Modelling Crowd Behaviour in the Polymensa using the Social Force Model

M. Vifian, M. Roggo, M. Aebli

GESS Presentations, December 2011

Outline

Introduction and Motivation
- Introduction, Motivation and Fundamental Questions
- Fundamental Questions

Description of Model
- Sources
- Description

Implementation

Simulation and Discussion

Summary and Outlook

Outline

Introduction and Motivation
- Introduction, Motivation and Fundamental Questions
- Fundamental Questions

Description of Model
- Sources
- Description

Implementation

Simulation and Discussion

Summary and Outlook

General Introduction and Motivation

- Mensa seems suboptimal.
- General motivation: mensa dynamics predictable?
- Matlab simulation
- Goal: Simulation with close resemblance to actual observed crowd behaviour.

General Introduction and Motivation

- Mensa seems suboptimal.
- General motivation: mensa dynamics predictable?
- Matlab simulation
- Goal: Simulation with close resemblance to actual observed crowd behaviour.

General Introduction and Motivation

- Mensa seems suboptimal.
- General motivation: mensa dynamics predictable?
- Matlab simulation
- Goal: Simulation with close resemblance to actual observed crowd behaviour.

General Introduction and Motivation

- Mensa seems suboptimal.
- General motivation: mensa dynamics predictable?
- Matlab simulation
- Goal: Simulation with close resemblance to actual observed crowd behaviour.

Outline

Introduction and Motivation
- Introduction, Motivation and Fundamental Questions
- Fundamental Questions

Description of Model
- Sources
- Description

Implementation

Simulation and Discussion

Summary and Outlook

Fundamental Questions.

- How do pedestrian groups with different destinations interact with one another?
- How do pedestrian groups interact when all pedestrians return to a cash register once they have obtained their meals?
- How accurately are pedestrian-dynamic models able to depict the empirically collected data?

Fundamental Questions.

- How do pedestrian groups with different destinations interact with one another?
- How do pedestrian groups interact when all pedestrians return to a cash register once they have obtained their meals?
- How accurately are pedestrian-dynamic models able to depict the empirically collected data?

Fundamental Questions.

- How do pedestrian groups with different destinations interact with one another?
- How do pedestrian groups interact when all pedestrians return to a cash register once they have obtained their meals?
- How accurately are pedestrian-dynamic models able to depict the empirically collected data?

Fundamental Questions.

- How do pedestrian groups with different destinations interact with one another?
- How do pedestrian groups interact when all pedestrians return to a cash register once they have obtained their meals?
- How accurately are pedestrian-dynamic models able to depict the empirically collected data?

Variables of Interests

- Time it takes for agents to enter and exit mensa.
- The number of times agent paths are blocked by other agents.
- Queue locations.

Variables of Interests

- Time it takes for agents to enter and exit mensa.
- The number of times agent paths are blocked by other agents.
- Queue locations.

Variables of Interests

- Time it takes for agents to enter and exit mensa.
- The number of times agent paths are blocked by other agents.
- Queue locations.

Outline

Introduction and Motivation
- Introduction, Motivation and Fundamental Questions
- Fundamental Questions

Description of Model
- Sources
- Description

Implementation

Simulation and Discussion

Summary and Outlook

Social Force Model Sources

- **Social Force Model for Pedestrian dynamics (1995)** – Dirk Helbing et al. First publication of Social Force Model.
- **Self-organized Pedestrian Crowd Dynamics (2005)** – Helbing et al. Extensions to Social Force Model.

Social Force Model Sources

- **Social Force Model for Pedestrian dynamics (1995)** – Dirk Helbing et al. First publication of Social Force Model.
- **Self-organized Pedestrian Crowd Dynamics (2005)** – Helbing et al. Extensions to Social Force Model.

Outline

Introduction and Motivation
- Introduction, Motivation and Fundamental Questions
- Fundamental Questions

Description of Model
- Sources
- Description

Implementation

Simulation and Discussion

Summary and Outlook

General

- All pedestrian interactions reduced to forces.
- Forces
  1. Destination
  2. Other Agents
  3. Boarders
  4. Objects of Attraction

General

- All pedestrian interactions reduced to forces.
- Forces
  1. Destination
  2. Other Agents
  3. Boarders
  4. Objects of Attraction

Destination Force

Unit Vector towards destination:

- $\vec{e}_\alpha(t) := \frac{\vec{r}^k_\alpha - \vec{r}_\alpha(t)}{||\vec{r}^k_\alpha - \vec{r}_\alpha(t)||}$

Force towards destination:

- $\vec{F}^0_\alpha(\vec{v}_\alpha, v^0_\alpha, \vec{e}_\alpha(t)) := \frac{1}{\tau_\alpha}(v^0_\alpha \vec{e}_\alpha - \vec{v}_\alpha)$

