Configuration Space

- Any admissible motion for the 3D mechanical system appears a collision-free path for a point in the CSpace
- Translating the ***continuous*** problem into a ***combinatorial*** one
- Capturing the ***topology*** of *CSfree* with ***graphs***

- The Piano Mover problem

- 80’s The complexity bottleneck

- 90’s The amazing efficiency of empirism

- **The Piano Mover problem: probabilistic approaches**
  - **Gradient descents and random walks**
  - **Sampling methods: PRM**
  - **Diffusion algorithms: RRT**

Probabilistic Motion Planning











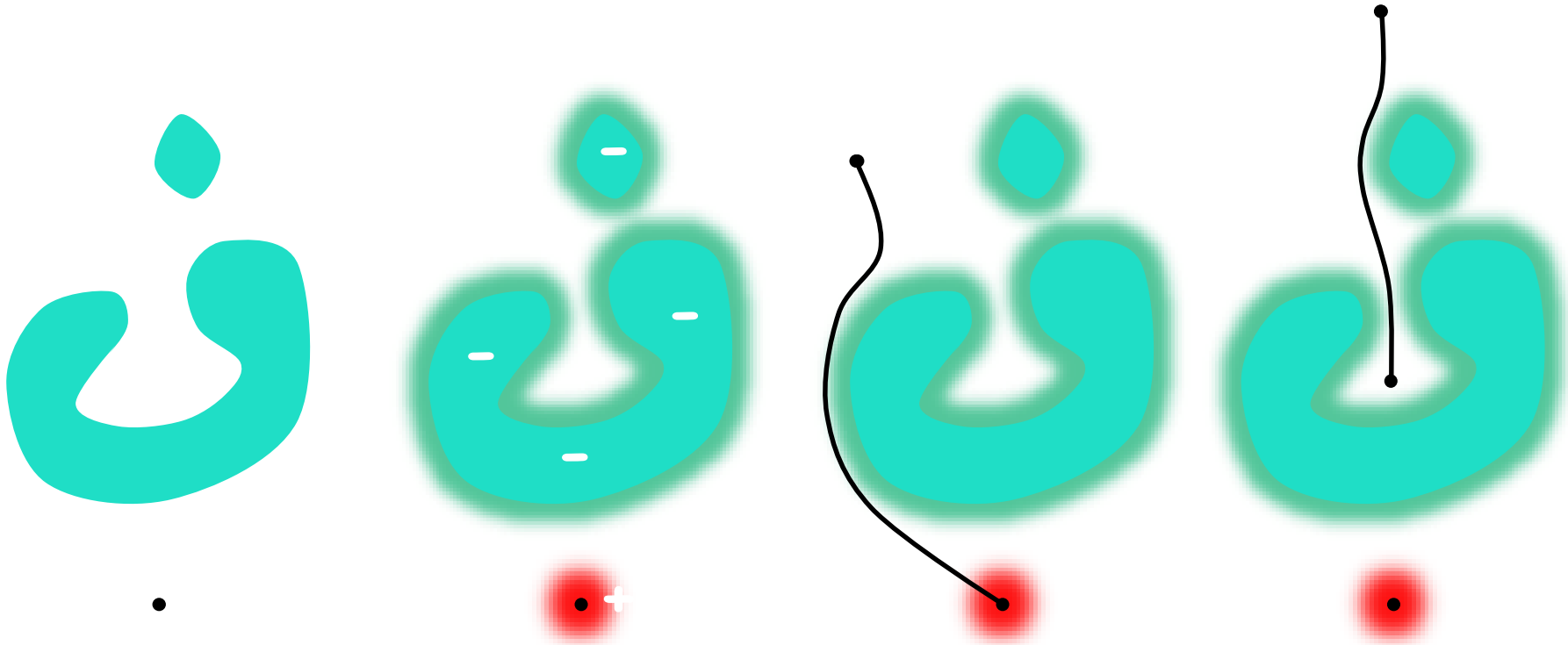




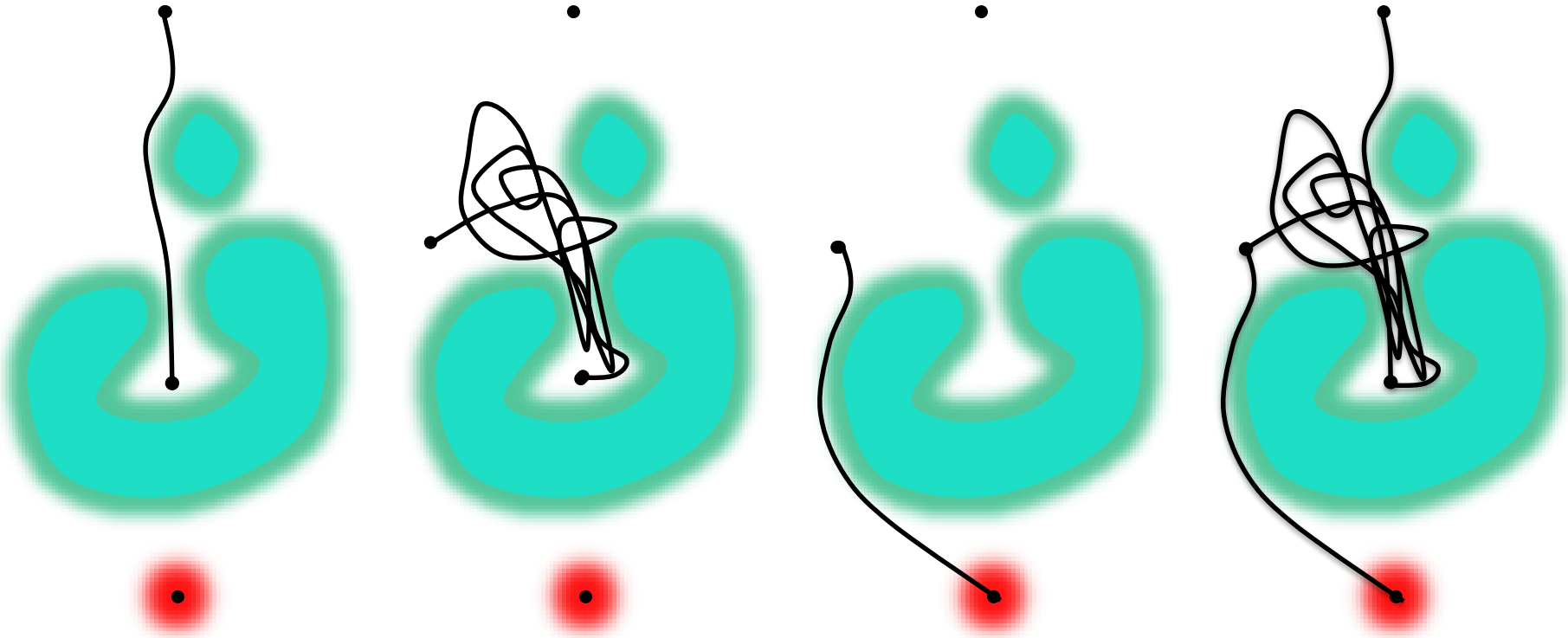


- **The Piano Mover problem: probabilistic approaches**

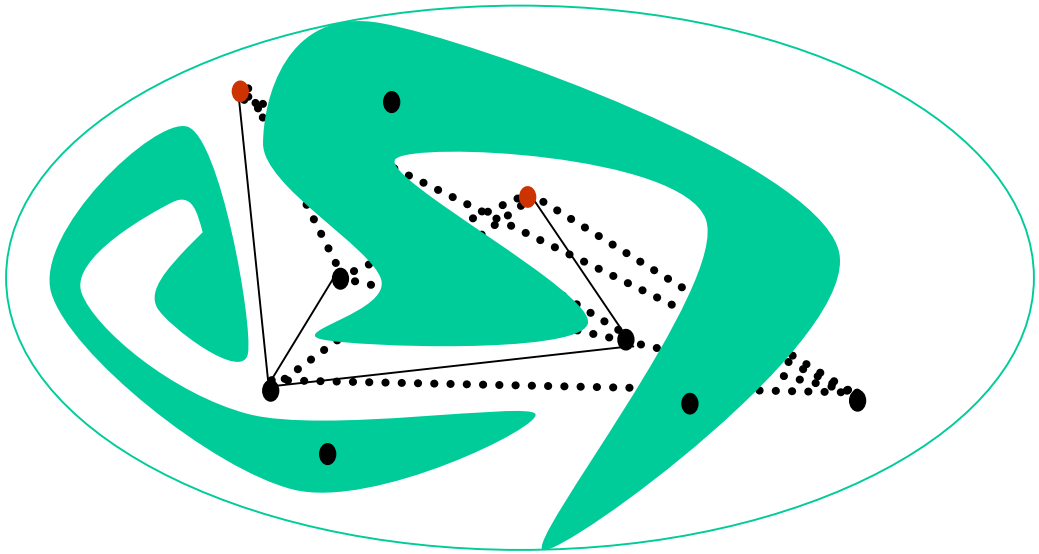
- **Gradient descents and random walks**
- **Sampling methods: PRM**
- **Diffusion algorithms: RRT**



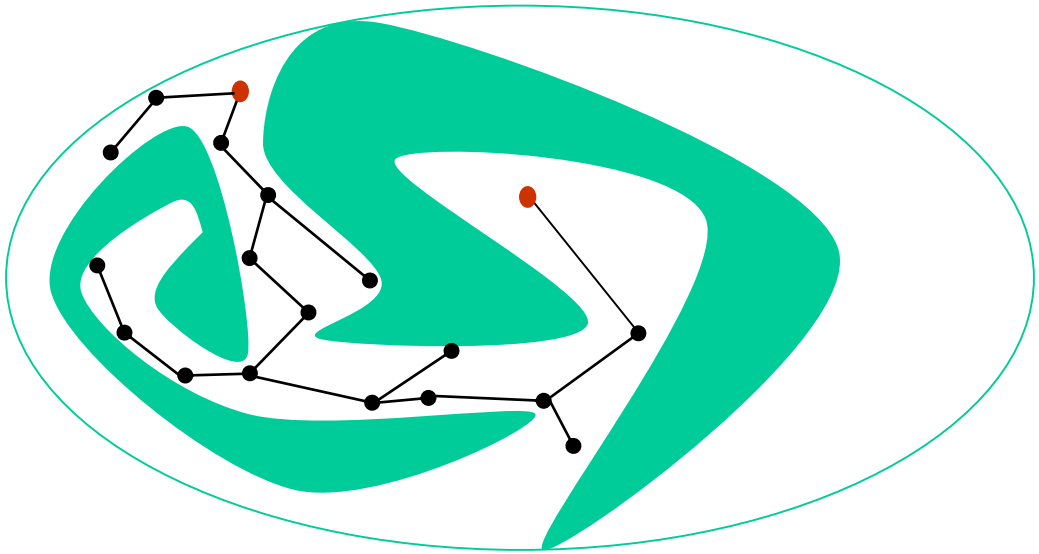
Probabilistic Motion Planning



- **The Piano Mover problem: probabilistic approaches**
