Enabling new technology

COMMIT/

Universiteit Utrecht

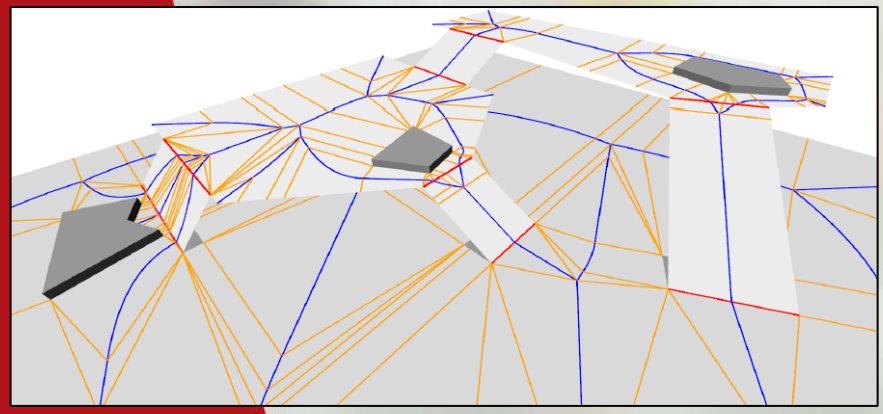
[Faculteit Bètawetenschappen
Informatica]

The Crowd Deciphered:

A computational simulation model of human navigation

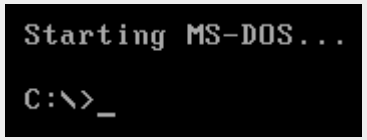
Dr. Roland Geraerts

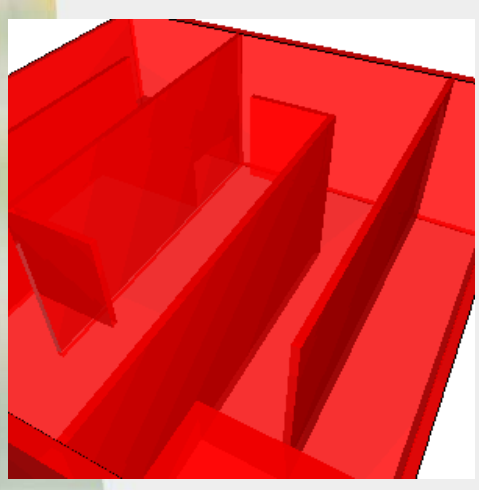
28 May 2016

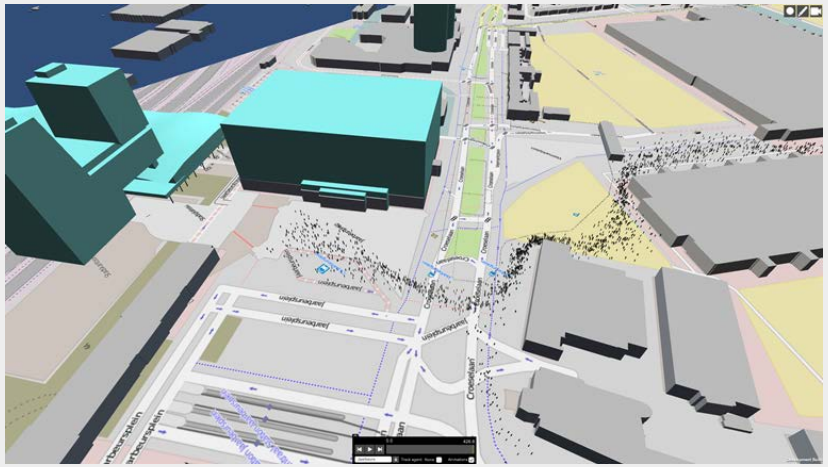


Who am I?

- Dr. Roland Geraerts
  - Computer games
  - Robotics background
  - Research on path planning and crowd simulation
  - Assistant professor
  - Crowd simulation software package







Social relevance of simulation

- Big-scale exercises (>500 people) are impractical
  - Has big impact on environment and surroundings
  - Costs considerable amount of time
  - Tests performed on a few scenarios only
- Crowd simulation is needed for
  - Simulations of the real world
    - Improving crowd flow, predicting crowd pressures, planning evacuation routes
  - Improving the immersion/realism in virtual worlds

Game worlds

Rebuilding of train station

Love Parade, Duisburg, 2010
21 deaths
510 injuries

Evacuation in sports stadiums

Universiteit Utrecht

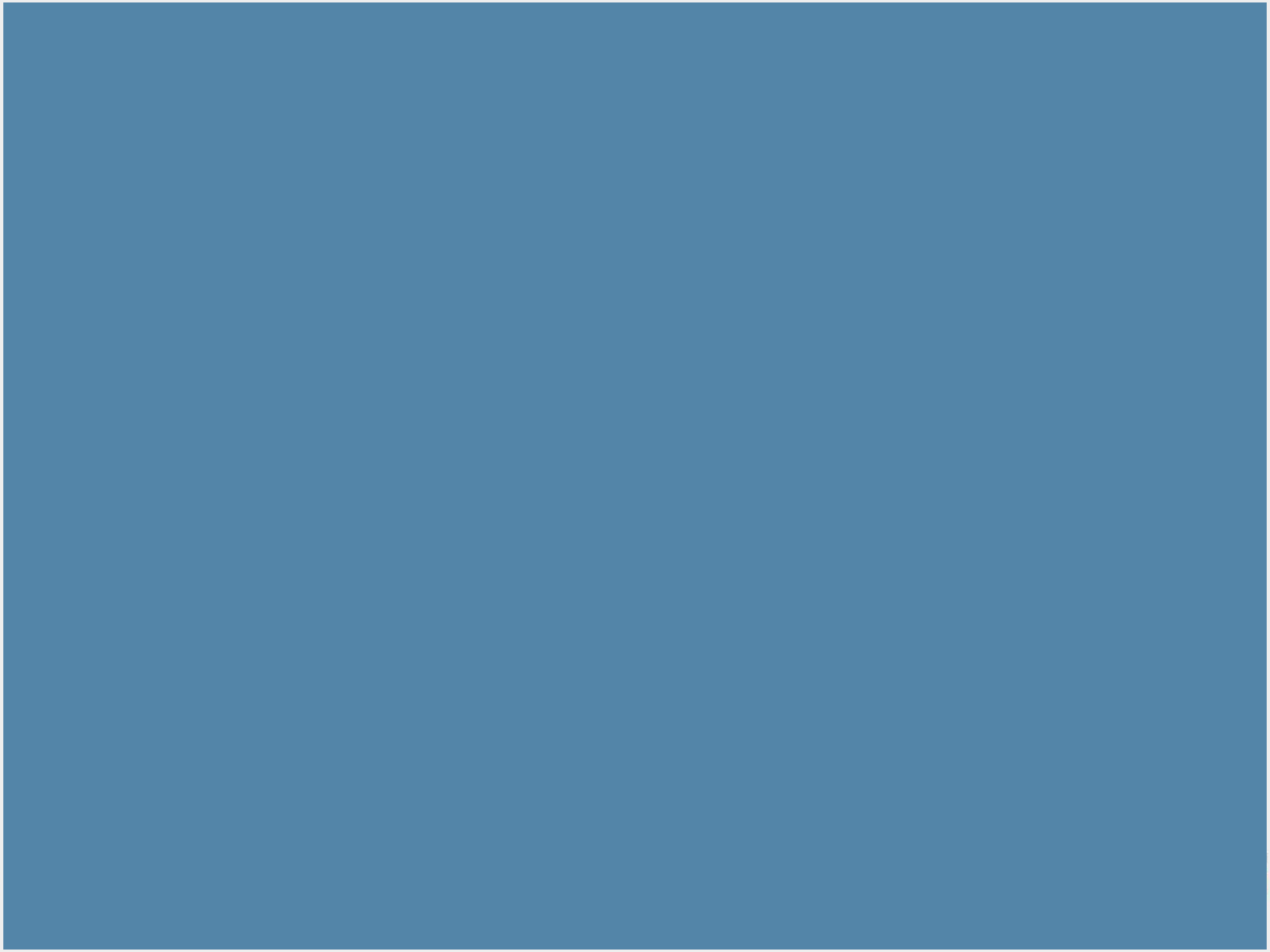
[Faculteit Bètawetenschappen
Informatica]

A computational model of human navigation

Challenge: Unify *dispersed models* for *realistic,* individual, small group, and collective human movements in *interactive, heterogeneous* environments.

- Dispersed models
  - Agent-based: individuals, but problems with high densities
  - Flow-based: no individuals, but good for high densities
- Realistic movements
  - Comprise collaboration, smooth and energy-efficient movement, collision avoidance, and dealing with unrealistic congestions.
- Interactive environment
  - Geometry can change dynamically, and the crowd has to react.
- Heterogeneous environment
  - People need to take logical, distinct, and realistic paths over heterogeneous terrains in the environment.

Are we there yet?



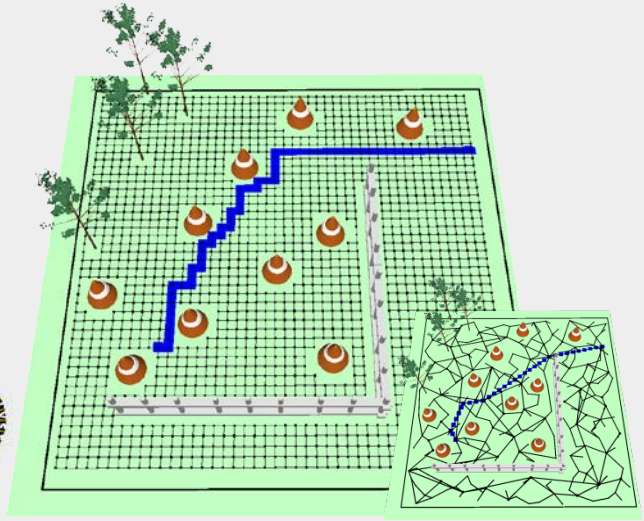
Path planning errors in *games*

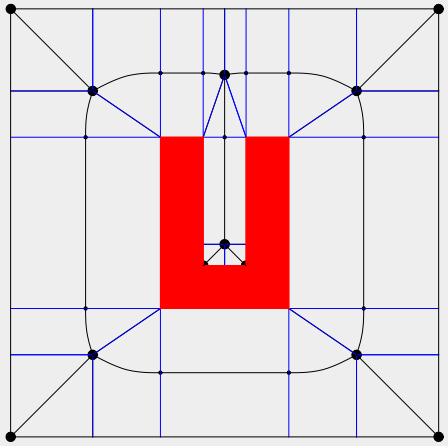
- Networks of waypoints are incorrect
  - Hand designed
  - Do not adapt to changes in the environment
  - Do not adapt to the type of character
- Local methods fail to find a route
  - Keep stuck behind objects
  - Lead to repeated motion
- Groups split up
  - Not planned as a coherent entity
- Paths are unnatural
  - Not smooth
  - Stay too close to network/obstacles
- Methodology is not general enough to handle all problems

Titan Quest: Immortal throne

Representation of the traversable environment

- We need a *paradigm shift*
  - from graph-based to surface-based navigation
- Graph-based navigation: little support for route deviation
  - Hard to avoid expected collision between humans
  - Hard to support differently sized humans/groups
  - Costly to deal with dynamic changes in the environment
  - Hard to efficiently deal with heterogeneous regions
  - Human navigation is surface-based

