + \\ + \sum_k \mathbf{F}_{ki}^{Ob}[n]w_i^{Ob} + \sum_{j(\neq i)} \mathbf{F}_{ji}^{Ot}[n]w_i^{Ot}$$

- All these forces are summed together with different weights determined by psychological and/or geometrical rules
- The force vector is therefore:

$$\mathbf{f}_i^{To} = \frac{\mathbf{F}_i^{To}}{|\mathbf{F}_i^{To}|}$$

-
- The new desired position for agent I is :

$$\mathbf{p}_i[n+1] = \mathbf{p}_i[n] + \alpha_i[n] \mathbf{v}_i[n] \left((1 - \beta_i[n]) \mathbf{f}_i^{To}[n] + \beta_i[n] \mathbf{F}_i^{Fa}[n] \right) T + \mathbf{r}_i[n]$$

$$\begin{aligned} \mathbf{F}_i^{To}[n] = & \mathbf{F}_i^{To}[n-1] + \mathbf{F}_i^{At}[n] w_i^{At} + \sum_w \mathbf{F}_{wi}^{Wa}[n] w_i^{Wa} + \\ & + \sum_k \mathbf{F}_{ki}^{Ob}[n] w_i^{Ob} + \sum_{j(\neq i)} \mathbf{F}_{ji}^{Ot}[n] w_i^{Ot} \end{aligned}$$

$$\mathbf{f}_i^{To} = \frac{\mathbf{F}_i^{To}}{|\mathbf{F}_i^{To}|}$$

Avoidance Force

□ Rectangle of influence

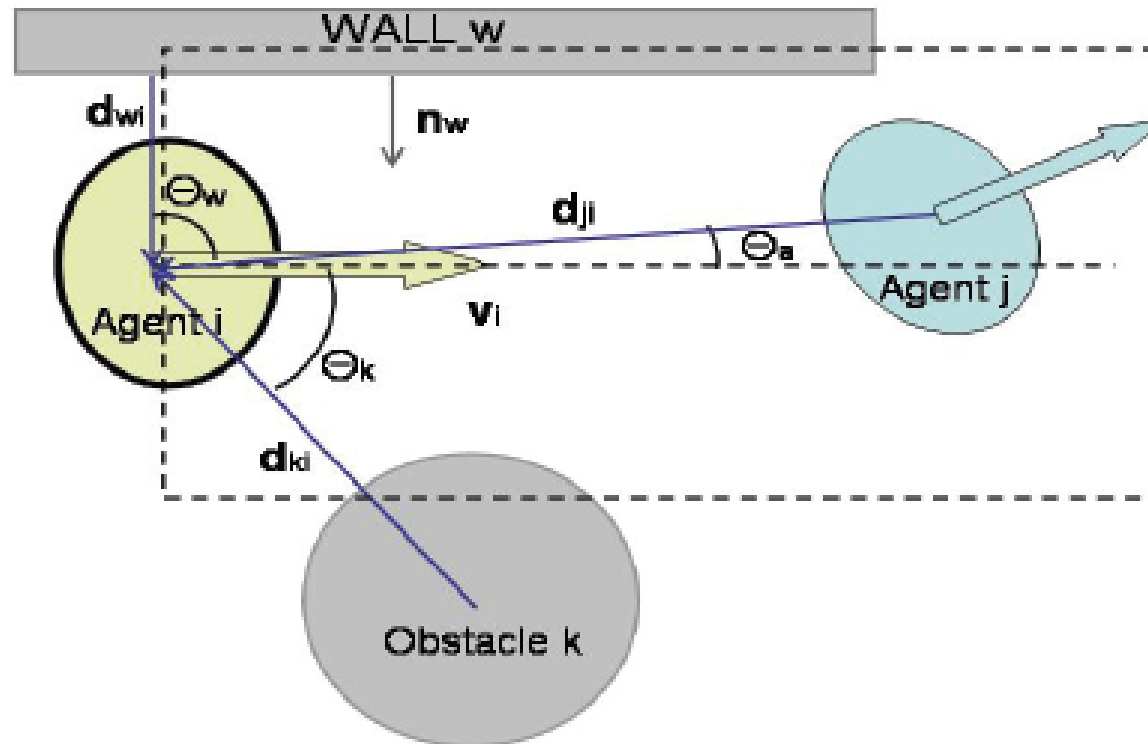


Figure 3: Perception for the yellow agent.