- Gradient descents and random walks
- **Sampling methods: PRM**
- Diffusion algorithms: RRT



- **The Piano Mover problem: probabilistic approaches**
- Gradient descents and random walks
- Sampling methods: PRM
- **Diffusion algorithms: RRT**



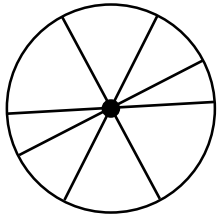
Combinatorial Topology and Visibility based Probabilistic Methods in Motion Planning

T. Siméon, J.P. Laumond, C. Nissoux,
Visibility based probabilistic roadmaps for motion planning
*Advanced Robotics Journa*l, Vol. 14, N° 6, 2000

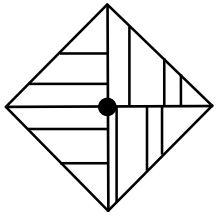
J.P. Laumond, T Siméon,
Notes on visibility roadmaps and path planning
in *Algorithmic and Computational Robotics: New Directions*
B. Donald, K. Lynch, D. Rus Eds, A.K. Peters, 2001

- Configuration Space Topology and Small Time Controllability

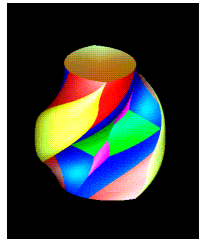
- Exemples:
  - linear interpolation,
  - Manhattan paths,
  - Reeds and Shepp paths,
  - flatness based methods



