Partial Searchlight Scheduling is Strongly PSPACE-complete

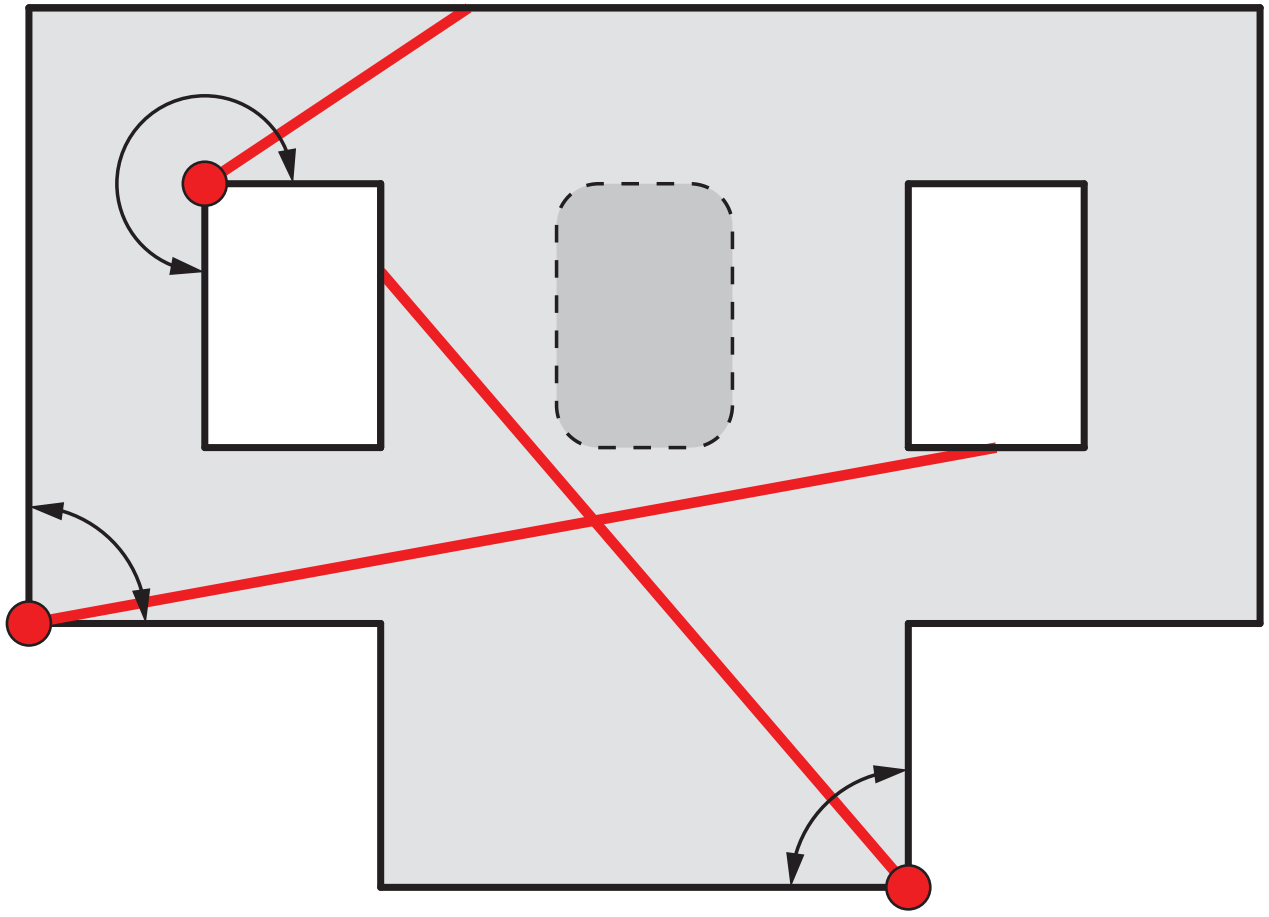
Giovanni Viglietta

Department of Computer Science, University of Pisa, Italy

Assisi - March 19$^{\text{th}}$, 2012

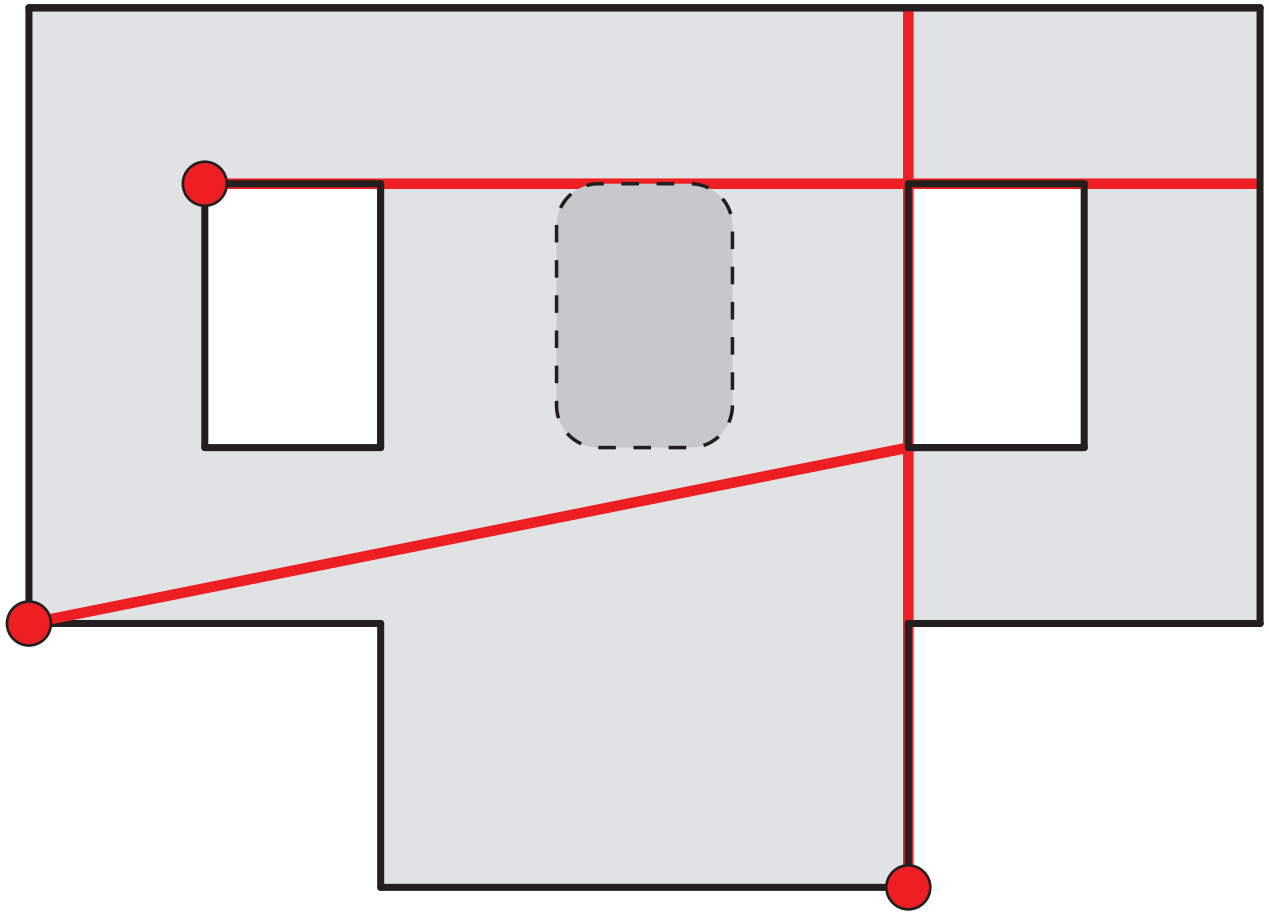
Partial Searchlight Scheduling Problem

- Can the three guards keep any moving intruder out of the dashed area by turning their searchlights?



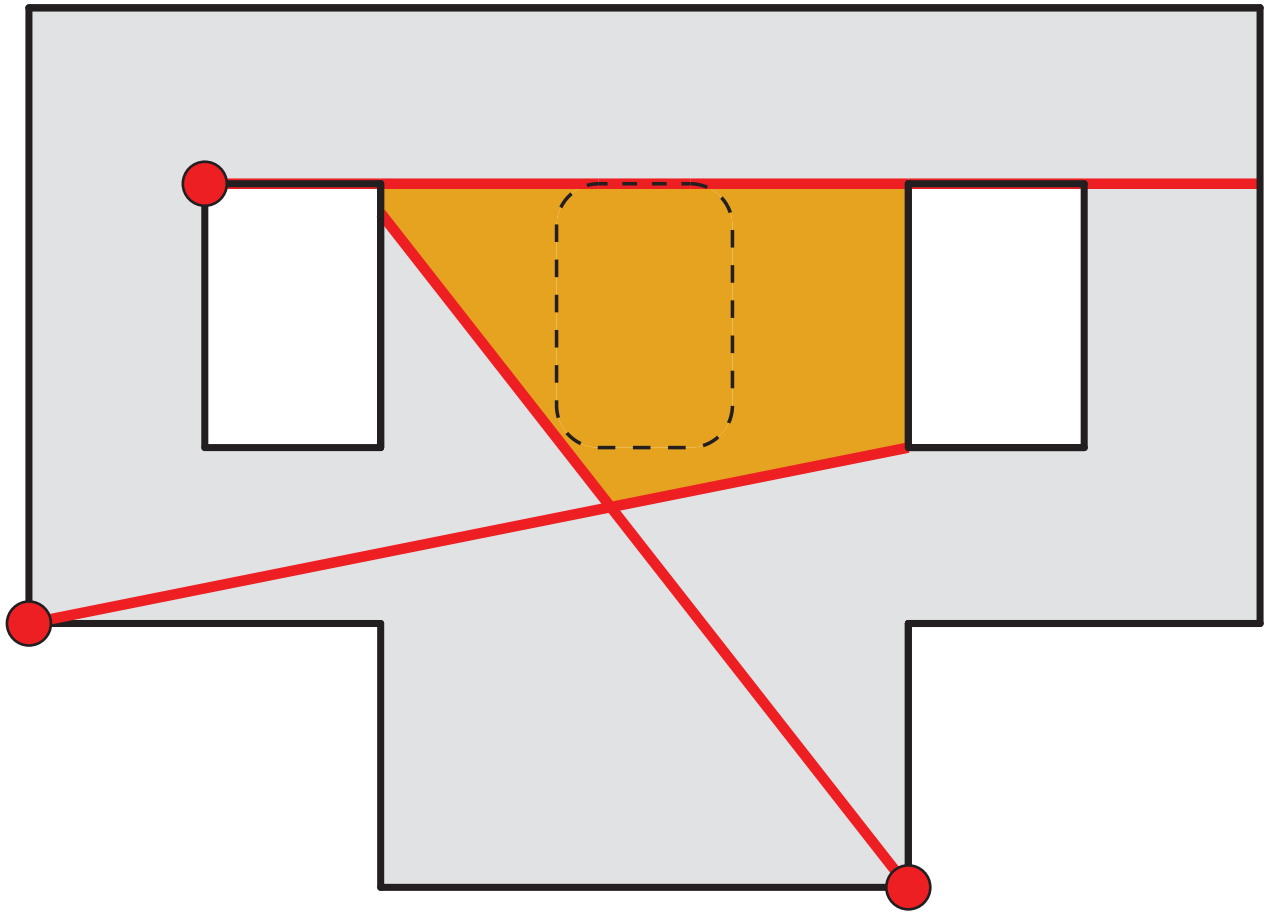
Partial Searchlight Scheduling Problem

- Can the three guards keep any moving intruder out of the dashed area by turning their searchlights?



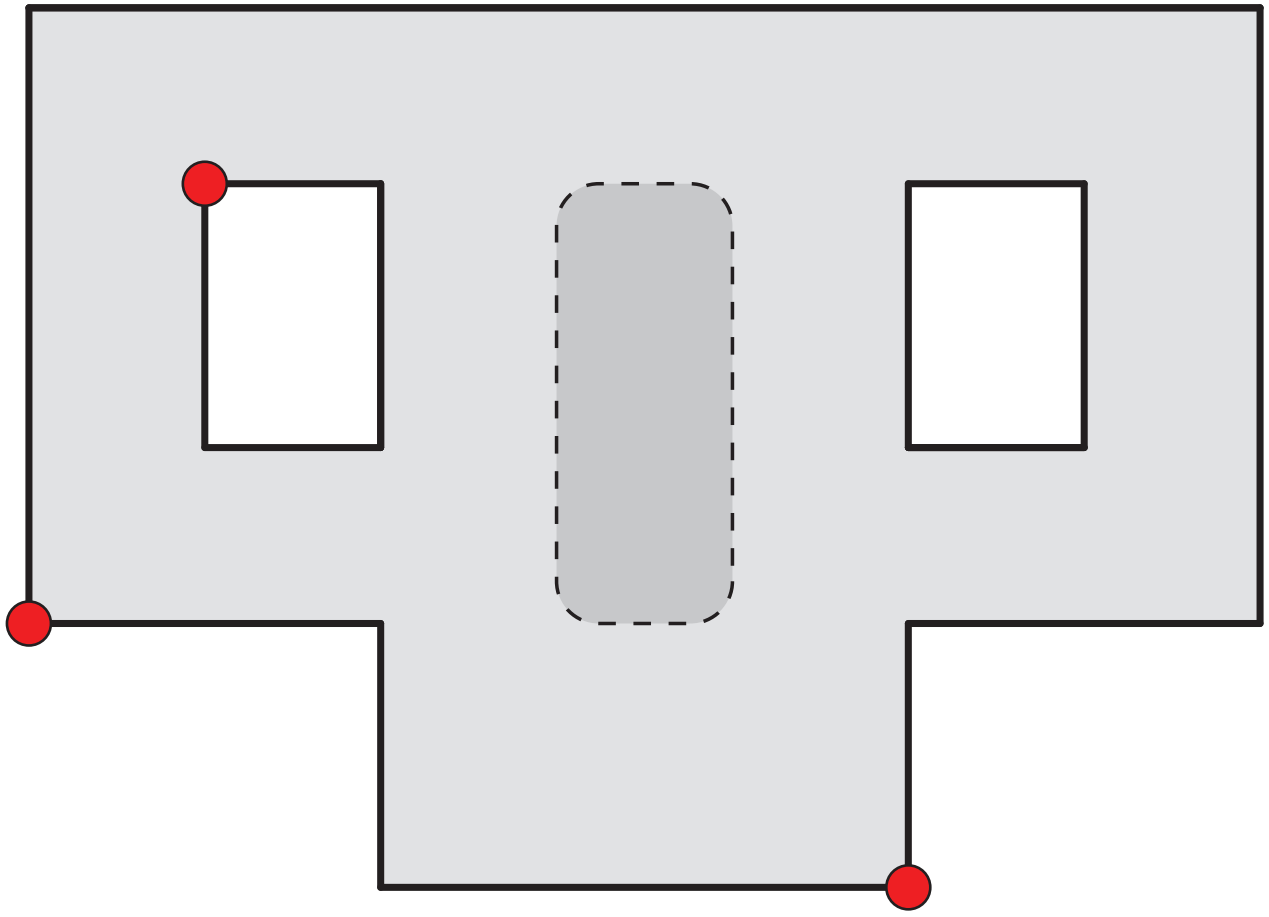
Partial Searchlight Scheduling Problem

- Can the three guards keep any moving intruder out of the dashed area by turning their searchlights?