-
- Perception: cell and portal graph
 - Cell: room
 - Cell contains a list of static and dynamic objects
 - Wall and Obstacle Avoidance:

$$\mathbf{F}_{ki}^{Ob} = \frac{(\mathbf{d}_{ki} \times \mathbf{v}_i) \times \mathbf{d}_{ki}}{|(\mathbf{d}_{ki} \times \mathbf{v}_i) \times \mathbf{d}_{ki}|} \quad \mathbf{F}_{wi}^{Wa} = \frac{(\mathbf{n}_w \times \mathbf{v}_i) \times \mathbf{n}_w}{|(\mathbf{n}_w \times \mathbf{v}_i) \times \mathbf{n}_w|}$$

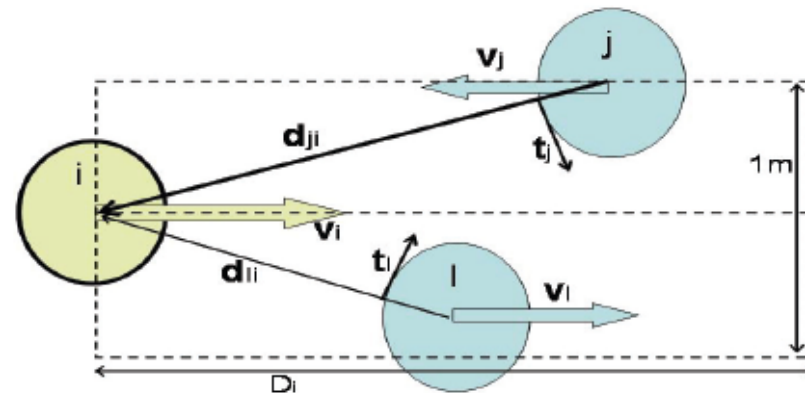
$$\mathbf{F}_i^{To}[n] = \mathbf{F}_i^{To}[n-1] + \mathbf{F}_i^{At}[n]w_i^{At} + \sum_w \mathbf{F}_{wi}^{Wa}[n]w_i^{Wa} + \sum_k \mathbf{F}_{ki}^{Ob}[n]w_i^{Ob} + \sum_{j(\neq i)} \mathbf{F}_{ji}^{Ot}[n]w_i^{Ot}$$

- Other Agent Avoidance
- $D_i = 3\text{m}$ for low density and 1.5m for high density

$$\mathbf{t}_j = \frac{(\mathbf{d}_{ji} \times \mathbf{v}_i) \times \mathbf{d}_{ji}}{|(\mathbf{d}_{ji} \times \mathbf{v}_i) \times \mathbf{d}_{ji}|}$$

$$\mathbf{F}_{ji}^{Ot} = \mathbf{t}_j w_i^d w_i^o$$

$$w_i^d = (d_{ji} - D_i)^2 \quad w_i^o = \begin{cases} 1.2 & \text{if } (\mathbf{v}_i \cdot \mathbf{v}_j) > 0 \\ 2.4 & \text{otherwise} \end{cases}$$



$$\mathbf{F}_i^{To}[n] = \mathbf{F}_i^{To}[n-1] + \mathbf{F}_i^{At}[n]w_i^{At} + \sum_w \mathbf{F}_{wi}^{Wa}[n]w_i^{Wa} + \sum_k \mathbf{F}_{ki}^{Ob}[n]w_i^{Ob} + \sum_{j(\neq i)} \mathbf{F}_{ji}^{Ot}[n]w_i^{Ot}$$