Destination Force

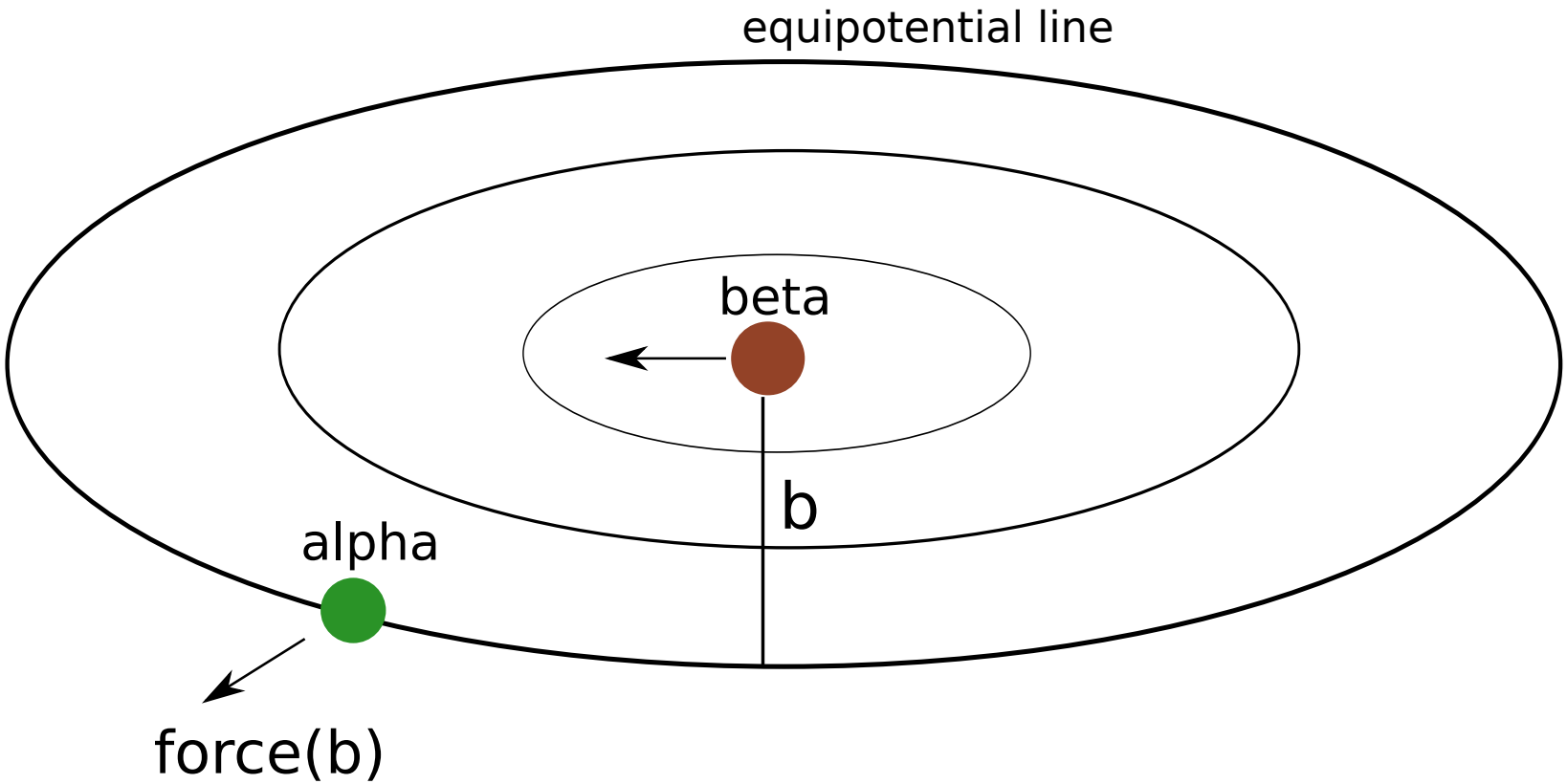
Unit Vector towards destination:

- $\vec{e}_\alpha(t) := \frac{\vec{r}^k_\alpha - \vec{r}_\alpha(t)}{||\vec{r}^k_\alpha - \vec{r}_\alpha(t)||}$

Force towards destination:

- $\vec{F}^0_\alpha(\vec{v}_\alpha, \vec{v}^0_\alpha, \vec{e}_\alpha(t)) := \frac{1}{\tau_\alpha}(v^0_\alpha \vec{e}_\alpha - \vec{v}_\alpha)$

Other Agents Force



Other Agents Force (alternative)

according to more recent Social Force model:

$$\begin{aligned}\vec{F}_{\alpha\beta}(t) = & A^1_\alpha \exp\left[(r_{\alpha\beta} - d_{\alpha\beta})/B^1_\alpha\right]\vec{n}_{\alpha\beta} \\ & \cdot\left(\lambda_\alpha + (1-\lambda_\alpha)\frac{1+\cos(\varphi_{\alpha\beta})}{2}\right) \\ & + A^2_\alpha \exp\left[(r_{\alpha\beta} - d_{\alpha\beta})/B^2_\alpha\right]\vec{n}_{\alpha\beta}\end{aligned}$$

Boundary Forces

Boundary Forces:

- boundry Repulsion Force $\vec{F}_{\alpha B} = -\nabla_{\vec{r_{\alpha B}}} U_{0\alpha B} e^{-||\vec{r}_{\alpha B}||/R}$

Total Forces

- **Note:** attractive object forces calculated in similar fashion.

Superposition of all Forces:

- $\vec{F}_\alpha(t) = \vec{F}^0_\alpha(\vec{v}_\alpha \vec{v}^0_\alpha \vec{e}_\alpha) + \sum_\beta \vec{F}_{\alpha\beta}(\vec{e}_\alpha, \vec{r}_\alpha - \vec{r}_\beta) + \sum_B \vec{F}_{\alpha B}(\vec{e}_\alpha, \vec{r}_\alpha - \vec{r}^\alpha_B) + \sum_i \vec{F}_{\alpha i}(\vec{e}_\alpha, \vec{r}_\alpha - \vec{r}_i, t)$

Using the force

Putting it all together

- $\frac{d\vec{w}_\alpha}{dt} := \vec{F}_\alpha(t) + \textit{fluctuations}$
- Actual velocity $\vec{v}_\alpha = \vec{w}_\alpha \cdot \begin{cases} 1 & , \text{ if } ||\vec{w}_\alpha|| < v_0^\alpha \\ \frac{v_o^\alpha}{||\vec{w}_\alpha||} & , \text{ otherwise} \end{cases}$