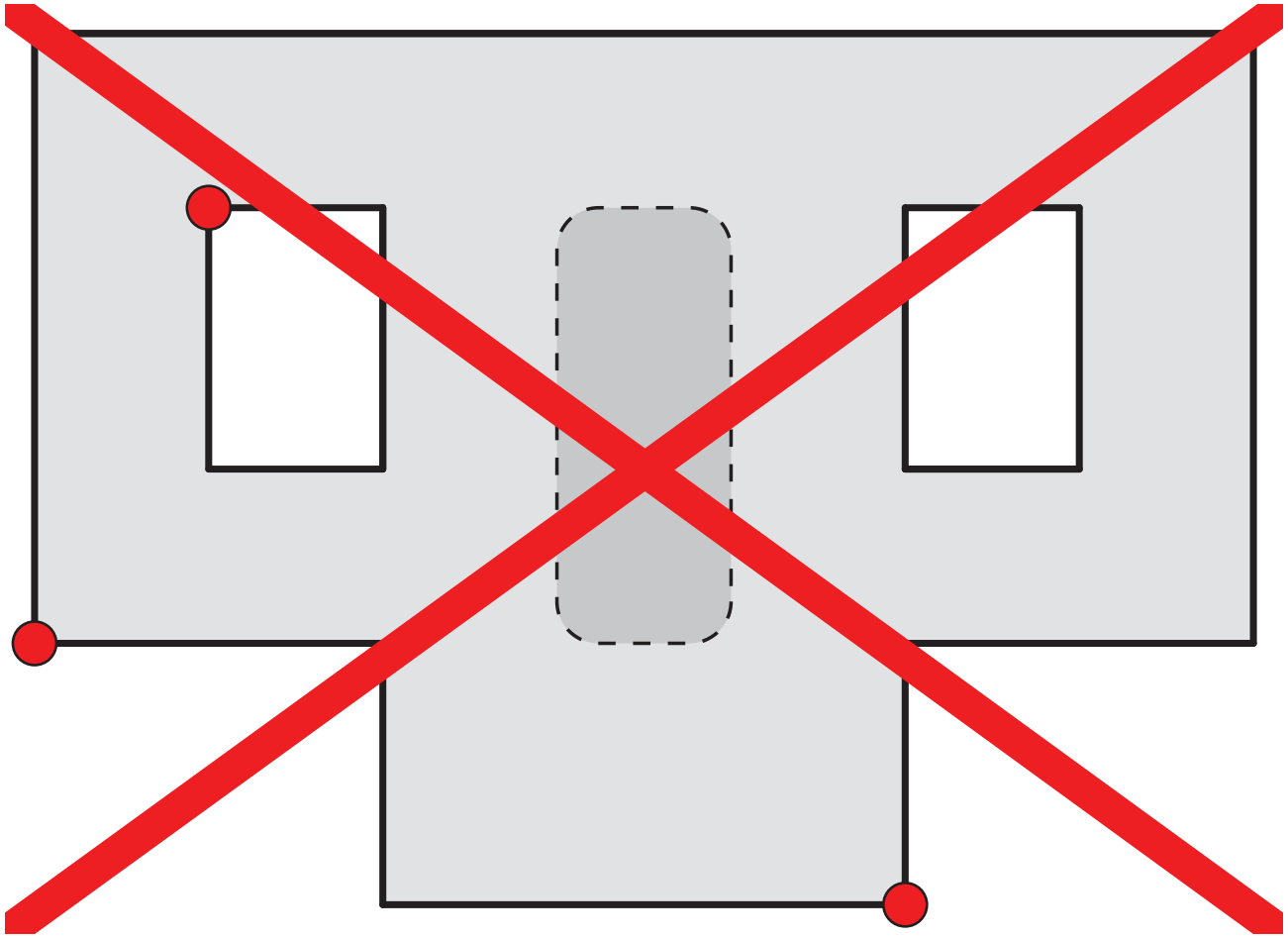
Partial Searchlight Scheduling Problem

- Can the three guards keep any moving intruder out of the dashed area by turning their searchlights?



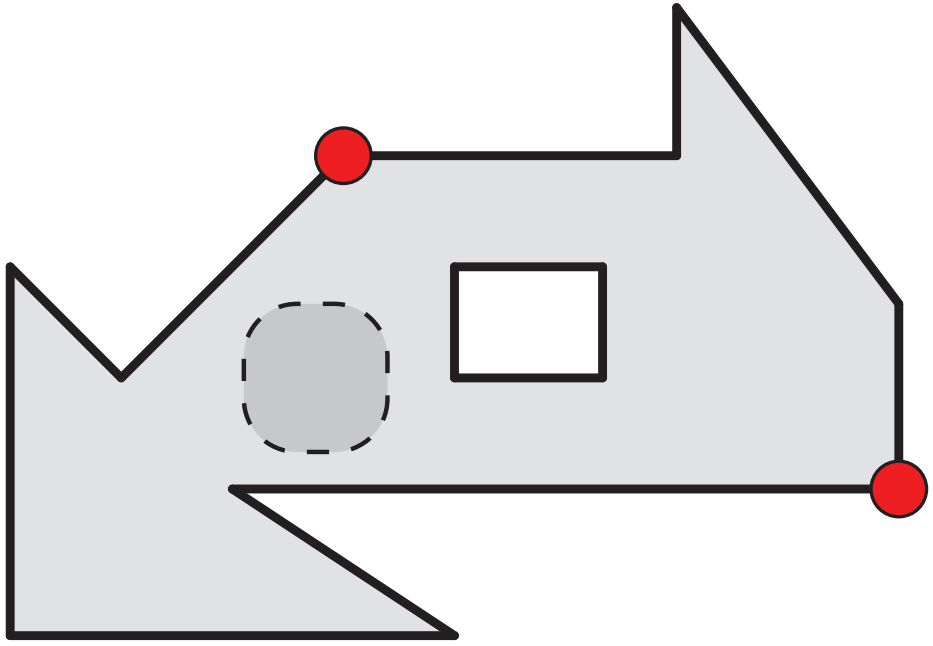
Partial Searchlight Scheduling Problem

- Can the three guards keep any moving intruder out of the dashed area by turning their searchlights?



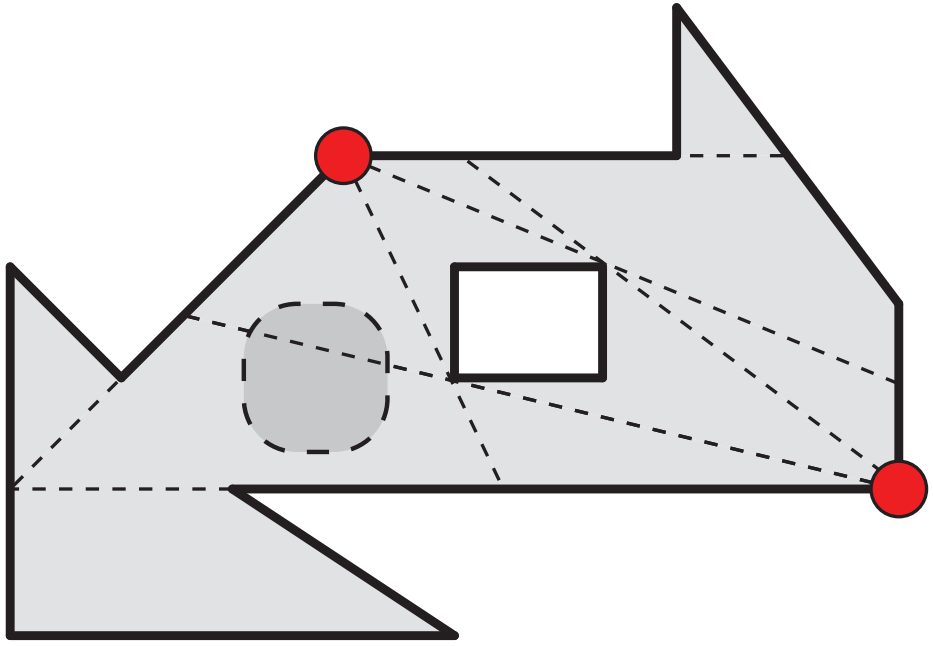
Complexity of Partial Searchlight Scheduling

- Deciding partial searchability is in **PSPACE**.



Complexity of Partial Searchlight Scheduling

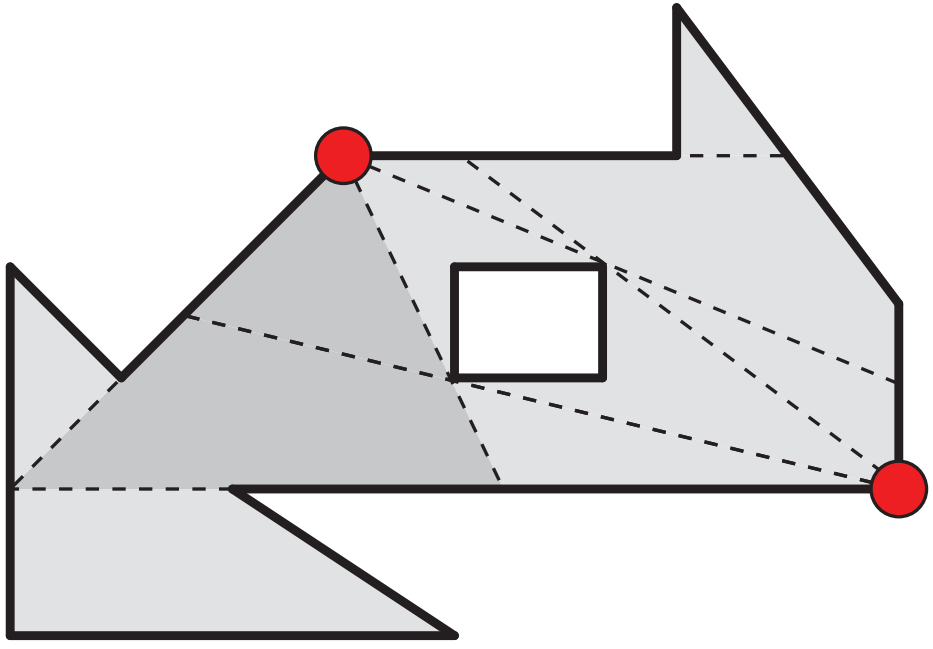
- Deciding partial searchability is in **PSPACE**.



- Identify *critical angles* and split the polygon into *cells*.

Complexity of Partial Searchlight Scheduling

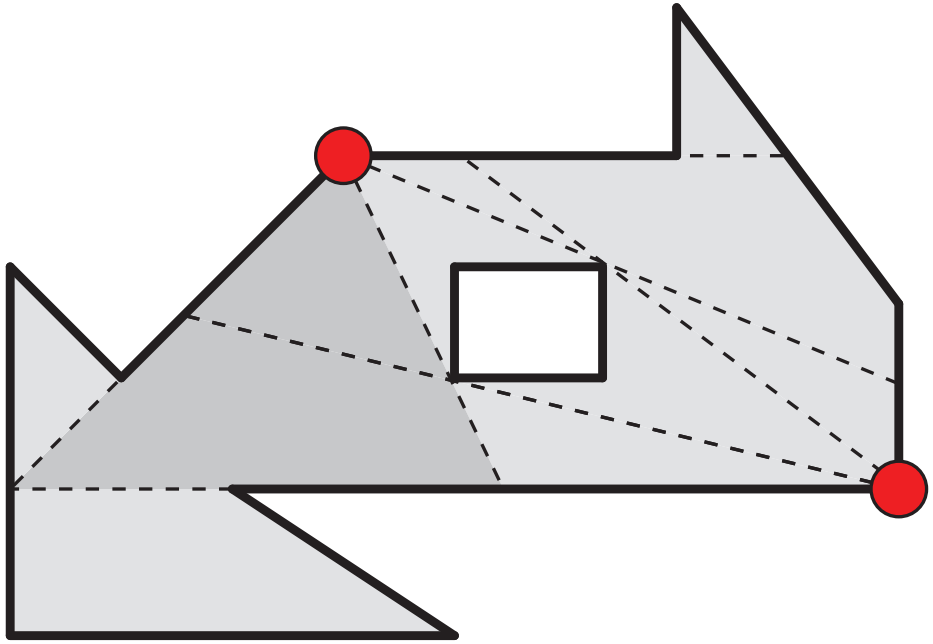
- Deciding partial searchability is in **PSPACE**.



- Identify *critical angles* and split the polygon into *cells*.
- Extend the target area to a set of cells.

Complexity of Partial Searchlight Scheduling

- Deciding partial searchability is in **PSPACE**.



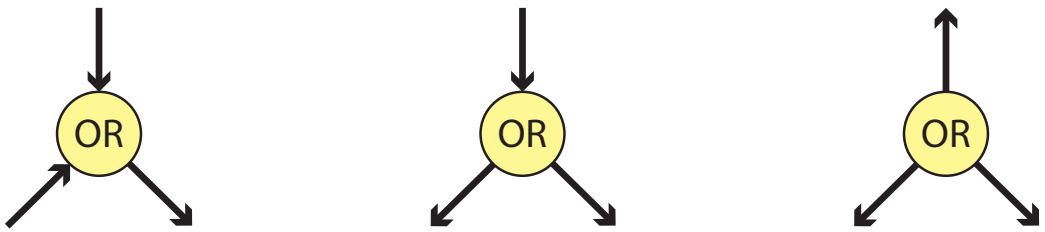
- Identify *critical angles* and split the polygon into *cells*.
- Extend the target area to a set of cells.
- Search the target area by turning one searchlight at a time, and stopping only at critical angles.

Nondeterministic Constraint Logic machines

- An NCL machine is a directed 3-regular graph, whose nodes may be of two types:

Nondeterministic Constraint Logic machines

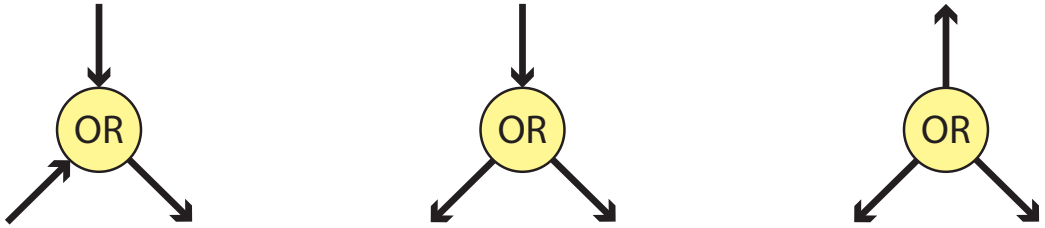
- An NCL machine is a directed 3-regular graph, whose nodes may be of two types:
  - **OR node.** At least one of its three incident edges must be directed inward.



YES YES NO

Nondeterministic Constraint Logic machines

- An NCL machine is a directed 3-regular graph, whose nodes may be of two types:
  - **OR node.** At least one of its three incident edges must be directed inward.



YES YES NO

  - **AND node.** Either the thick edge is directed inward, or both thin edges are directed inward.

YES YES YES NO

Nondeterministic Constraint Logic machines

- A *legal move* is the reversal of an edge, preserving the above constraints.
- Moves may be *asynchronous*.

Nondeterministic Constraint Logic machines

- A *legal move* is the reversal of an edge, preserving the above constraints.
- Moves may be *asynchronous*.

- NCL machines have two *distinguished edges*, e_a and e_b.
- Each edge has a *target orientation*.

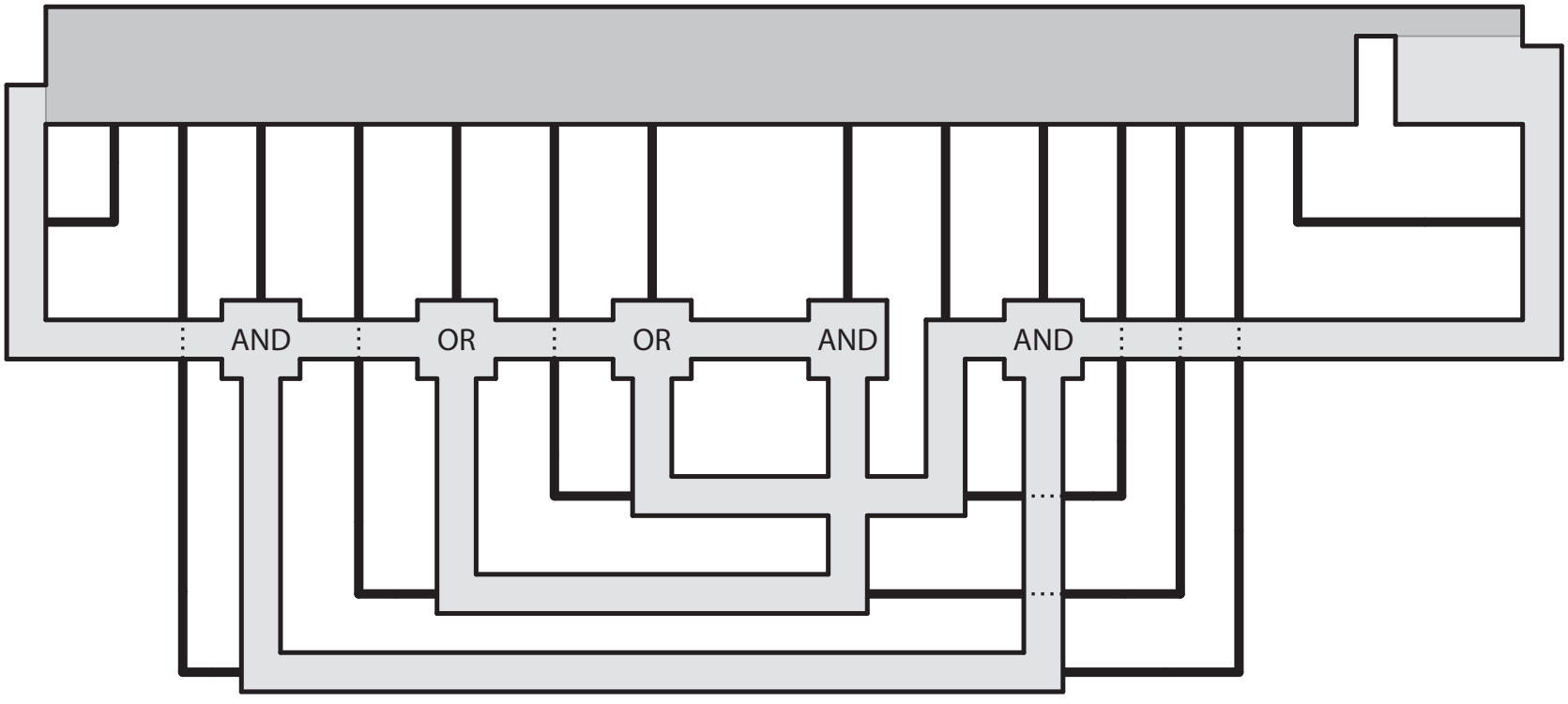
Nondeterministic Constraint Logic machines

- A *legal move* is the reversal of an edge, preserving the above constraints.
- Moves may be *asynchronous*.

