

Optimal Grid Exploration by Asynchronous Oblivious Robots

Franck Petit

INRIA/LIP6-CNRS/UPMC

Impossibility Results (I)

Impossibility Results (I)

✓ *Impossible if $k < 3$*

Impossibility Results (I)

✓ *Impossible if $k < 3$*

Remark

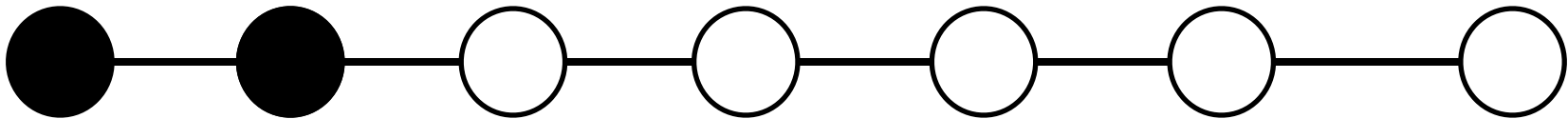
Any terminal configuration of any (probabilistic or deterministic) exploration protocol for a grid of n nodes using $k < n$ oblivious robots contains at least one tower.

Impossibility Results (I)

✓ *Impossible if $k < 3$*

Remark

Any terminal configuration of any (probabilistic or deterministic) exploration protocol for a grid of n nodes using $k < n$ oblivious robots contains at least one tower.

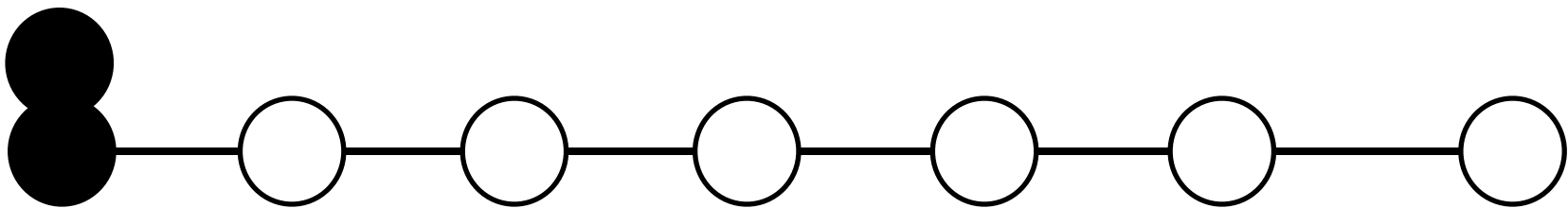


Impossibility Results (I)

✓ *Impossible if $k < 3$*

Remark

Any terminal configuration of any (probabilistic or deterministic) exploration protocol for a grid of n nodes using $k < n$ oblivious robots contains at least one tower.

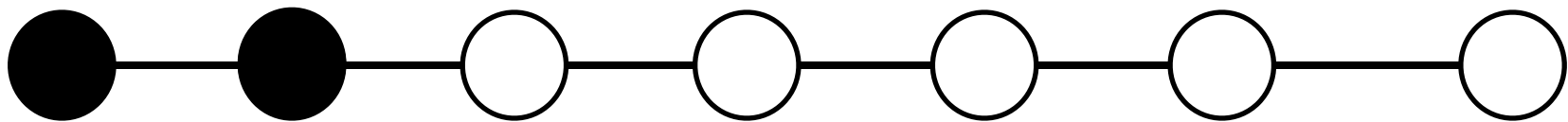


Impossibility Results (I)

✓ *Impossible if $k < 3$*

Remark

Any terminal configuration of any (probabilistic or deterministic) exploration protocol for a grid of n nodes using $k < n$ oblivious robots contains at least one tower.



Initial configuration

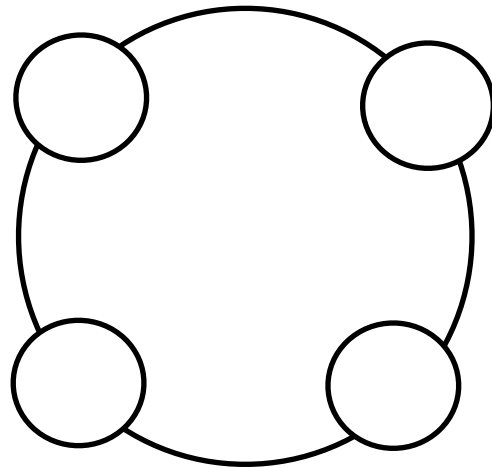
Impossibility Results (2)

Impossibility Results (2)

- ✓ *Impossible for (2,2)-grid if $k=3$*

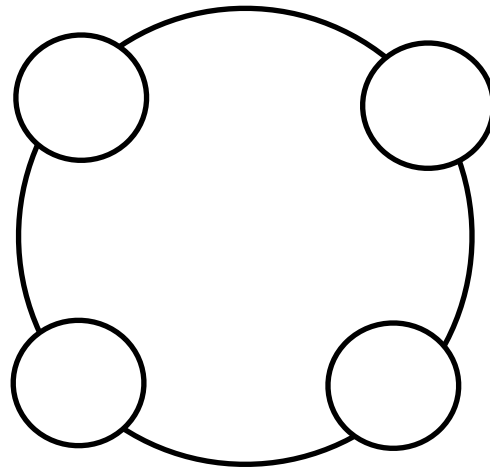
Impossibility Results (2)

✓ *Impossible for (2,2)-grid if $k=3$*



Impossibility Results (2)

✓ *Impossible for (2,2)-grid if $k=3$*



Impossible [Devismes 2009]

Impossibility Results (3)

✓ *Impossible for (3,3)-grid if $k=3$*

Impossibility Results (3)

✓ *Impossible for (3,3)-grid if $k=3$*

□ Tower of size 3 \longrightarrow At most one new node is visited

Impossibility Results (3)

✓ *Impossible for (3,3)-grid if $k=3$*

❑ Tower of size 3 \longrightarrow At most one new node is visited

❑ Tower of size 2

- The tower remains idle \longrightarrow 4 new nodes are visited
(7 distinct nodes are visited in total)

Impossibility Results (3)

✓ *Impossible for (3,3)-grid if $k=3$*

❑ Tower of size 3 \longrightarrow At most one new node is visited

❑ Tower of size 2

• The tower remains idle \longrightarrow 4 new nodes are visited
(7 distinct nodes are visited in total)

Impossible since $n=6$

Impossibility Results (3)

✓ *Impossible for (3,3)-grid if $k=3$*

❑ Tower of size 3 \longrightarrow At most one new node is visited

❑ Tower of size 2

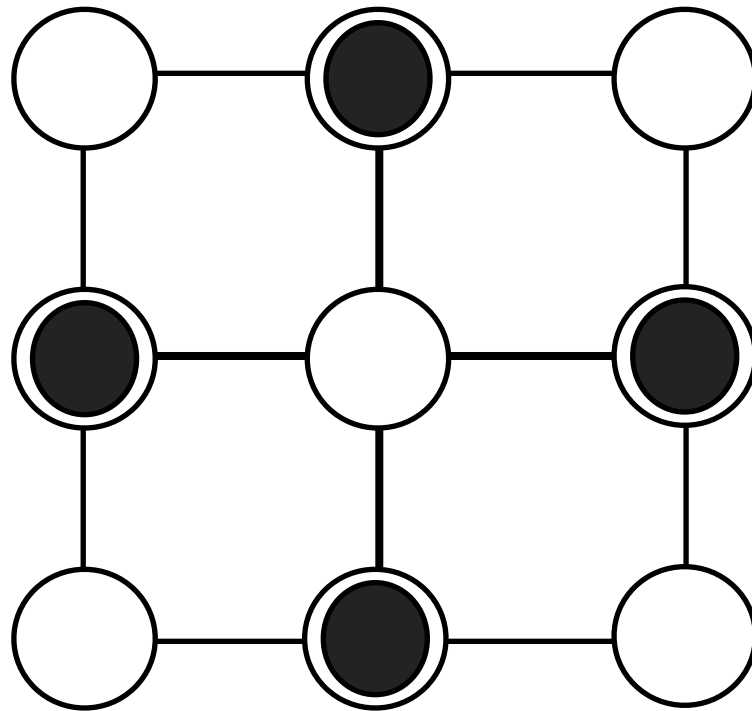
• The tower remains idle \longrightarrow 4 new nodes are visited
(7 distinct nodes are visited in total)