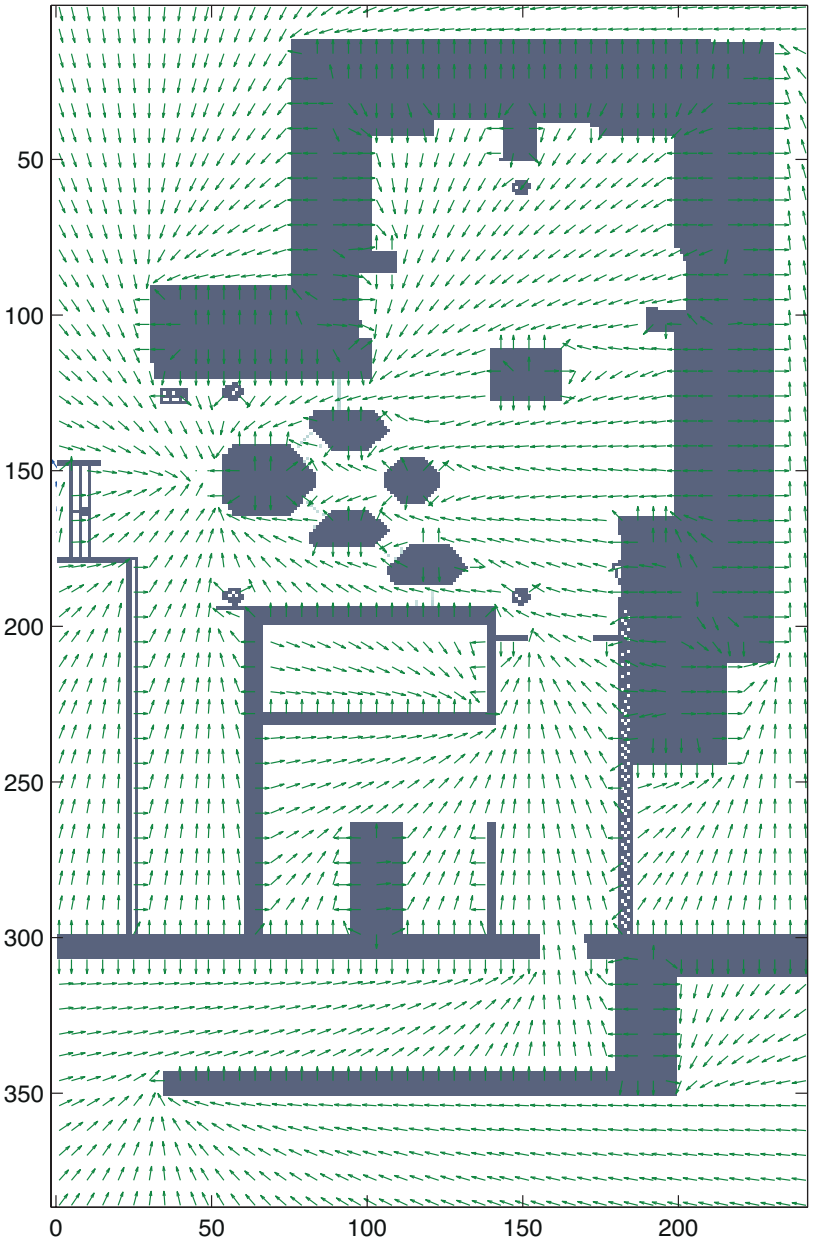
Structure of our Program

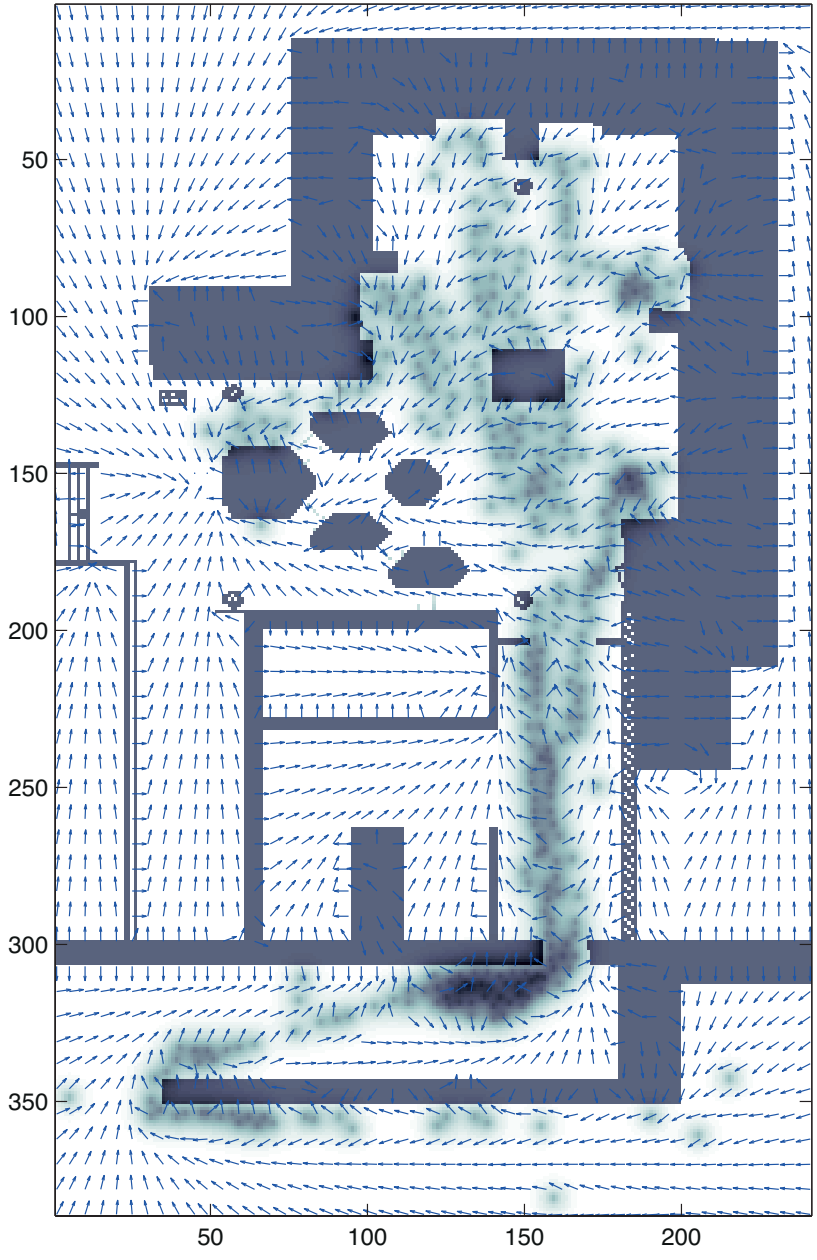
- No objects
  - Objects need loops
  - Matrices are faster (usually)
- All agents stored inside one matrix
  - ```
    % | Agent 1 | Agent 2 | Agent 3
    %-----------------------------------------
    % 1 position x |
    % 2 position y |
    % 3 speed x |
    % 4 speed y |
    % 5 desired speed v0|
    % 6 goal |
    % 7 last counter
    % 8 ETR from FM
    % 9 Red Carpet...
    ```
- Rather scripts than functions
  - Functions: pass by value
  - Global Variables: pass by “reference”
    ```

Simulation: Euler Method