Euclidean

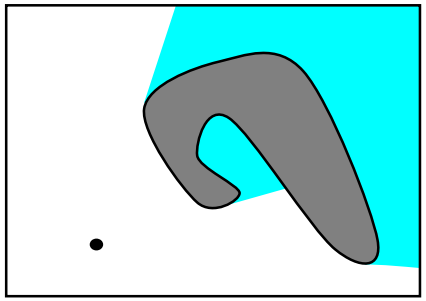


Manhattan

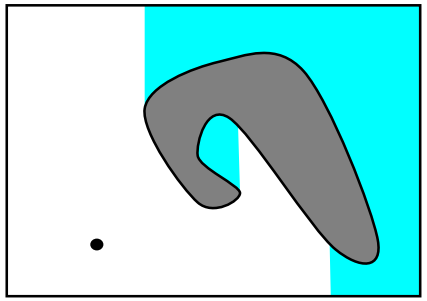


Reeds-Shepp

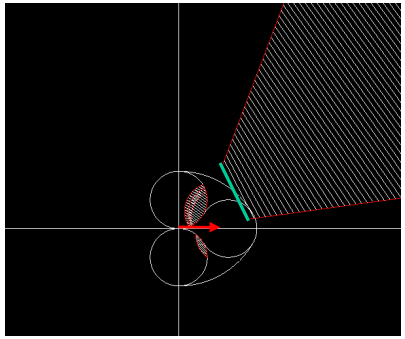
- Same topology, different visibility sets



Euclidean



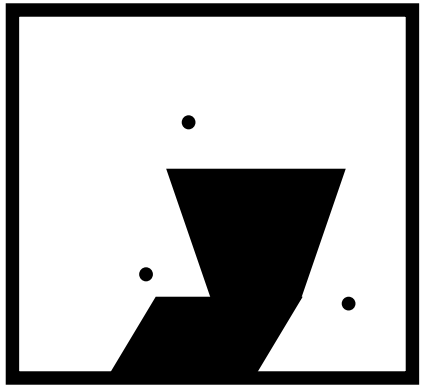
Manhattan



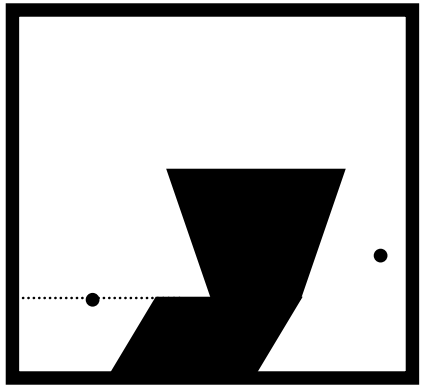
Reeds-Shepp

- Combinatorial topology (1)

- Existence and robustness of finite coverage



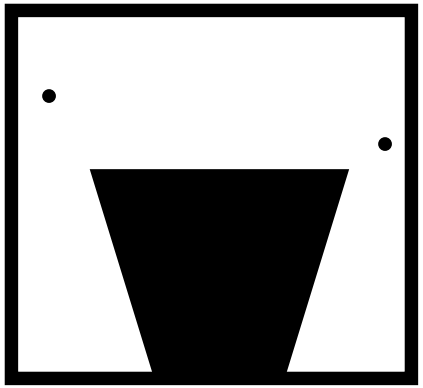
Euclidean



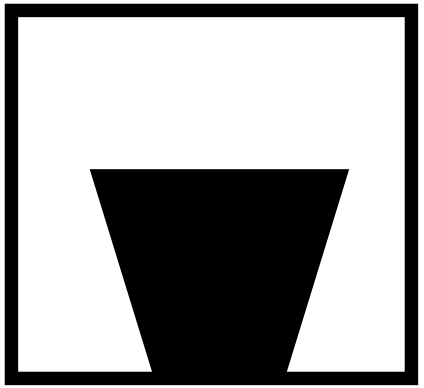
Manhattan

- Combinatorial topology (1)

- Existence and robustness of finite coverage



Euclidean



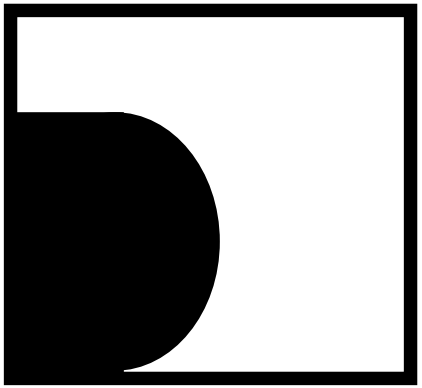
Manhattan

No

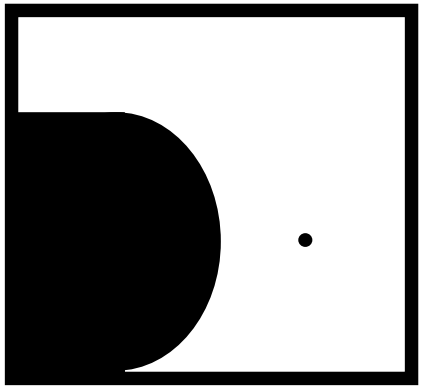
- Combinatorial topology (1)

- Existence and robustness of finite coverage

No



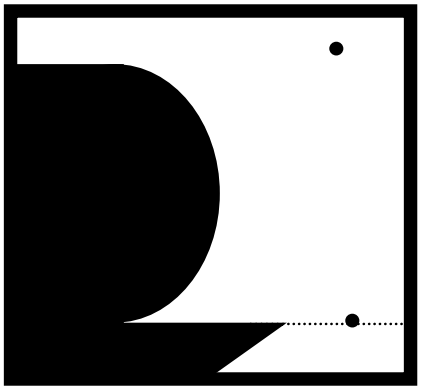
Euclidean



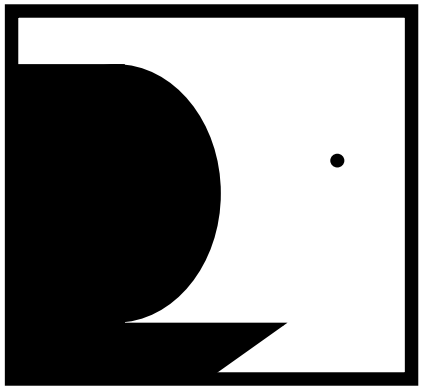
Manhattan

- Combinatorial topology (1)