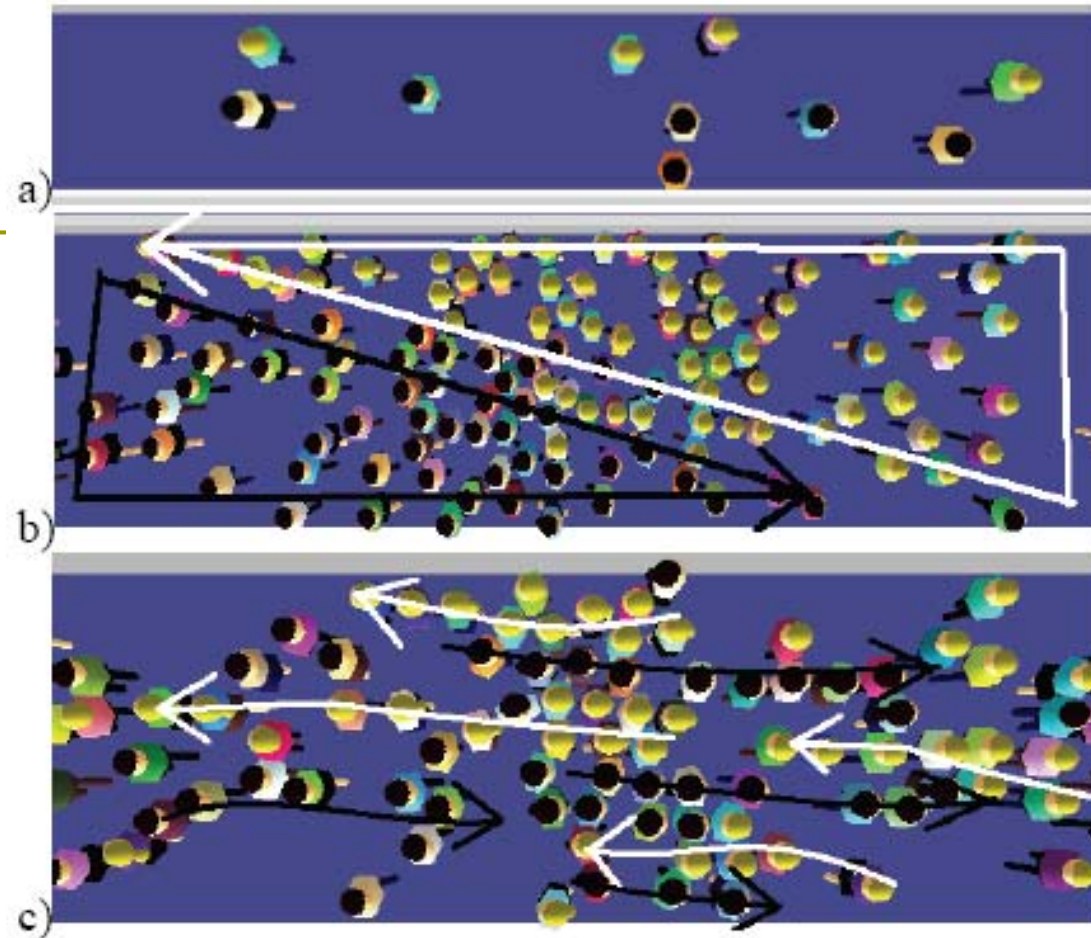


Figure 5: *Bi-directional flows. People with blonde hair walk towards the left, while dark-haired people walk towards the right. (a) low-density flows, (b) high-density without altering the viewing rectangle and right preference, (c) high-density with HiDAC.*

Repulsion Forces

- When overlapping occurs, a collision response force is applied.

$$\mathbf{r}_i[n] = \sum_w \mathbf{F}_{wi}^{R-Wa}[n] + \sum_k \mathbf{F}_{ki}^{R-Ob}[n] + \lambda \sum_{j(\neq i)} \mathbf{F}_{ji}^{R-Ot}[n]$$

$$\mathbf{F}_{wi}^{R-Wa}[n] = \frac{\mathbf{n}_w (r_i + \varepsilon_i - d_{wi}[n])}{d_{wi}[n]}$$

$$\mathbf{F}_{ki}^{R-Ob}[n] = \frac{(\mathbf{p}_i[n] - \mathbf{p}_k[n])(r_i + \varepsilon_i + r_k - d_{ki}[n])}{d_{ki}[n]}$$

$$\mathbf{F}_{ji}^{R-Ot}[n] = \frac{(\mathbf{p}_i[n] - \mathbf{p}_j[n])(r_i + \varepsilon_i + r_j - d_{ji}[n])}{d_{ji}[n]}$$

