Pathfinding and Routing
NAIL137

Pathfinding and crowd simulation in games

Jiří Švancara

FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

Pathfinding in games

- Requirements
  - Fast enough
  - Short paths
  - Reasonable paths
- Stakes
  - ☺ No broken hardware
  - ☹ Lagging games
  - ☹ Bad gaming experience

Environment - 2D grid

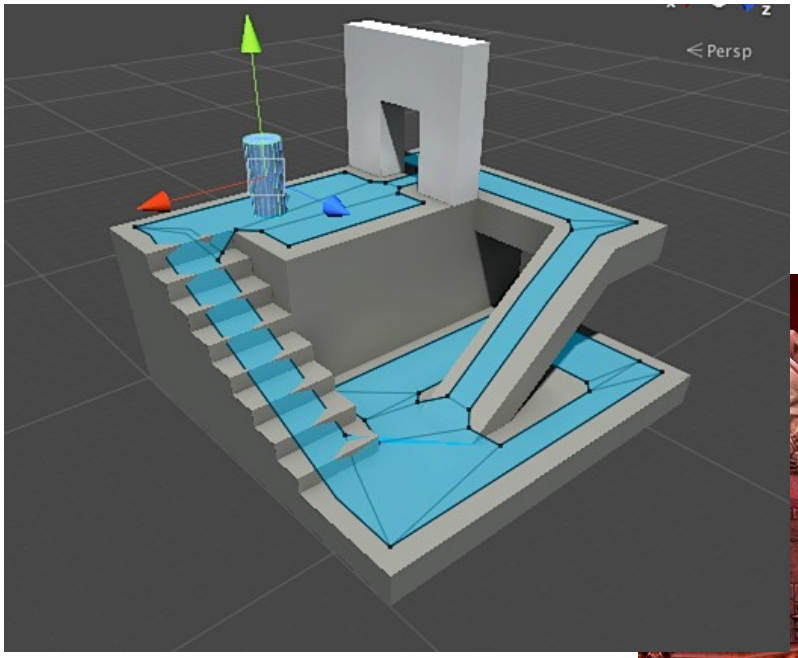
Environment - skewed 2D grid

https://www.youtube.com/watch?v=ouctGk-vf0c

Environment - hexagons (bestagons)