- Existence and robustness of finite coverage



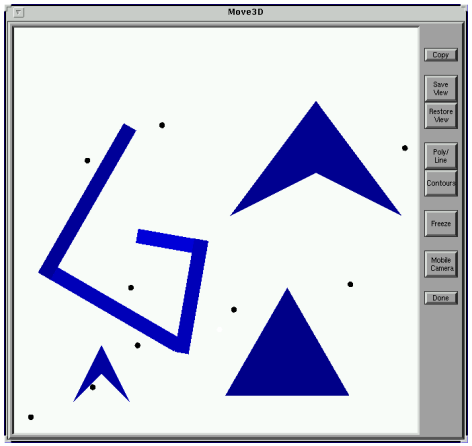
Euclidean



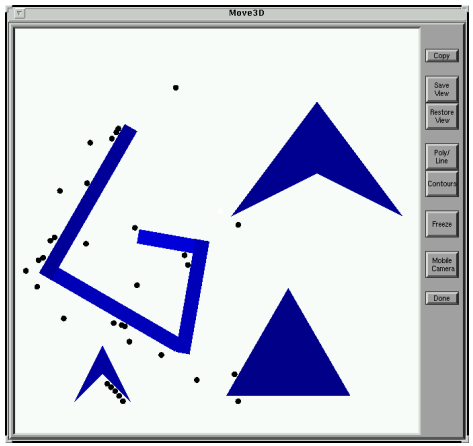
Manhattan

- Combinatorial topology (2)

- Optimal coverage (related to art gallery problem)



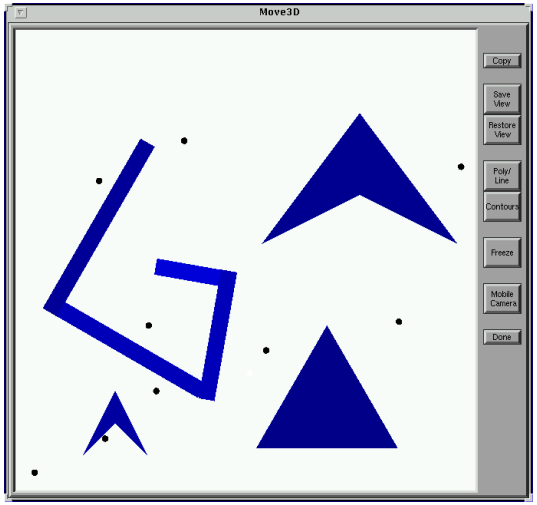
Euclidean



Manhattan

- Combinatorial topology (2)

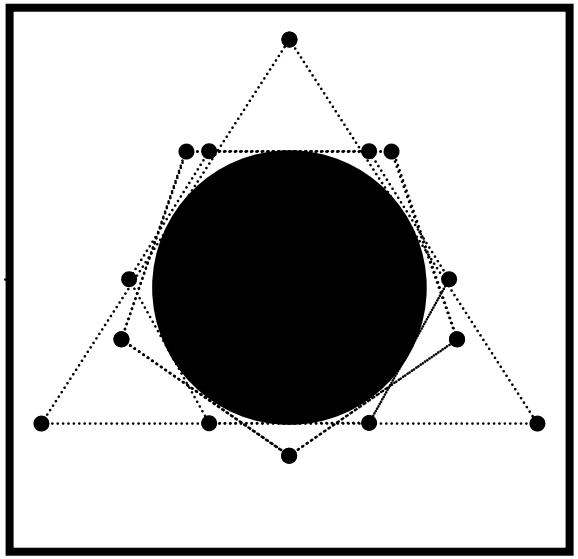
- Optimal coverage: finite? bounded?



Euclidean and 2D polygonal obstacles: finite and bounded

- Combinatorial topology (3)

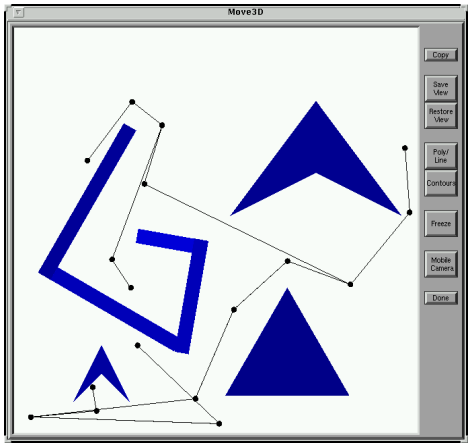
- Optimal coverage: finite? bounded?



Euclidean: finite and unbounded

- Combinatorial topology (4)

- From (optimal) coverage to (visibility) roadmaps



Guards + Connectors

- Computational challenge

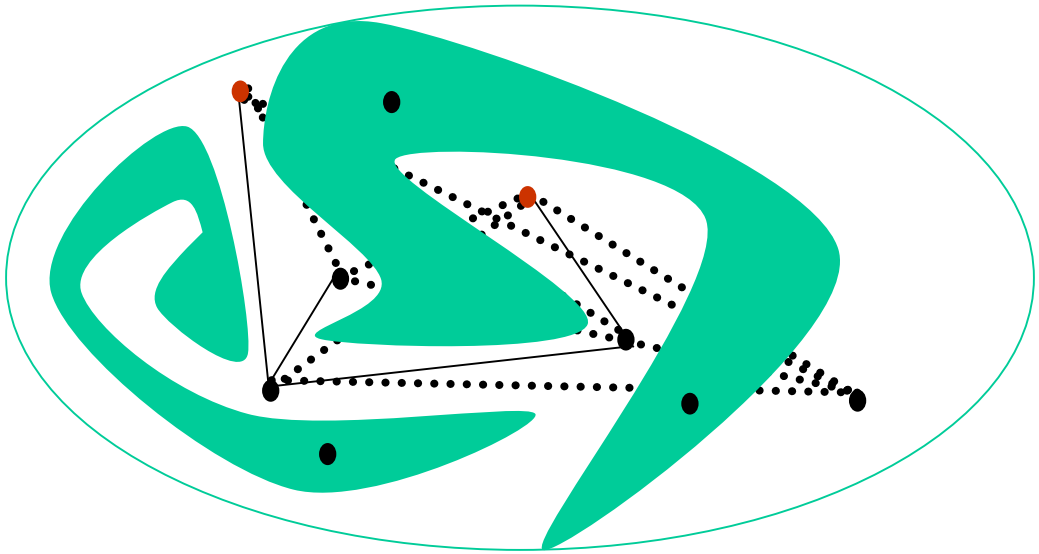
- No explicit knowledge of *CS*-obstacles
- No explicit knowledge of visible (reachable) sets