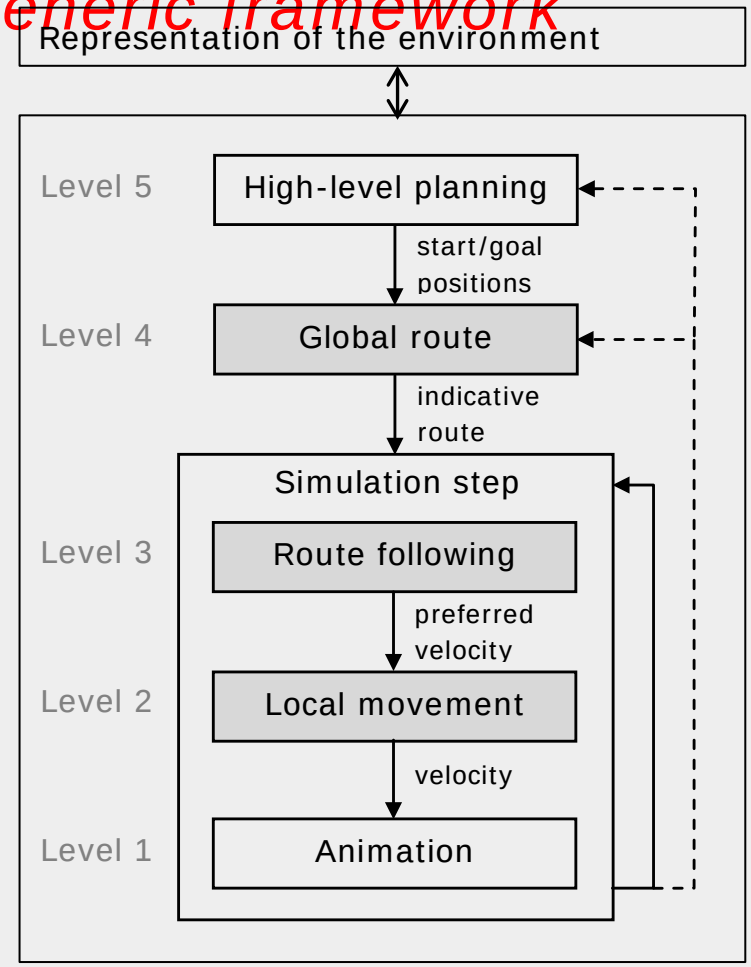




Crowd simulation framework

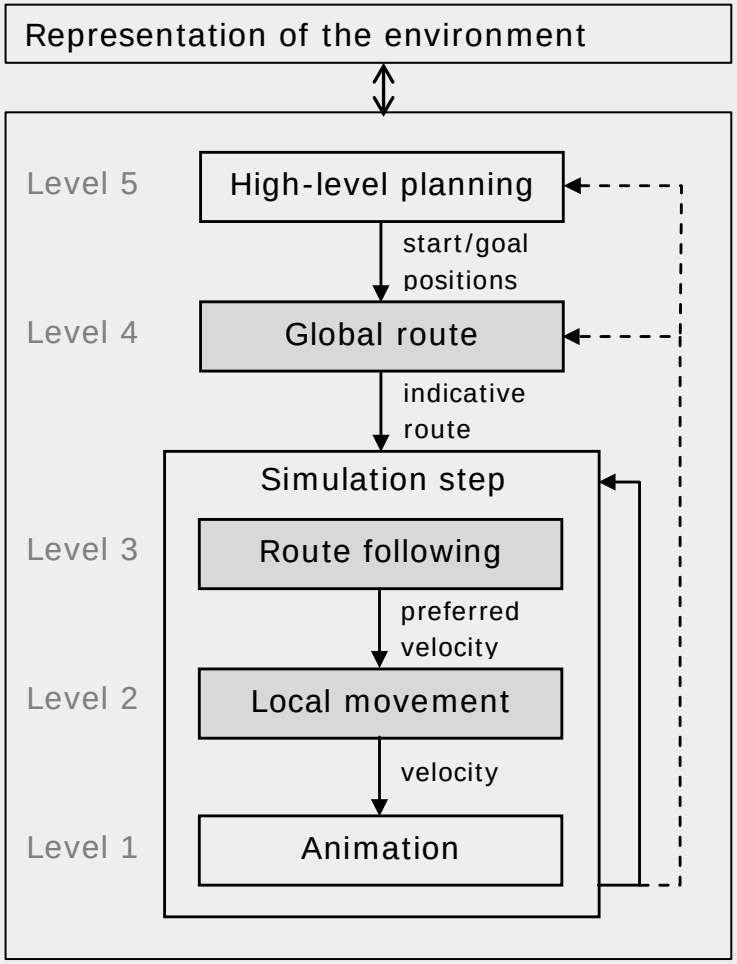
We need a fast and generic framework

- Representation environment
- Level 5
  - Plans actions
- Level 4
  - Creates indicative routes
- Level 3
  - Traverses the routes
  - Yields speed/direction pairs
- Level 2
  - Adapts routes
  - E.g. to avoid collisions
- Level 1
  - Moves the characters



Crowd simulation framework

- Representation environment
- Level 5
  - Plans actions
- Level 4
  - Creates indicative routes
- Level 3
  - Traverses the routes
  - Yields speed/direction pairs
- Level 2
  - Adapts routes
  - E.g. to avoid collisions
- Level 1
  - Moves the characters



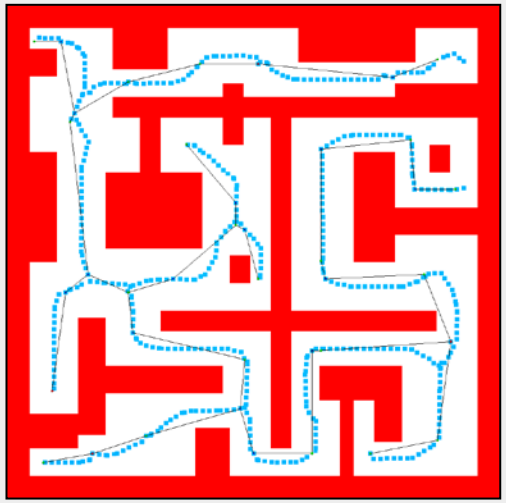
Representation of the traversable environment

- Requirements
  - Path existence
  - 100% coverage of the navigable space
  - All cycles
  - Fast computation and small storage
  - Fast query time during simulation
  - *Flexible: surfaces*

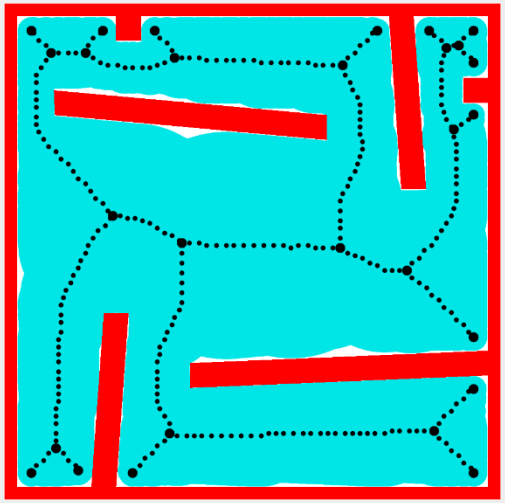
Representation of the traversable environment

- I did many attempts…

Mathematical morphology



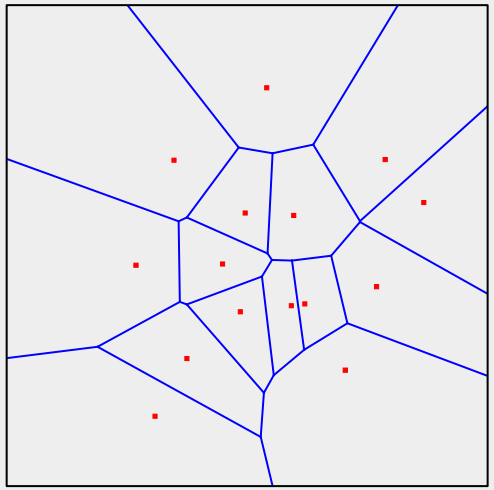
Sampling and retraction on the medial axis…



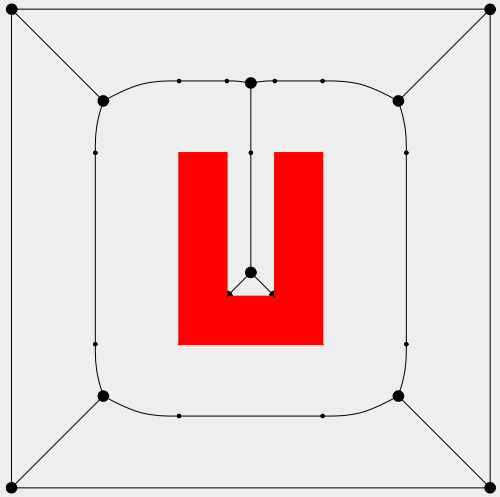
…adding clearance disks

Representation of the traversable environment

- Navigation mesh
  - Voronoi diagram / Medial axis



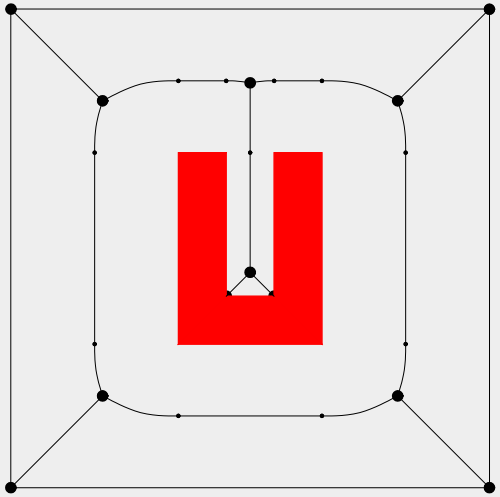
Voronoi sites: red points

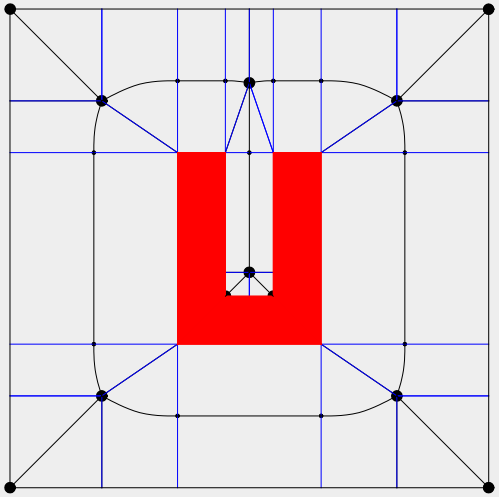


Voronoi sites: red/black lines

Representation of the traversable environment

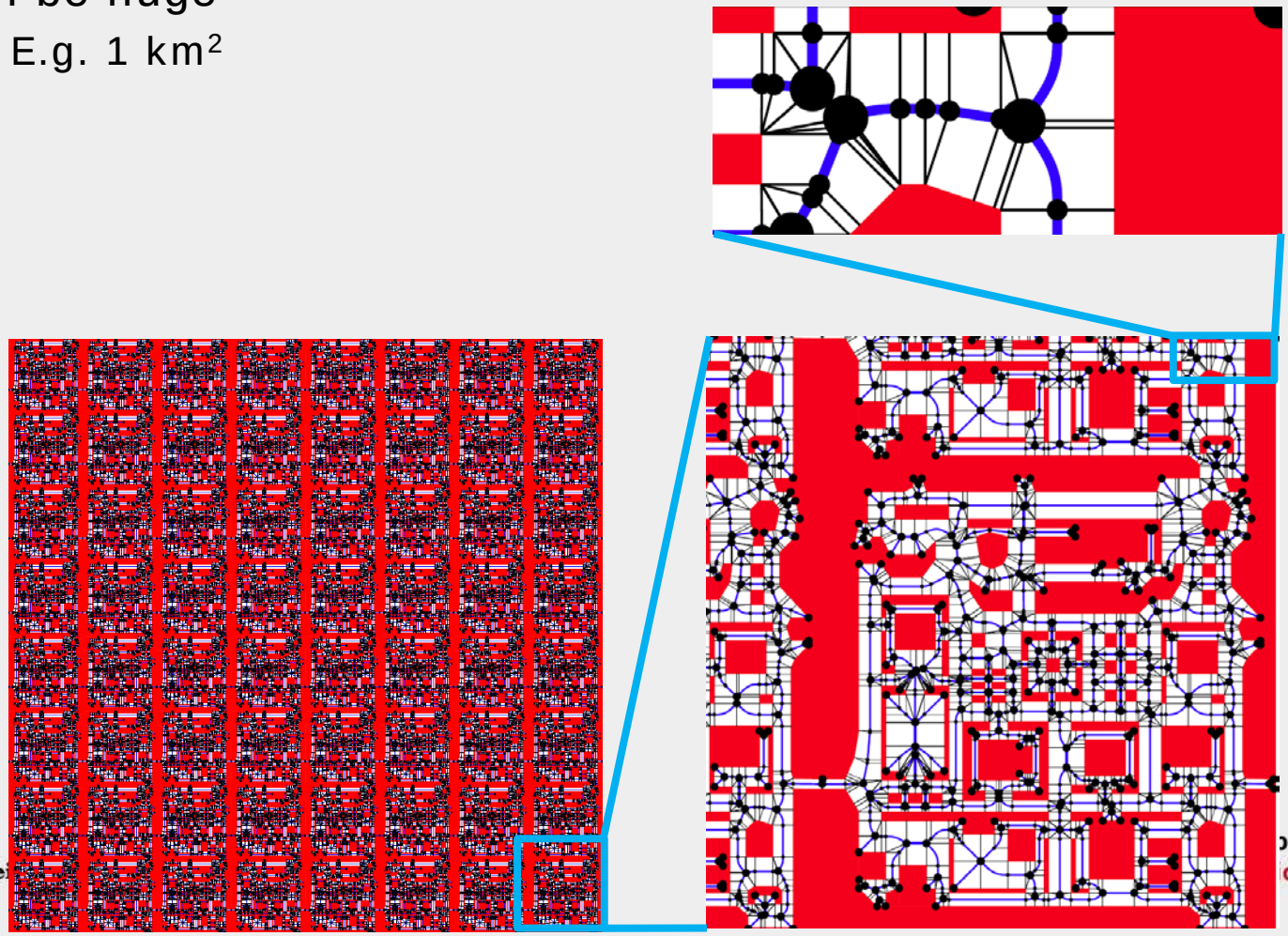
- Navigation mesh
  - Voronoi diagram / Medial axis
  - From a graph to a surface representation
    - Closest point annotation





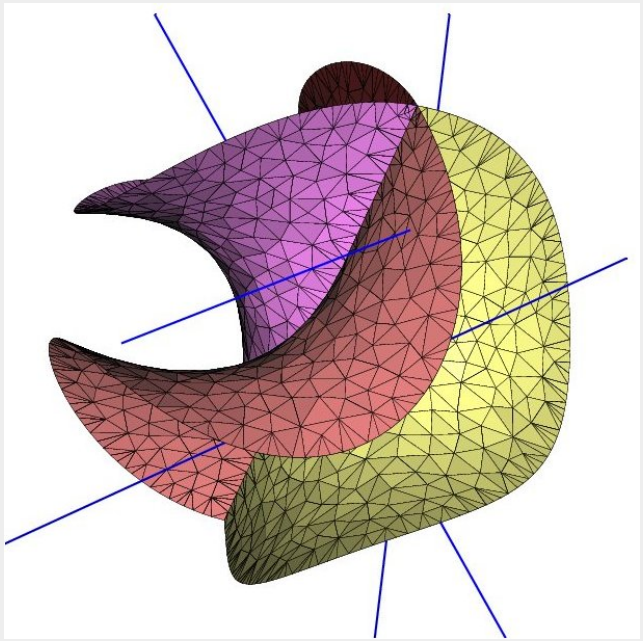
Representation of the traversable environment

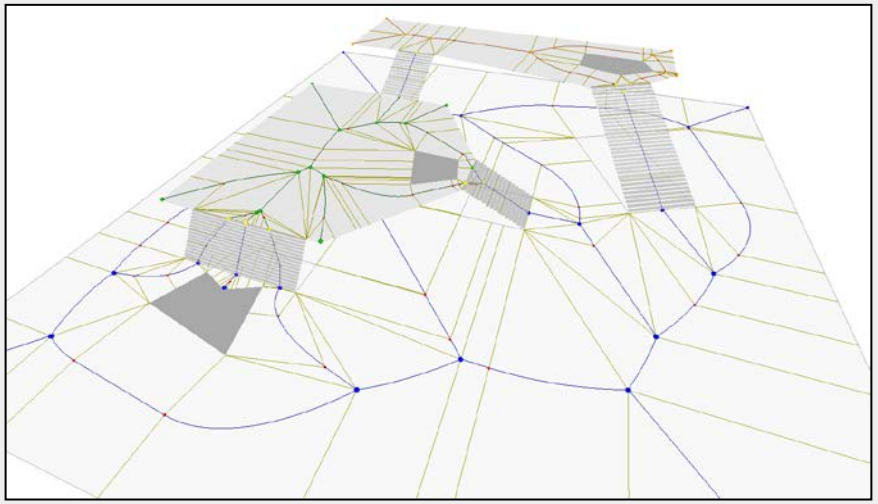
- Can be huge
  - E.g. 1 km^2



What about 3D environments?