Impossible since $n=6$

• The tower moves \longrightarrow Multiplicity of size 3

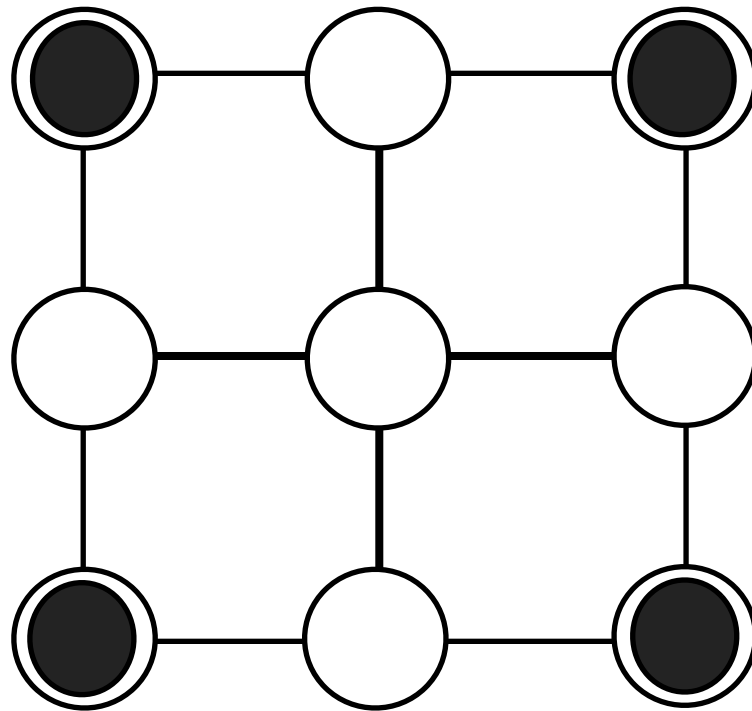
Impossibility Results (4)

✓ *Impossible for (3,3)-grid if $k=4$*



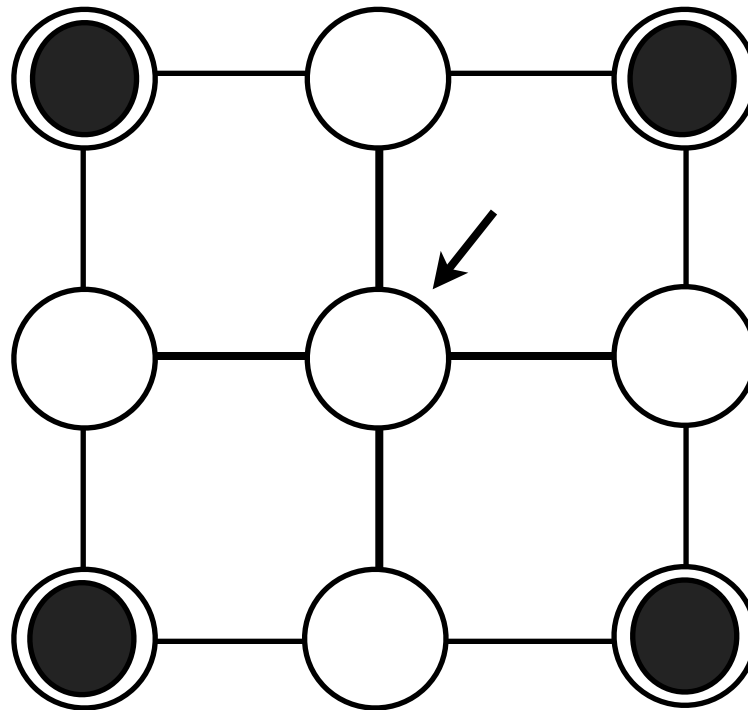
Impossibility Results (4)

✓ *Impossible for (3,3)-grid if $k=4$*



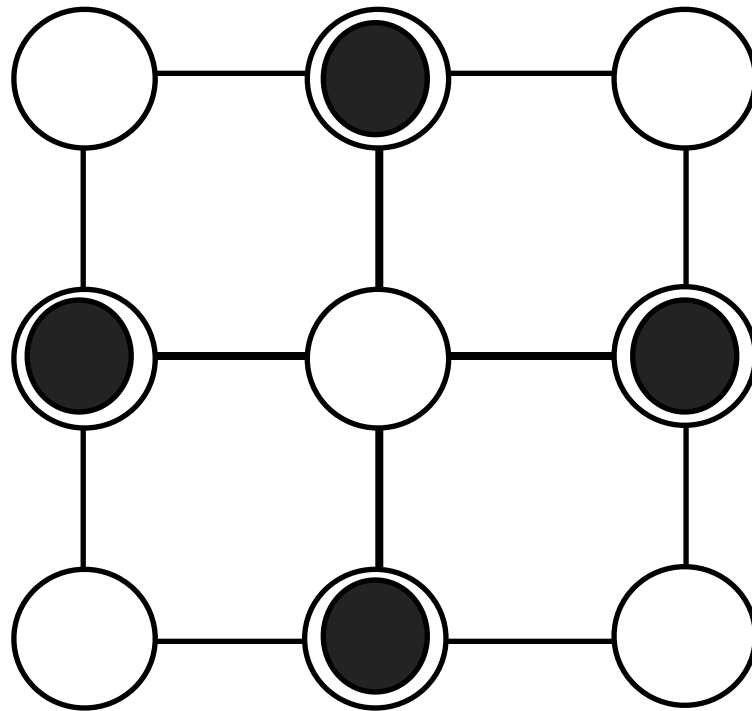
Impossibility Results (4)

✓ *Impossible for (3,3)-grid if $k=4$*



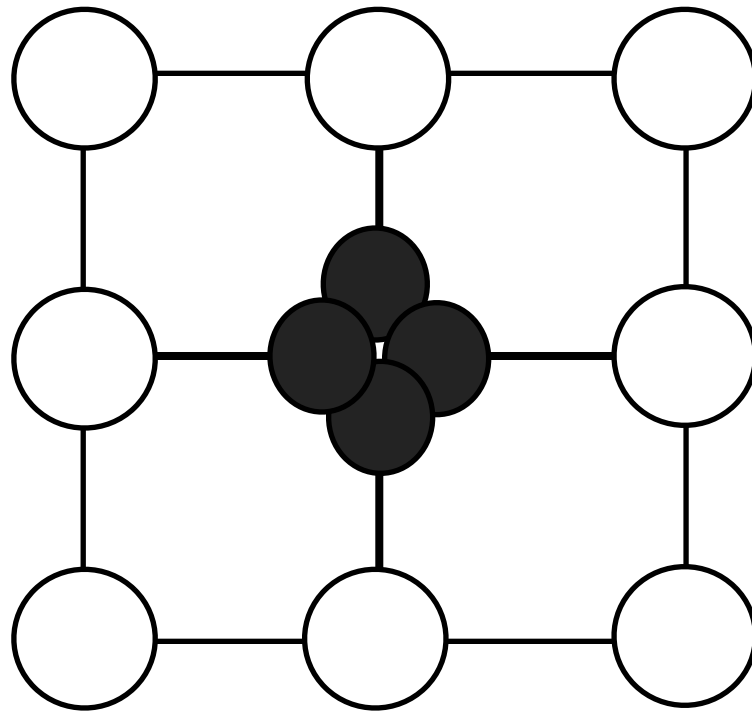
Impossibility Results (4)