Environment - mesh

- Area is split into convex polygons

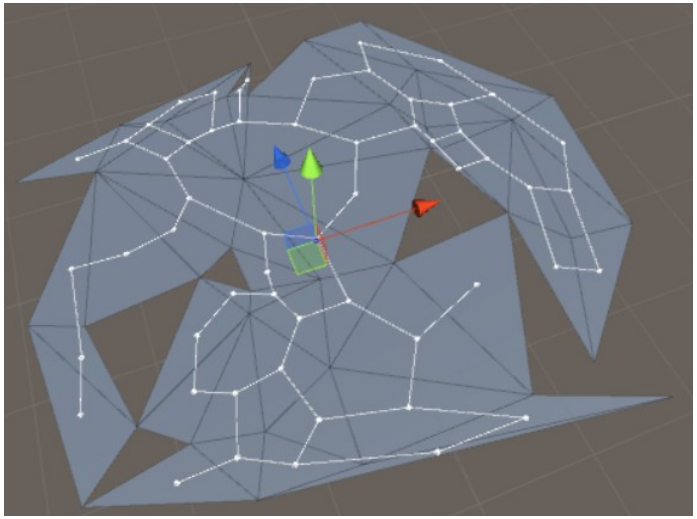


Single agent navigation - grid

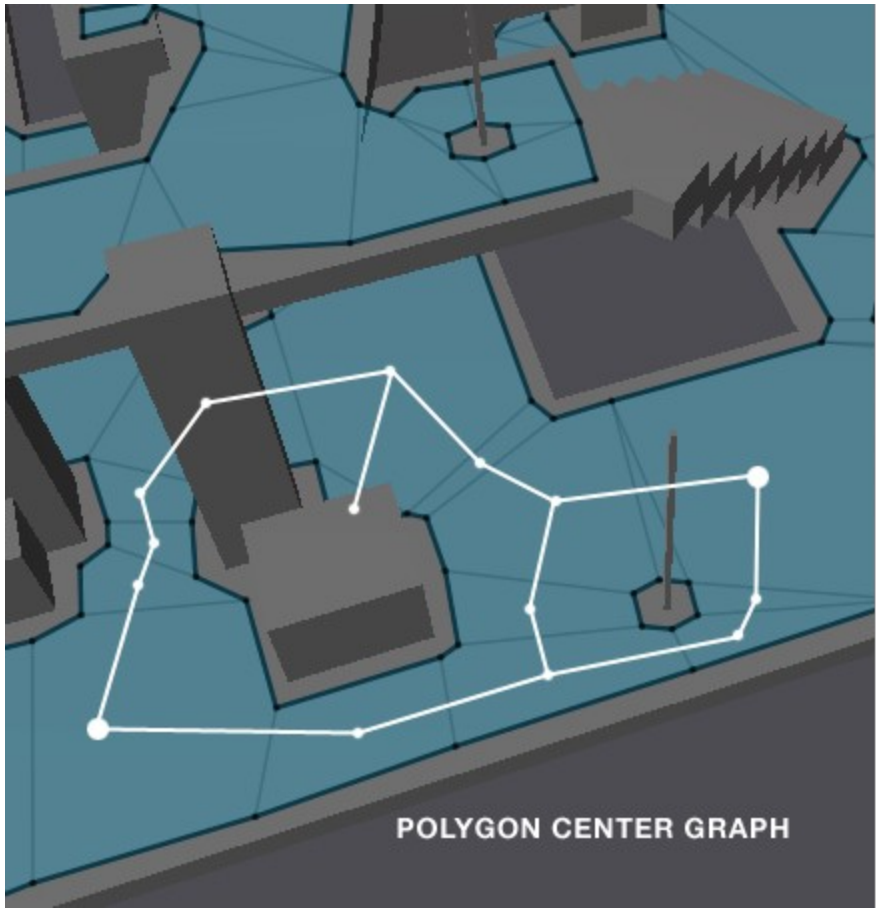
- (weighted) A*

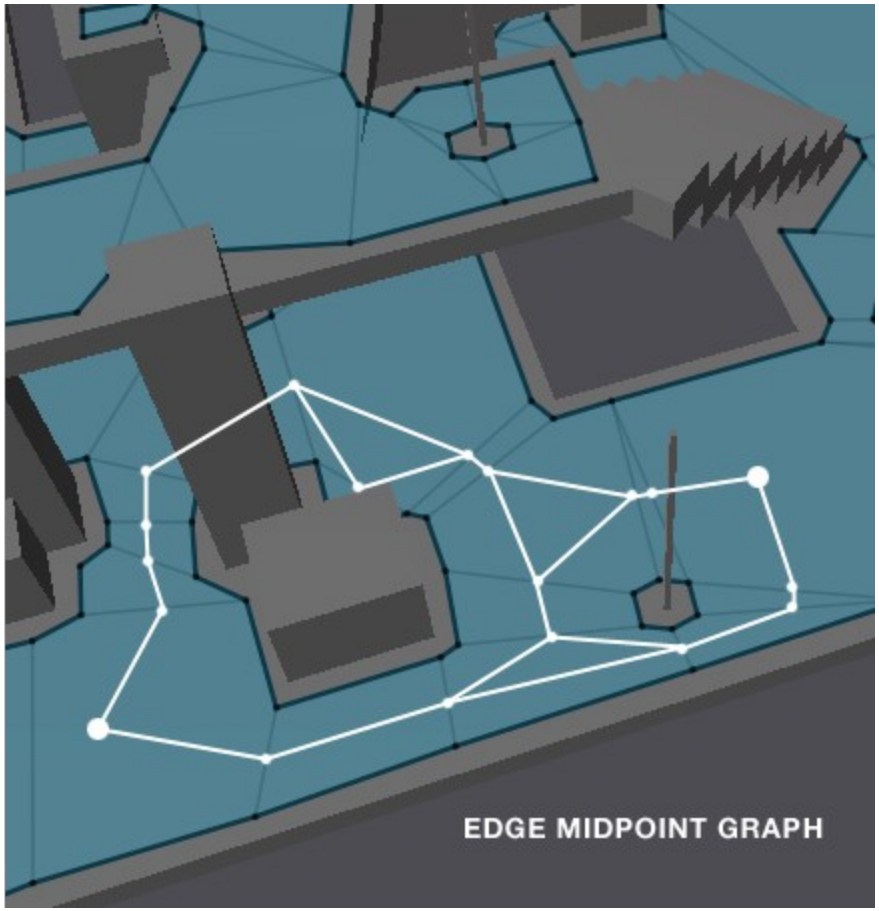
Single agent navigation - navmesh

- Area is a node (center of the area)
- Connect neighboring nodes (weights by true distance)
- Inside area, go in straight line
- Move to different area via centers (A*)