- Computational challenge

- Probabilistic method ingredients:
  - A collision checker
  - A steering method

- Two type methods:
  - Learning *CS* topology by *sampling*
  - Answering single query by *diffusion*

Computational challenge

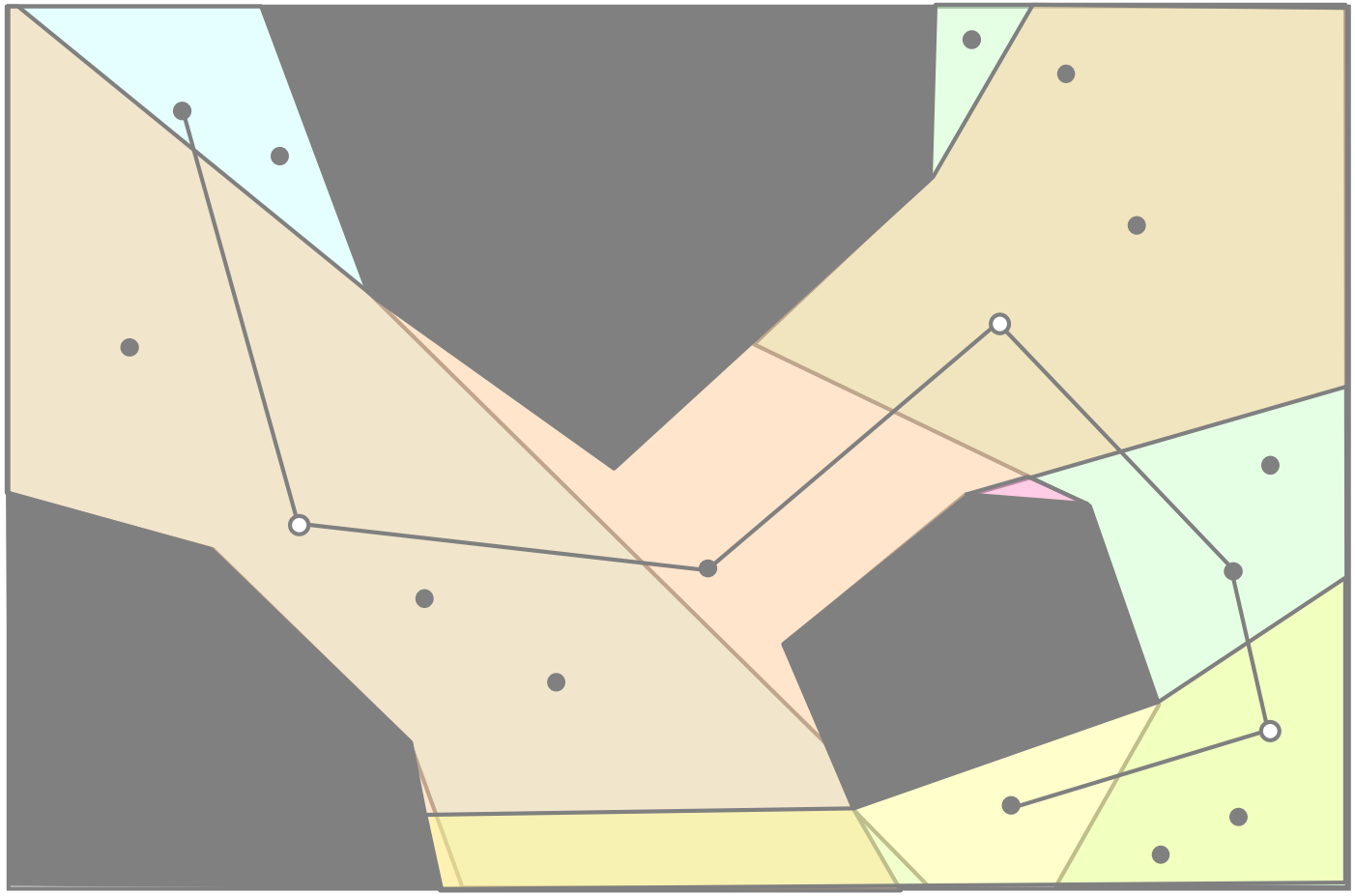


Random sampling

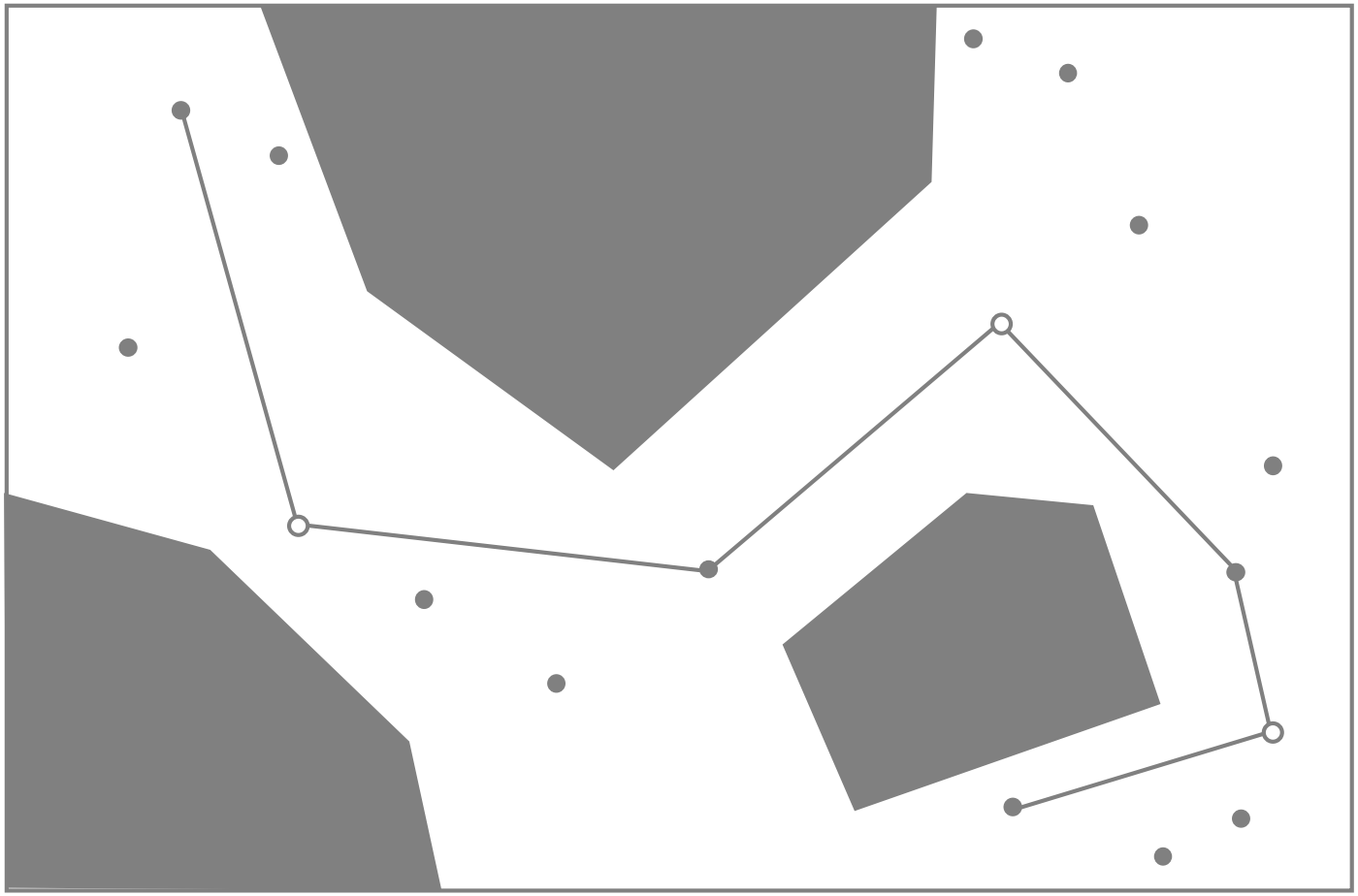
- Visibily based sampling

- Algorithm:
  - Generate guards and connectors randomly
  - Stop after *#try* failures