✓ *Impossible for (3,3)-grid if $k=4$*



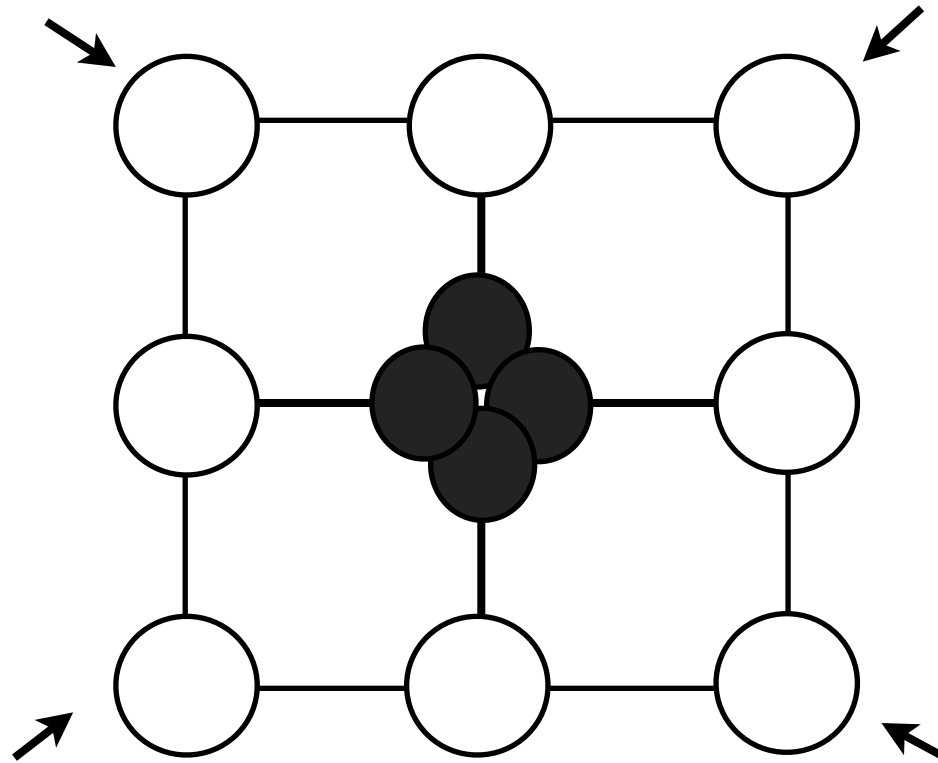
Impossibility Results (4)

✓ *Impossible for (3,3)-grid if $k=4$*



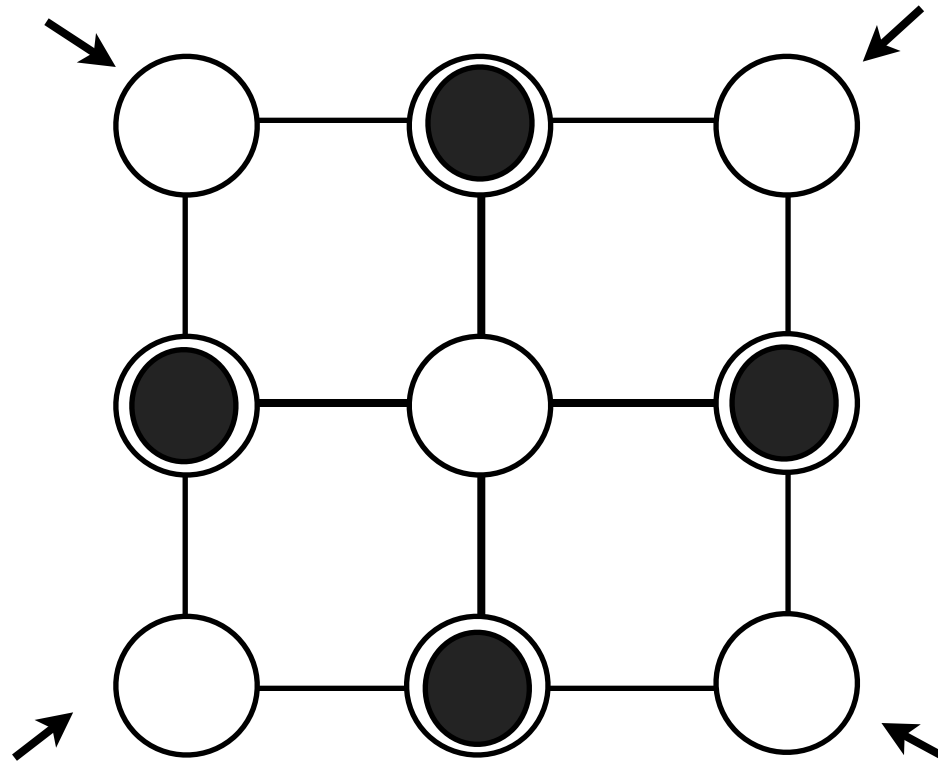
Impossibility Results (4)

✓ *Impossible for $(3,3)$ -grid if $k=4$*



Impossibility Results (4)