- 3D Voronoi diagram?
  - No – create a multi-layered Voronoi diagram

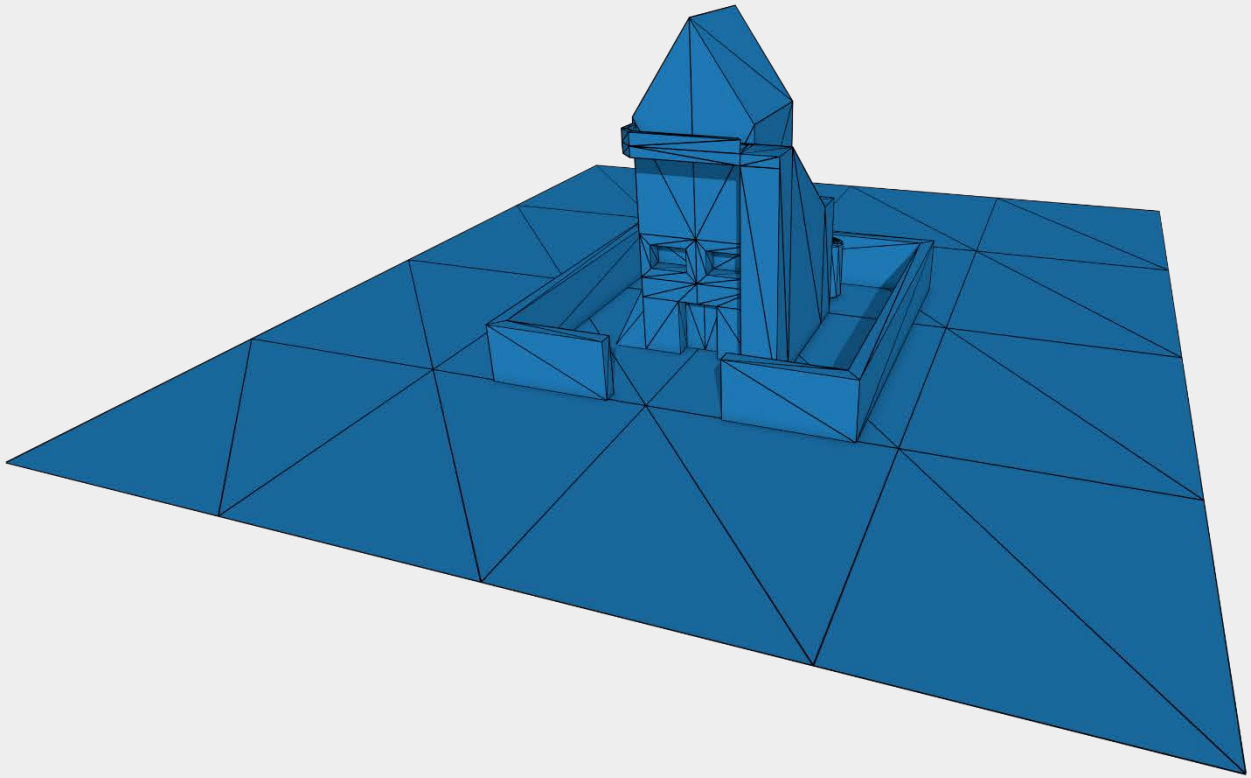




M. Hemmer, O. Setter, and D. Halperin. *Constructing the Exact Voronoi Diagram of Arbitrary Lines in Space with Fast Point-Location.* RR-7273, INRIA. 2010, pp.19.

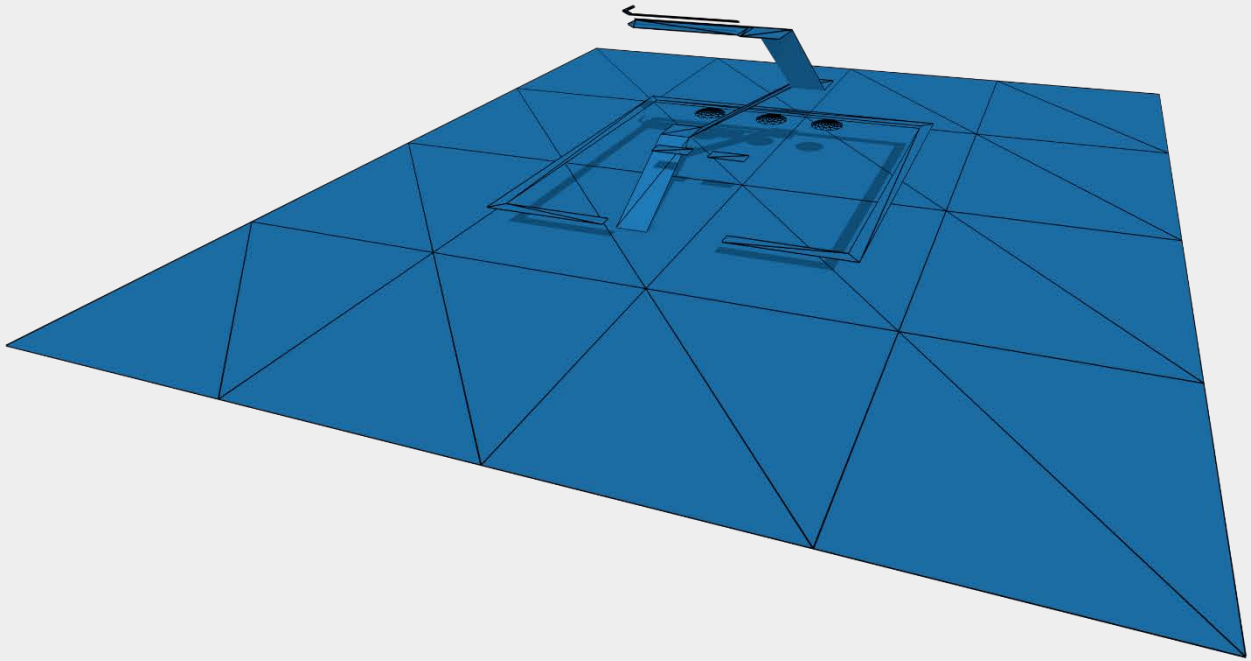
What about 3D environments?

- Preprocessing steps that extract the walkable areas
  - Input environment



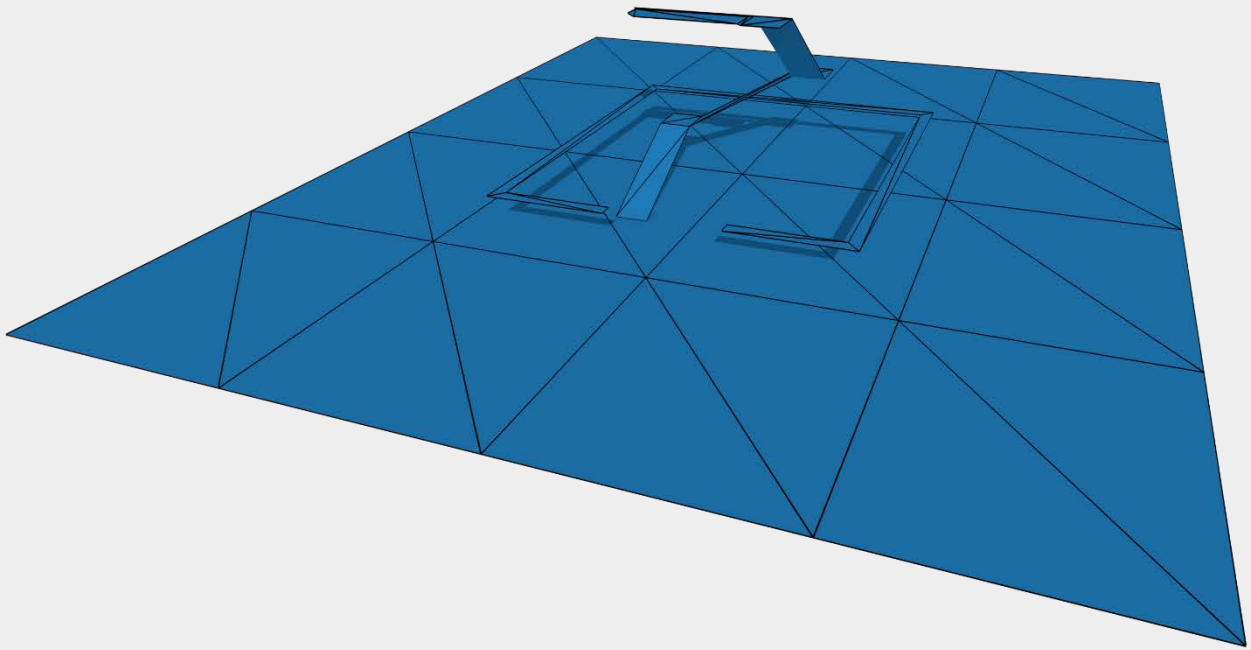
What about 3D environments?

- Preprocessing steps that extract the walkable areas
  1. Remove steep polygons



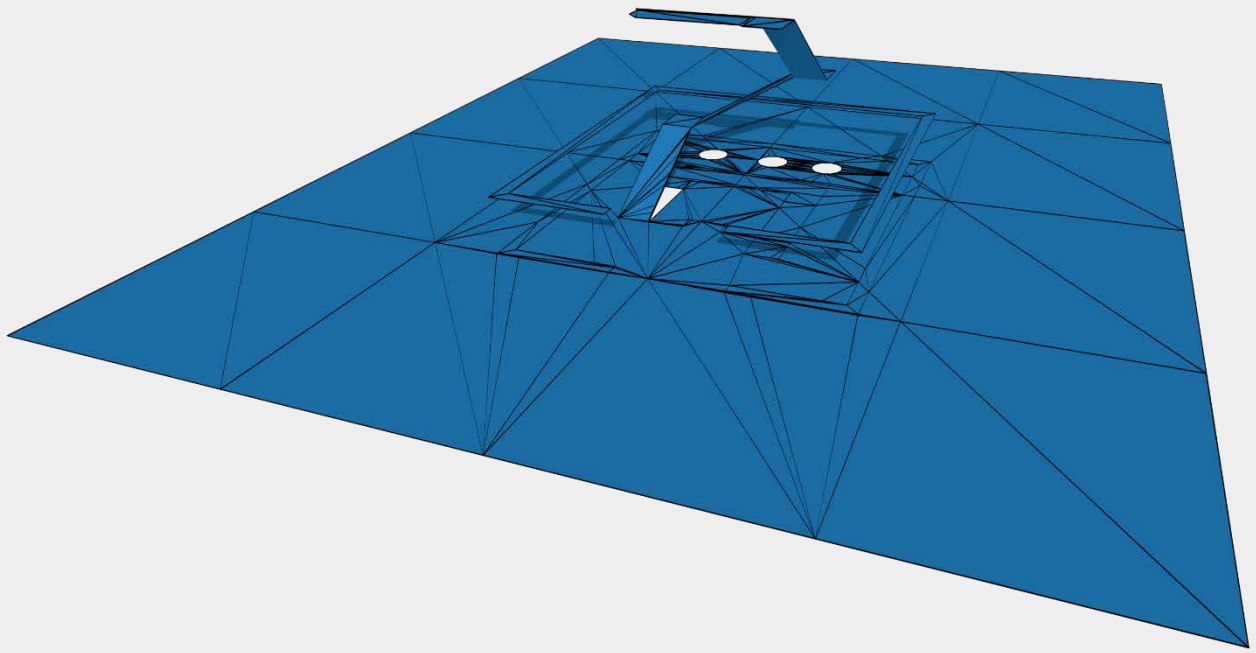
What about 3D environments?