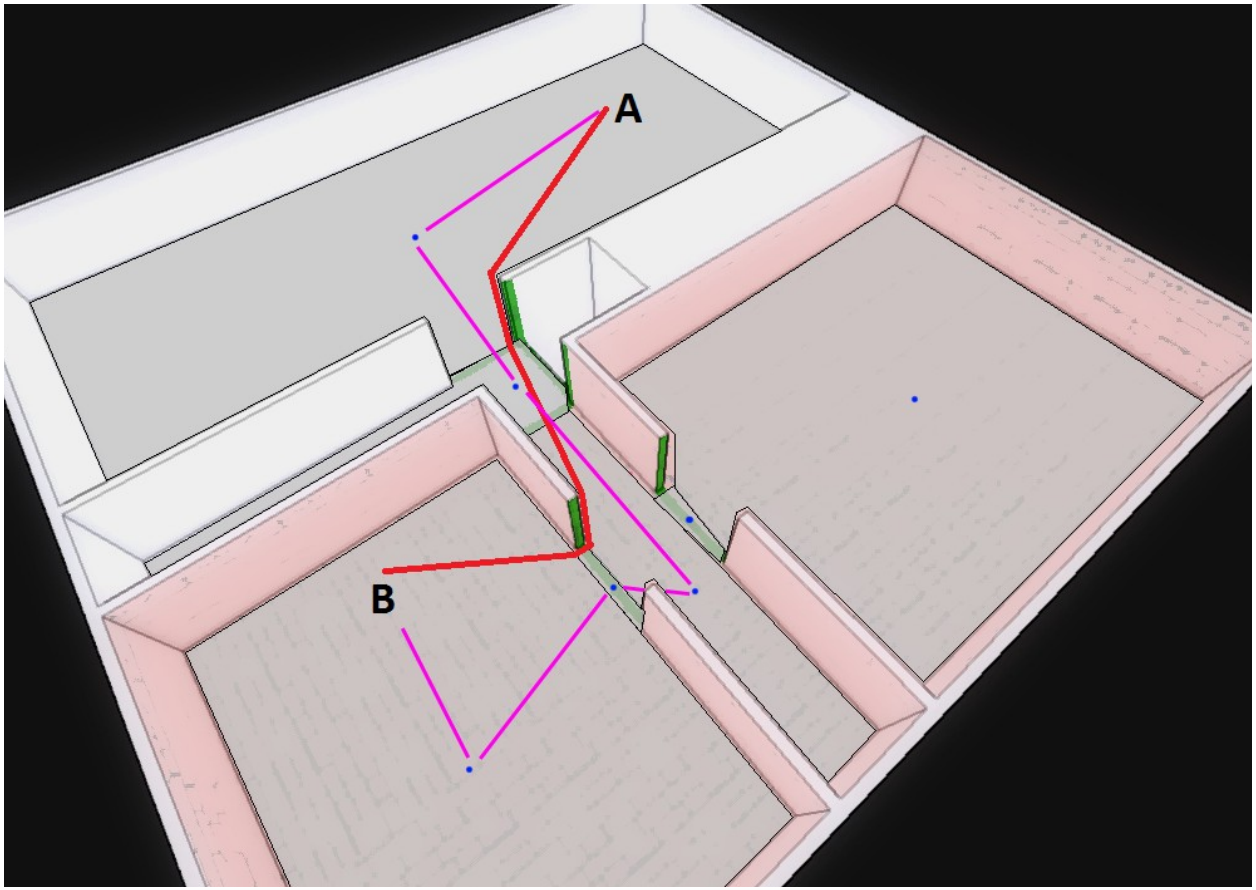
Navmesh alternative connection





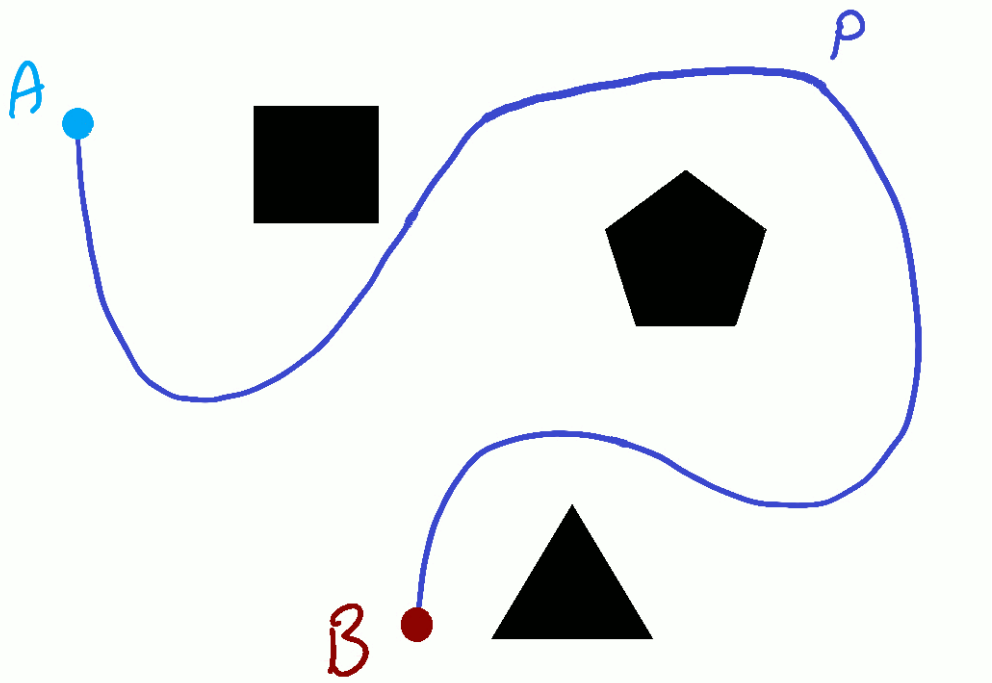
Navmesh improve found paths

- String pulling (funnel algorithm)



Funnel algorithm

- Sequence of edges to cross
- Move the crossing point left/right while not hitting obstacles



https://medium.com/@reza.teshnizi/the-funnel-algorithm-explained-visually-41e374172d2d

Handling more agents

1) Single agents only (collisions are ok)
2) Multiple single agents, refine paths when problems arise
3) Crowd simulation - flocking