✓ *Impossible for $(3,3)$ -grid if $k=4$*



Algorithm

Algorithm

□ Setting

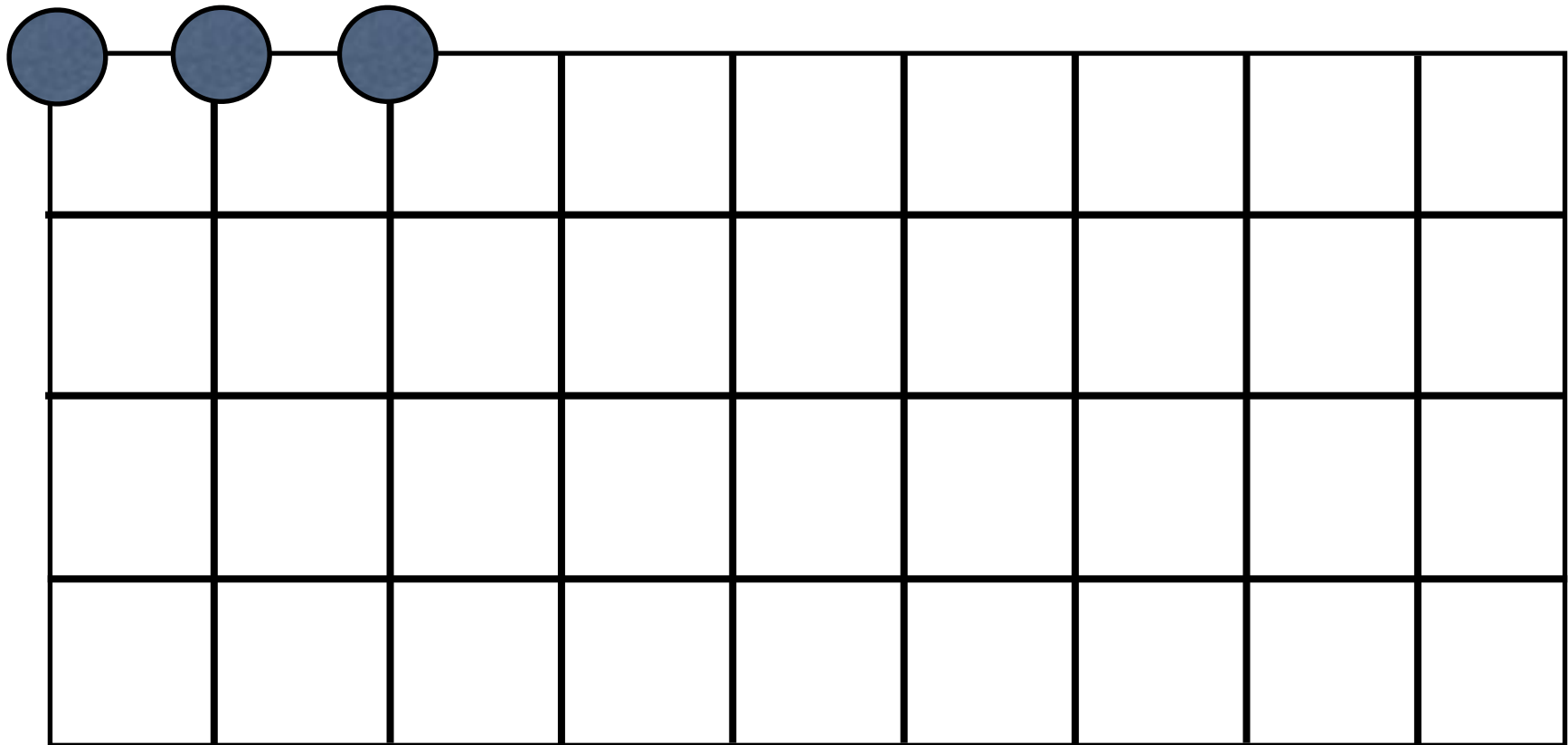
- ✓ (i,j) -grids such that $j > 3$
- ✓ $k=3$
- ✓ Towerless initial configuration

Algorithm

Algorithm

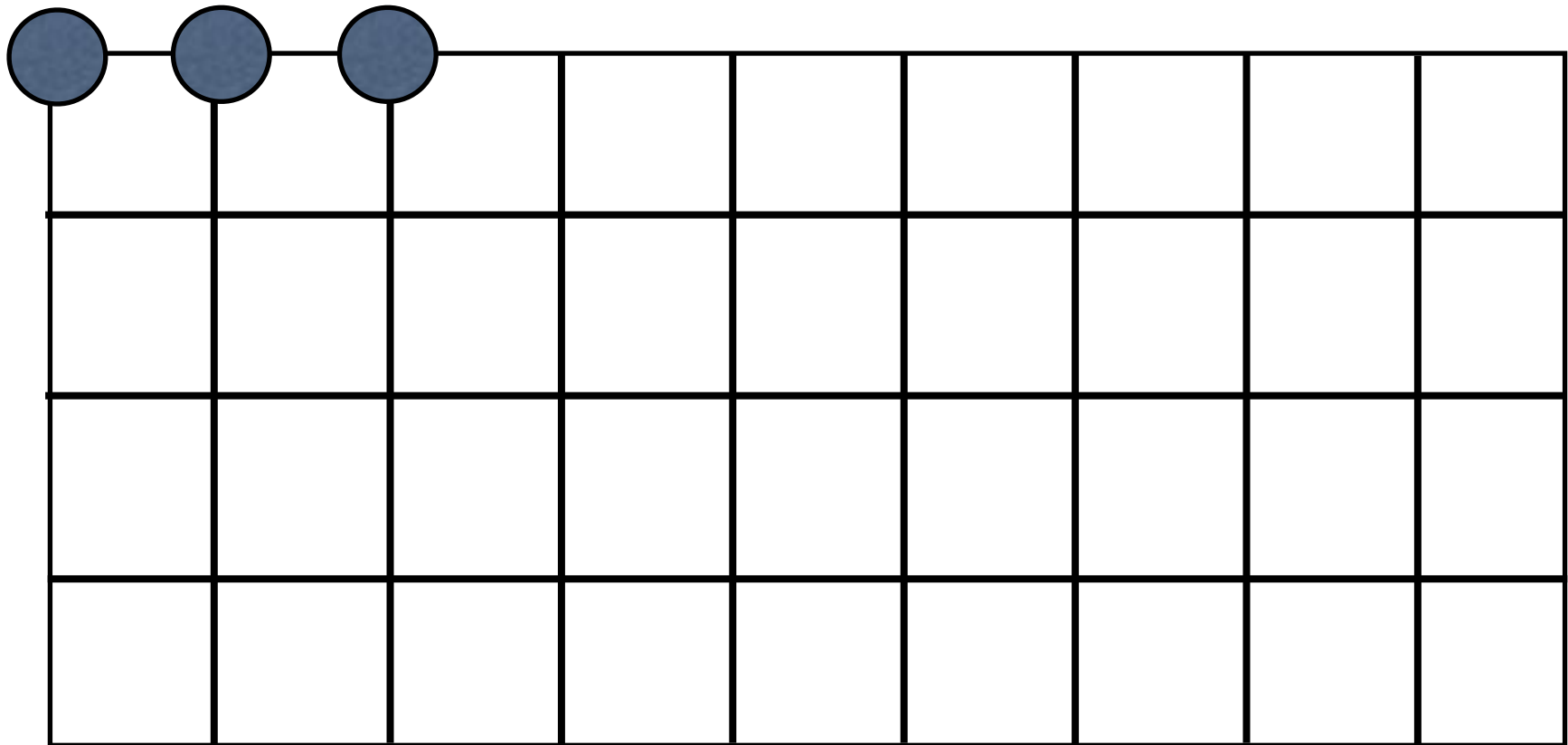
- ❑ Phase 1: Set-Up phase
- ❑ Phase 2: Orientation phase
- ❑ Phase 3: Exploration phase