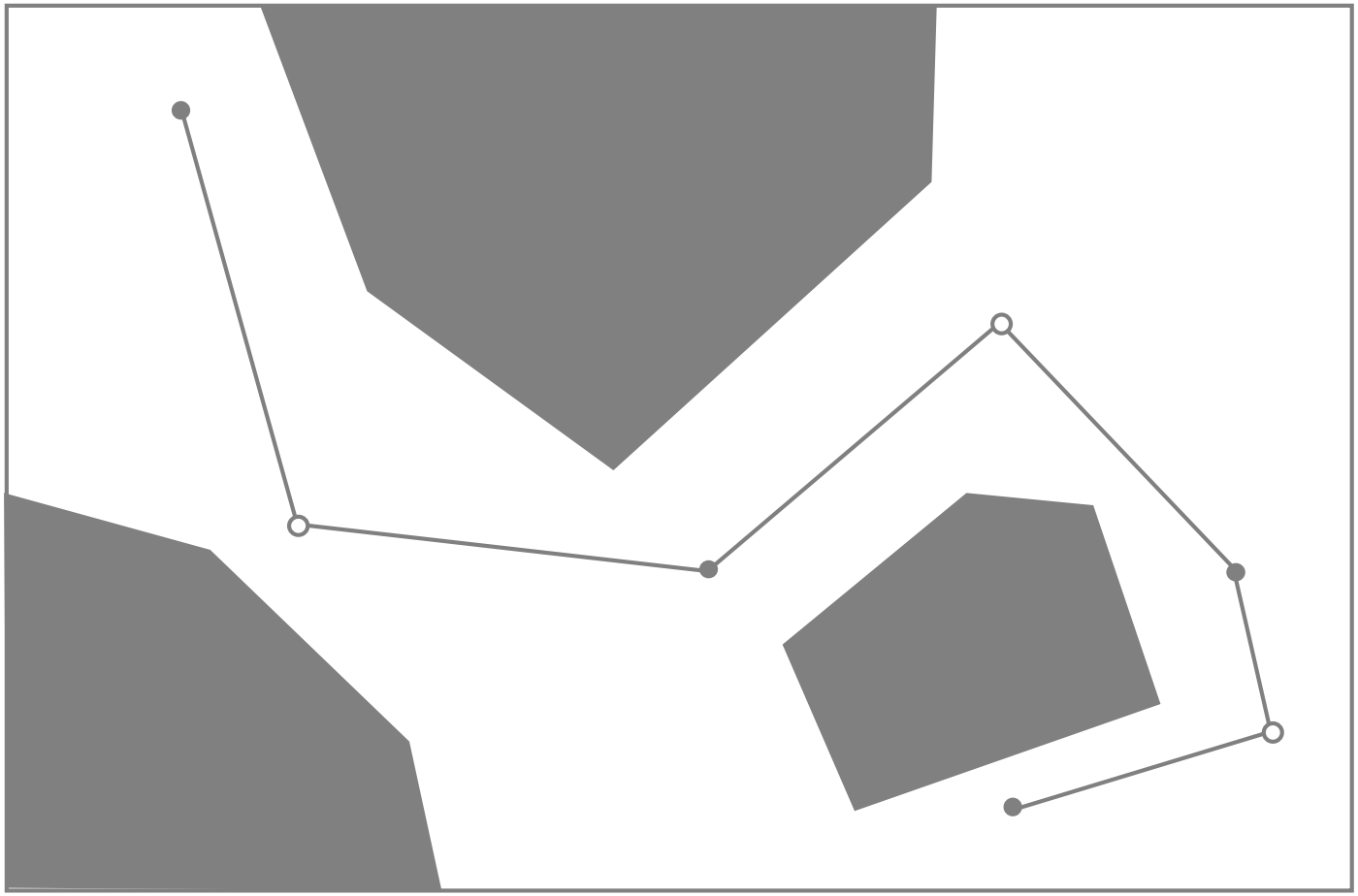
Visibily based sampling



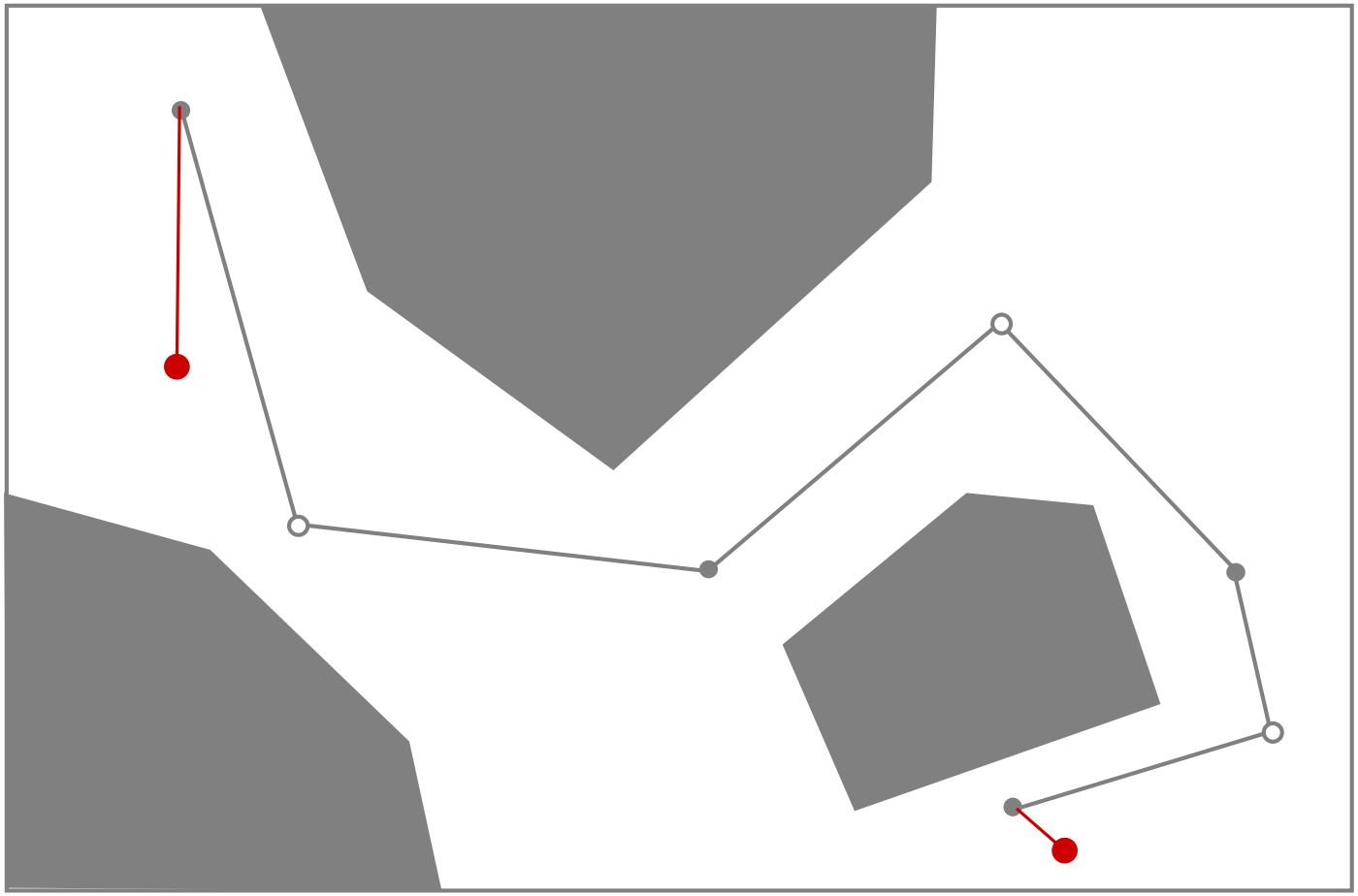
Visibily based sampling



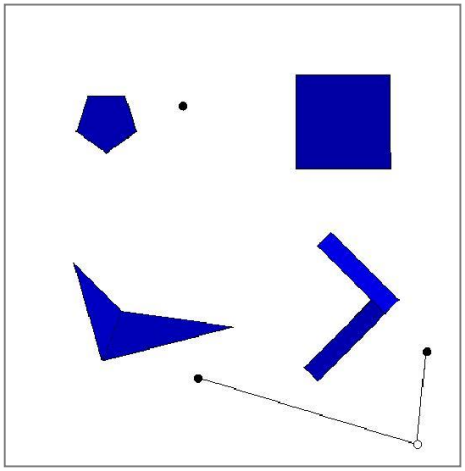
- Visibily based sampling