- Calculate forces
  - `agents_f(:,agentID) = agents_force(A,agentID);`
    `agents_p(:,agentID) =`
    `potential_force(round(A(1,agentID)),round(A(2,agentID)),`
- Resulting Velocities
  - `A(3:4,agentID) = (agents_p(:,agentID)...`
    `*agents_f(:,agentID)...`
    `+10 *[(rand(1)-.5);(rand(1)-.5)])...`
    `*timestep;`
- New positions: $\vec{x_{n+1}} = \vec{x_n} + \vec{v} \cdot \Delta t$
  - `deltaPos=A(3:4,:)*timestep;`
    `A(1:2,:)=A(1:2,:)+deltaPos;`

Desired Direction





Agent Forces

- Distances to other Agents: A single matrix multiplication
  - `distances=A(1:2,:)-A(1:2,alpha)*ones(1,size(A,2))`
- Neglect agents outside 1 meter
  - `close_agents = sqrt(sum(r_alphabeta_matrix.^2))<sight;`
  - `agent_others = A(:,close_agents);`
- Calculate resulting interaction forces
  - `F_agents = A2*sum(...`
    `(ones(2,1)...`
    `*exp((2+tray_factor*(agent_alpha(6)==1)...`
    `*sigma*ones(1,agent_number_back-1)-sum(r_alphabeta_matri`
    `+sigma*tray_factor*(agent_others(6,:)==1))/B2)...`
    `).*e_beta_matrix,2);`

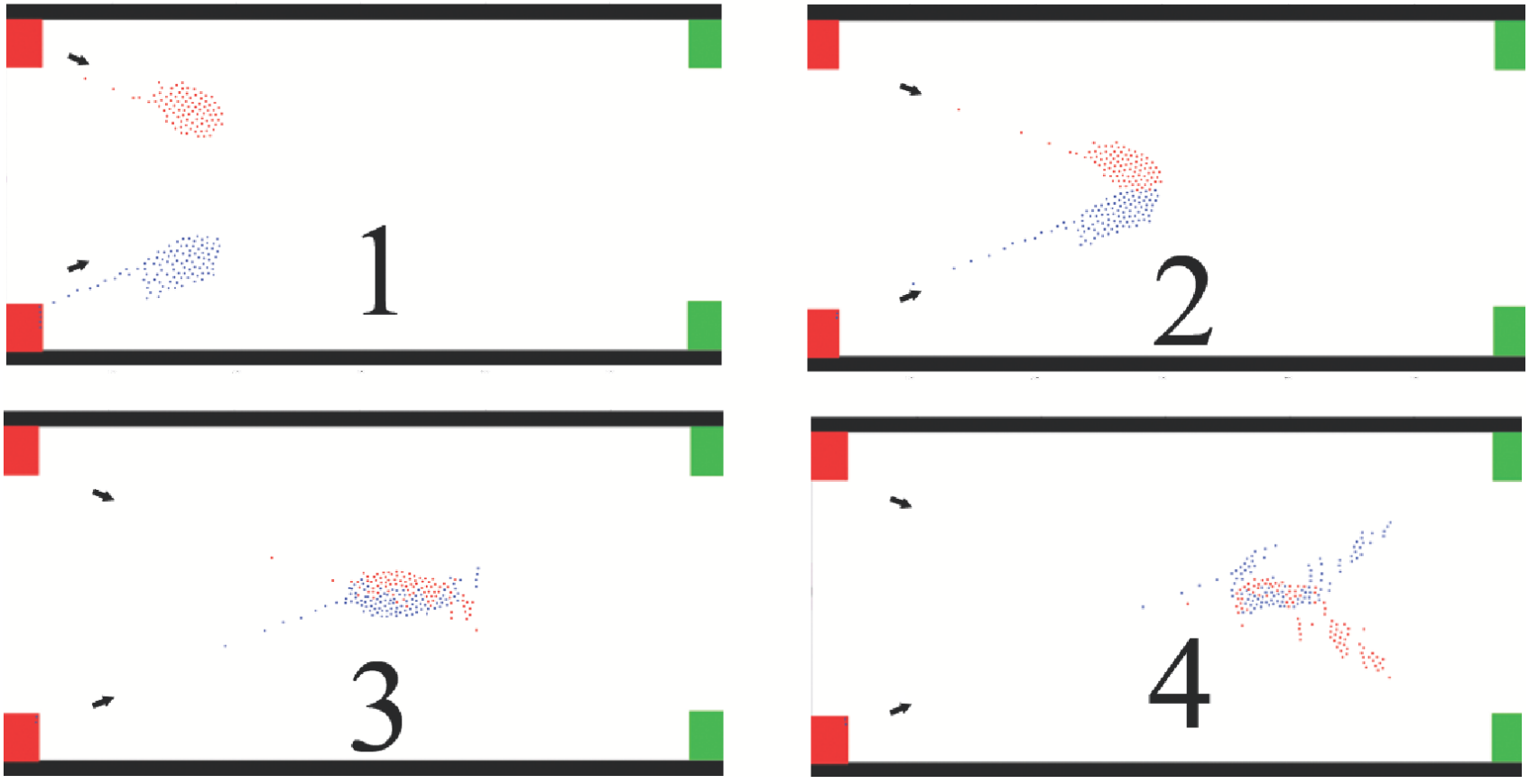
Simple Queueing

- Algorithm
  - Look around. Who is headed towards the same goal?
  - Who ist last in line?
  - Join him/her
- Implementation
  - Fast Marching offers the expected time until reaching a certain goal