- NCL machines have two *distinguished edges*, e_a and e_b.
- Each edge has a *target orientation*.

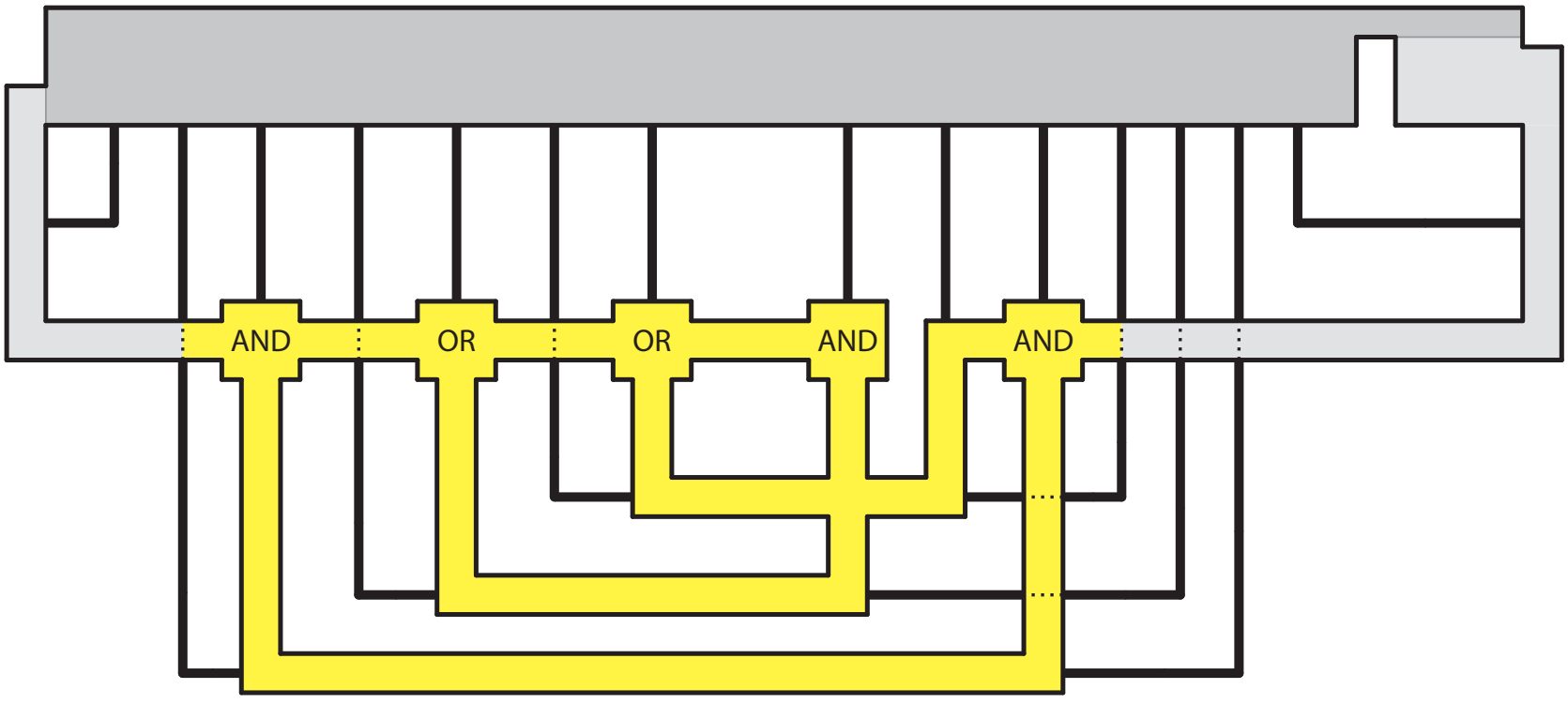
- It is **PSPACE**-complete to decide if legal configurations A and B exist, such that:
  - e_a is in its target orientation in A;
  - e_b is in its target orientation in B;
  - there is a legal sequence of asynchronous moves from A to B.

From NCL machines to orthogonal polygons



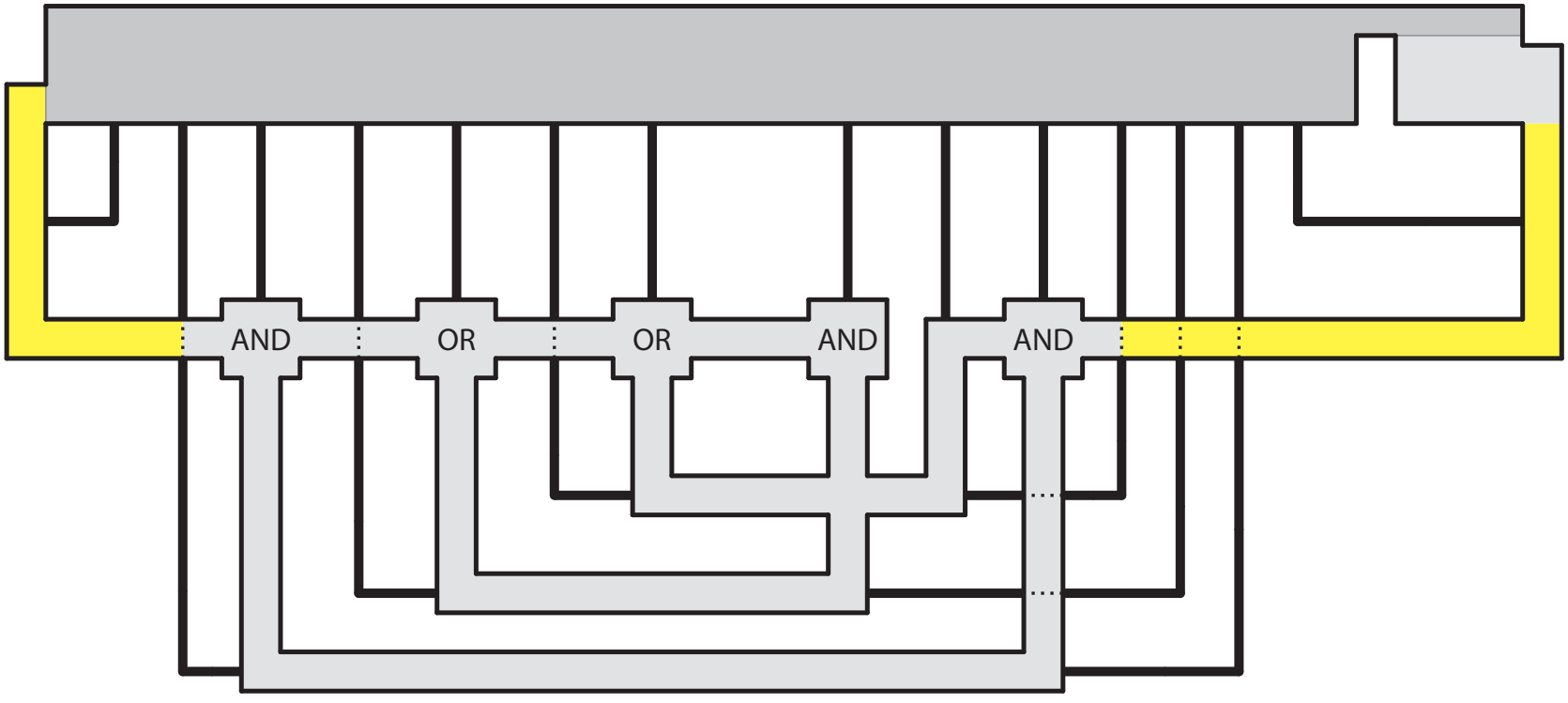
- Reduction overview

From NCL machines to orthogonal polygons



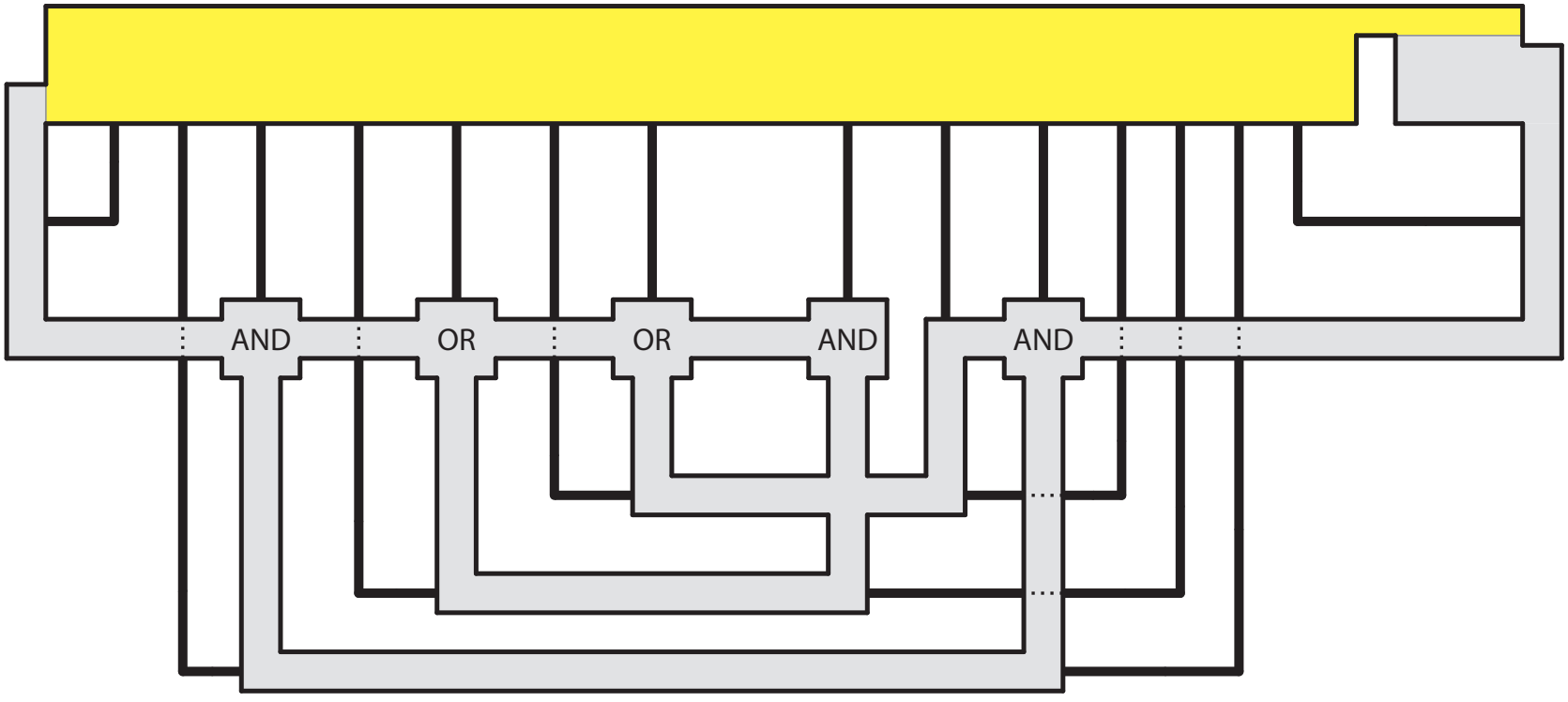
- NCL network (each "segment" contains a guard)