- Preprocessing steps that extract the walkable areas
  2. Remove small regions



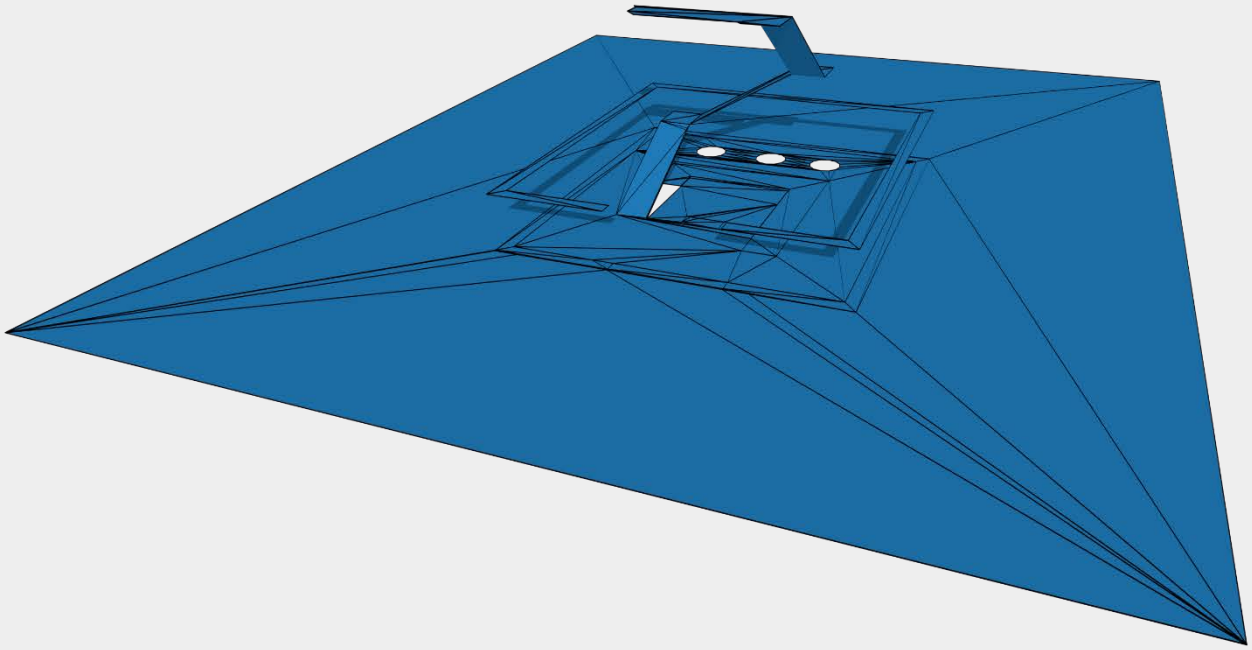
What about 3D environments?

- Preprocessing steps that extract the walkable areas
  3. Check vertical clearance (using all geometry)



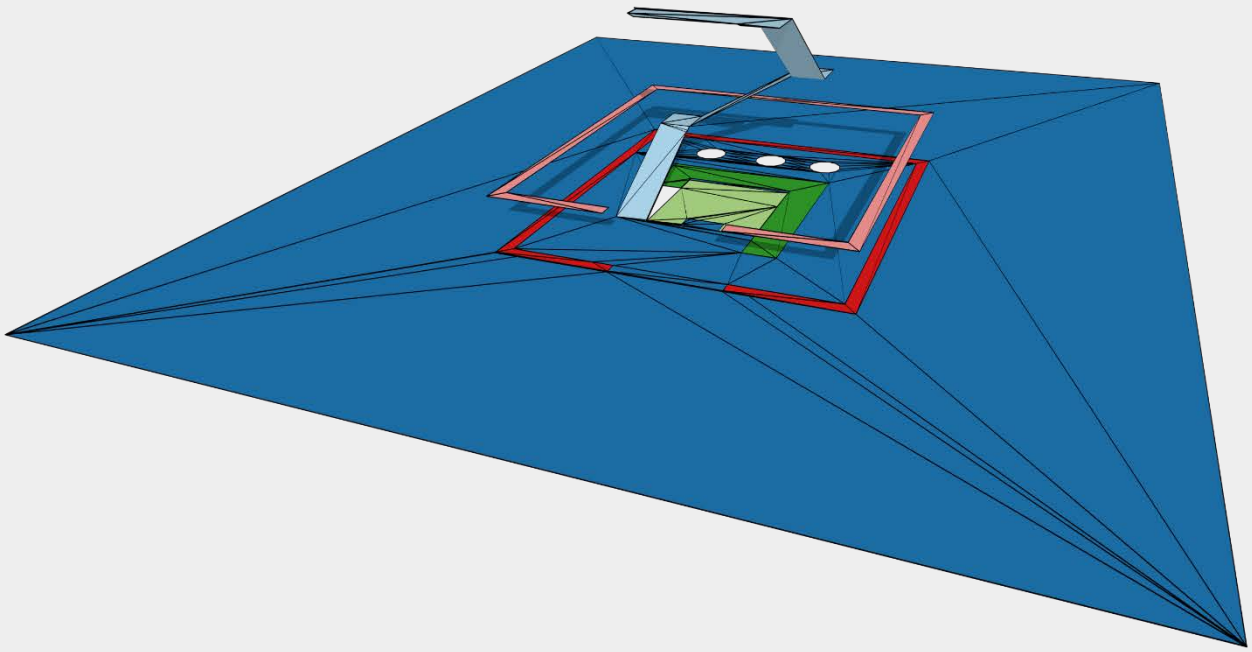
What about 3D environments?

- Preprocessing steps that extract the walkable areas
  4. Simplify triangulation



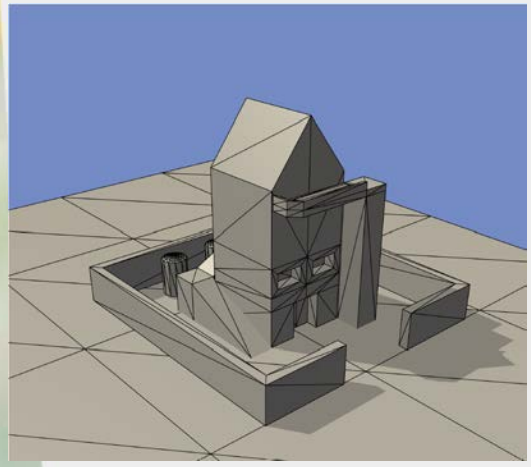
What about 3D environments?

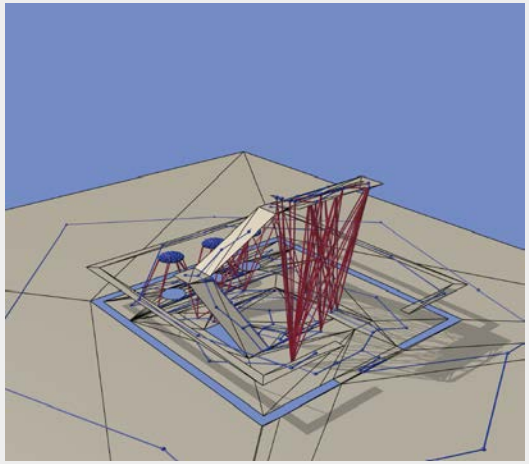
- Preprocessing steps that extract the walkable areas
  5. Separate into layers

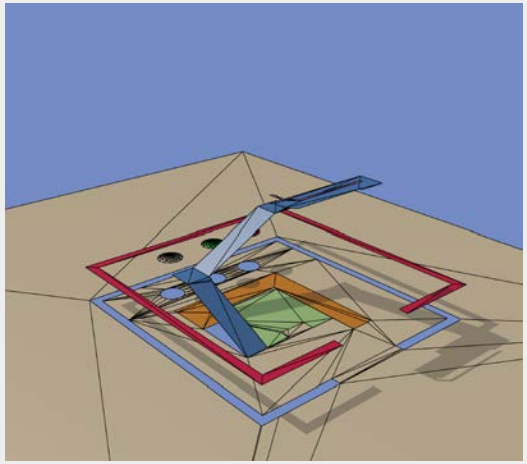


What about 3D environments?

- Separate into layers, s.t. number of connections is minimized
  - Instance of the multi-commodity minimal-cut problem
  - Cannot use function parameter tractable (FPT) algorithms
  - Is an NP-hard problem
    - Use heuristics to split environment into 2D layers

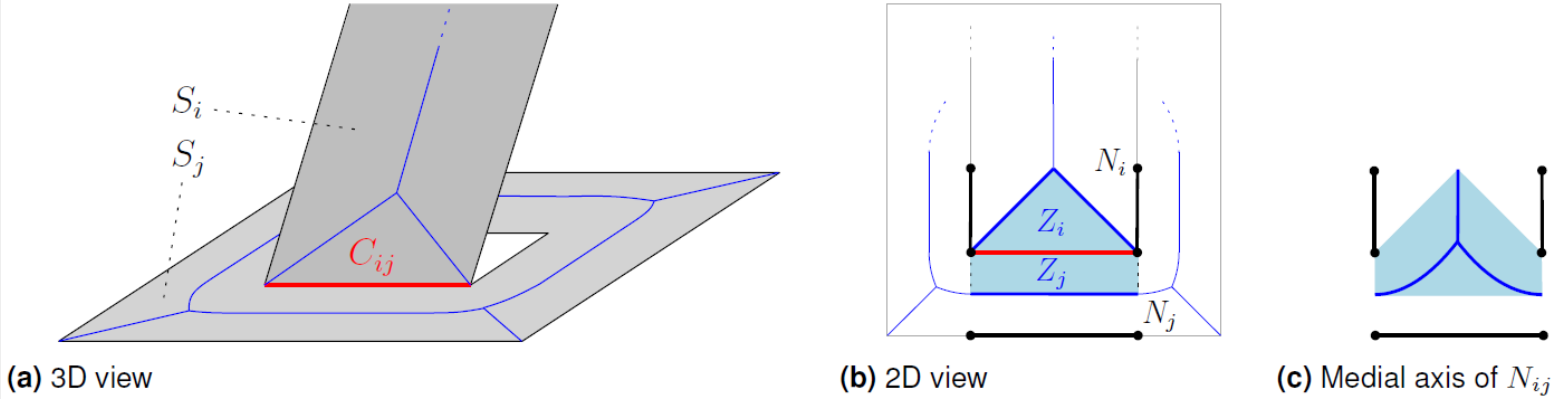




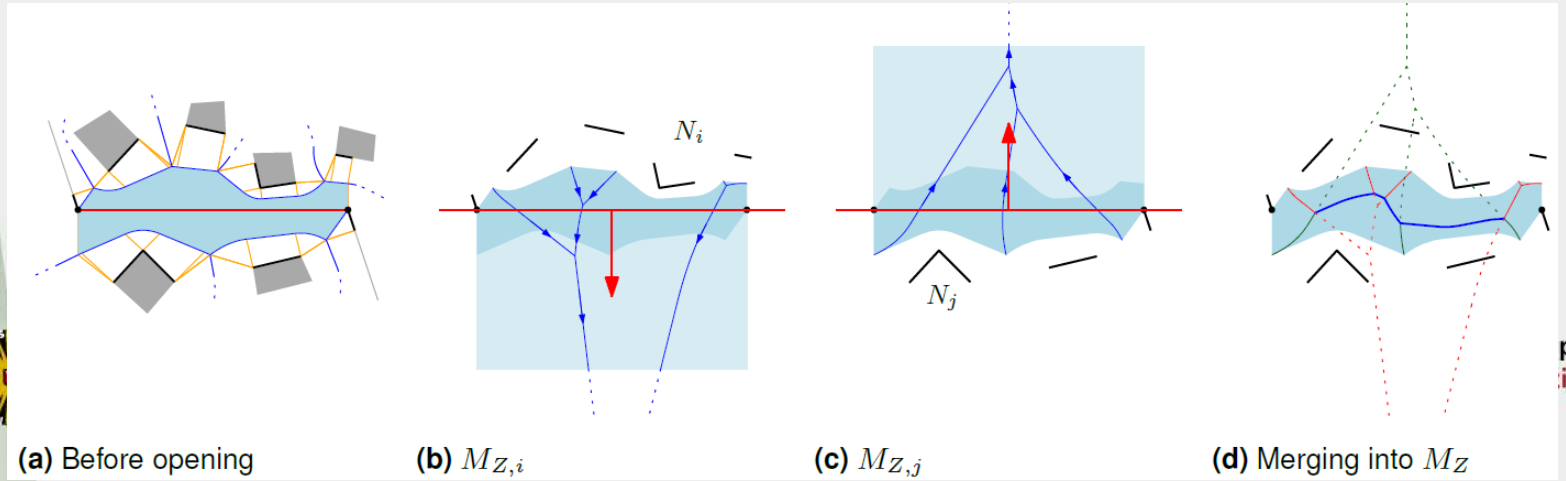


What about 3D environments?