Algorithm



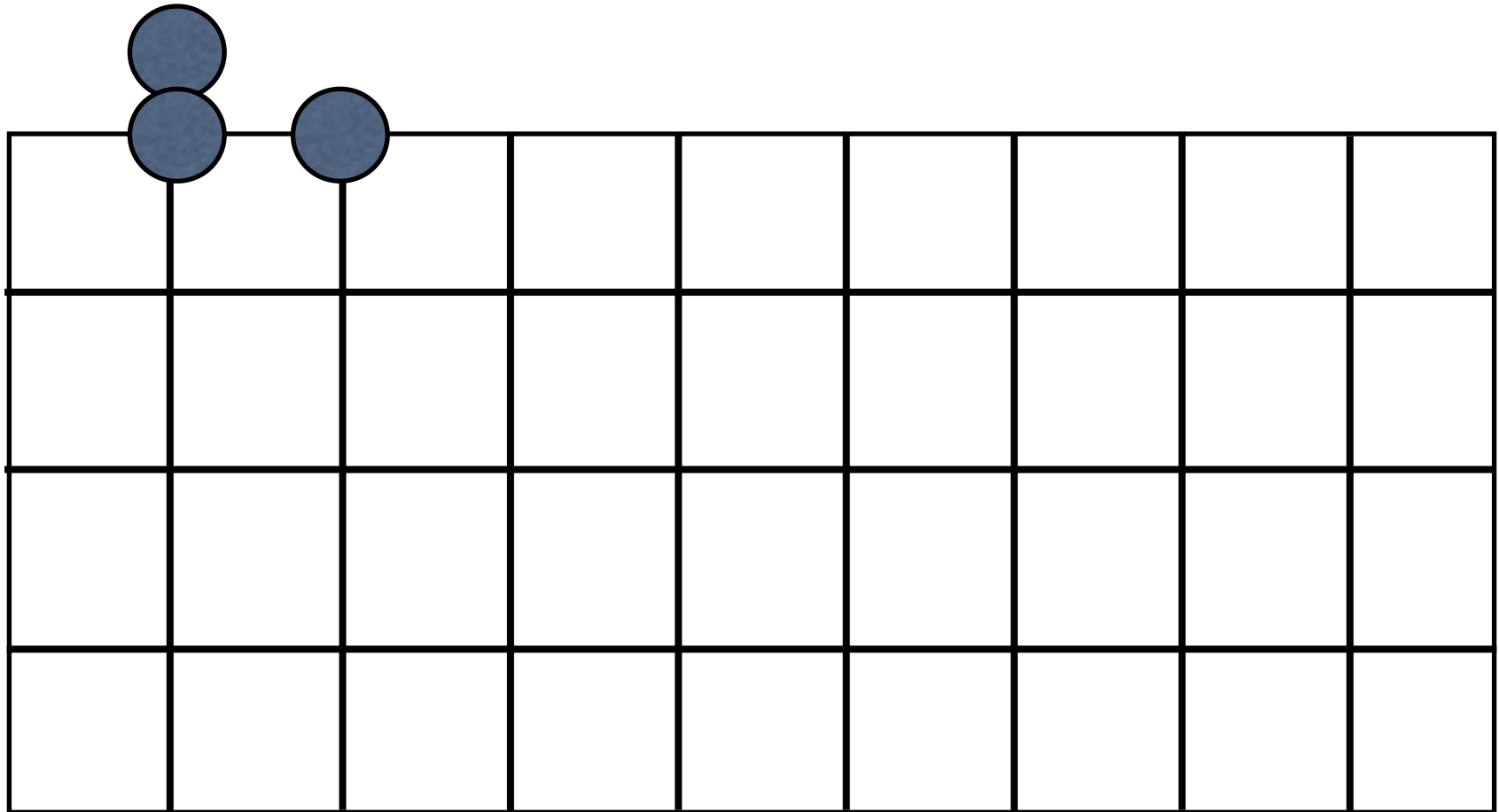
Algorithm

❑ Phase 2: Orientation phase



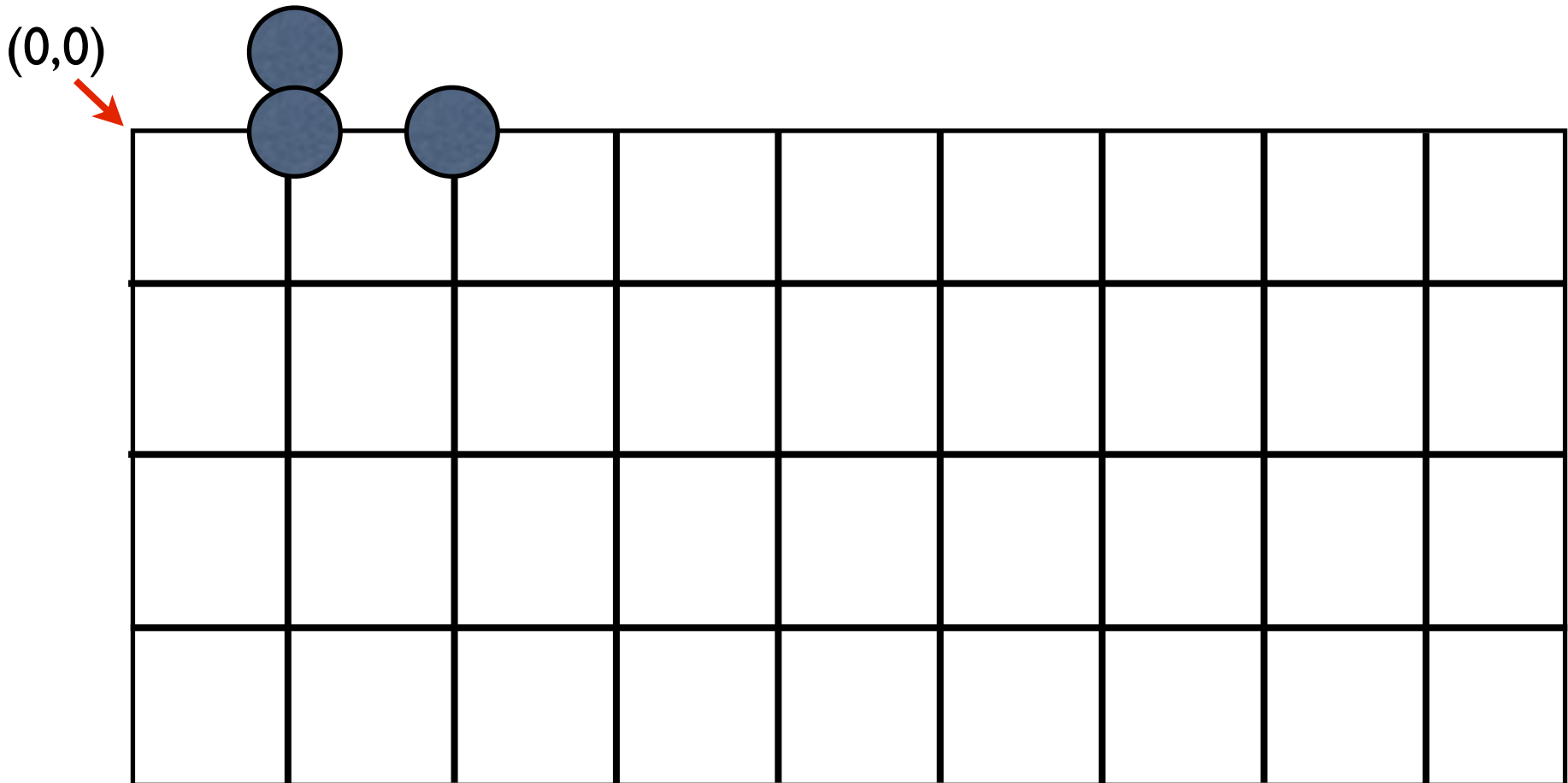
Algorithm

□ Phase 2: Orientation phase



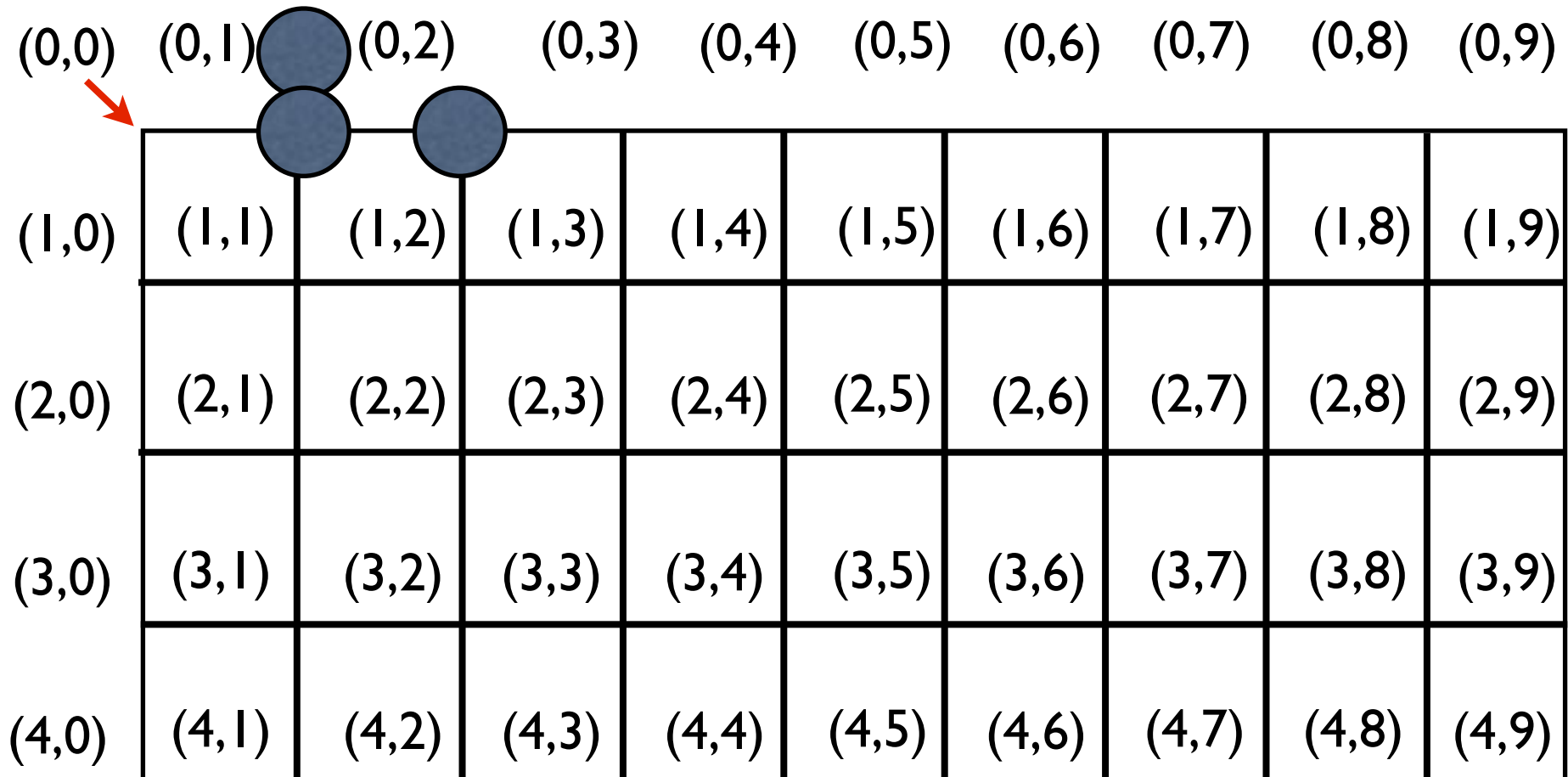
Algorithm

□ Phase 2: Orientation phase



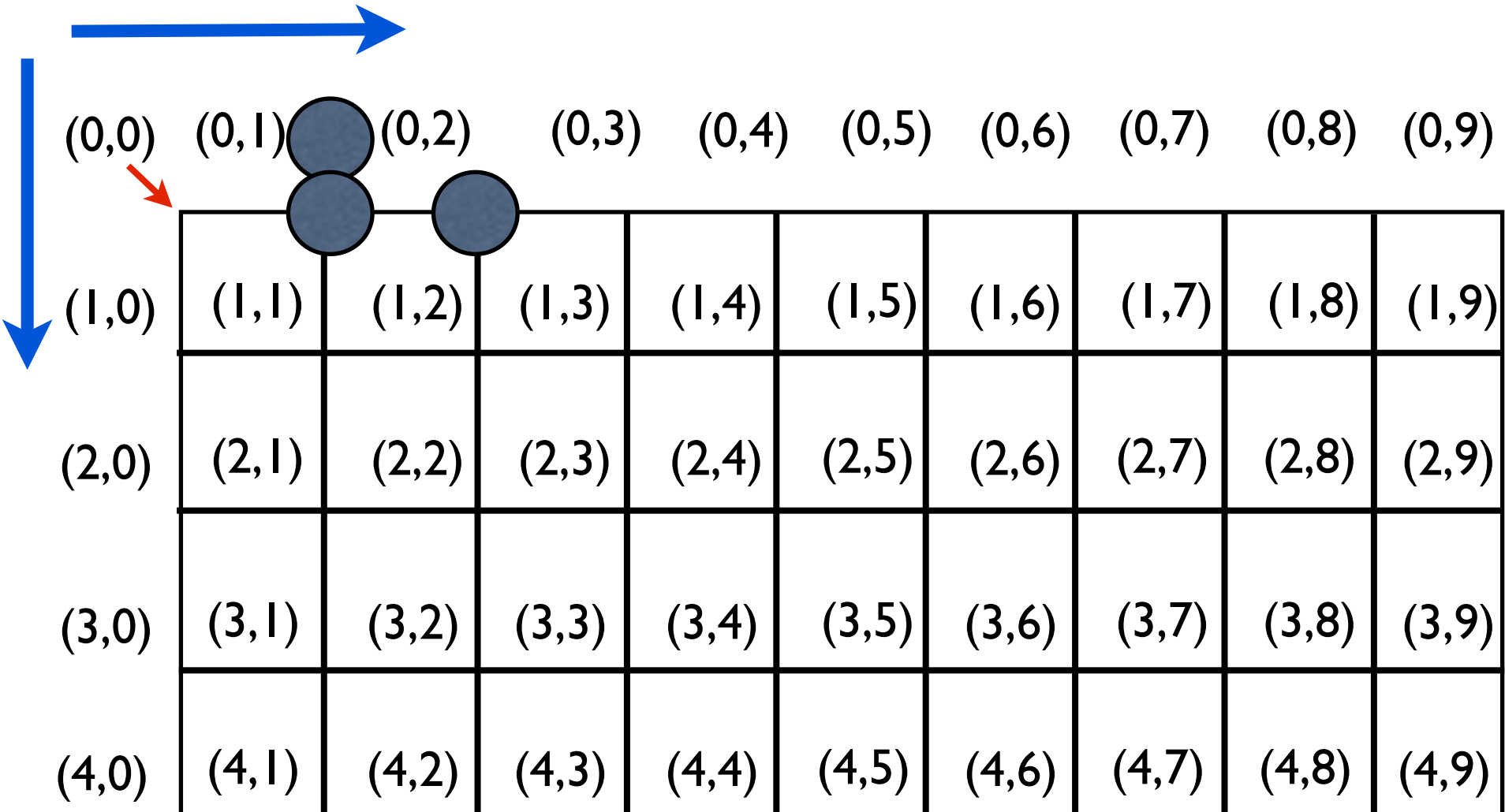