From NCL machines to orthogonal polygons



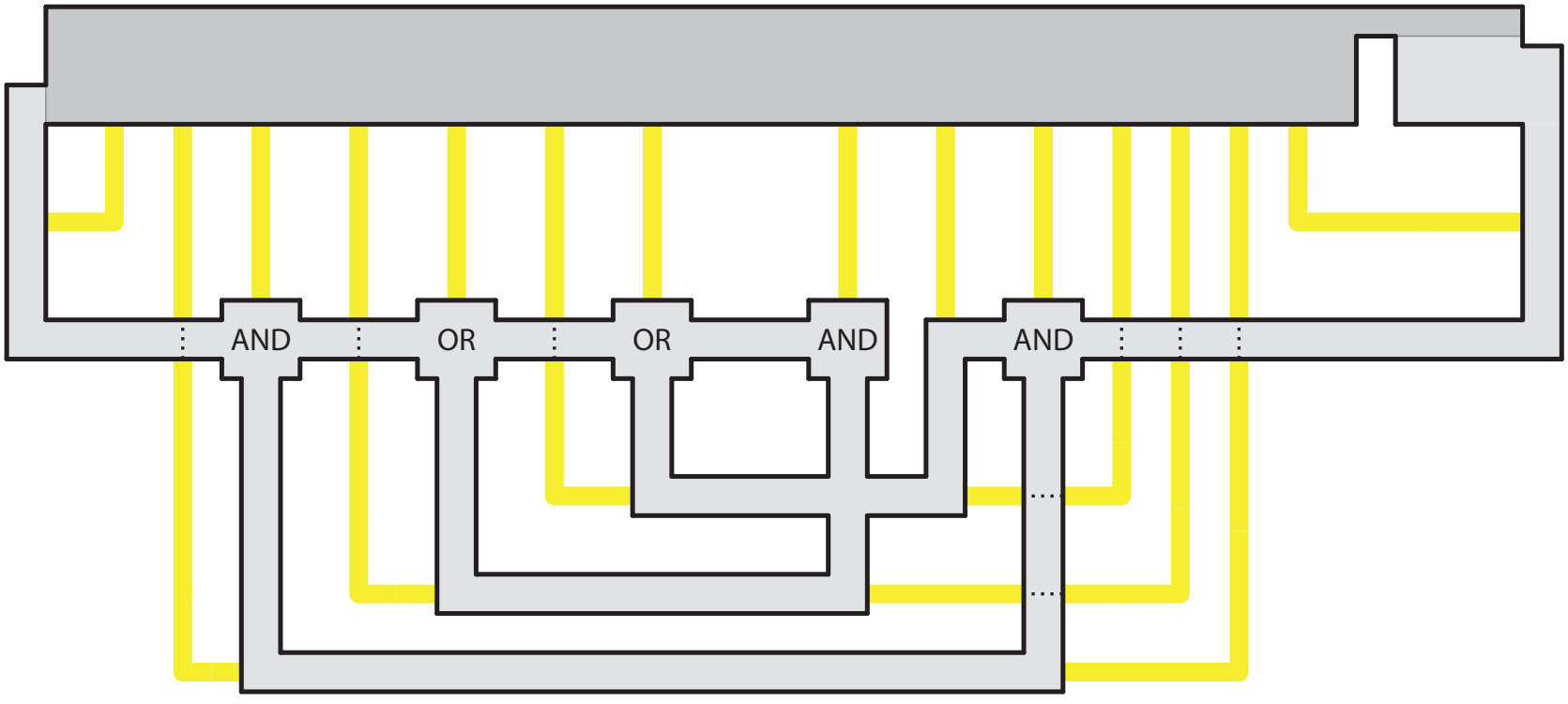
- Distinguished edges

From NCL machines to orthogonal polygons



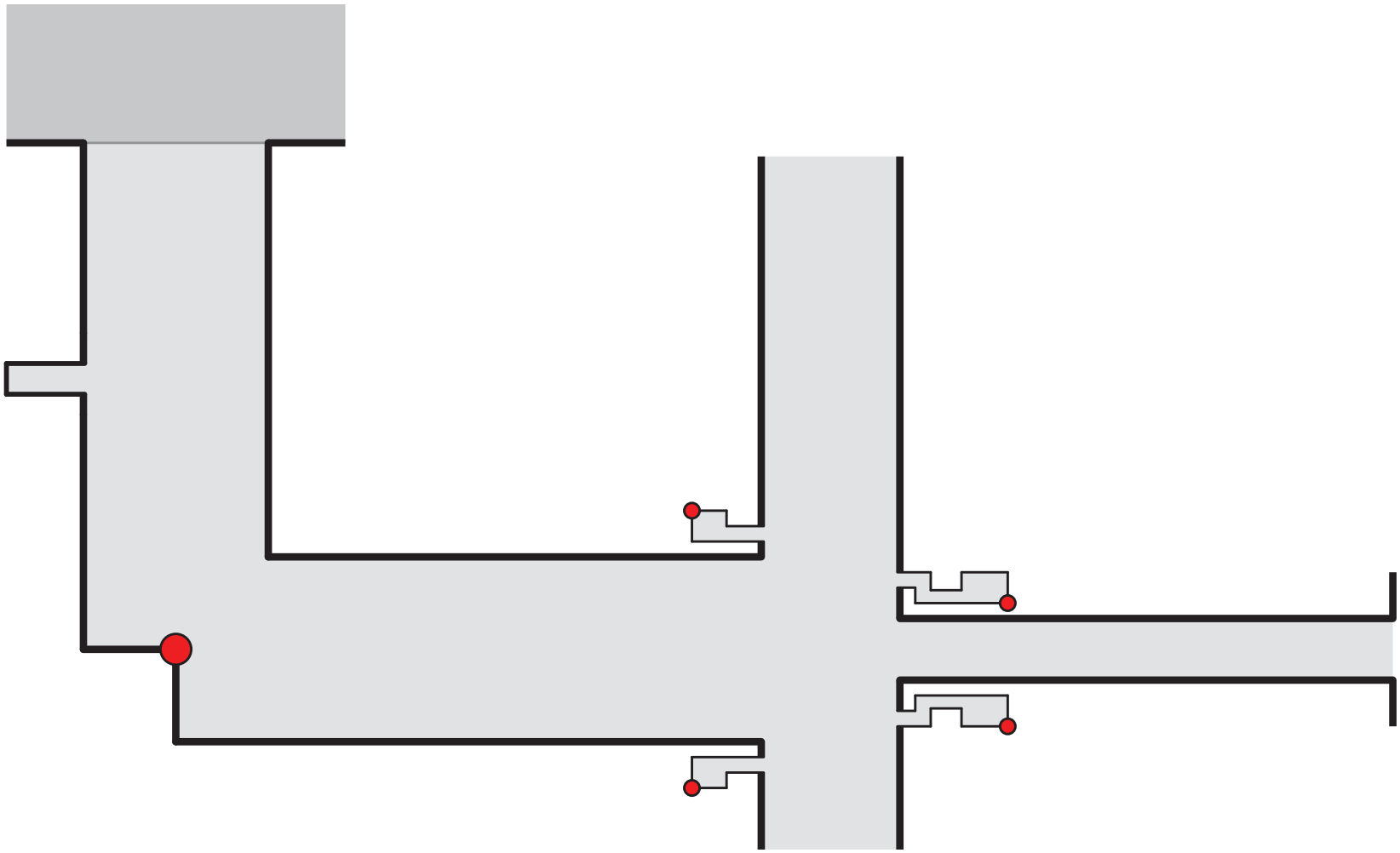
- Target area

From NCL machines to orthogonal polygons



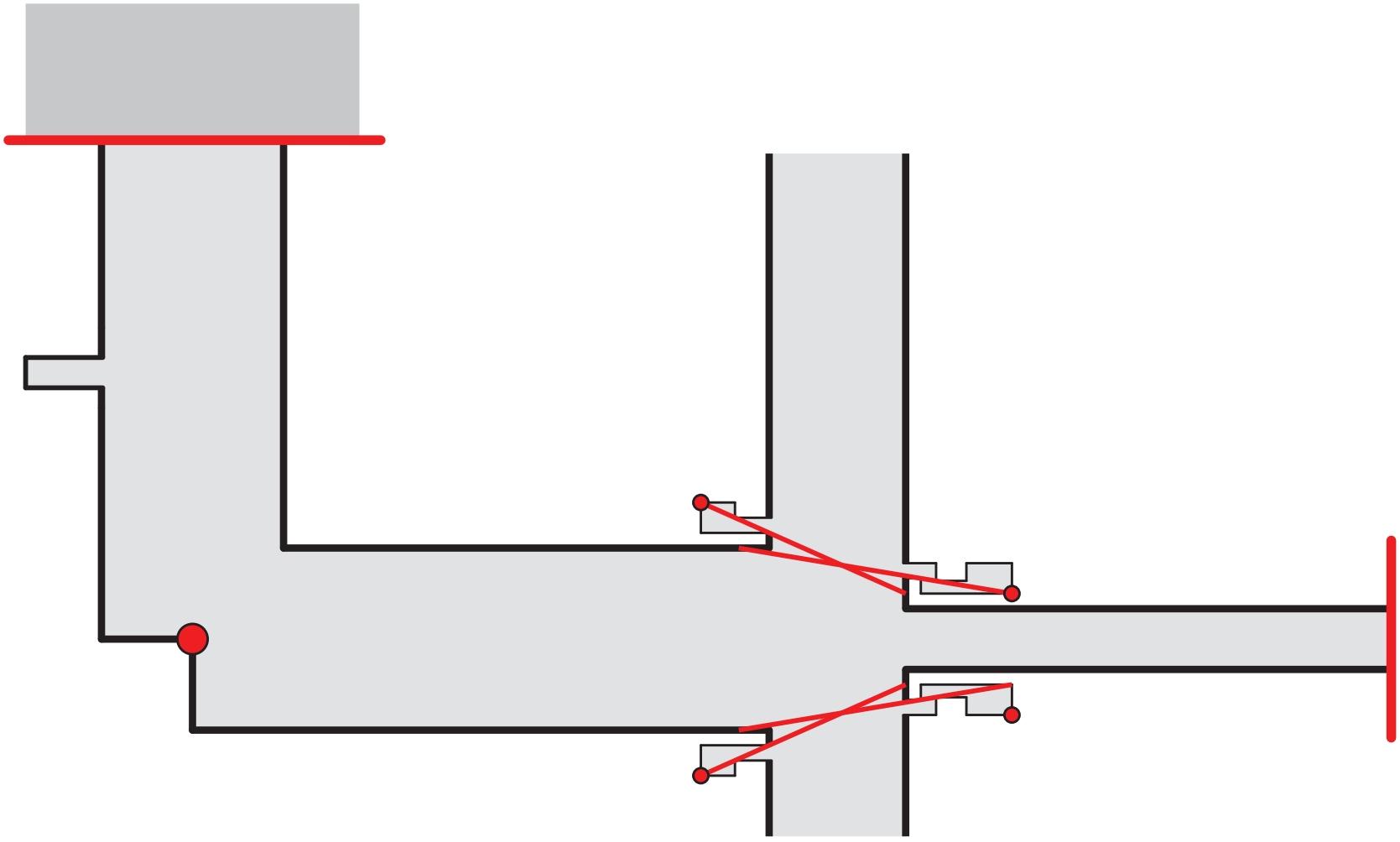
- Pipes (each one contains a guard)

Crossings between pipes and NCL edges



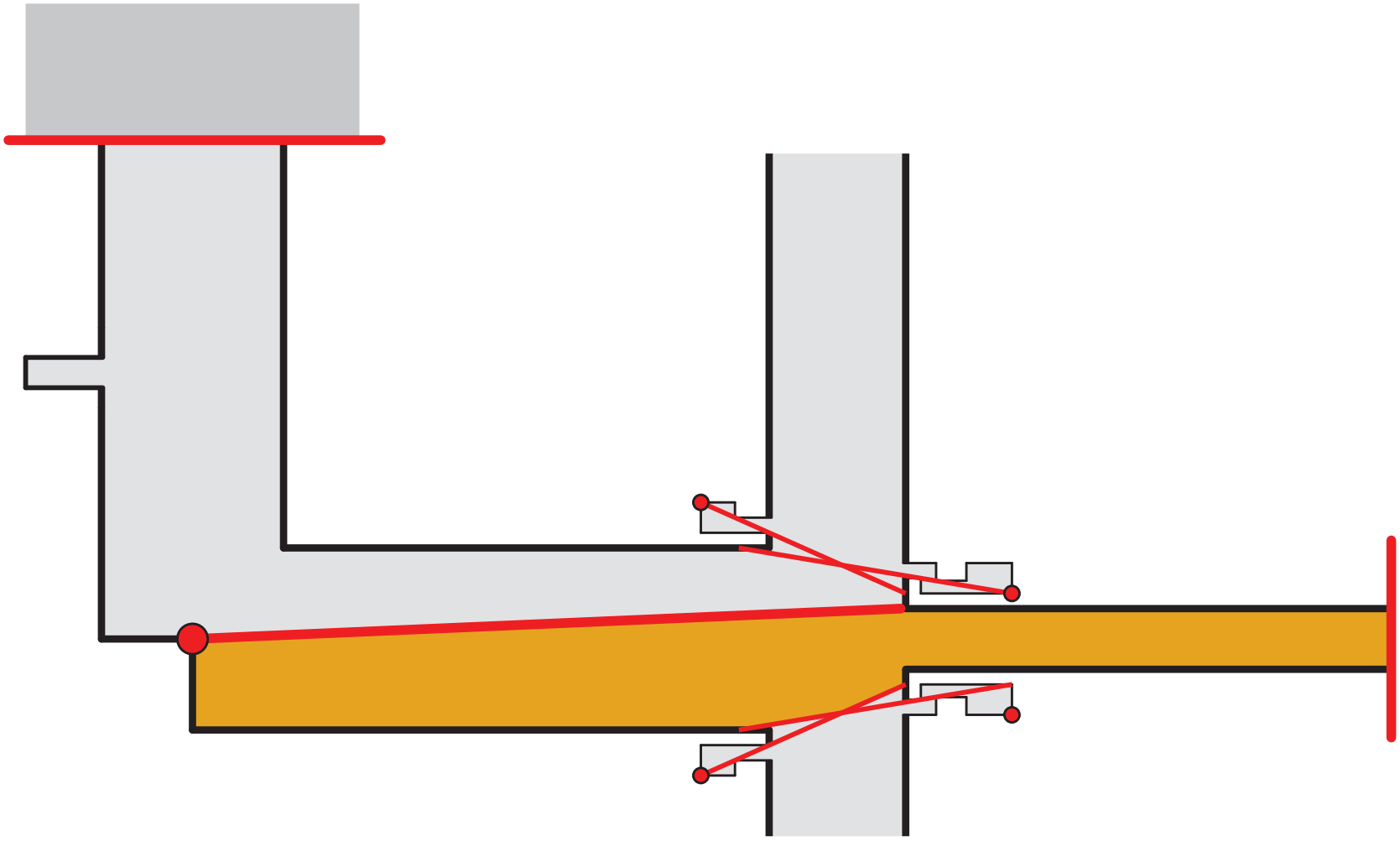
- Construction overview

Crossings between pipes and NCL edges



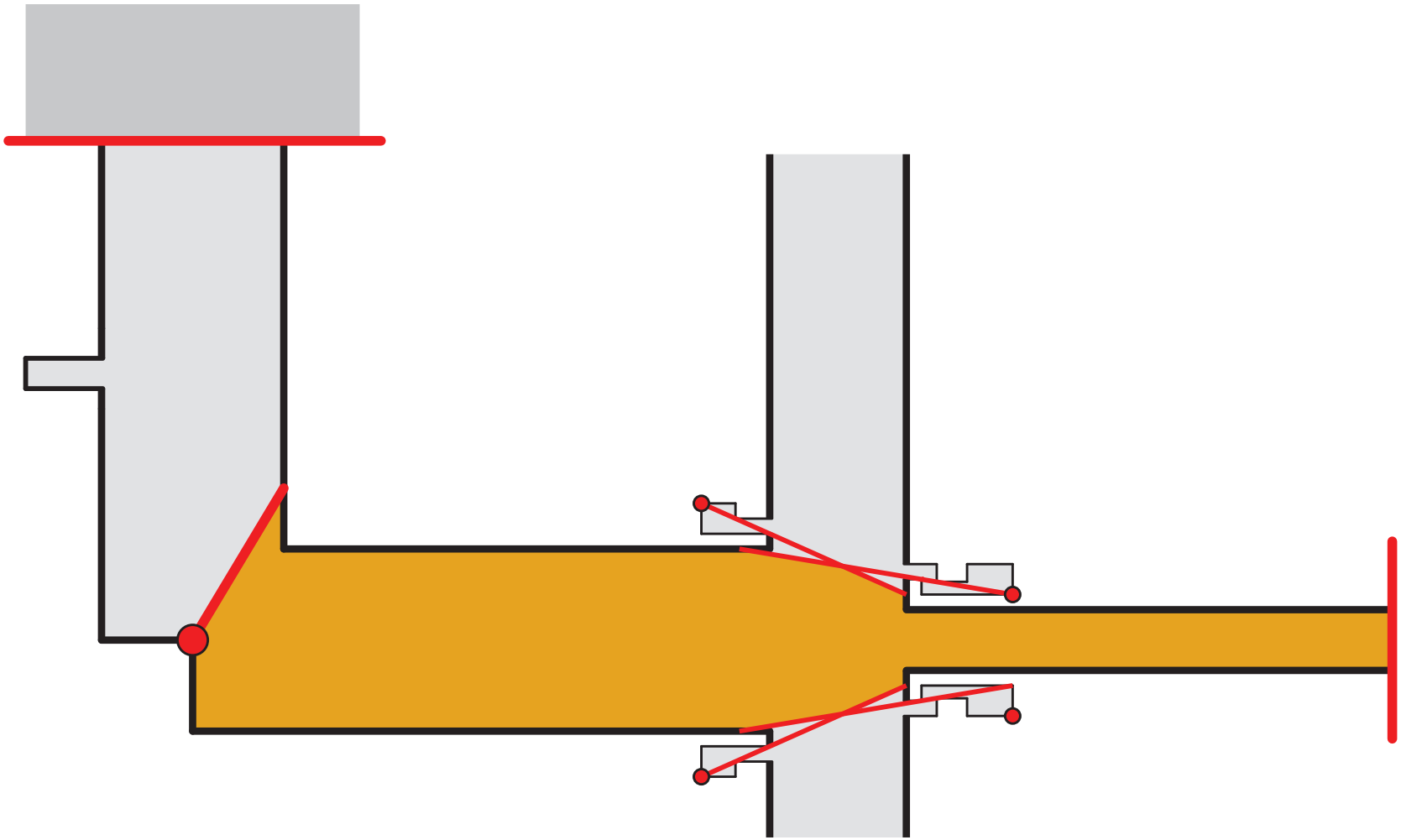
- External searchlights are closing the pipe's ends