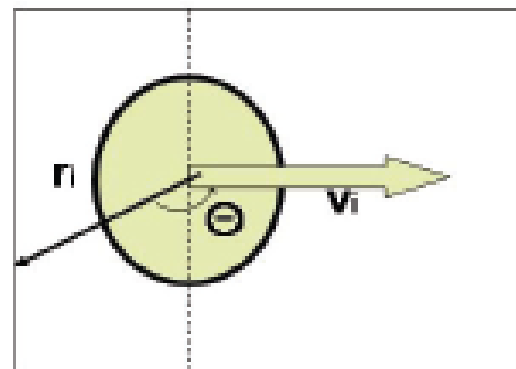
$$\mathbf{F}_i^{To}[n] = \mathbf{F}_i^{To}[n-1] + \mathbf{F}_i^{At}[n]w_i^{At} + \sum_w \mathbf{F}_{wi}^{Wa}[n]w_i^{Wa} + \sum_k \mathbf{F}_{ki}^{Ob}[n]w_i^{Ob} + \sum_{j(\neq i)} \mathbf{F}_{ji}^{Ot}[n]w_i^{Ot}$$

Solution to “shaking” in High-Density

□ “stopping rules”

- Personality
- Direction of other agent
- Current situation (panic/normal)
- Applies when repulsion forces from other agents is against desired direction and the situation is normal
- Set a timer to avoid deadlock
- When StoppingRule is true, α_i is 0.

$$\mathbf{p}_i[n+1] = \mathbf{p}_i[n] + \alpha_i[n] \mathbf{v}_i[n] \left((1 - \beta_i[n]) \mathbf{f}_i^{To}[n] + \beta_i[n] \mathbf{F}_i^{Fa}[n] \right) T + \mathbf{r}_i[n]$$



$$\Theta = \text{angle}(v_i, r)$$

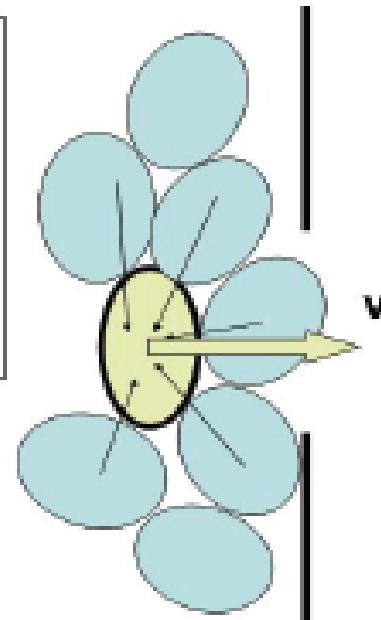
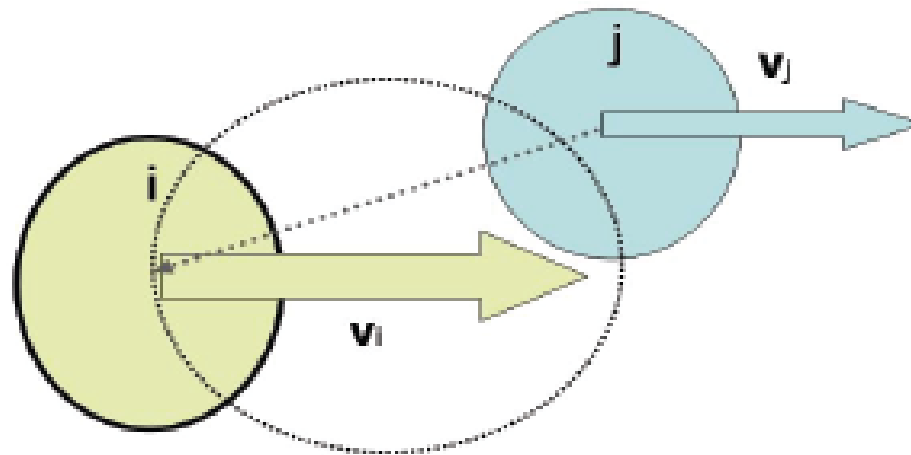