Algorithm

□ Phase 2: Orientation phase



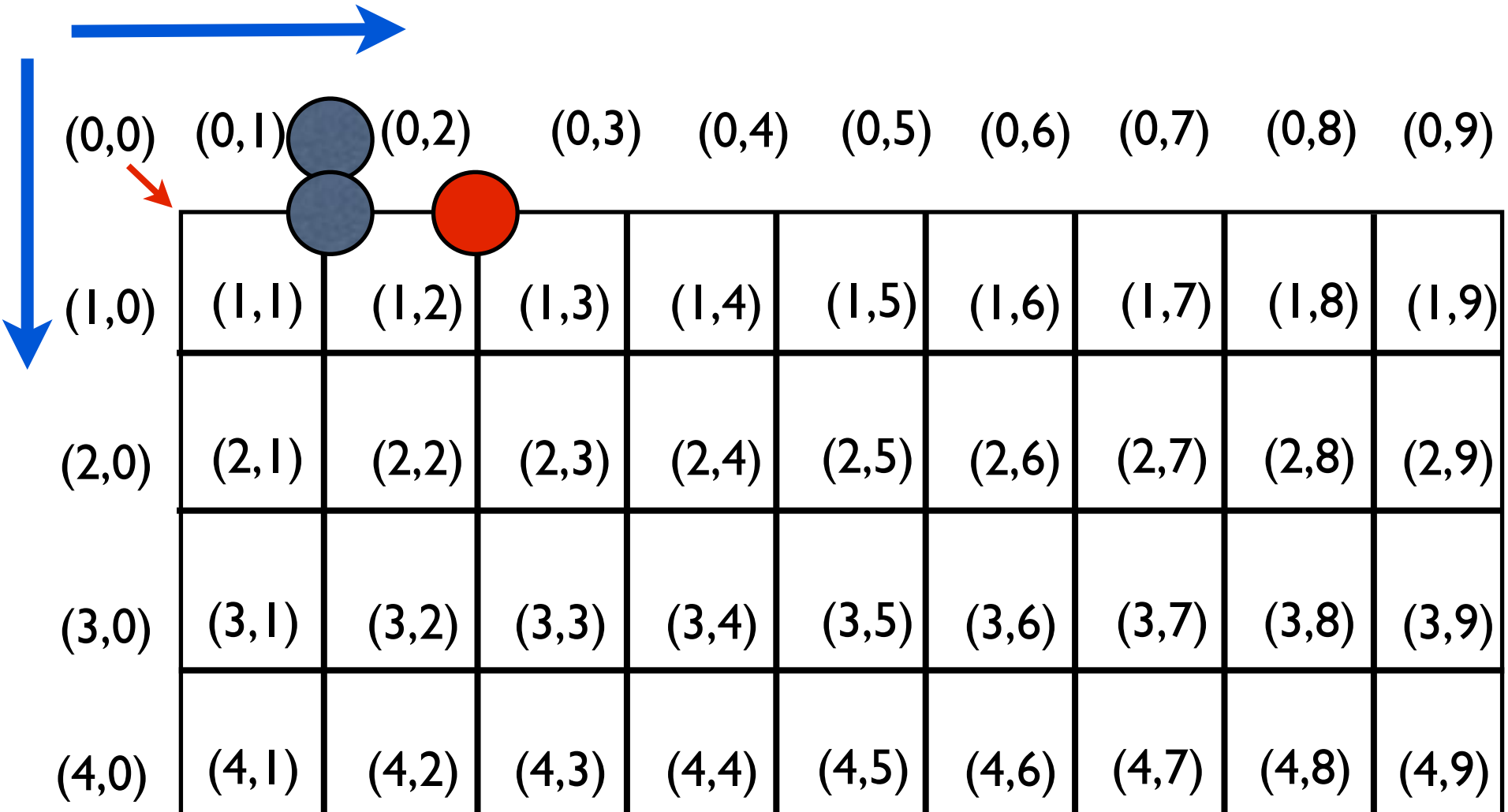
Algorithm

□ Phase 2: Orientation phase



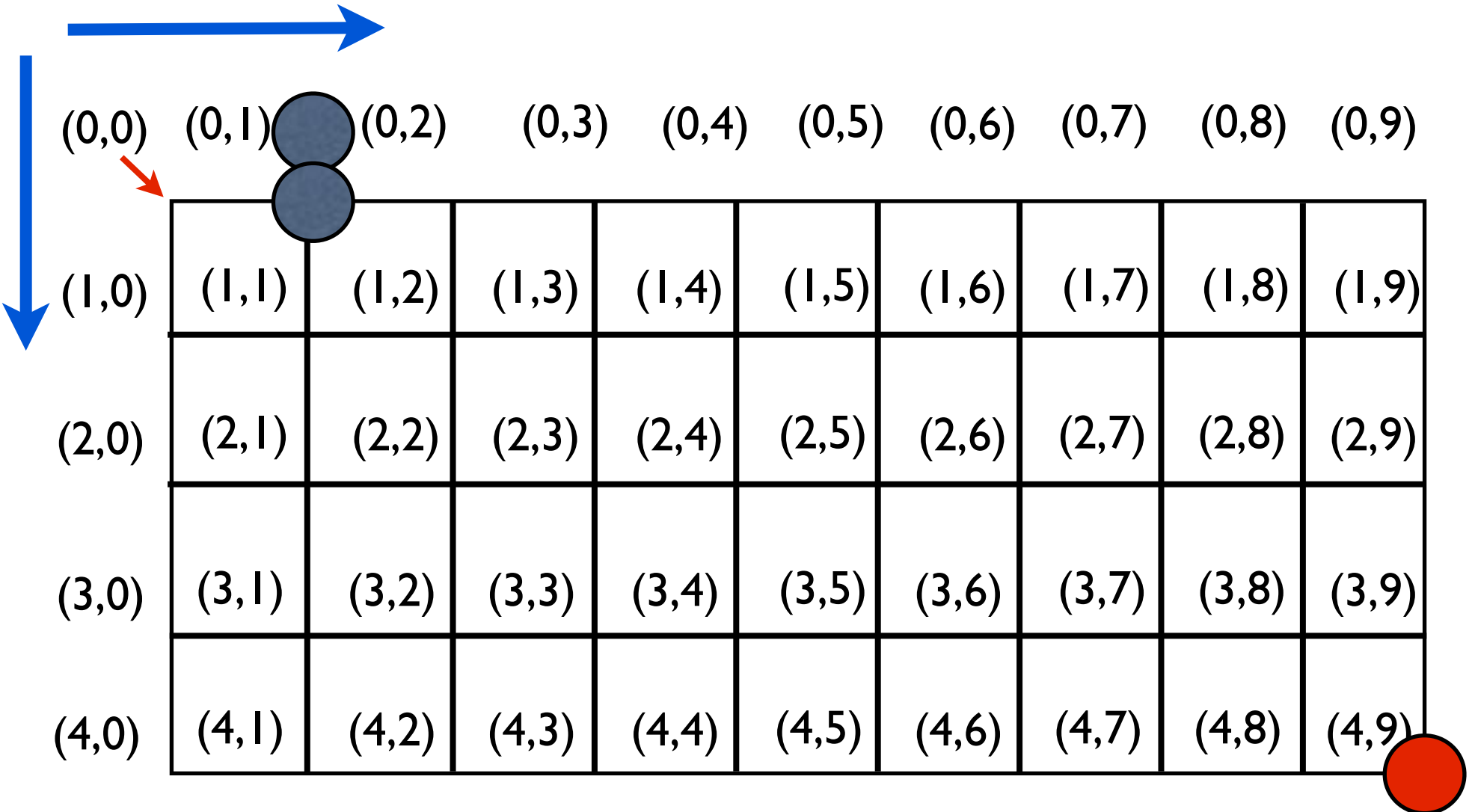
Algorithm

❑ Phase 3: Exploration phase



Algorithm

□ Phase 3: Exploration phase



Algorithm

Algorithm

□ Phase I: Set-Up

Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

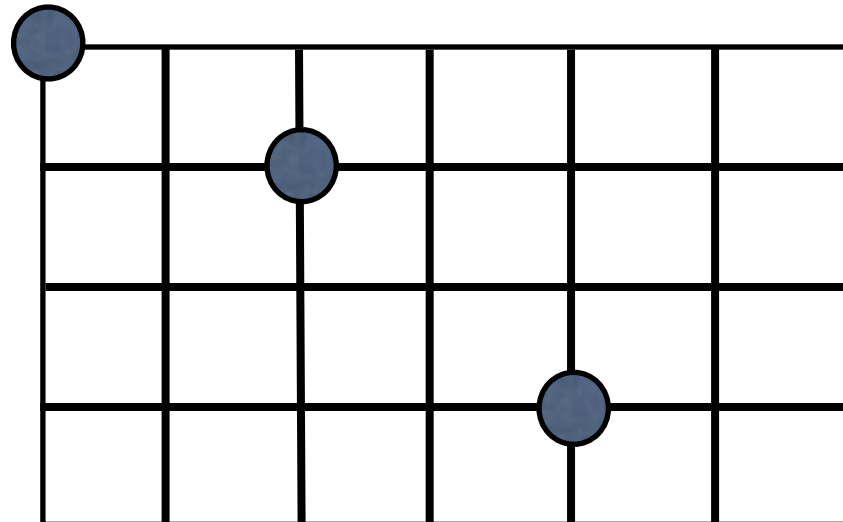
There is **exactly** one robot at a corner of the grid

Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

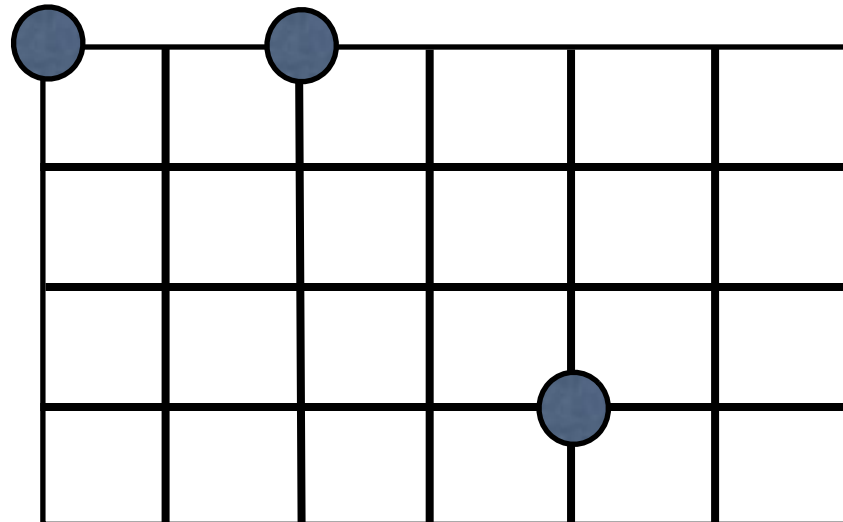


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