Crossings between pipes and NCL edges



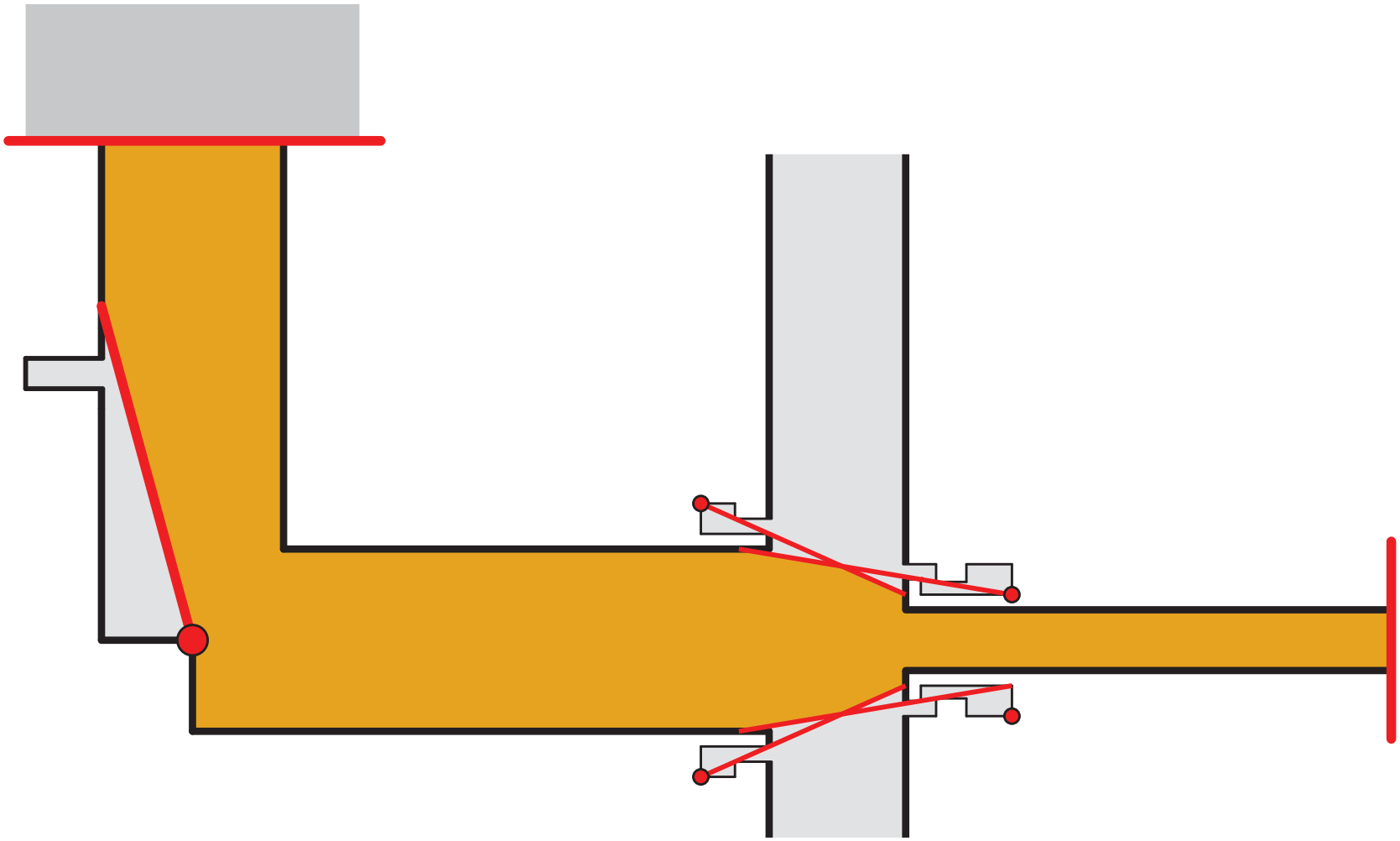
- Search begins

Crossings between pipes and NCL edges



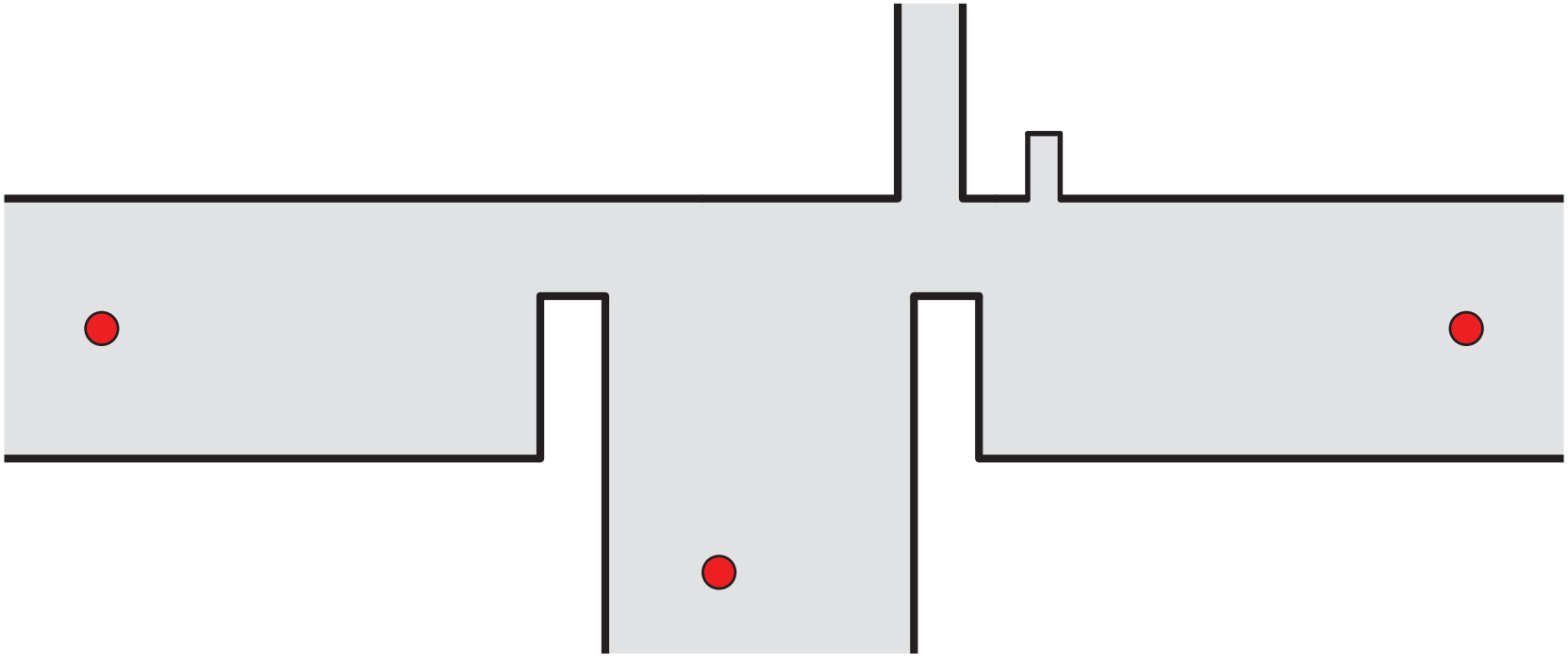
- Search continues

Crossings between pipes and NCL edges



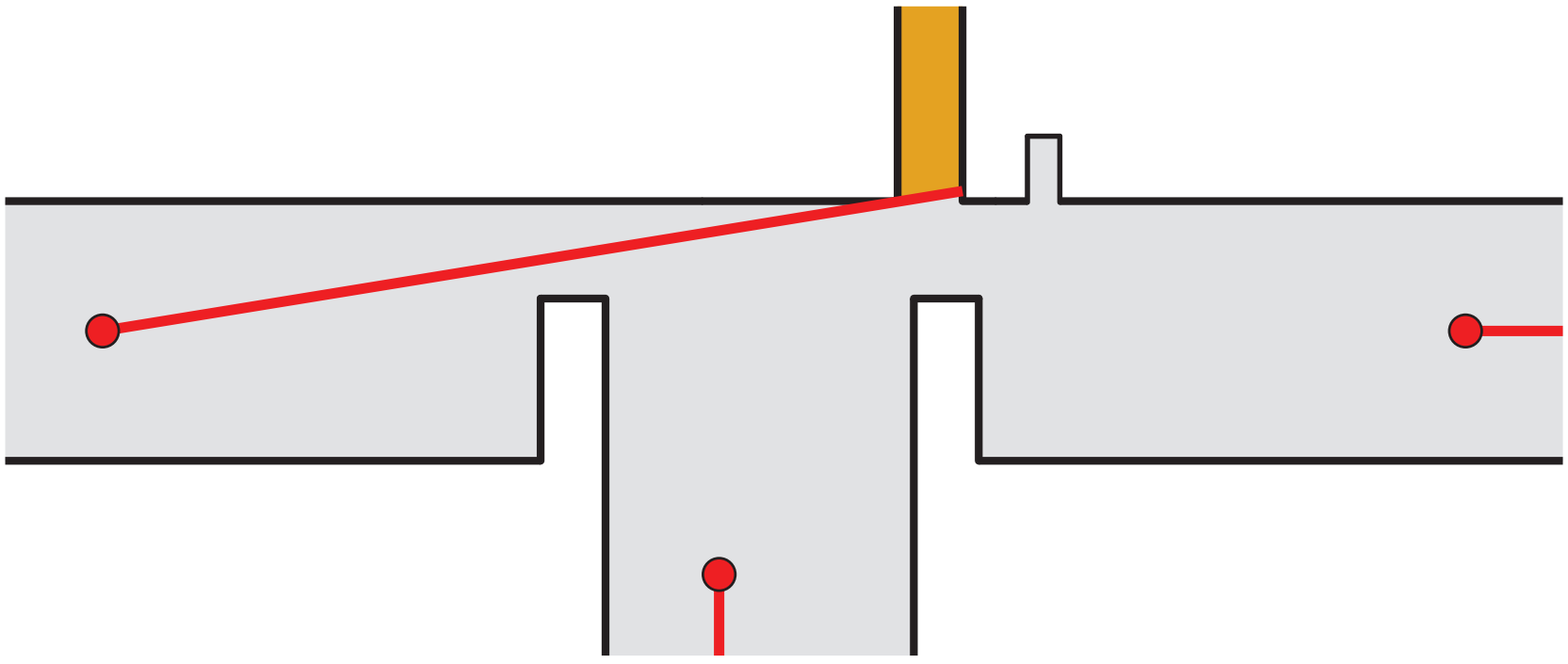
- Search ends

OR nodes



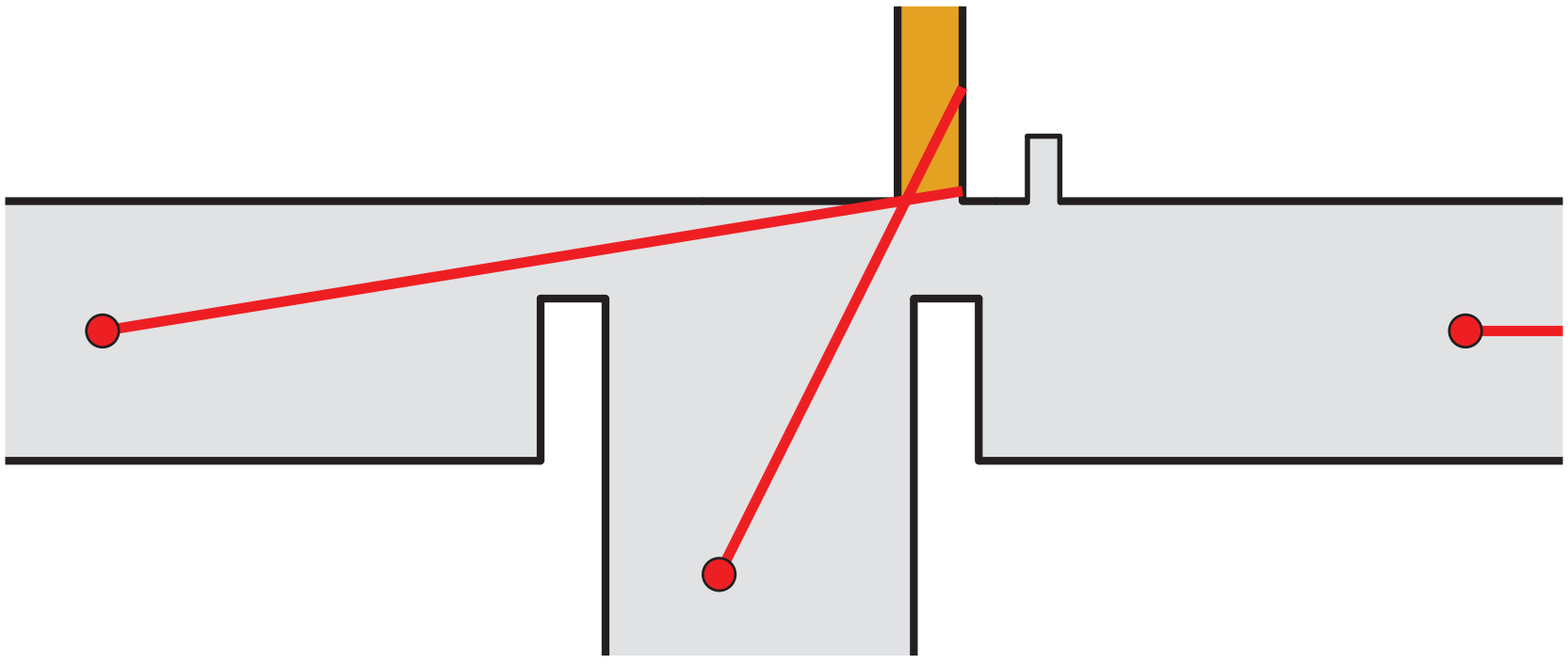
- Construction overview

OR nodes



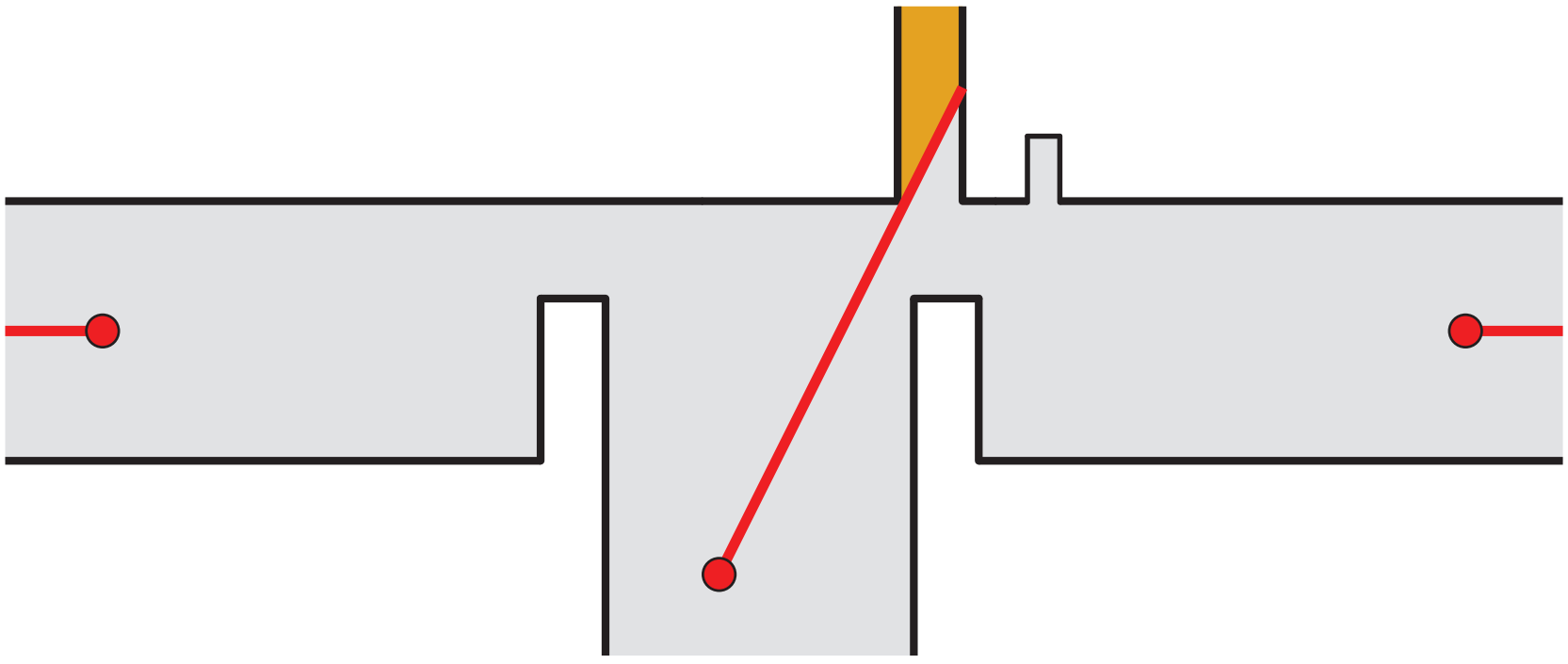
- Legal configuration (pipe still clear)