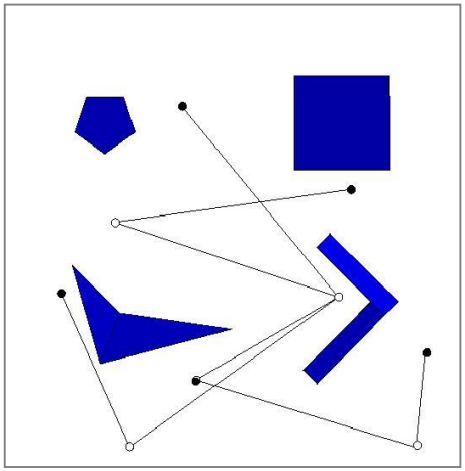
Visibily based sampling



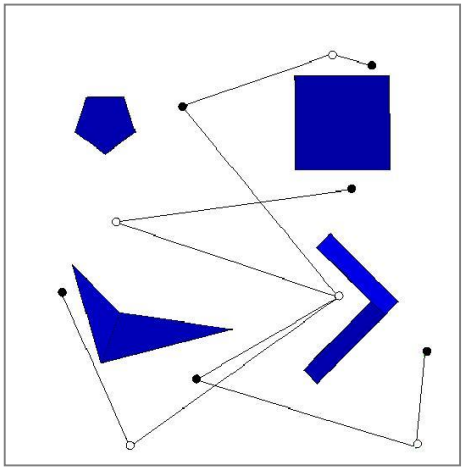
- Visibily based sampling
- Algorithm:
  - Generate guards and connectors randomly
  - Stop after *#try* failures (*#try=10)*