6. For each 2D layer, create a 2D navigation mesh
7. Stitch them together into a multi-layered navigation mesh



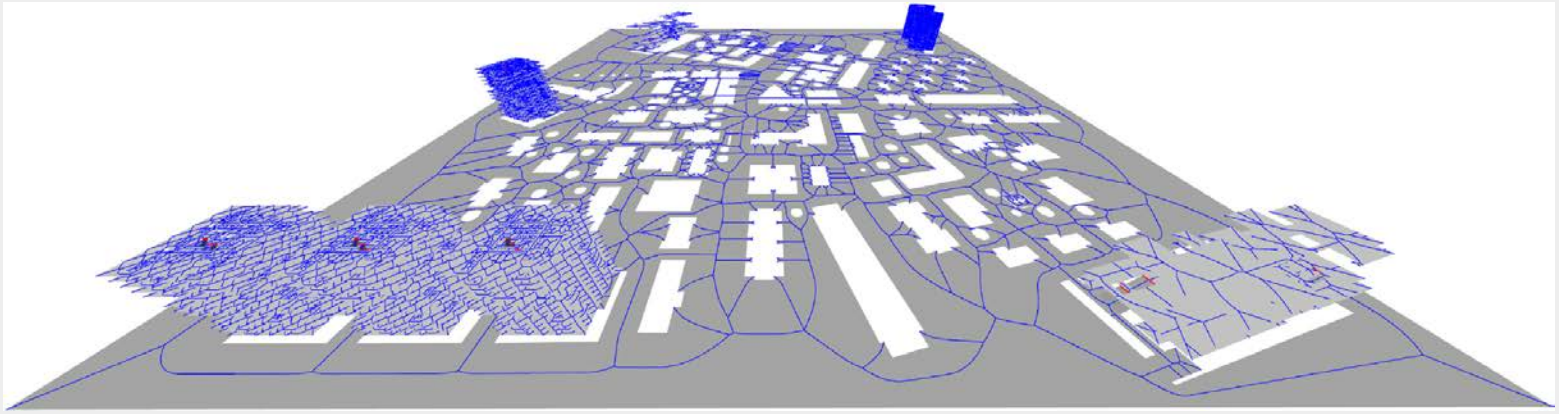
(a) 3D view **(b)** 2D view **(c)** Medial axis of N_{ij}

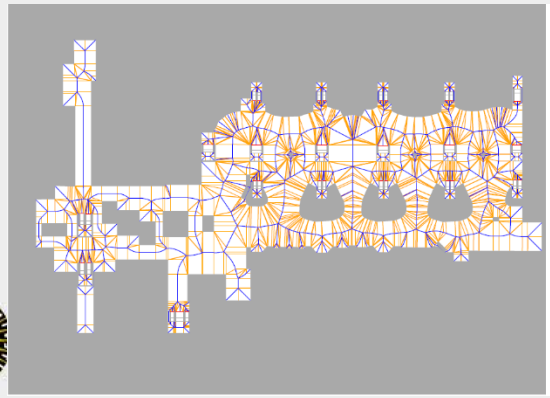


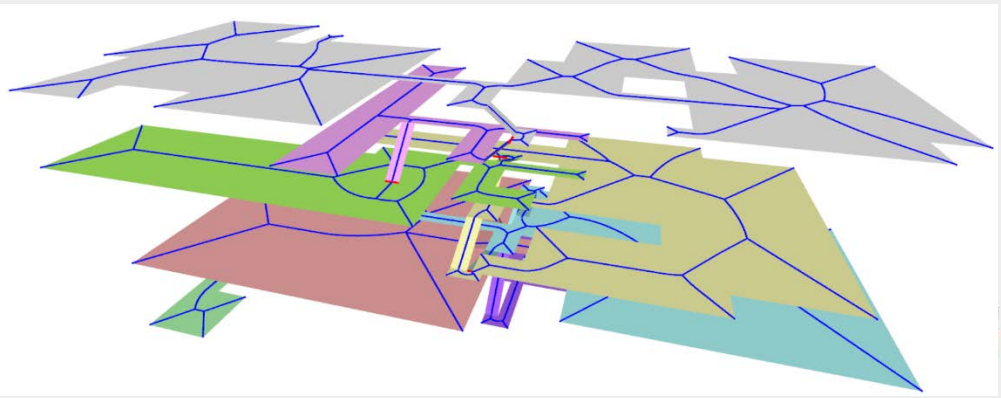
(a) Before opening **(b)** $M_{Z,i}$ **(c)** $M_{Z,j}$ **(d)** Merging into M_Z

Representation of the traversable environment

- Can be *really* huge
  - E.g. many km^2

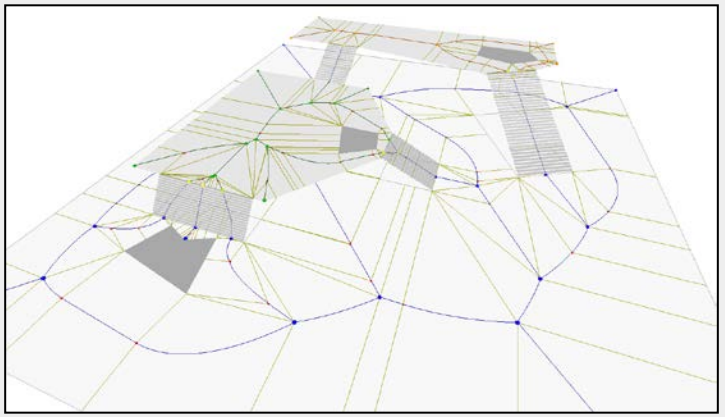




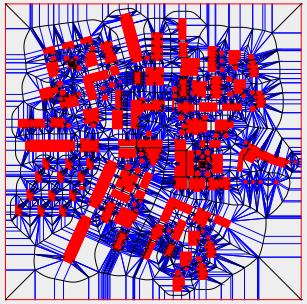


Representation of the traversable environment

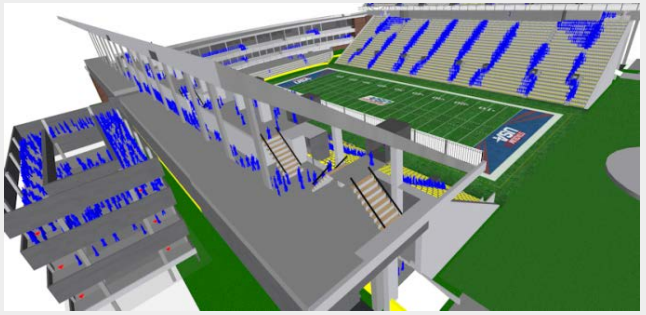
- Navigation mesh
  - Fast to compute



10 ms



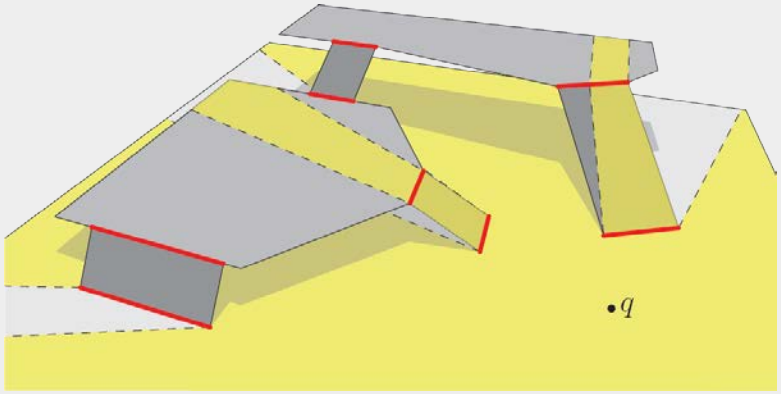
115 ms

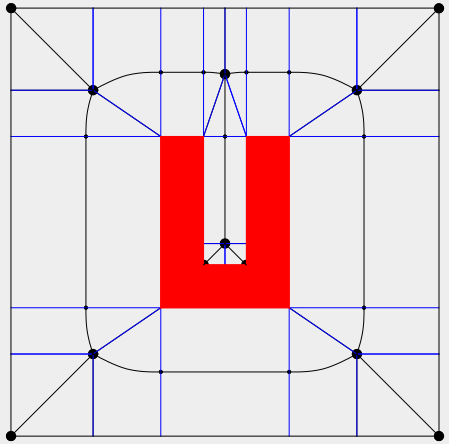


3 s

Representation of the traversable environment

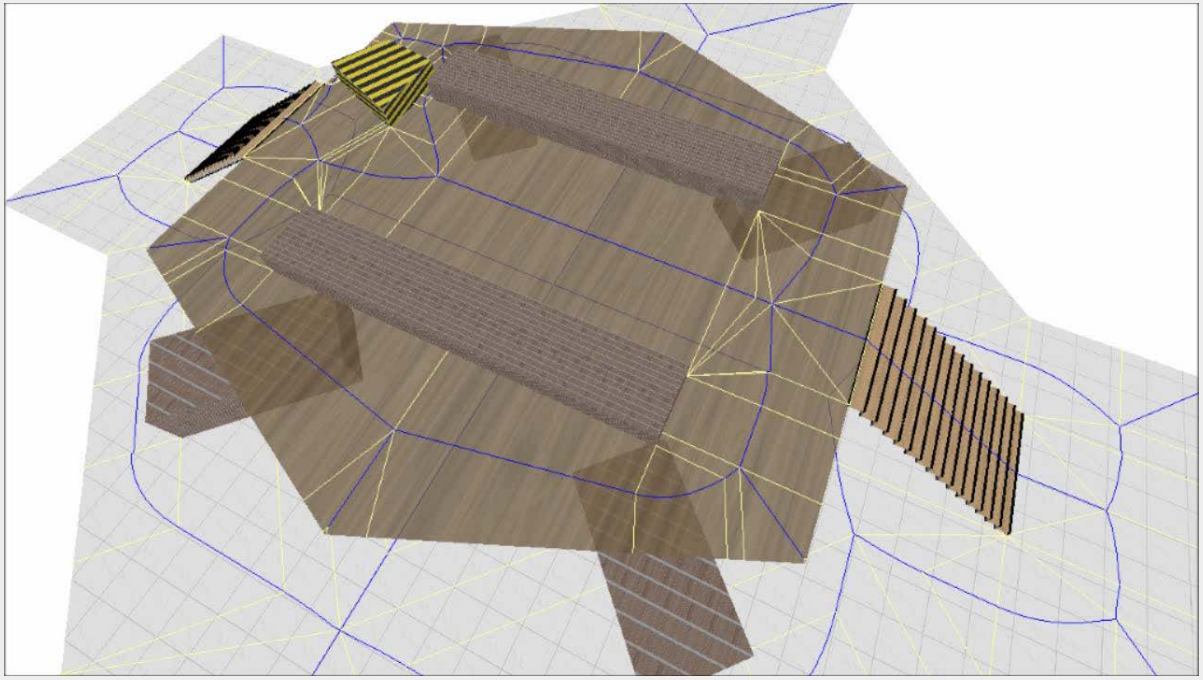
- Navigation mesh
  - Exact representation
    - Captures 100% of the free space
    - Captures all homotopically different routes (cycles)
  - Allows fast extraction of global routes and final paths
  - Nice mathematical properties
    - Fast to compute – O(n log n log k), with k connections
    - Small data structure – O(n)
    - Nearest obstacle computation – O(1)
    - 2D algorithms also work in multi-layered environments





Representation of the traversable environment

- Navigation mesh
  - Handles dynamic changes
    - Update costs < 1 ms



Representation of the traversable environment

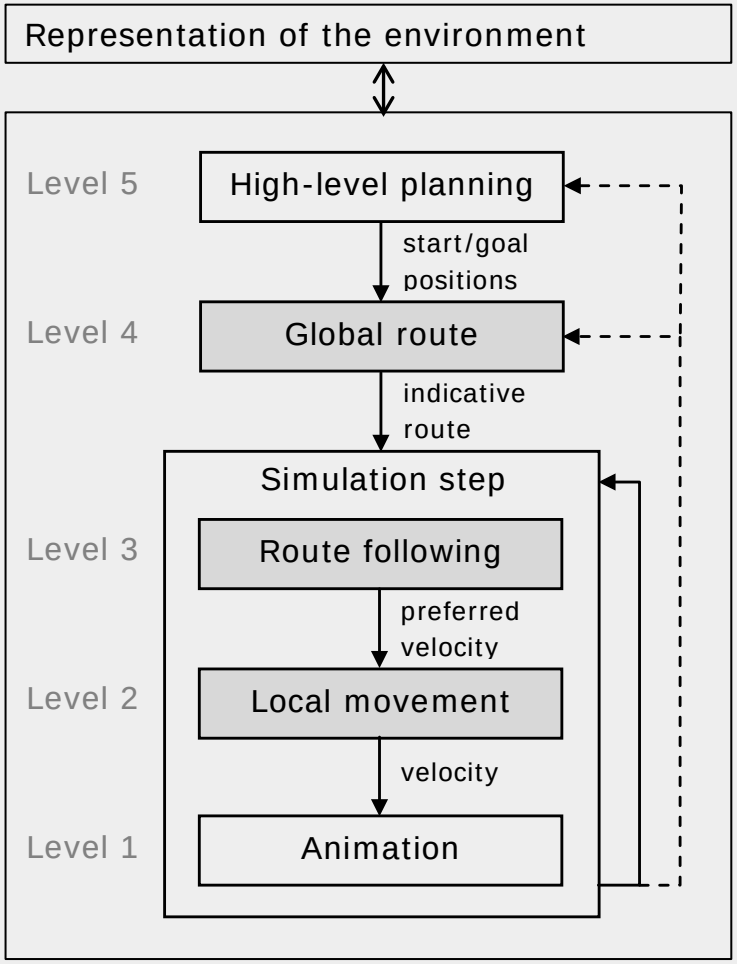
- Navigation mesh
  - Handles dynamic changes
    - Demo at Campus Party (Utrecht Region)