OR nodes



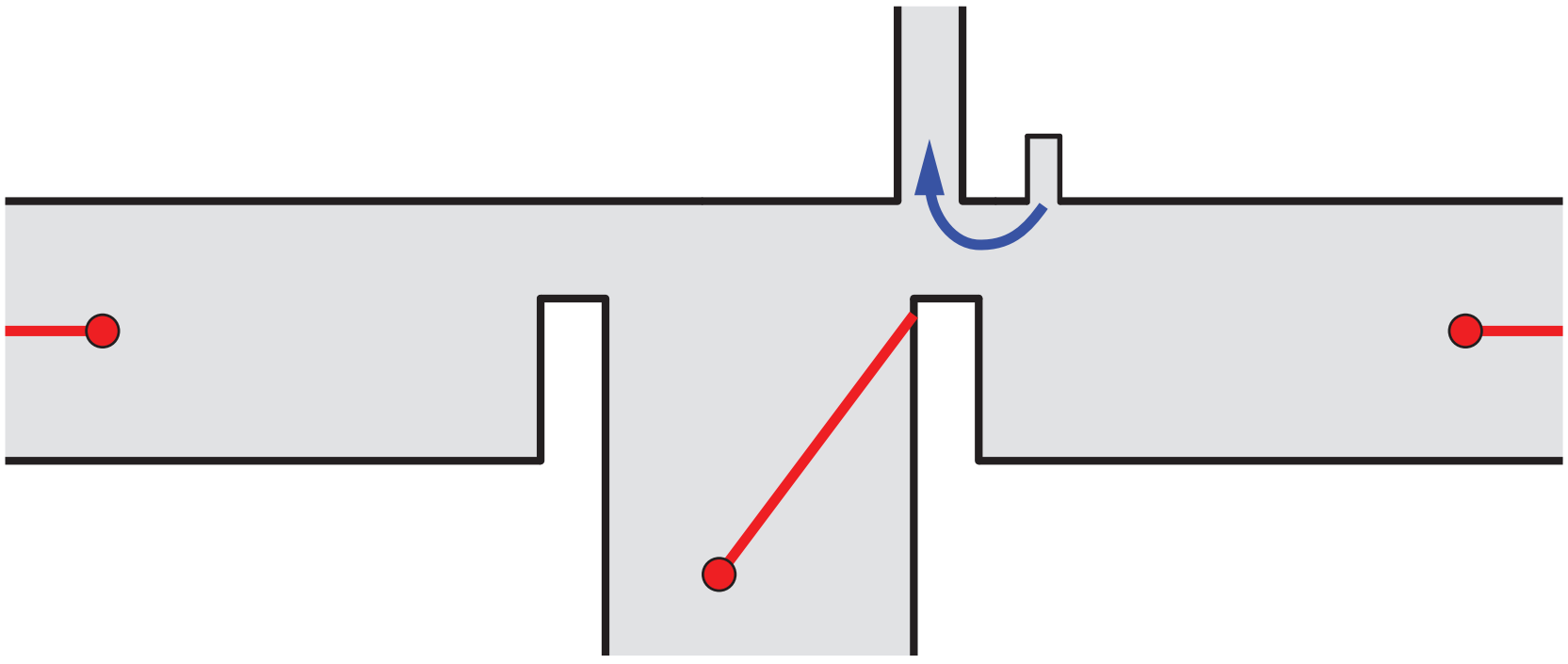
- Legal configuration (pipe still clear)

OR nodes



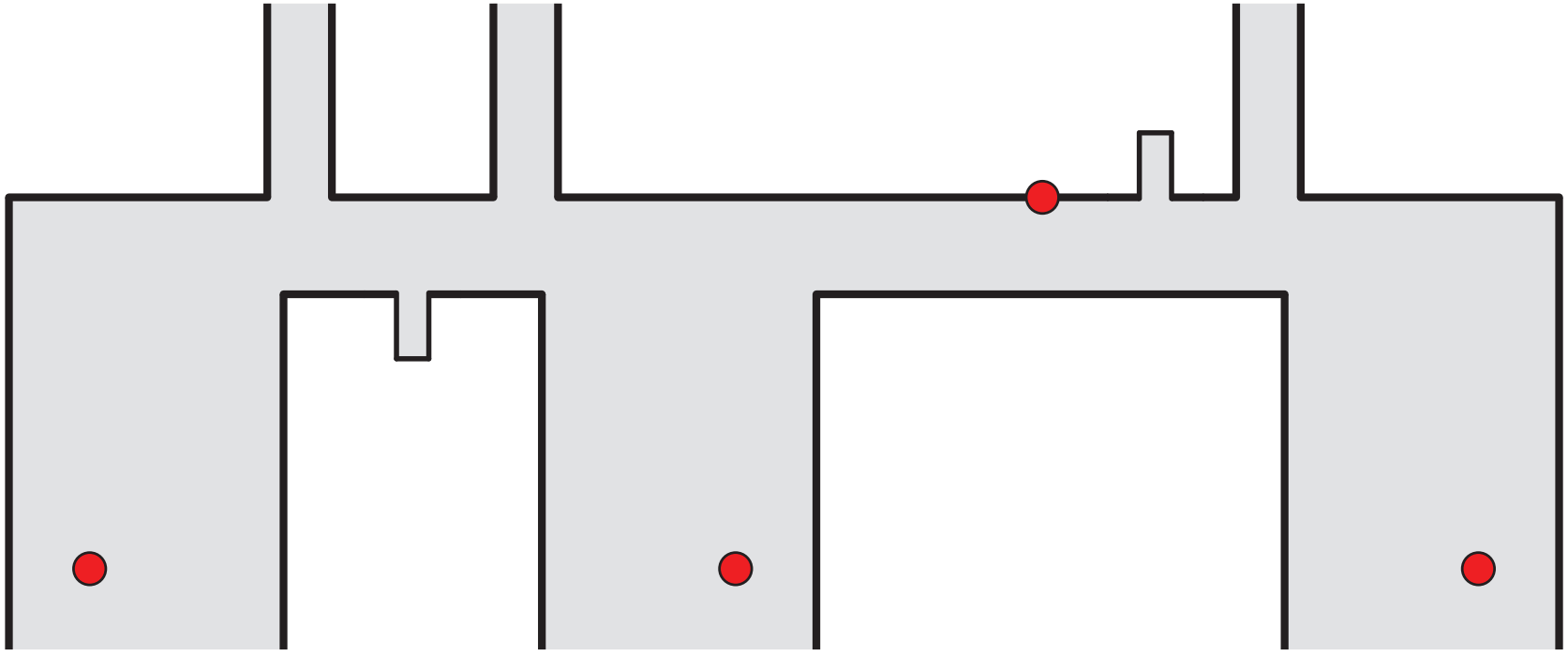
- Legal configuration (pipe still clear)

OR nodes



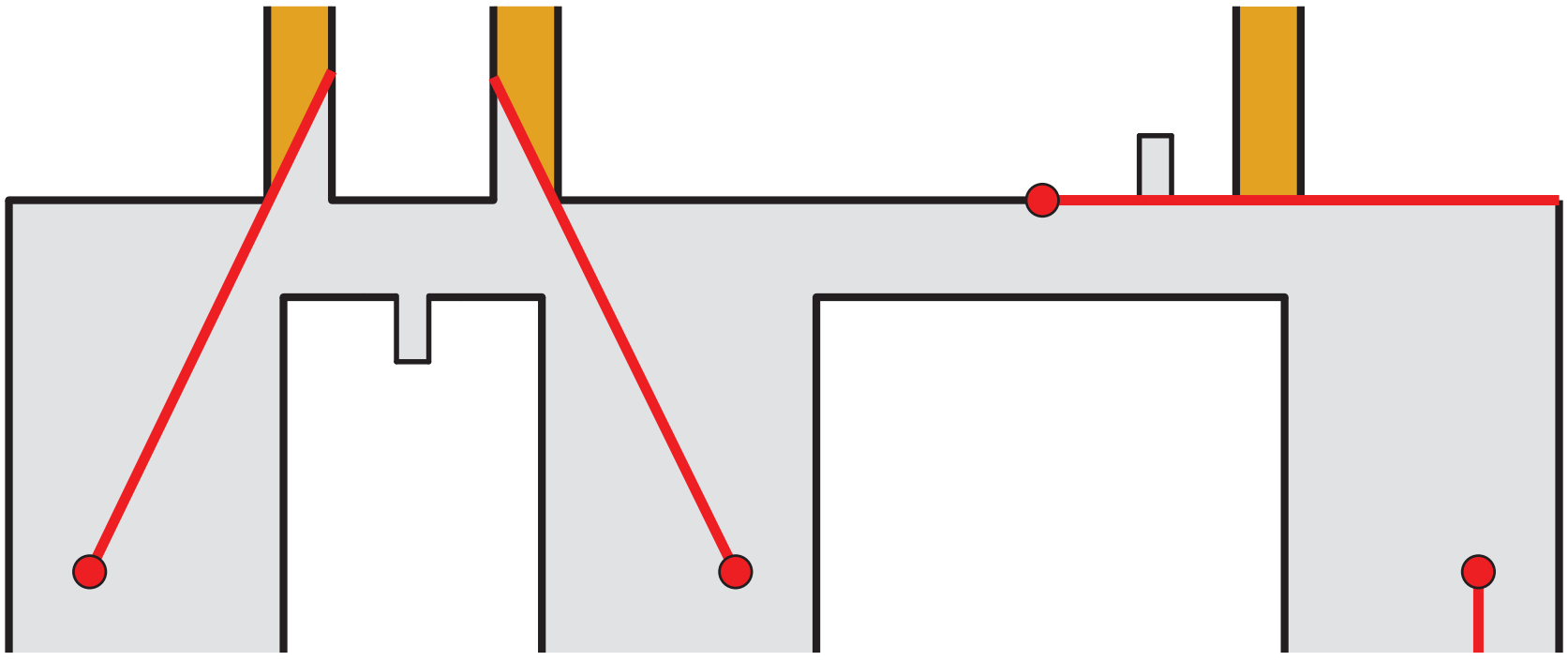
- Illegal configuration (pipe recontaminated)

AND nodes



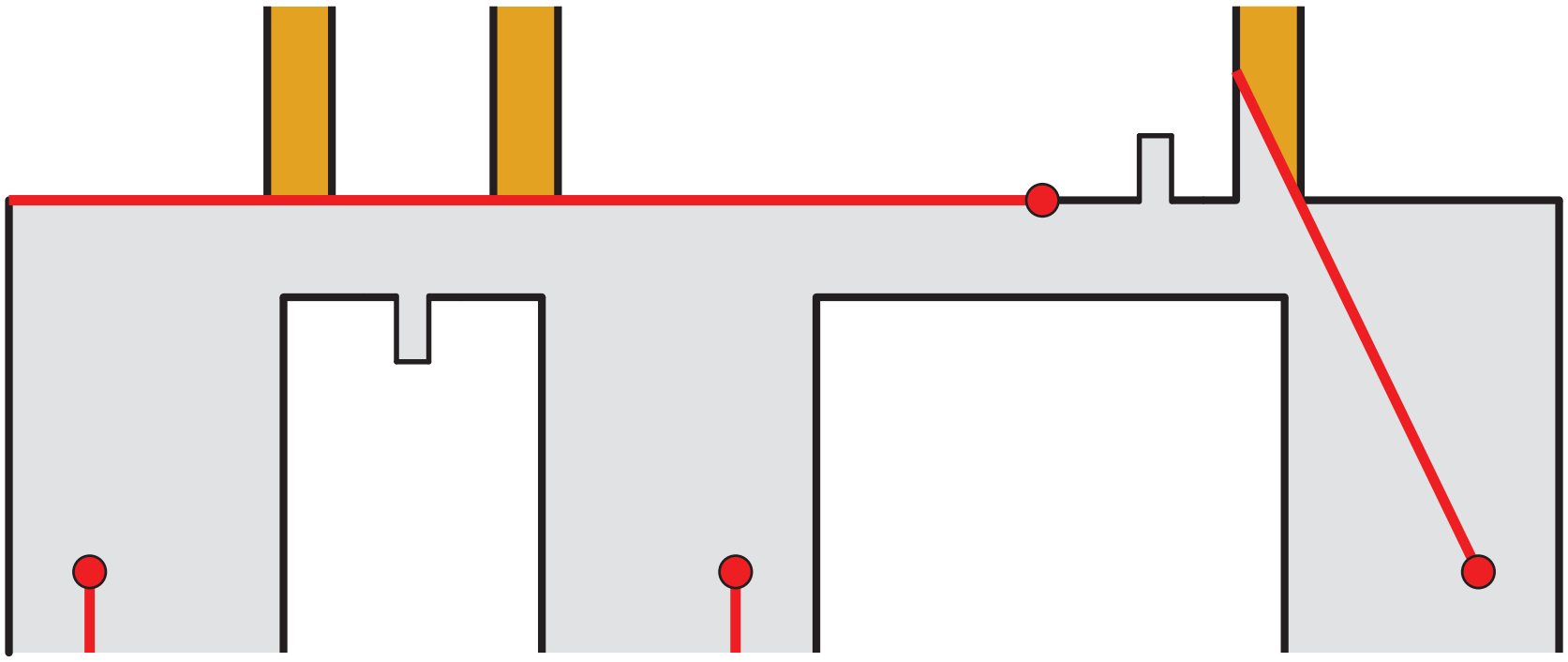
- Construction overview

AND nodes



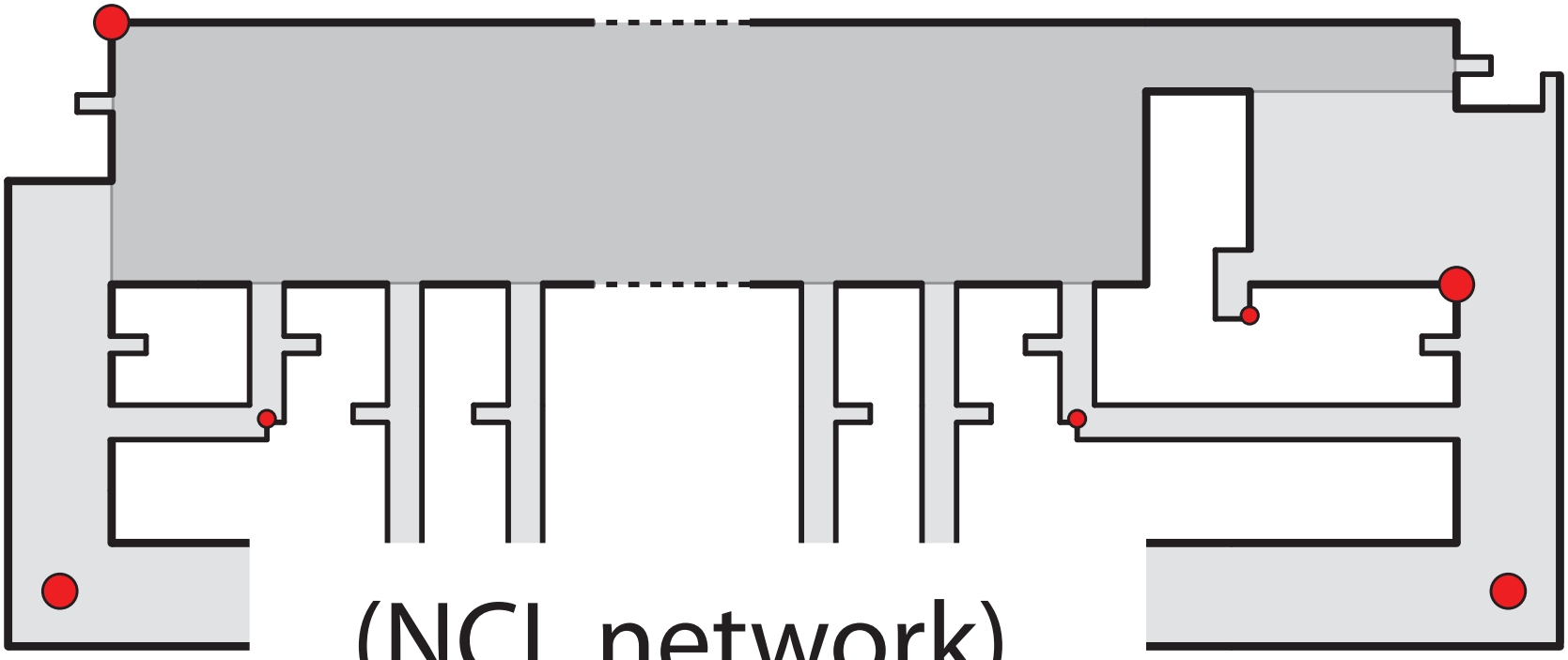
- Legal configuration (pipes still clear)