  - Take those closer to the goal than you (less time expected)
  - Of those take the last (max time)

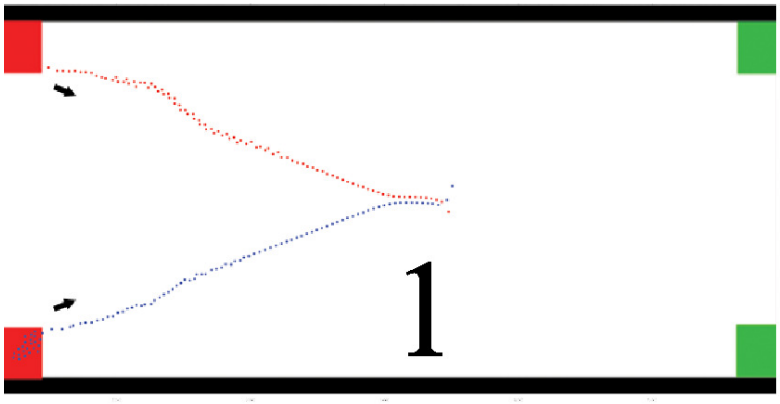
Testing Heuristics

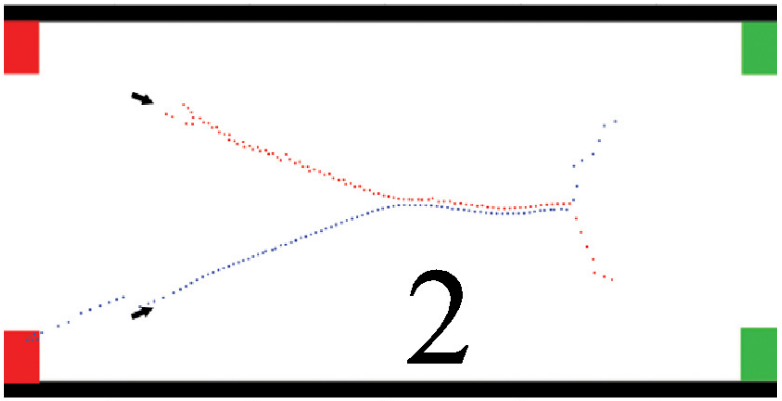
- Basic social force model
- additional Queuing
- additional real-time-Path-planning

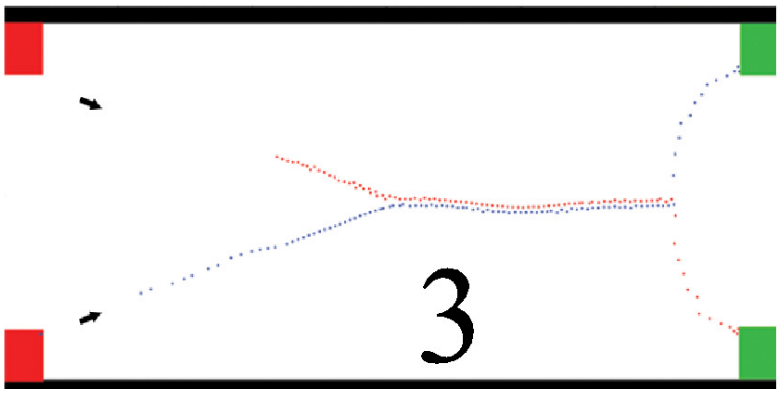
Basic Social Force Model

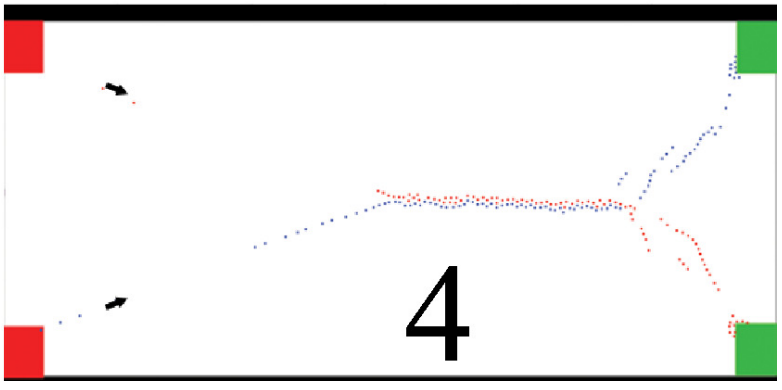


Queuing

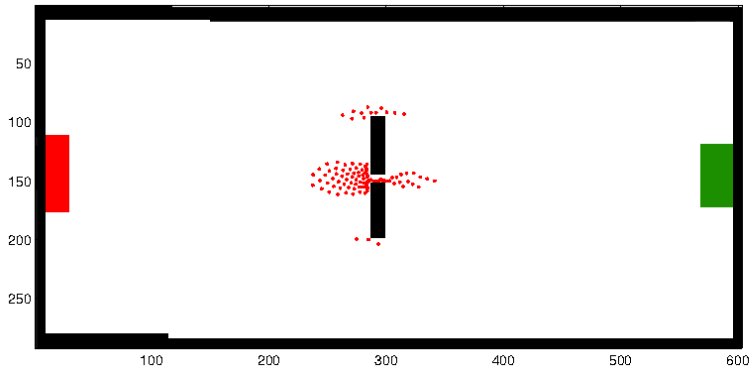


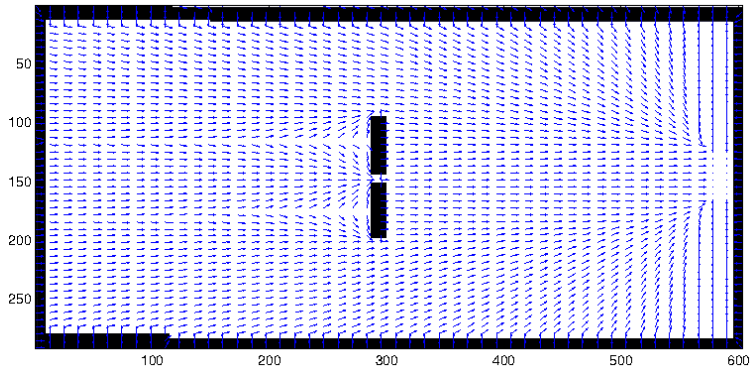


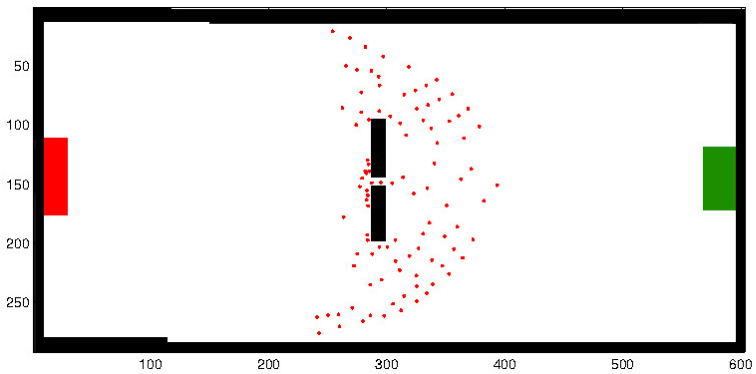


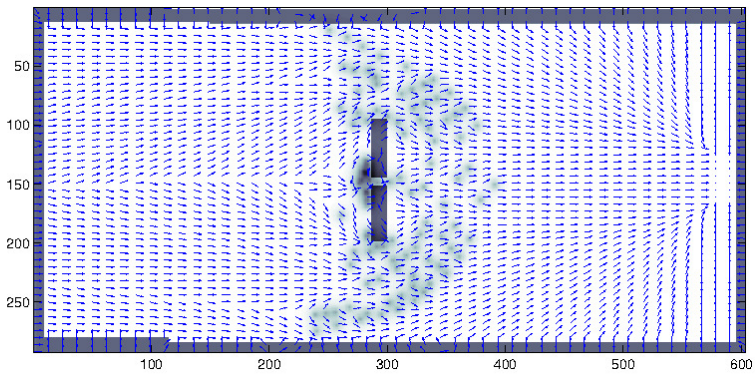


Real-time Path Planning