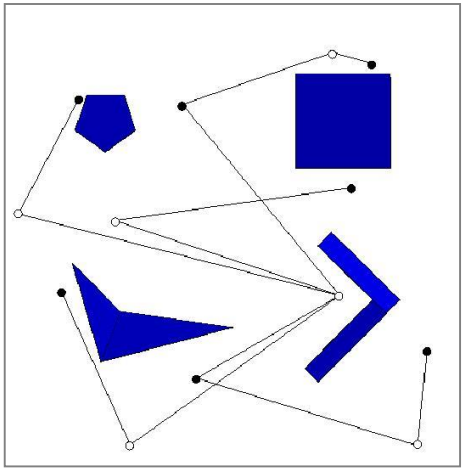
- Visibily based sampling
- Algorithm:
  - Generate guards and connectors randomly
  - Stop after *#try* failures (*#try=100)*



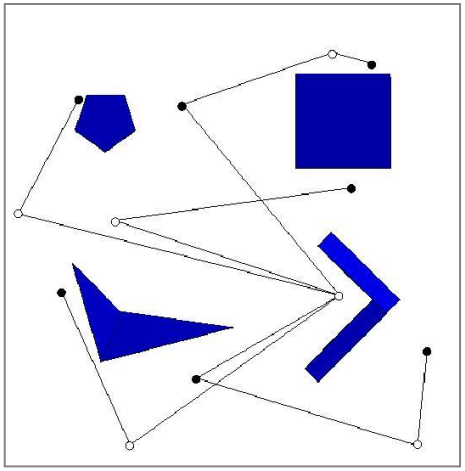
- Visibily based sampling
- Algorithm:
  - Generate guards and connectors randomly
  - Stop after *#try* failures (*#try=200)*



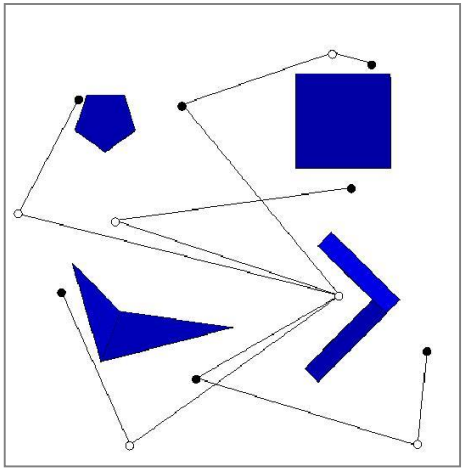
- Visibily based sampling
- Algorithm:
  - Generate guards and connectors randomly
  - Stop after *#try* failures (*#try=500)*



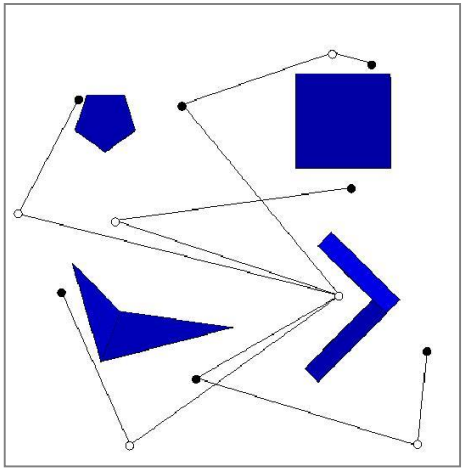
- Visibily based sampling
- Algorithm:
  - Generate guards and connectors randomly
  - Stop after *#try* failures (*#try=1000)*



- Visibily based sampling
- Algorithm:
  - Generate guards and connectors randomly
  - Stop after *#try* failures (*#try=10000)*



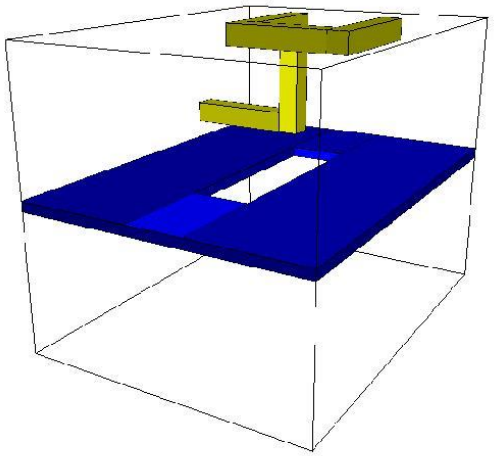
- Visibily based sampling
- Algorithm:
  - Generate guards and connectors randomly
  - Stop after *#try* failures (*#try=1000000000!)*

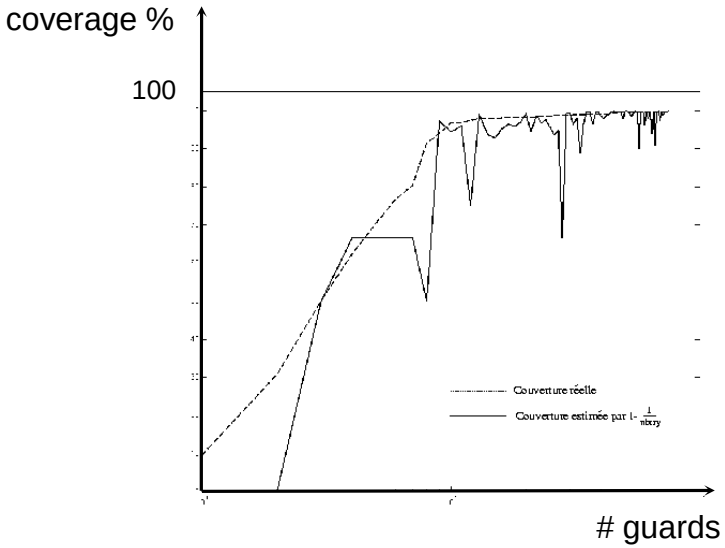


- Visibily based sampling
- Algorithm:
  - Generate guards and connectors randomly
  - Stop after #*try* failure
- Theorems:
  - The estimated percentage of non-covered free-space is $\#try^{-1}$
  - Probability to find an existing path increases as an exponential function of time

- Visibily based sampling

- Possible online estimation of *#try*





- Visibily based sampling

- Real time demonstrations