AND nodes



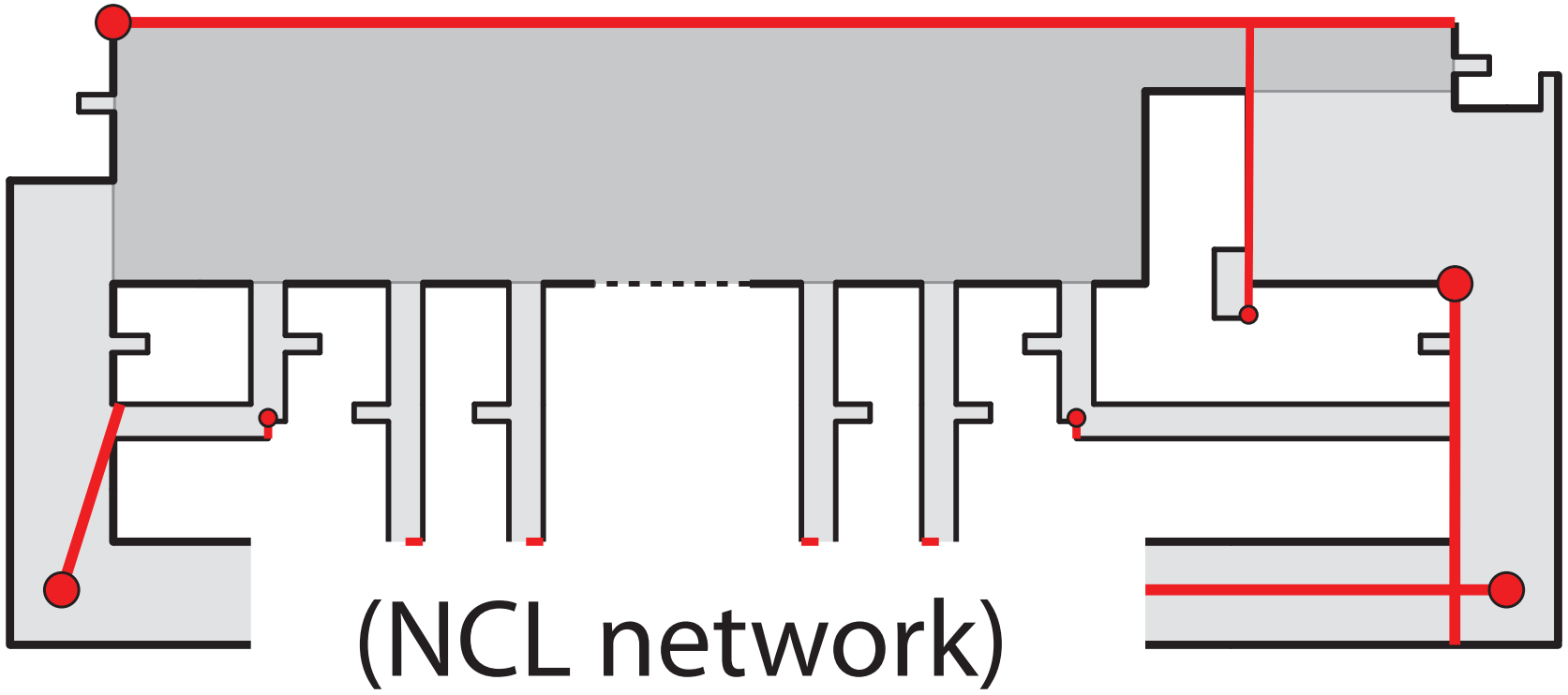
- Legal configuration (pipes still clear)

Clearing process

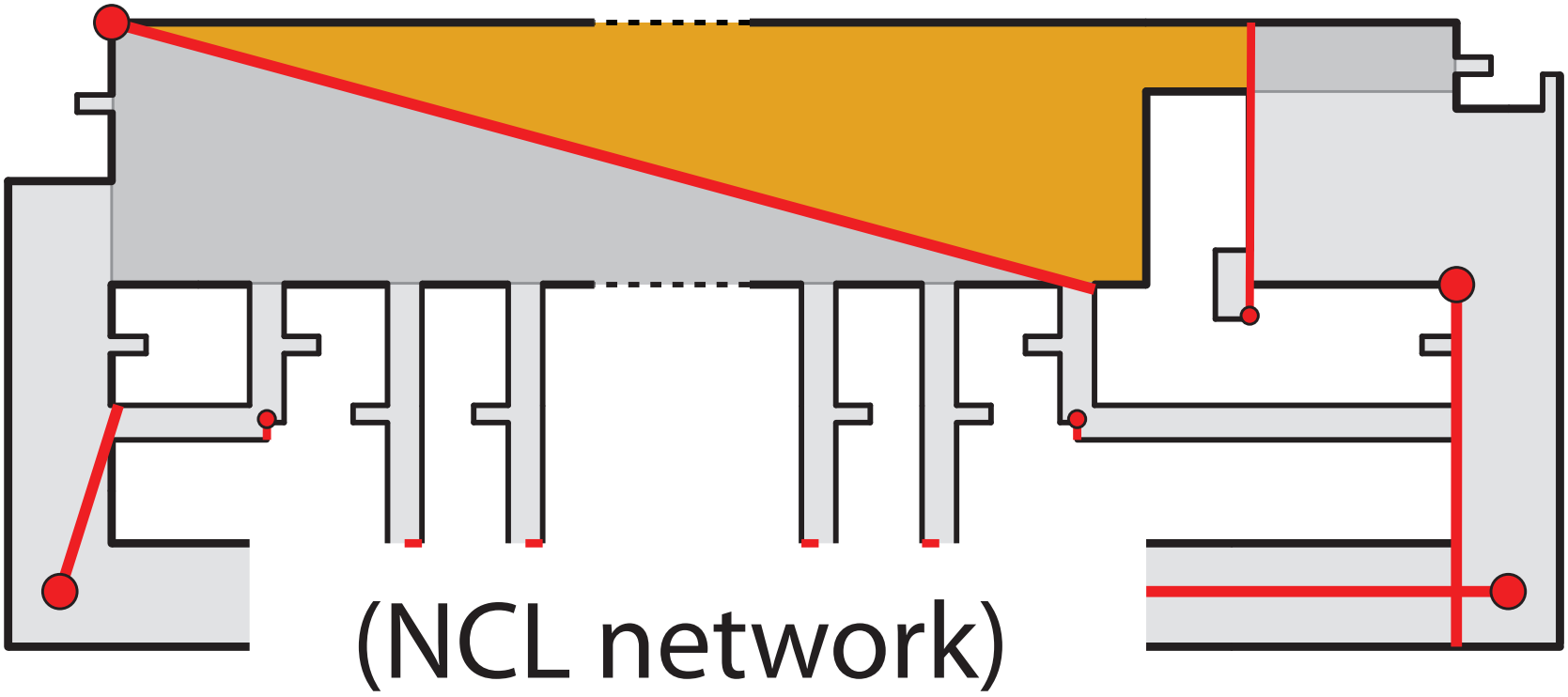


(NCL network)