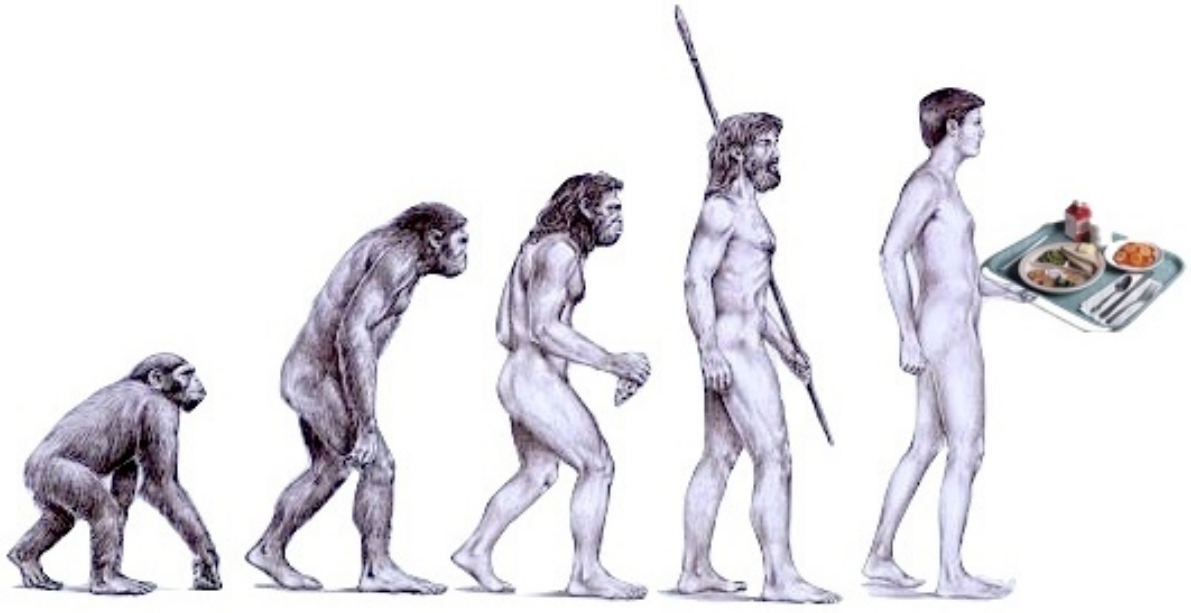


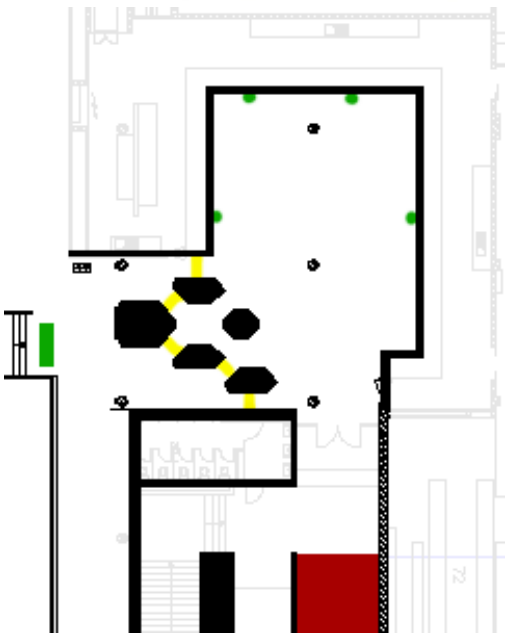


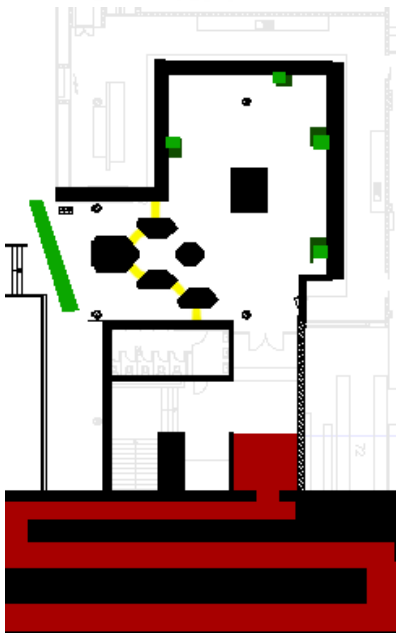


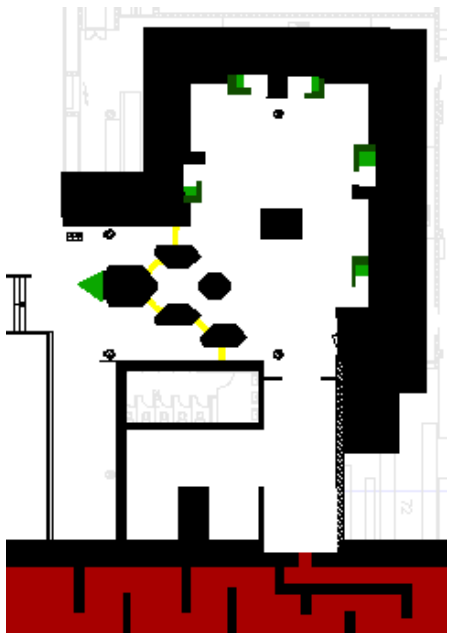
- Path planning takes 250 ms

The Real Mensa

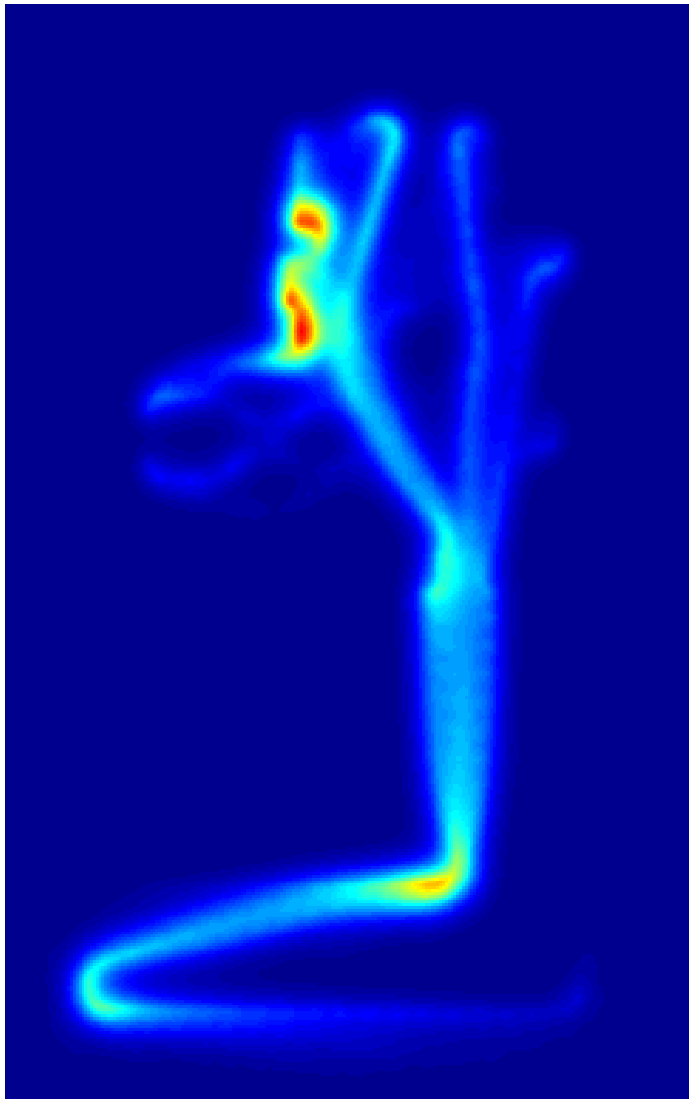


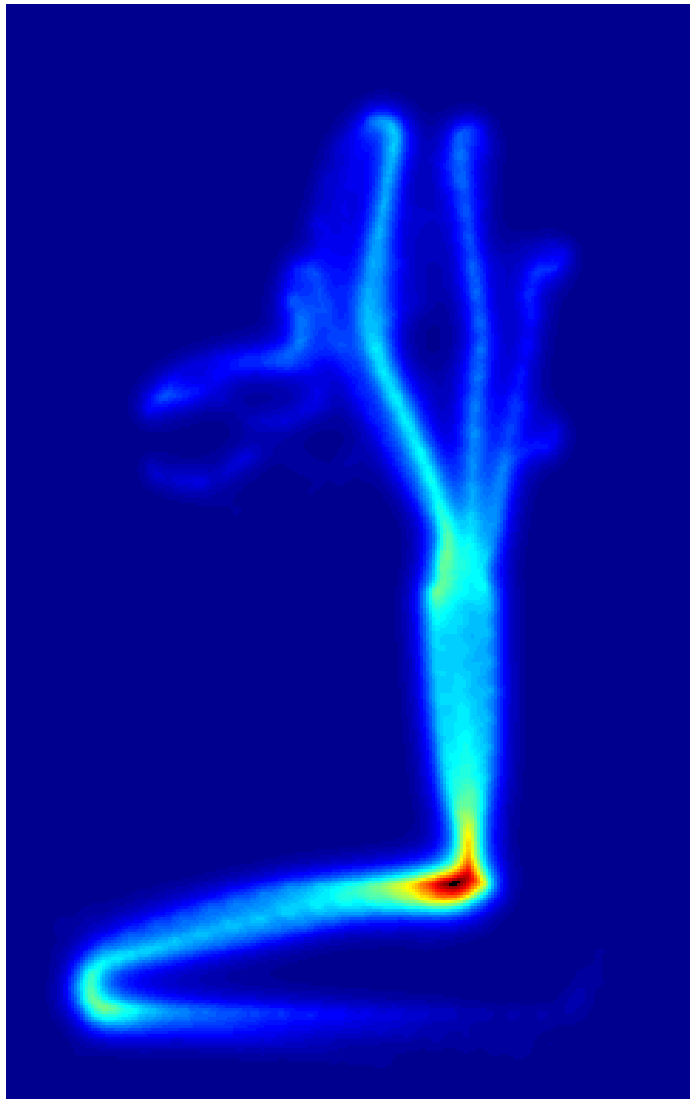




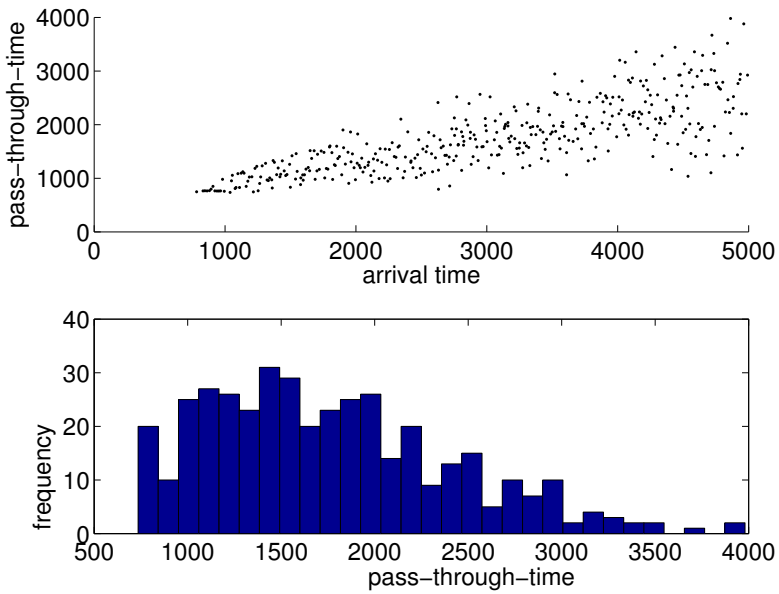


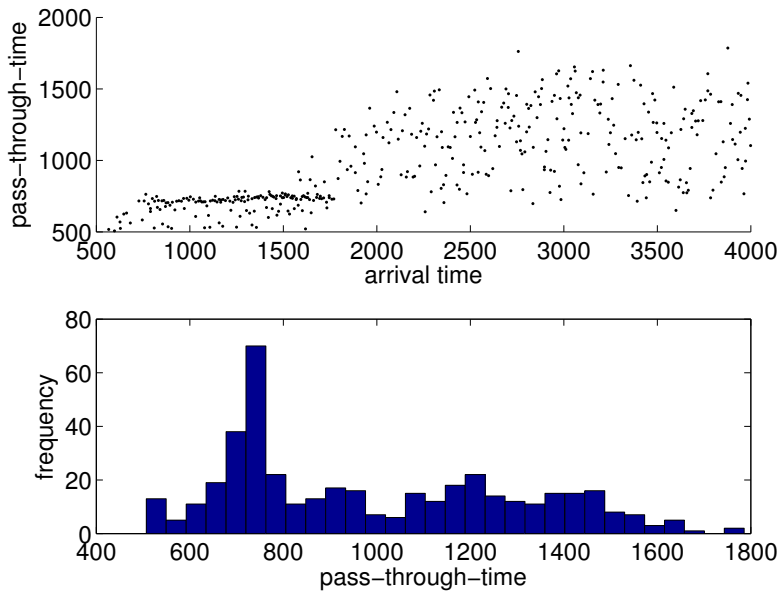
Finding The Balance



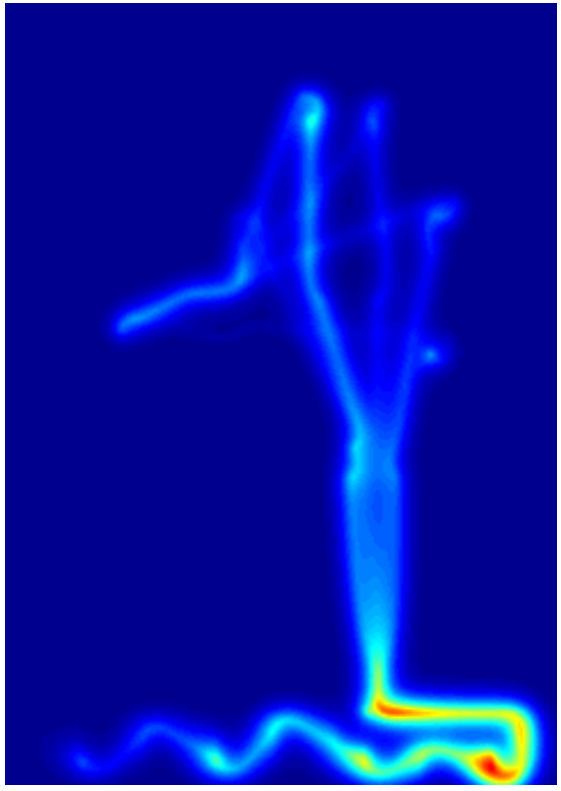


Measuring The Balance

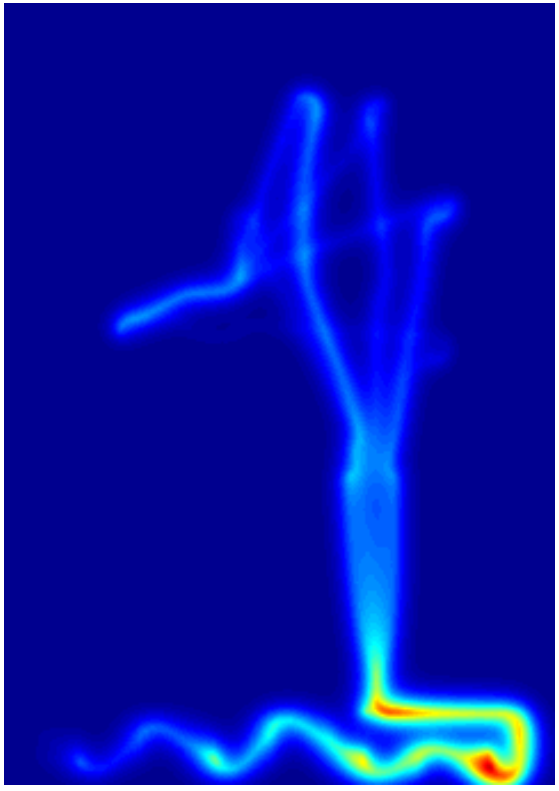




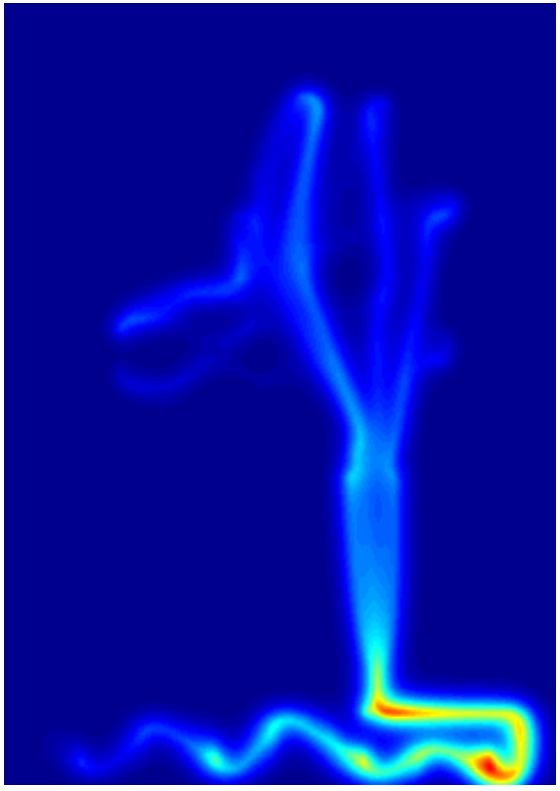
Comparison In The Mensa



Basic Model



+ Queueing



+ Q., + real-time FM

Video demo

- Struggling to get through waiting lines
- No waiting at the serving counters and checkouts

Summary and Outlook

- Fetching food is way more than bare queueing
- We were able to reproduce some characteristics of the polymensa-Crowd
- Simulating 2 Frames/second is tedious

Summary and Outlook

- Fetching food is way more than bare queueing
- We were able to reproduce some characteristics of the polymensa-Crowd
- Simulating 2 Frames/second is tedious

Summary and Outlook

- Fetching food is way more than bare queueing
- We were able to reproduce some characteristics of the polymensa-Crowd
- Simulating 2 Frames/second is tedious