Path planning errors in *games*

Pathfinding challenges with large groups

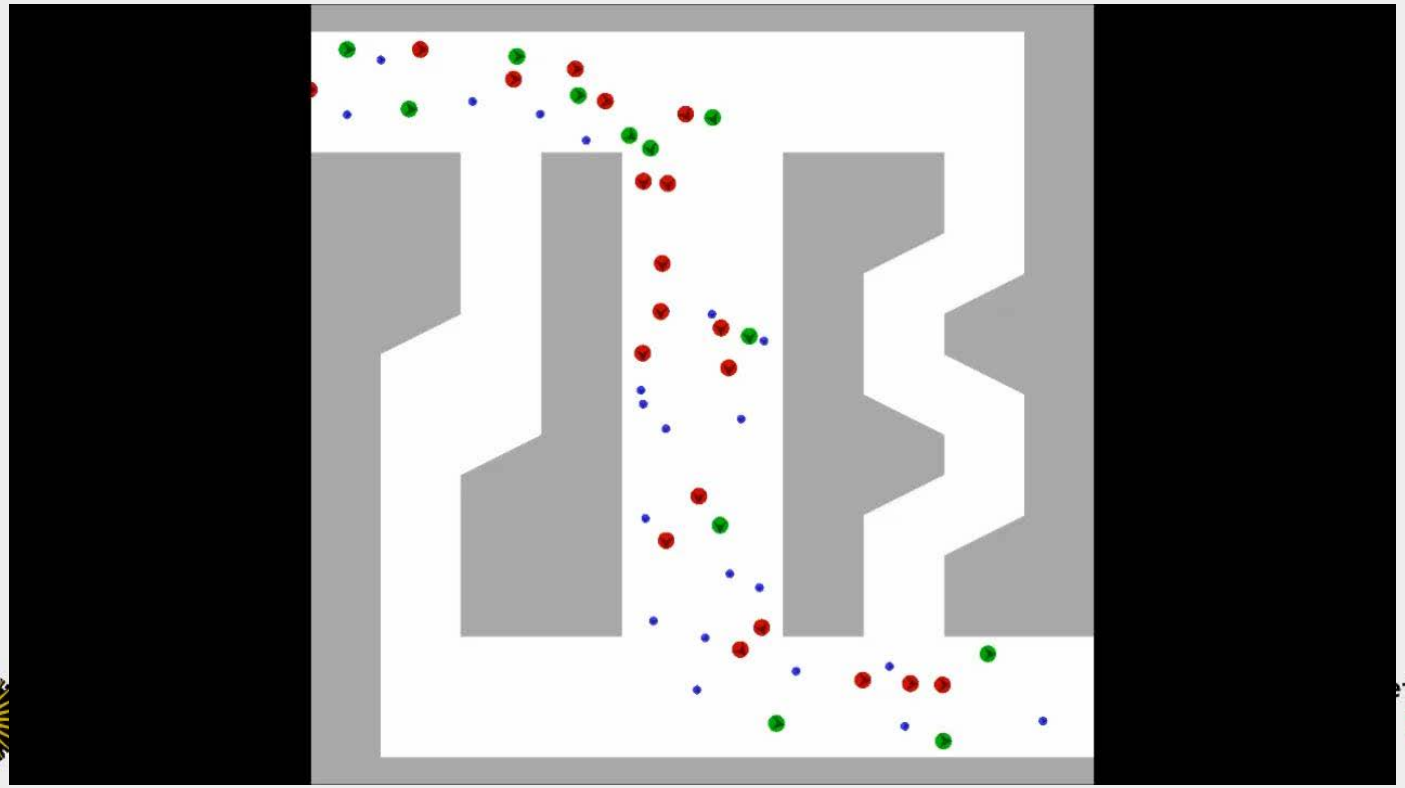
Crowd simulation framework

- Representation environment
- Level 5
  - Plans actions
- Level 4
  - Creates indicative routes
- Level 3
  - Traverses the routes
  - Yields speed/direction pairs
- Level 2
  - Adapts routes
  - E.g. to avoid collisions
- Level 1
  - Moves the characters



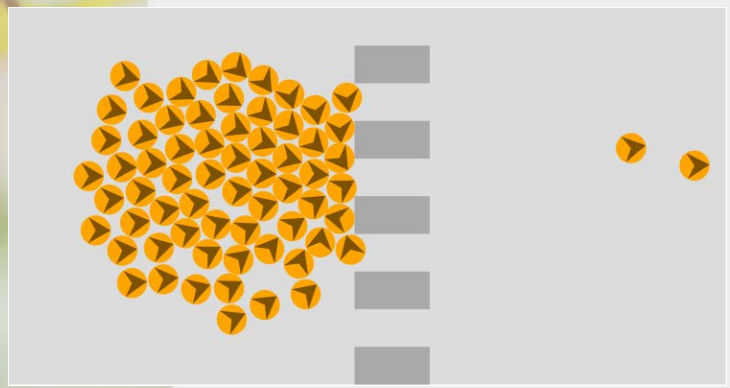
Action planning

- Splits up a task into geometric queries
  - Example: dynamic updates of the crowd

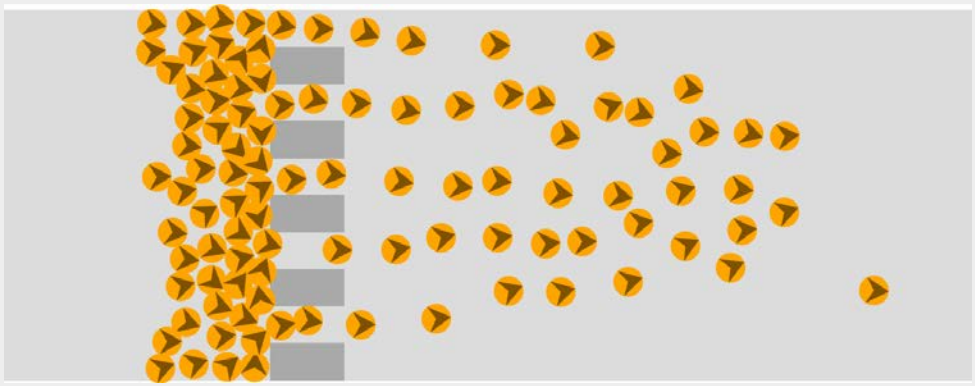


Action planning

- Splits up a task into geometric queries
  - Example: dynamic updates of the crowd



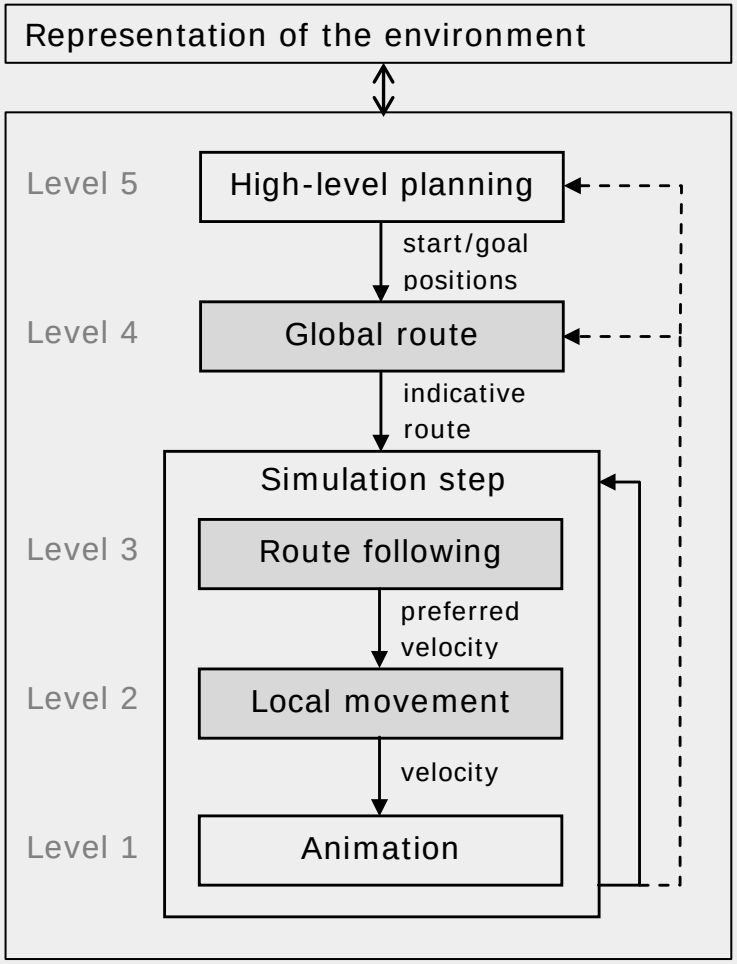
Standard behavior: pedestrians take the same ov-terminal / escalator



Improved behavior: pedestrians take the same ov-terminal / escalator

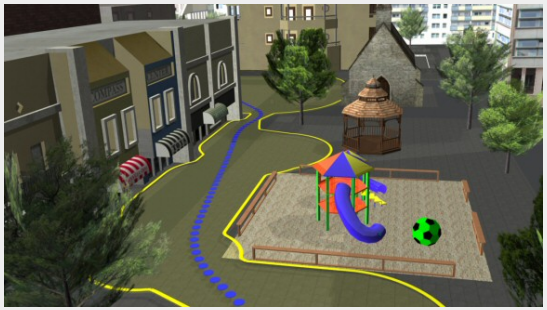
Crowd simulation framework

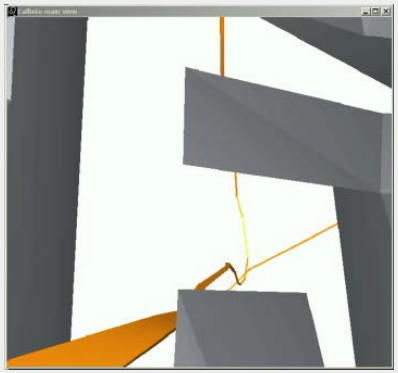
- Representation environment
- Level 5
  - Plans actions
- Level 4
  - Creates indicative routes
- Level 3
  - Traverses the routes
  - Yields speed/direction pairs
- Level 2
  - Adapts routes
  - E.g. to avoid collisions
- Level 1
  - Moves the characters

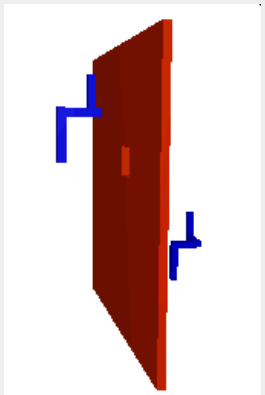


Indicative routes

- A path planning algorithm should NOT compute a path
  - A one-dimensional path limits the character's freedom
  - Humans don't do that either
- It should produce
  - An Indicative/Preferred Route
    - Guides character to goal
- It uses a corridor

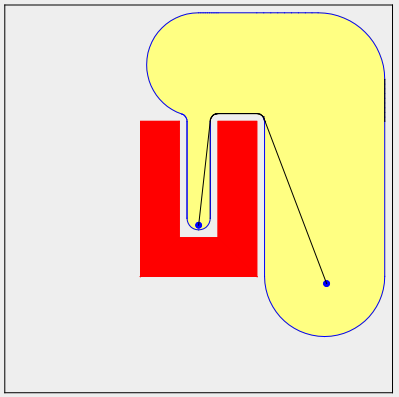


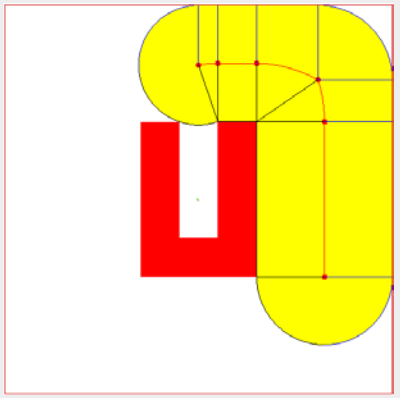




Computing Indicative Routes

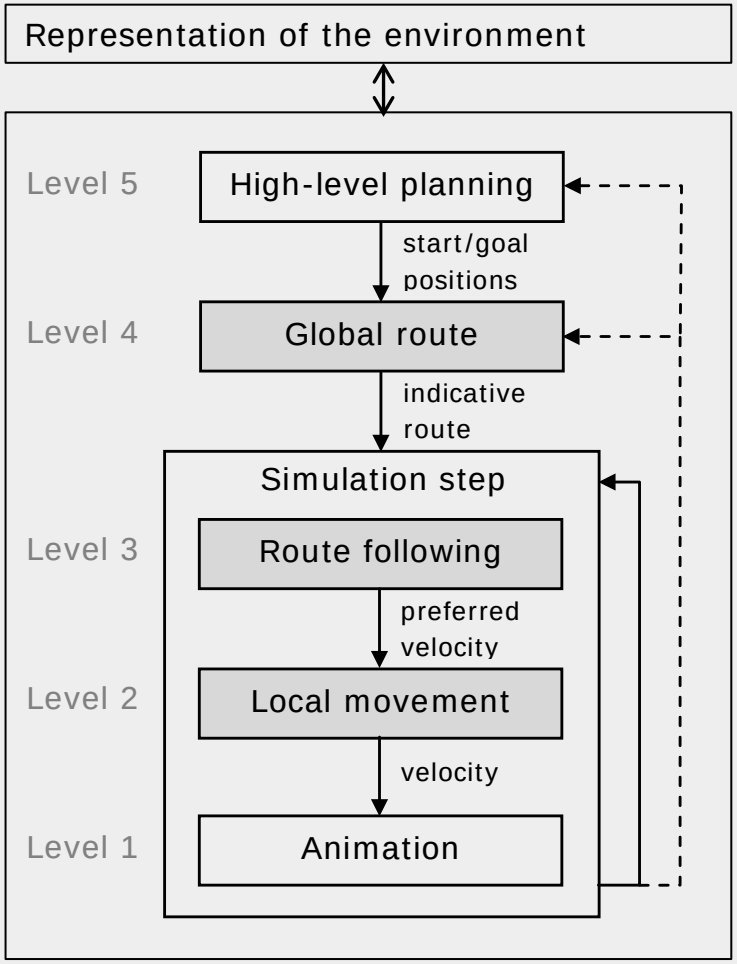
- Short path with clearance
- Short path with weighted regions
  - Unsolvable in the Algebraic Computation Model over the Rational Numbers