Single agents with collisions

https://www.youtube.com/watch?v=LAis1rKtT3A

Single agents with collisions + formations

https://www.youtube.com/watch?v=qI10g3Qmwac

Avoid collision by refining paths

1) Each unit plans path by A* (avoid known obstacles)
2) If the next square is empty, move
3) If it is occupied, find new path

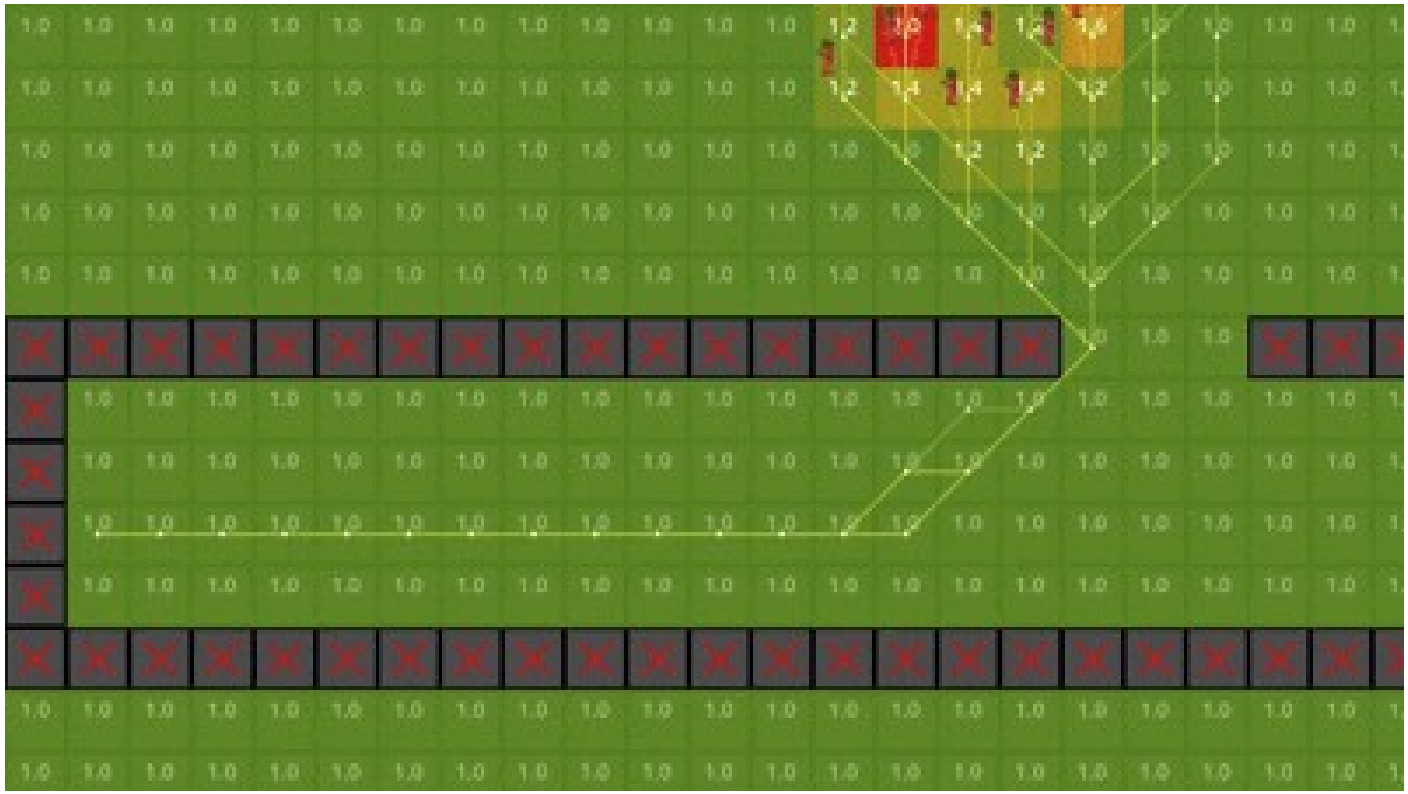
Known improvements

- Do not replan immediately, friendlies are probably moving as well
- If friendlies are not moving, “wiggle” them to allow through