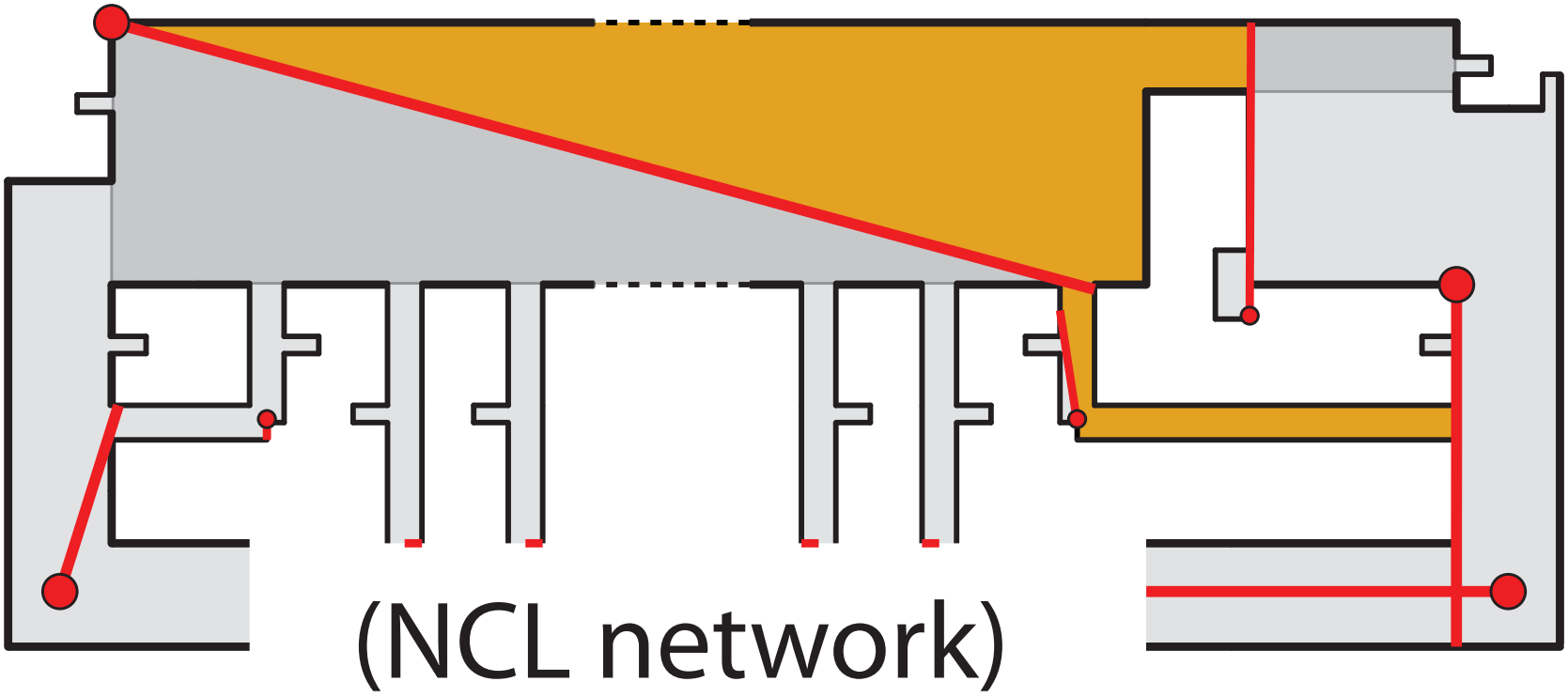
Clearing process



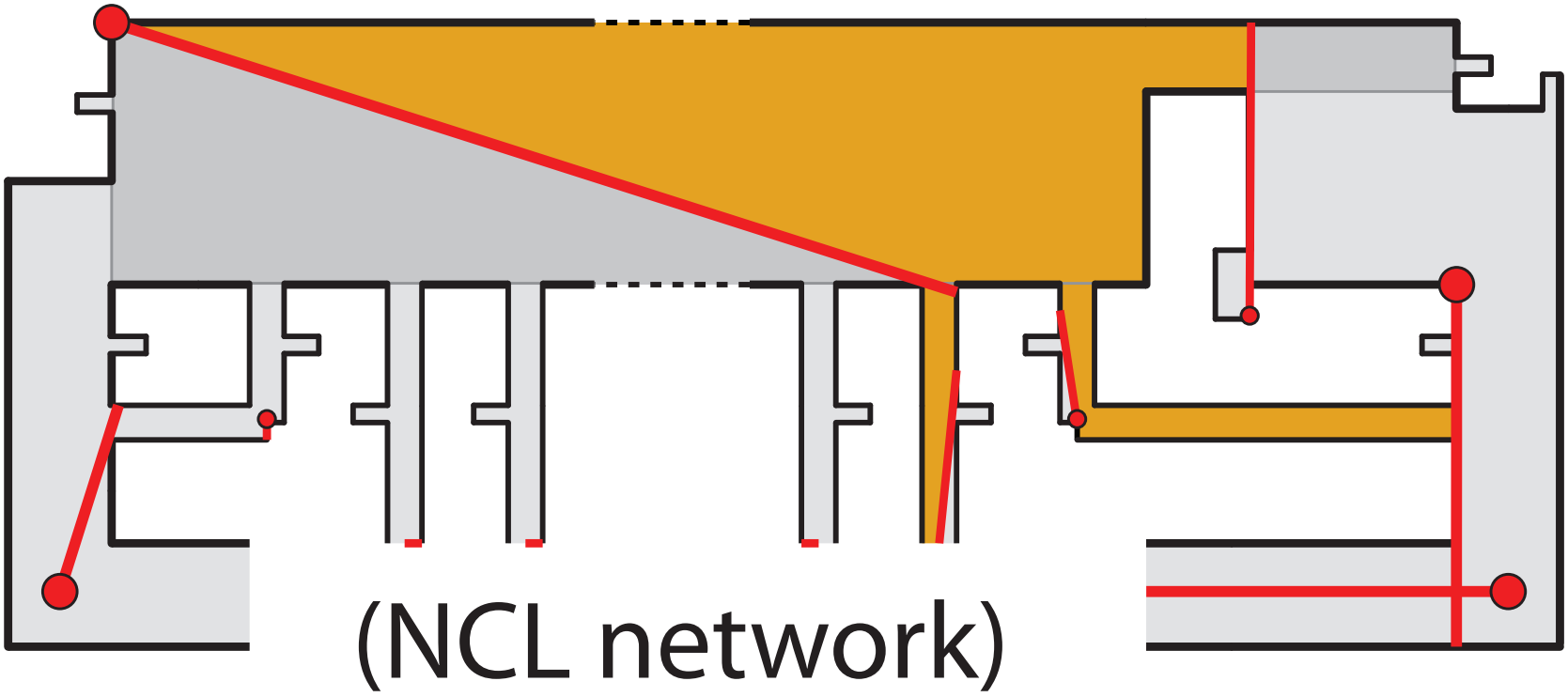
Clearing process



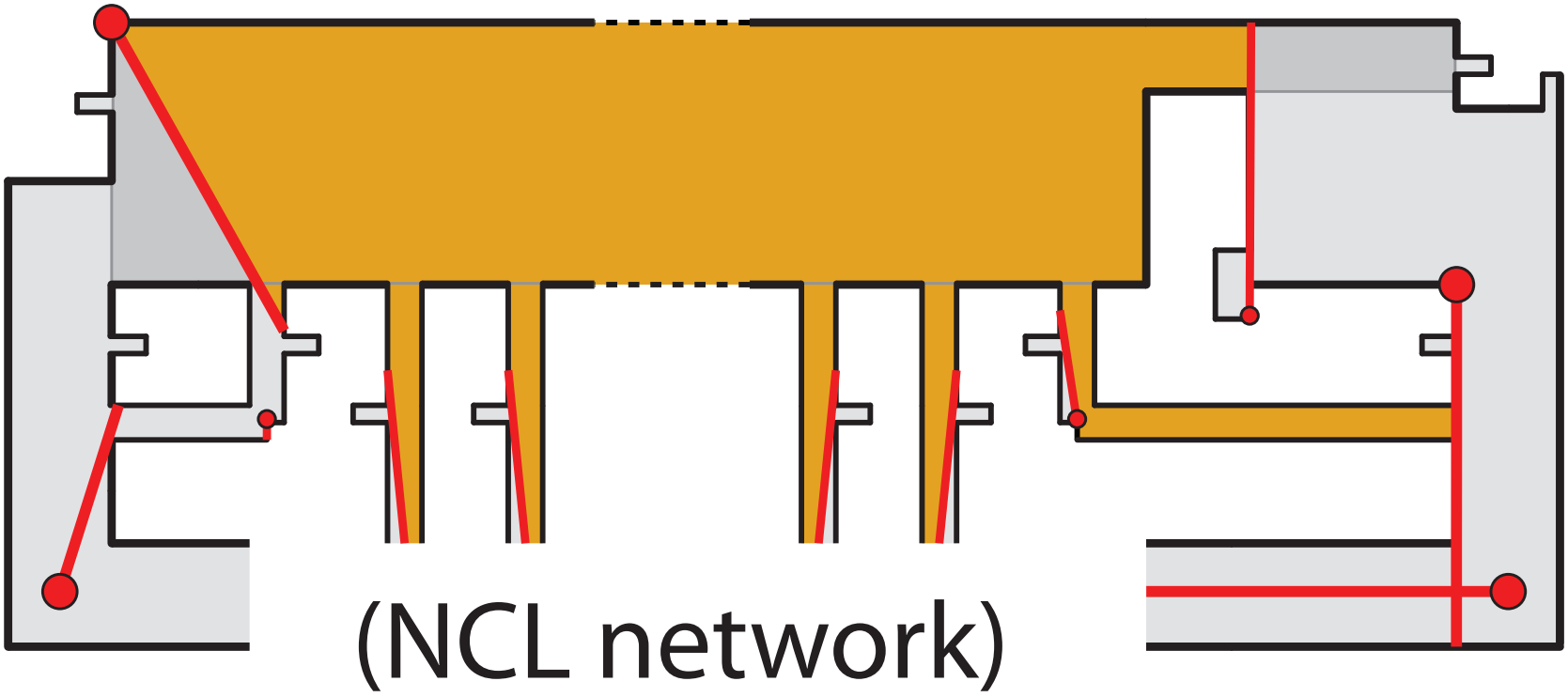
Clearing process



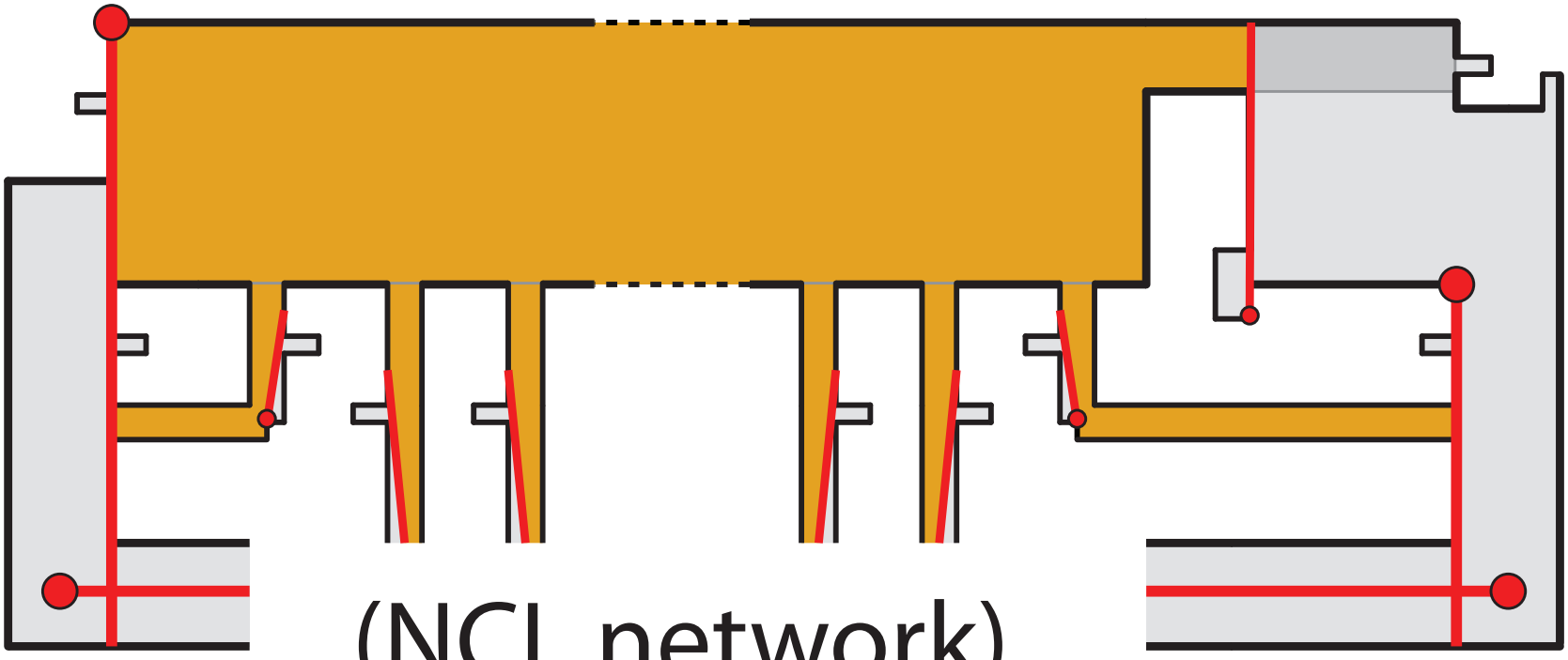
Clearing process



Clearing process

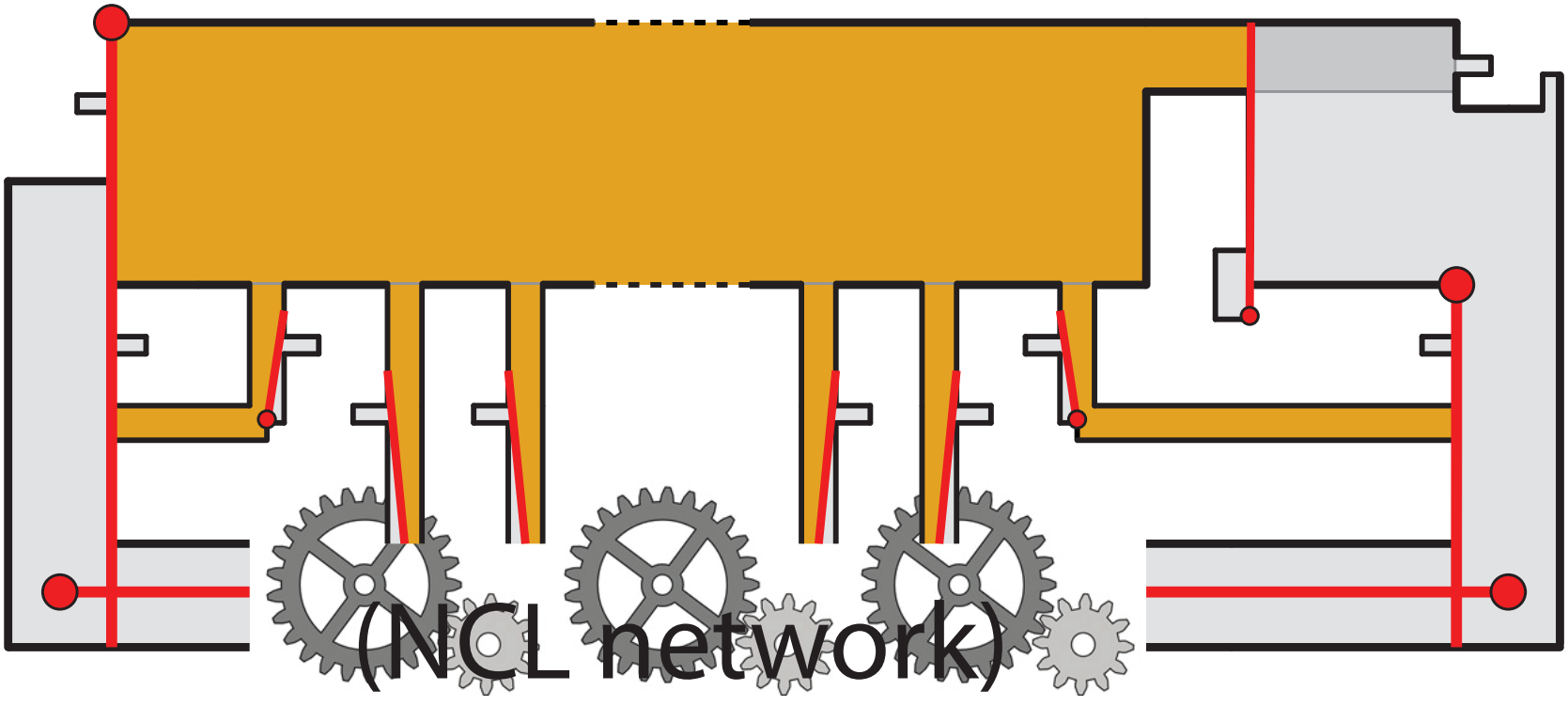


Clearing process

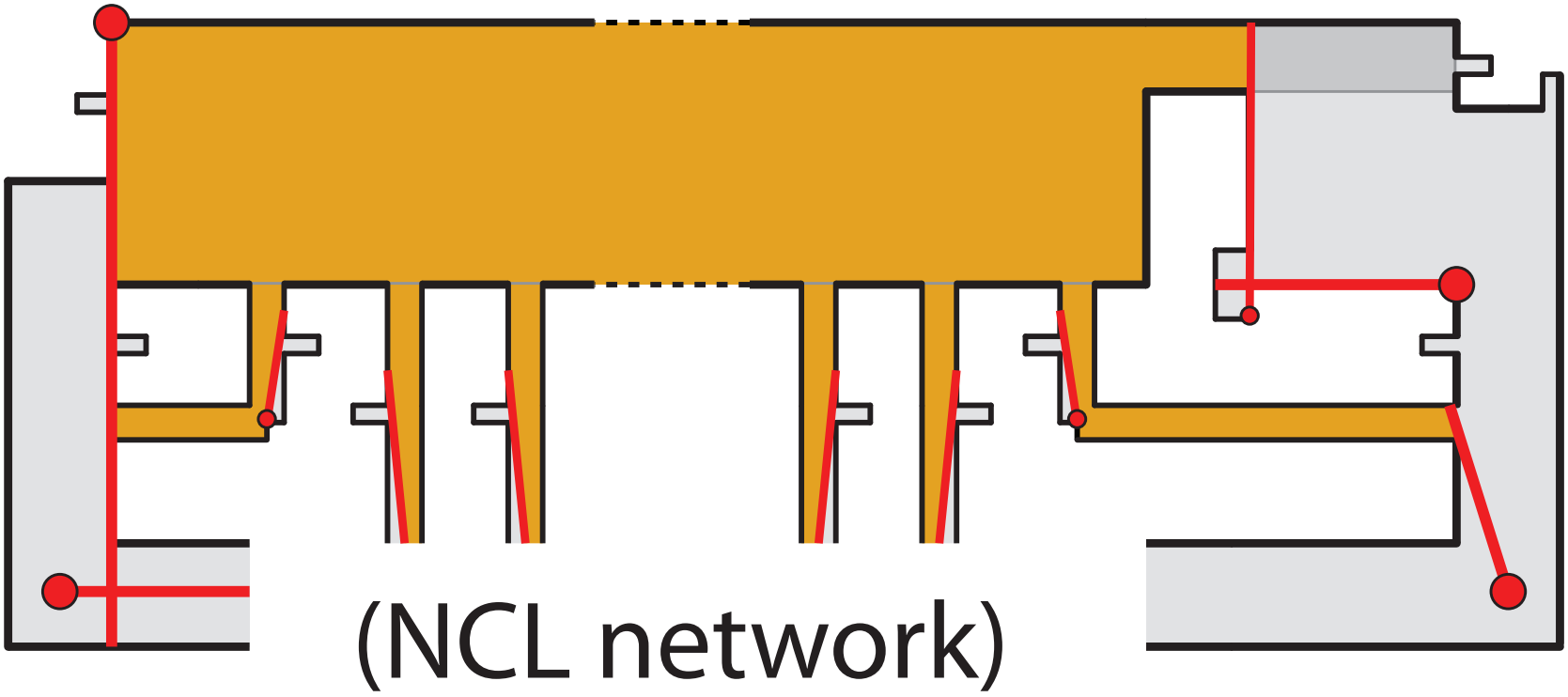


(NCL network)

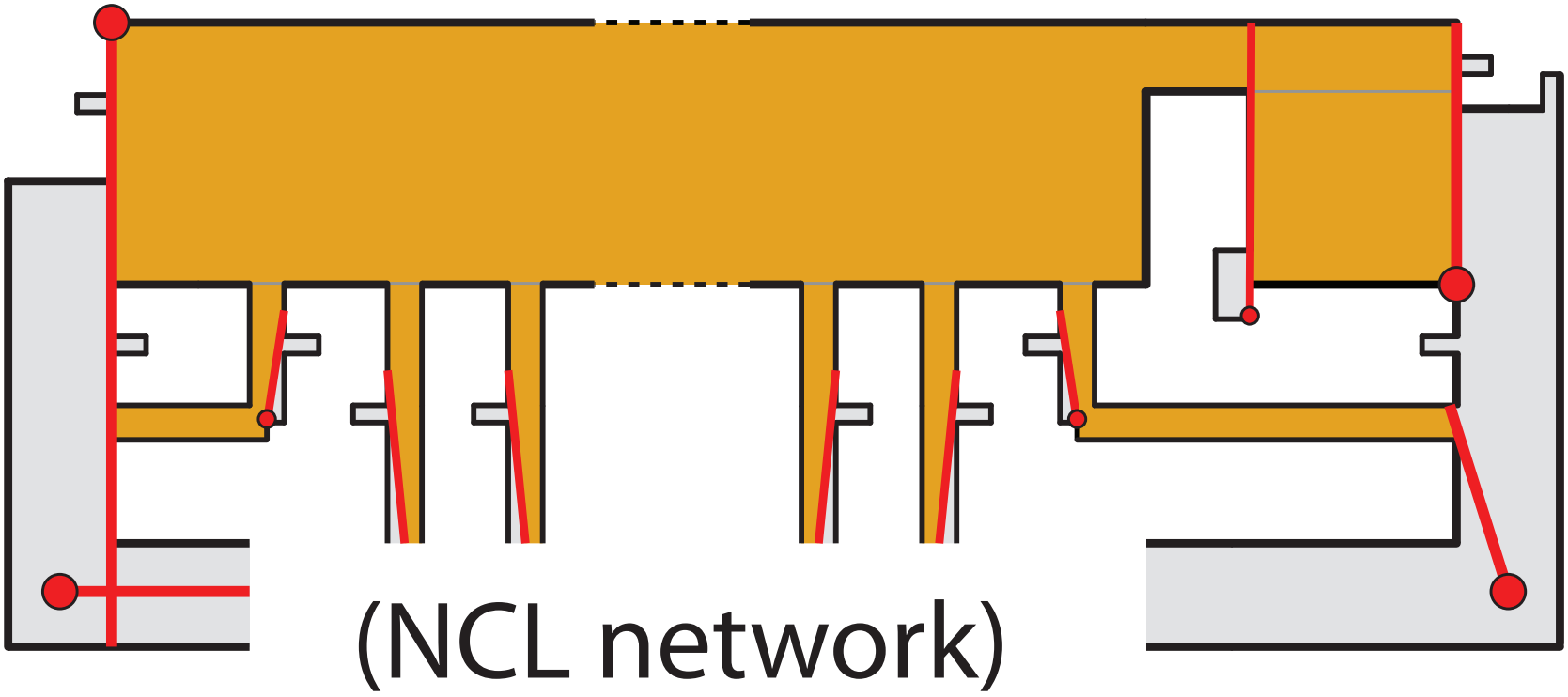
Clearing process



Clearing process



Clearing process



Conclusions and open problems

- Partial Searchlight Scheduling is strongly **PSPACE**-complete, even for orthogonal polygons and rectangular target areas.

Conclusions and open problems

- Partial Searchlight Scheduling is strongly **PSPACE**-complete, even for orthogonal polygons and rectangular target areas.

- Full Searchlight Scheduling is (obviously) in **PSPACE**.
  Is it also **PSPACE**-complete?

Conclusions and open problems

- Partial Searchlight Scheduling is strongly **PSPACE**-complete, even for orthogonal polygons and rectangular target areas.
- Full Searchlight Scheduling is (obviously) in **PSPACE**.
  Is it also **PSPACE**-complete?
- Generalizing to 3D polyhedral environments, Partial Searchlight Scheduling stays (obviously) **PSPACE**-hard, but is it in **PSPACE**?

Conclusions and open problems

- Partial Searchlight Scheduling is strongly **PSPACE**-complete, even for orthogonal polygons and rectangular target areas.
- Full Searchlight Scheduling is (obviously) in **PSPACE**.
  Is it also **PSPACE**-complete?
- Generalizing to 3D polyhedral environments, Partial Searchlight Scheduling stays (obviously) **PSPACE**-hard, but is it in **PSPACE**?
- 3D Full Searchlight Scheduling is **NP**-hard for general polyhedra, but is it in **NP**?
  What about orthogonal polyhedra?

References

R. A. Hearn and E. D. Demaine.
PSPACE-completeness of sliding-block puzzles and other problems through the nondeterministic constraint logic model of computation.
Theor. Comput. Sci. **343** (2005), 72–96, special issue "Game Theory Meets Theoretical Computer Science."

K. J. Obermeyer, A. Ganguli, and F. Bullo.
A complete algorithm for searchlight scheduling.
Int. J. Comput. Geom. Ap. **21** (2011), 101–130.

K. Sugihara, I. Suzuki, and M. Yamashita.
The searchlight scheduling problem.
SIAM J. Comput. **19** (1990), 1024–1040.

G. Viglietta.
Searching polyhedra by rotating half-planes.
Preprint (2011), `http://arxiv.org/abs/1104.4137`.