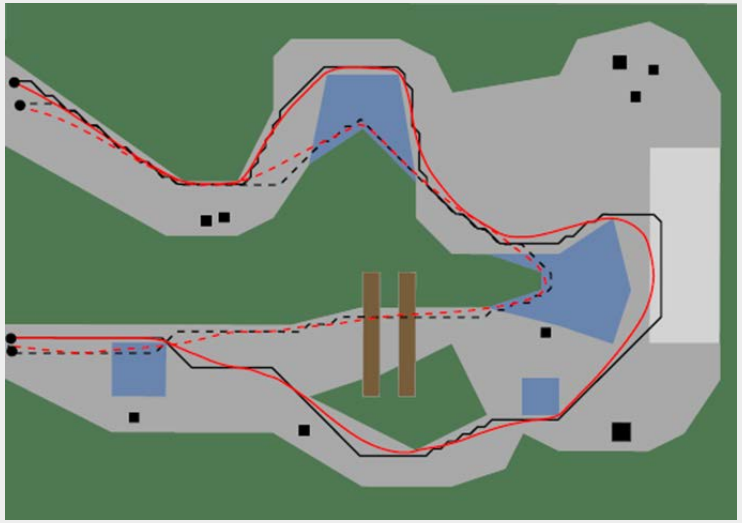
Crowd simulation framework

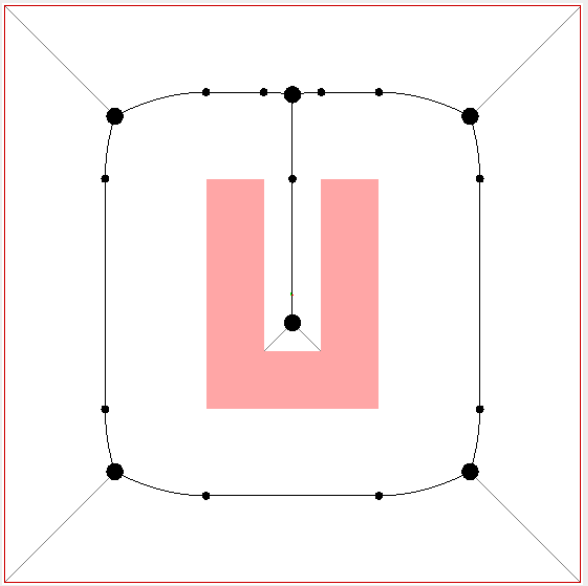
- Representation environment
- Level 5
  - Plans actions
- Level 4
  - Creates indicative routes
- Level 3
  - Traverses the routes
  - Yields speed/direction pairs
- Level 2
  - Adapts routes
  - E.g. to avoid collisions
- Level 1
  - Moves the characters

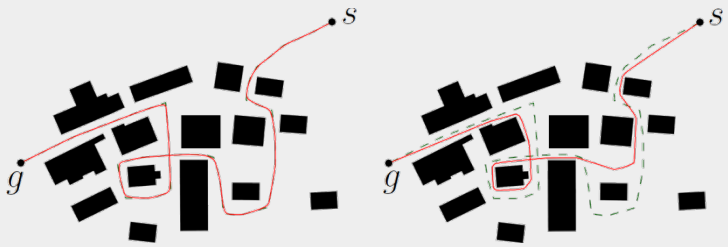


Traversing the routes

- Modified Indicative Routes And Navigation (MIRAN)
- Supports
  - heterogeneous terrains
  - separate character profiles
  - customized smoothing

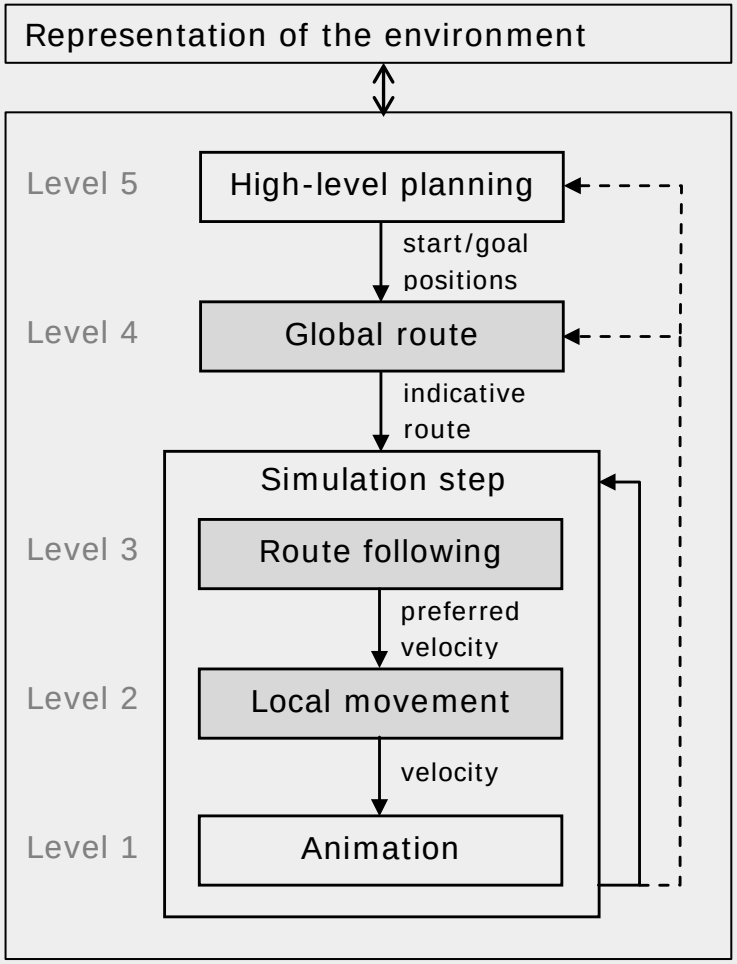






Crowd simulation framework

- Representation environment
- Level 5
  - Plans actions
- Level 4
  - Creates indicative routes
- Level 3
  - Traverses the routes
  - Yields speed/direction pairs
- Level 2
  - Adapts routes
  - E.g. to avoid collisions
- Level 1
  - Moves the characters



What is realistic collision-avoidance behavior?

Smack the pony s01x02

What is realistic collision-avoidance behavior?

Crowd prank in Japan

Adapting the routes: Collision avoidance

- Our model is derived from experiments in the MOCAP lab

PhD students Wouter van Toll and Norman Jaklin are making recordings in the motion capture lab.

Adapting the routes

- Supports high-quality collision avoidance

Adapting the routes: Social groups

- The group members stay close and visible to each other

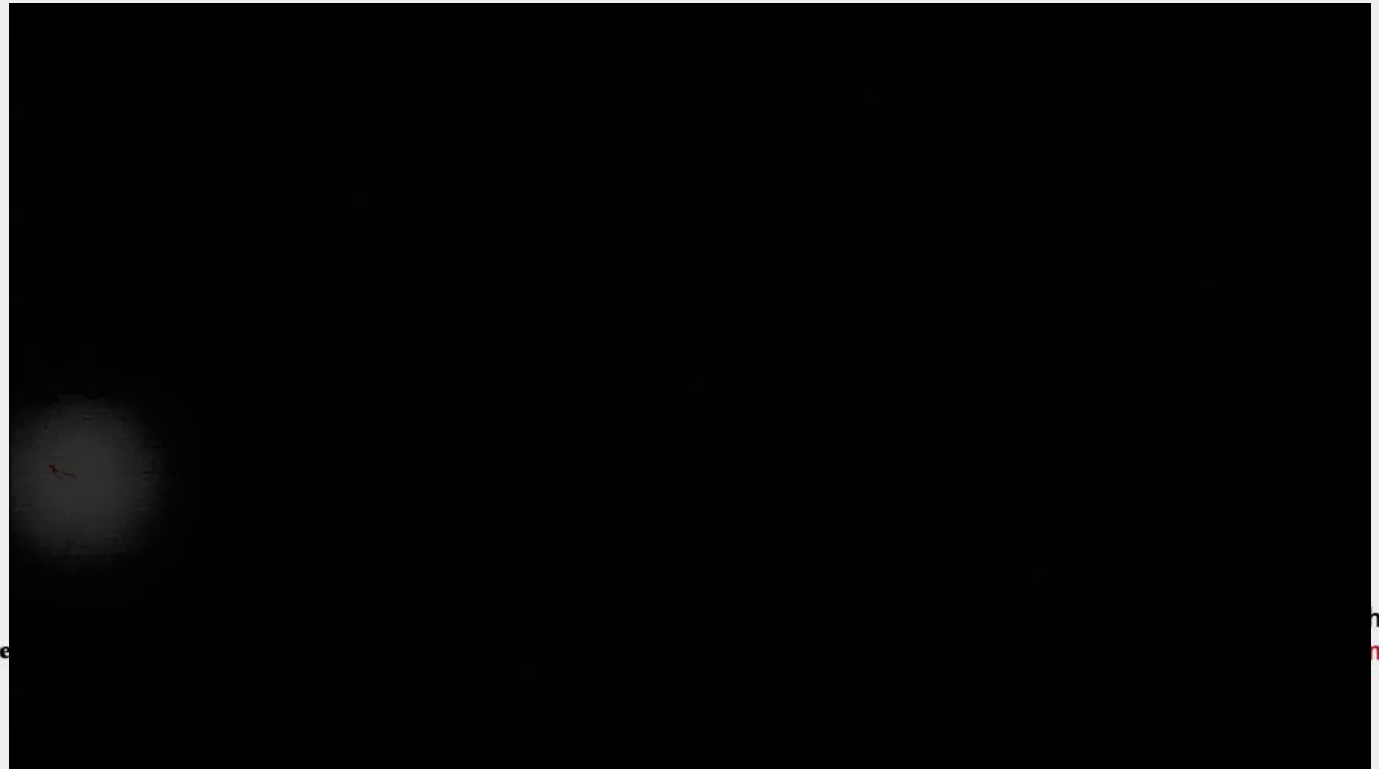
Adapting the routes: Moving through a dense crowd

- People making room for a passing individual

By S. Stüvel and A. Egges.

Adapting the routes:
Unification of individual and collective movements

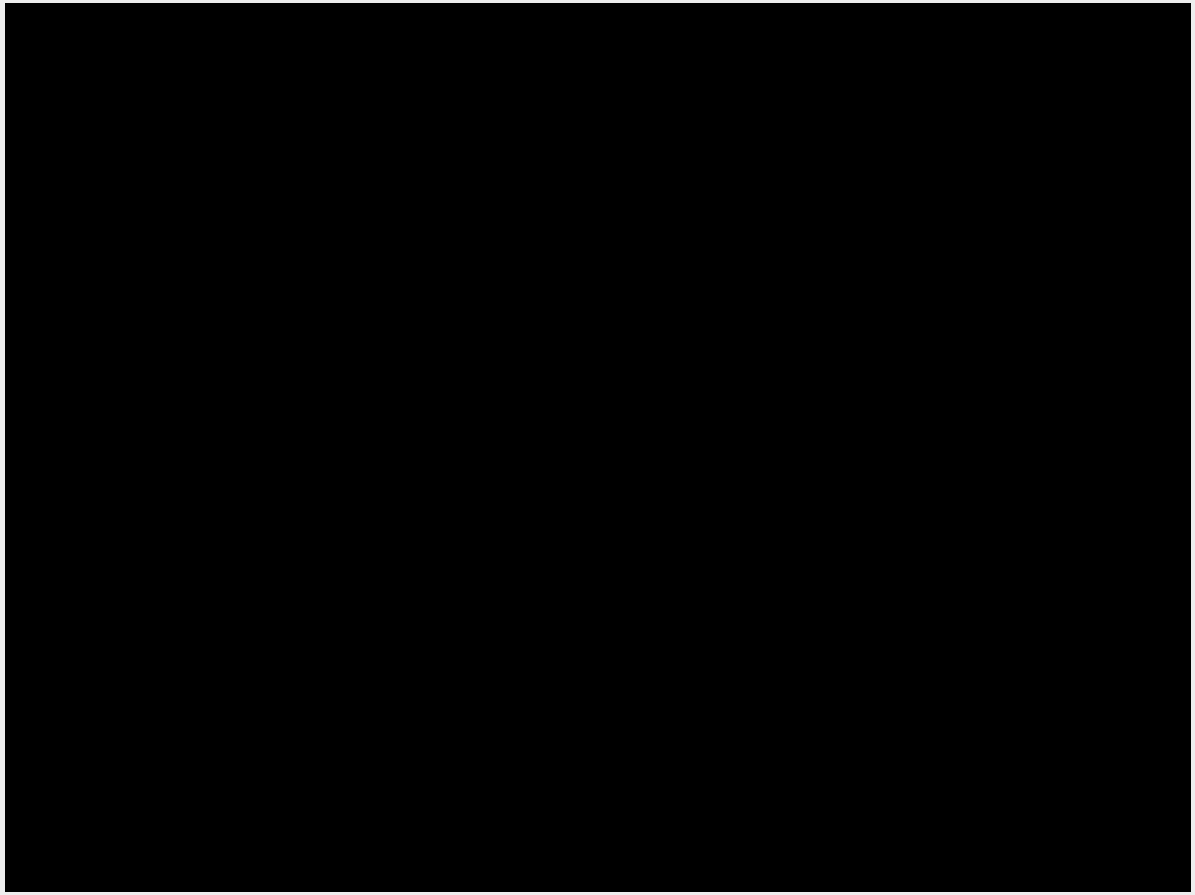
- Our stream-based model allows local coordination, based on a character's *incentive*
  - Deviation from the local flow
  - Local density
  - Internal motivation
  - Spent time to reach goal



Adapting the routes:
Unification of individual and collective movements

- Our stream-based model allows local coordination, based on a character's *incentive*
  - Deviation from the local flow
  - Local density
  - Internal motivation
  - Spent time to reach goal