https://www.youtube.com/watch?v=S-VAL7Epn3o

Known improvements

- Nodes with movements are more expensive
- Change state of the moving unit



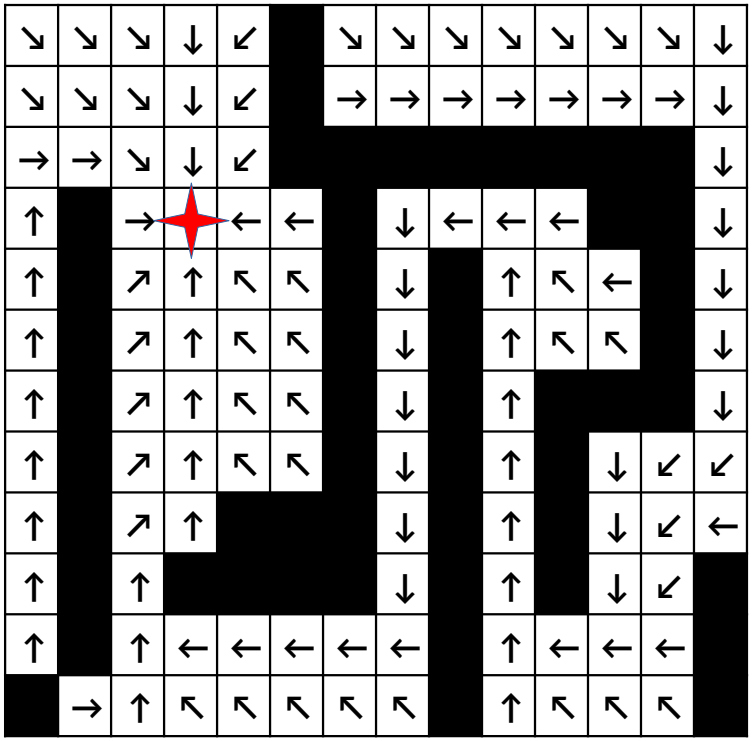
https://www.youtube.com/watch?v=fVPRW1XHd8g

Notoriously bad example

https://www.youtube.com/watch?v=jJ7URDgCza8
https://www.youtube.com/watch?v=24134OFfvFI