Figure 6: *Example of repulsion forces which are necessary to apply braking forces.*

Organized behavior - queuing

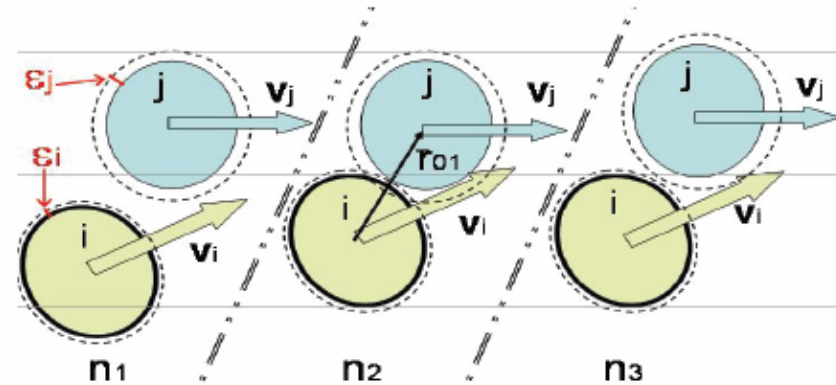
□ Influence disk

- Drive the temporal waiting behavior
- Work similar to stopping rules



7: Area of influence for waiting behaviors.

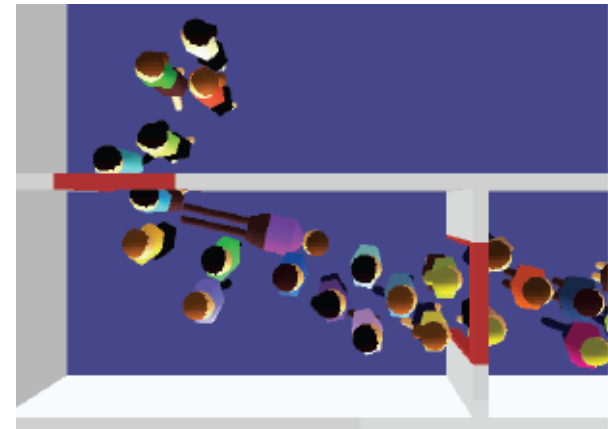
Pushing Behavior



- ❑ Pushing behavior: a type of collision response.
- ❑ During an organized situation, individuals wait for space available before moving; but when in panic, they try to move until they collide with other individuals who impede forward progress.
- ❑ By combining both behaviors simultaneously for a heterogeneous crowd, we observe an emergent behavior where some individuals that do not respect personal space will get very close to other agents and push them away in order to open a path through a dense crowd.
- ❑ The effect of being pushed away is achieved by applying collision response forces and different personal space thresholds (ϵ_i , ϵ_j).

Falling and becoming Obstacles

- ❑ When the majority of pushing forces affecting one individual are approximately in the same direction, the agent will receive a sum of forces with magnitude high enough to make it lose equilibrium.
- ❑ Fallen agents represent a different type of obstacle: it is an obstacle that should be avoided, but if necessary (or unavoidable) can be stepped over.
- ❑ Fallen individuals => a rectangular obstacle
- ❑ When other agents approach this new obstacle, weak tangential forces are applied in order to walk around the fallen agent, but repulsive forces are not applied. Therefore, when the crowd is extremely dense and the pushing forces from behind are strong, the result is that agents may walk over the body on the floor.