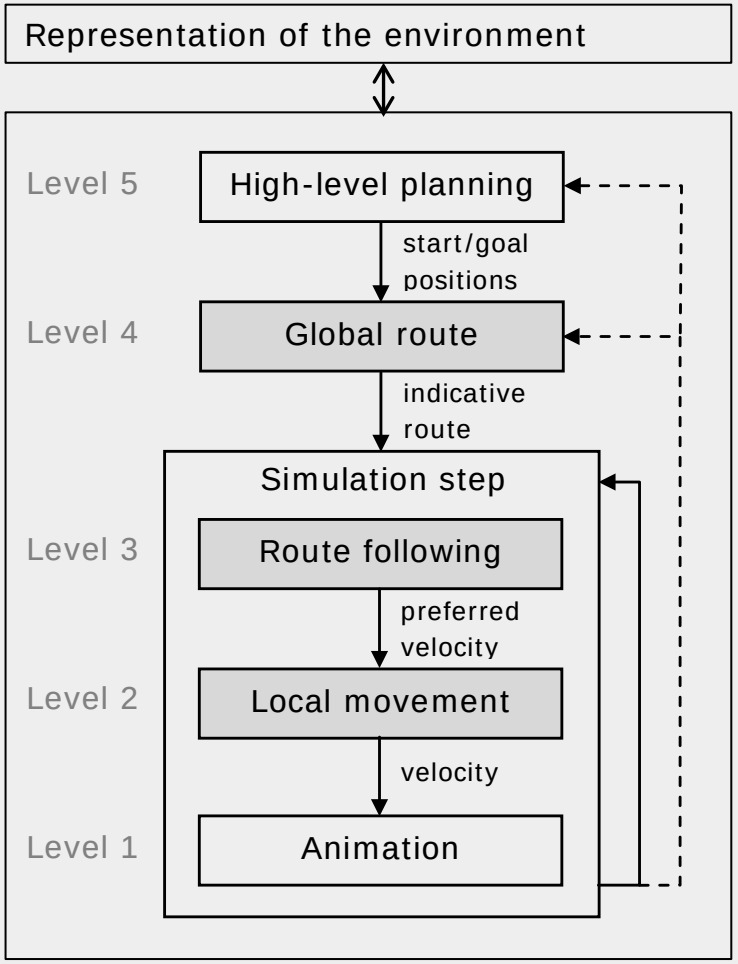
So what is realistic collision avoidance?



Rush Hour; by Fernando Livschitz

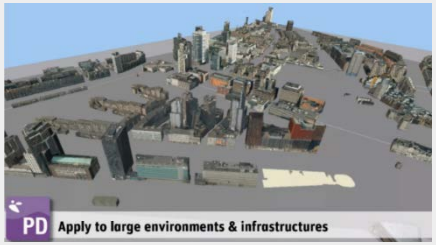
Crowd simulation framework

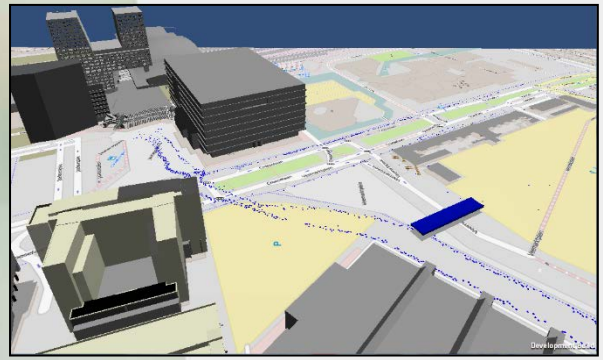
- Representation environment
- Level 5
  - Plans actions
- Level 4
  - Creates indicative routes
- Level 3
  - Traverses the routes
  - Yields speed/direction pairs
- Level 2
  - Adapts routes
  - E.g. to avoid collisions
- Level 1
  - Moves the characters

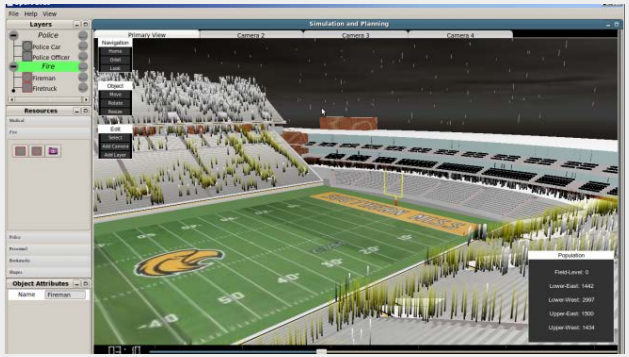


Software package

- Freely available for researchers
- Simulates 65K characters in real-time
- Simulates 1 million people on 1 PC
- Already used in the industry
  - GreenDino (Car simulator)
  - InControl (Pedestrian dynamics, SportEvac)
  - Movares (Reach)
    - Improving the crowd flow of Grand Départ
    - Evacuation studies Noord/Zuidlijn
    - Crowd flows in the Apenheul







Open research questions

- Validation of the models
- Simulation of 1.000.000 people in real-time
- Real-time prediction during a real-time event

Messages

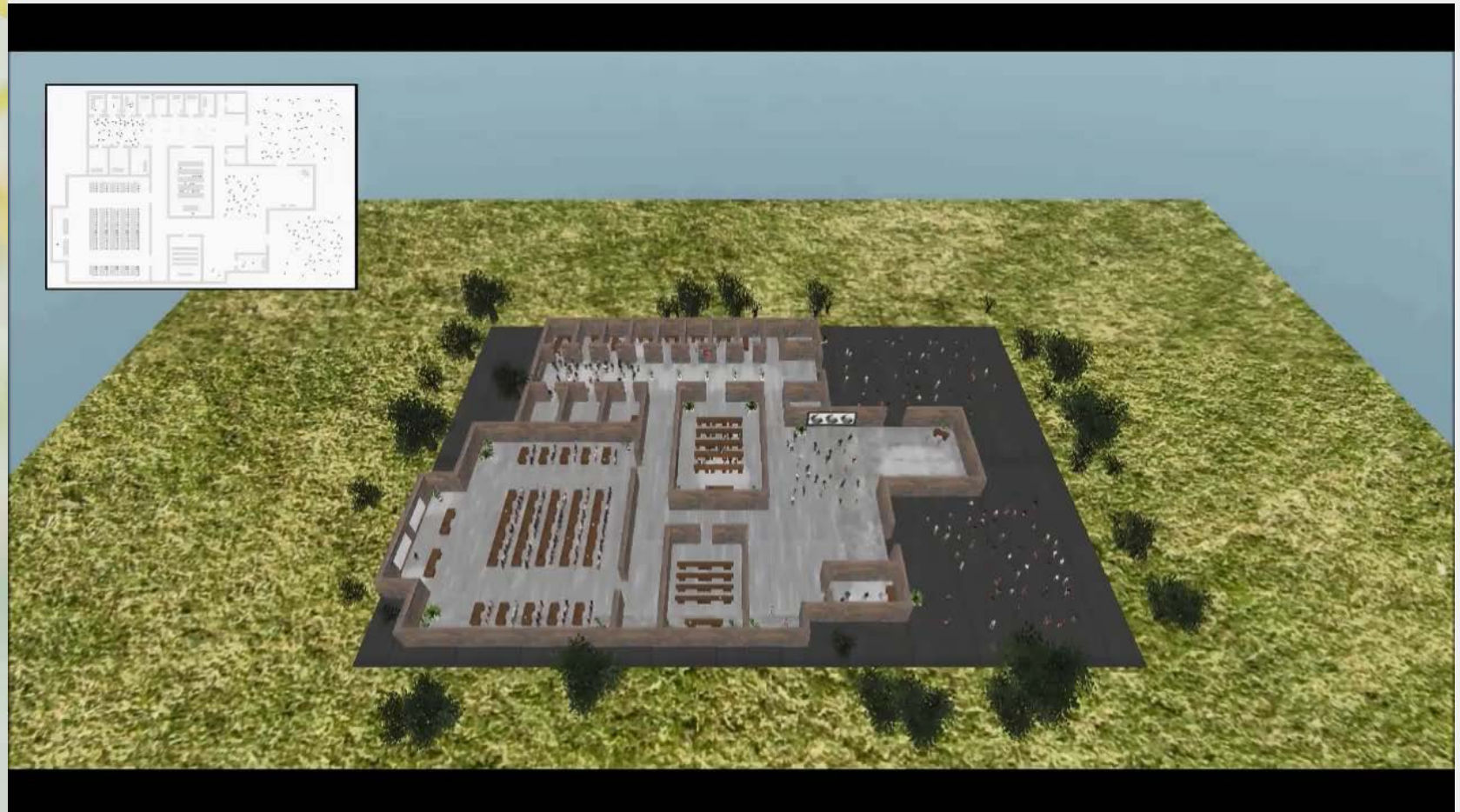
- For efficiently and flexibly simulating crowds, we need
  - a generic and efficient representation of the navigable areas;
  - a framework of (at least) 5 complexity levels.
- Methods must be compatible with surface-based navigation at all levels (paradigm shift!)
  - so a graph-based approach is not going to be sufficient
- A path planning algorithm should not compute a path
- Our simulation software is freely available for researchers

List of contributors

- Staff
  - *Roland Geraerts*
  - Marjan van den Akker
  - Han Hoogeveen
  - Frank van der Stappen
  - Mark Overmars
  - Marc van Kreveld
- PhD students
  - Arthur van Goethem
  - Norman Jaklin
  - Ioannis Karamouzas
  - Wouter van Toll
  - Arne Hillebrand
- MSc students
  - Angelos Kremyzas
  - Peter Prins
  - Simon Rosman
  - Marijn van der Zwan
  - Martijn Koenis
  - Mihai Polak
- Scientific programmer
  - Angelos Kremyzas
- Companies
  - Movares
  - GreenDino
  - InControl

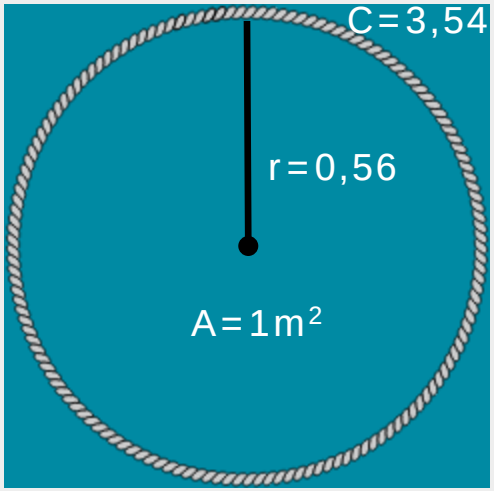
Crowd management

- Collect much information, study many scenario's…



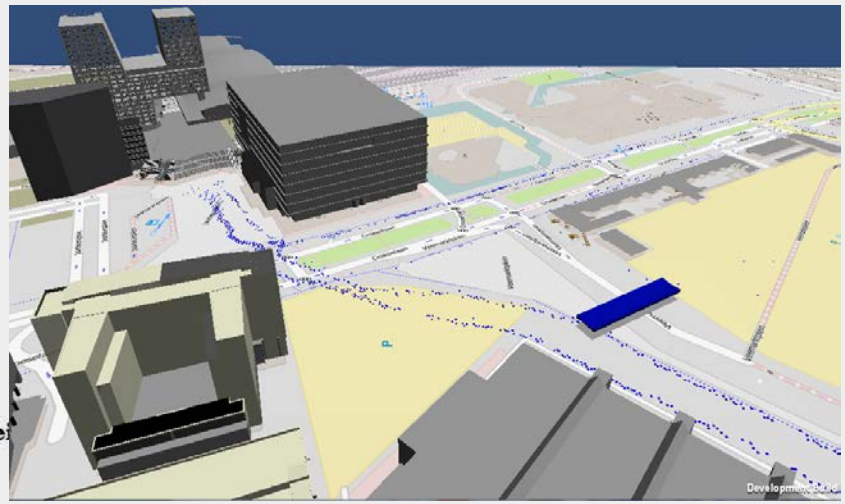
Crowd management

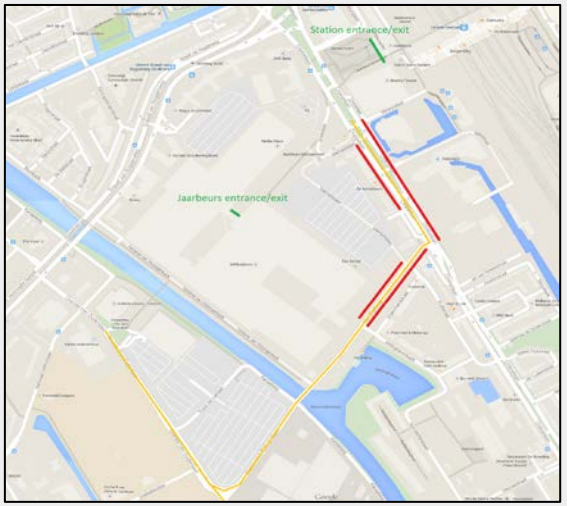
- …*measure*, and act.
- But what should we measure?
- It's time for an experiment!



Crowd management

- Le Tour demo of one scenario
  - Around 75% of the visitors are expected to come from the train station
  - Many flows
    - From train station to Jaarbeurs (±100.000)
    - From train station to viewing area along the stage route (±250.000)
    - From Jaarbeurs to train station (±20.000)
    - From Jaarbeurs to viewing area along the stage route (±80.000)





Contact

- Roland Geraerts, R.J.Geraerts@uu.nl
- http://www.staff.science.uu.nl/~gerae101/