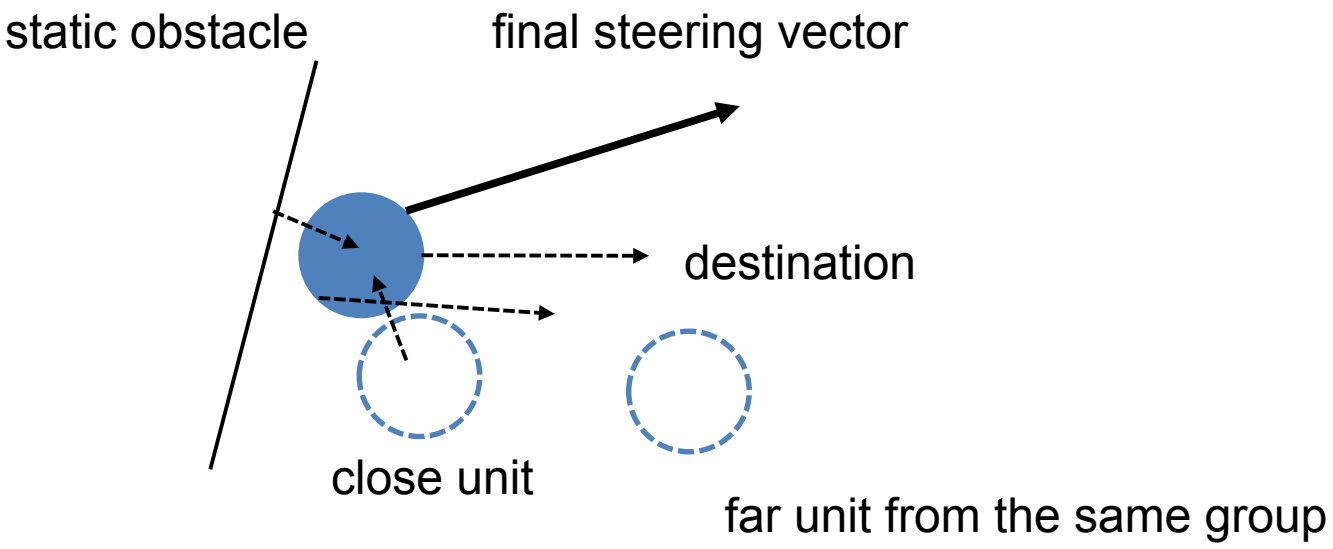
Flow field (goal-based) pathfinding

- No need to calculate for each unit independently
- The goal is the same for all!



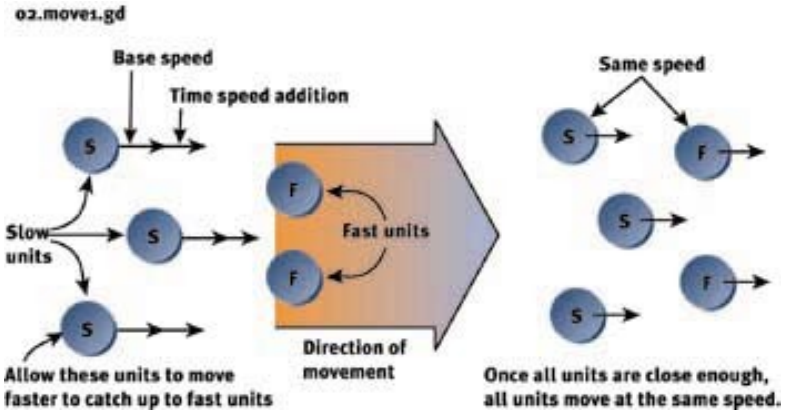
Crowd simulation - flocking

- Agents need to move smoother, together, but not bump into each other
- The environment is „continuous“
- Use navigation mesh or way points
- Consider only small surrounding area

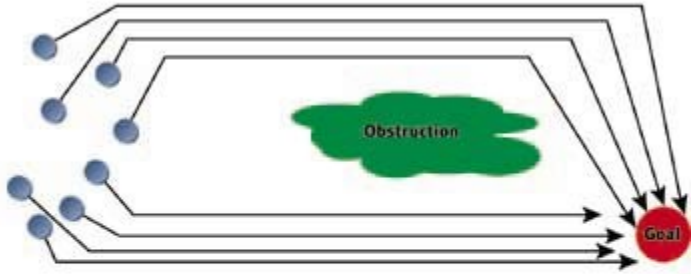


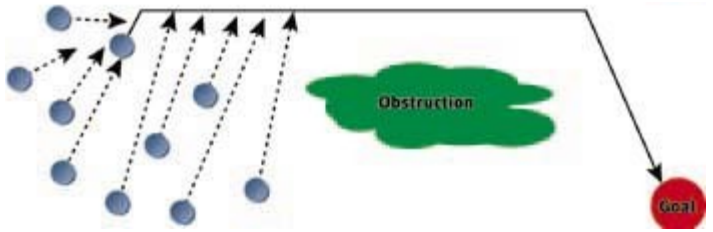
Flocking considerations

- Agents travel the same speed



- Agents take the same path





https://ap011y0n.github.io/Group-Movement/

Flocking

https://www.youtube.com/watch?v=HWRfm8ocl3g

Crowd simulation - issues

https://www.youtube.com/watch?v=fGikLWjkE3A