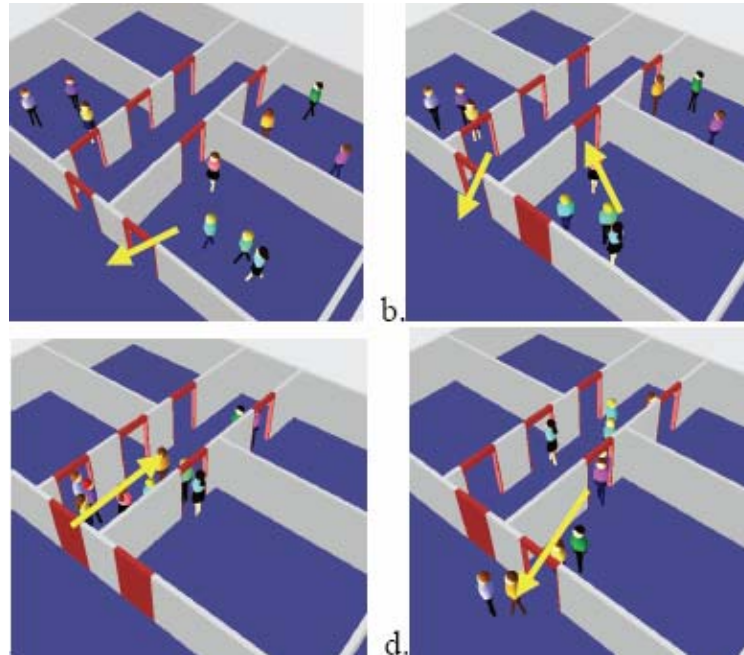
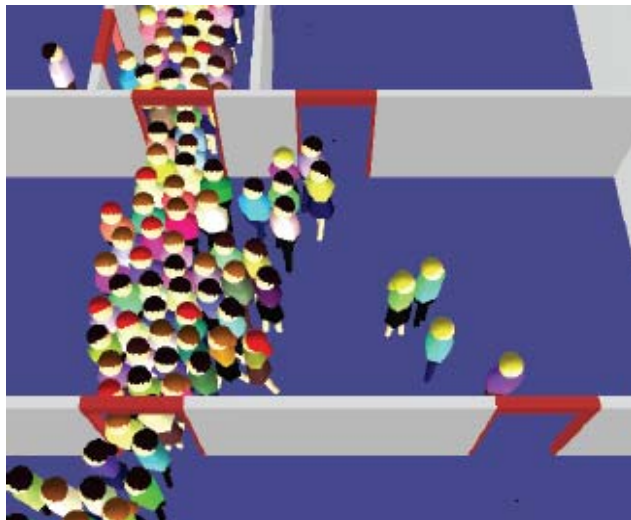


Panic Propagation

- ❑ HiDAC can simulate an emergency evacuation.
- ❑ Agent personality and levels of panic.
- ❑ To propagate panic, we use either communication between agents (managed by the High-level behavior module), or perception to detect relevant changes in low-level behaviors, such as increasing crowd densities and number of people pushing or both.

Avoiding bottlenecks and interactive changes in the environment

- When dealing with high-density crowds in buildings, bottlenecks can appear in the portals.
- Find an alternative path: high-level module
 - Base on its current position (doors, obstacles)
 - knowledge that the agent has about the internal connectivity of the building.



Results

- ❑ 2.99 GHz Intel Xeon with 2GB of RAM measuring frame rates both for simulation only and for simulation and 3D rendering.
- ❑ When doing only simulation, HiDAC can handle up to
- ❑ 1800 agents with a frame rate of 25Hz.
- ❑ Simulation and 3D rendering using an NVIDIA Quadro FX 3400/4400 graphics system can achieve 25 frames/second (not using GPU rendering) for up to 600 simple 3D virtual human figures (“crayon figures”) each with about 100 vertices.
- ❑ For the frame rate tests, we used a large complex environment with 85 rooms and 53,448 vertices overall.

Conclusion

- ❑ Novel extensions to social forces models by adding stopping rules and influence region controls.
- ❑ Uses the best features of both rule-based and social forces systems, while eliminating their disadvantages.
- ❑ The implementation allows real-time simulations for hundreds of individualized agents.
- ❑ Future work:
 - Map low level parameters to agent properties that are much more intuitive
 - Add in agent actions other than locomotion.