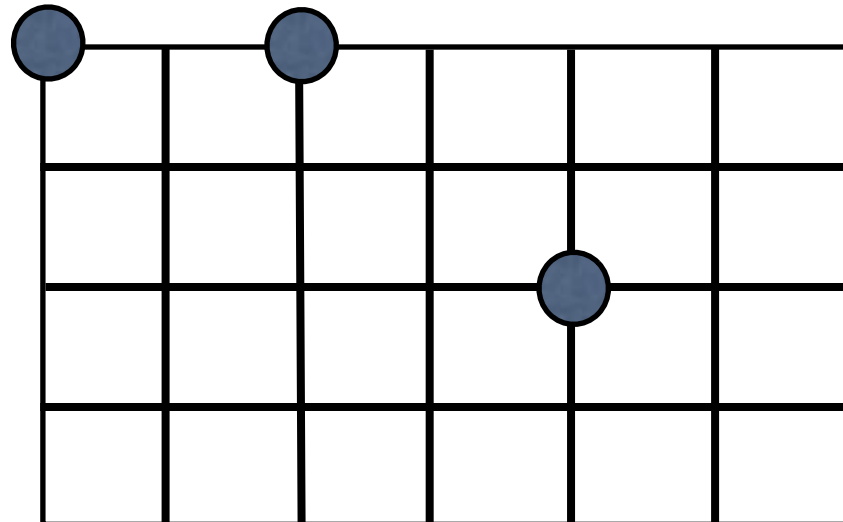


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

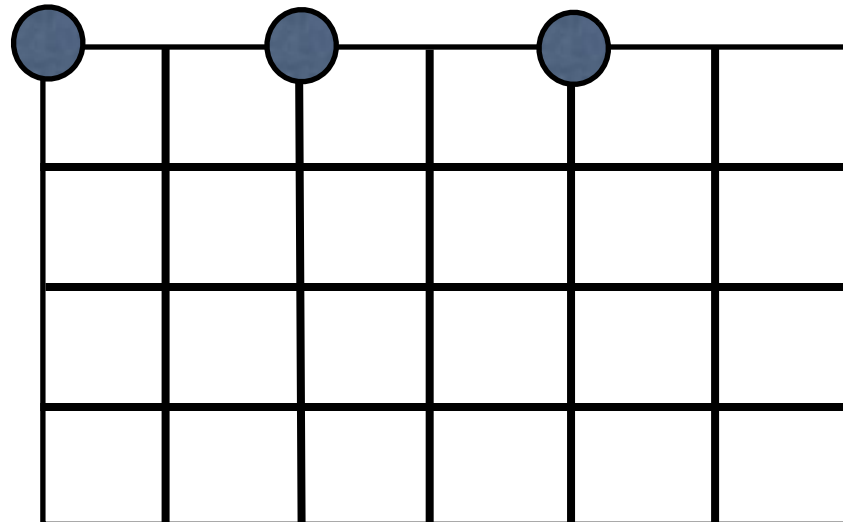


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

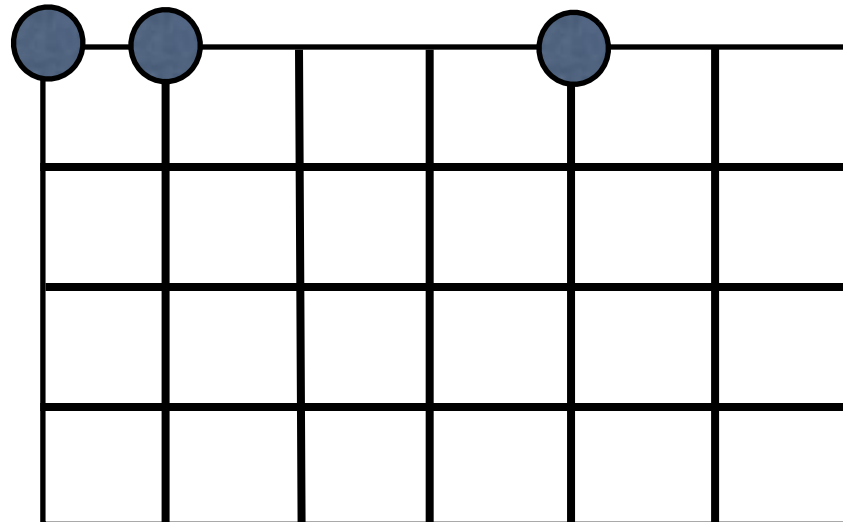


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

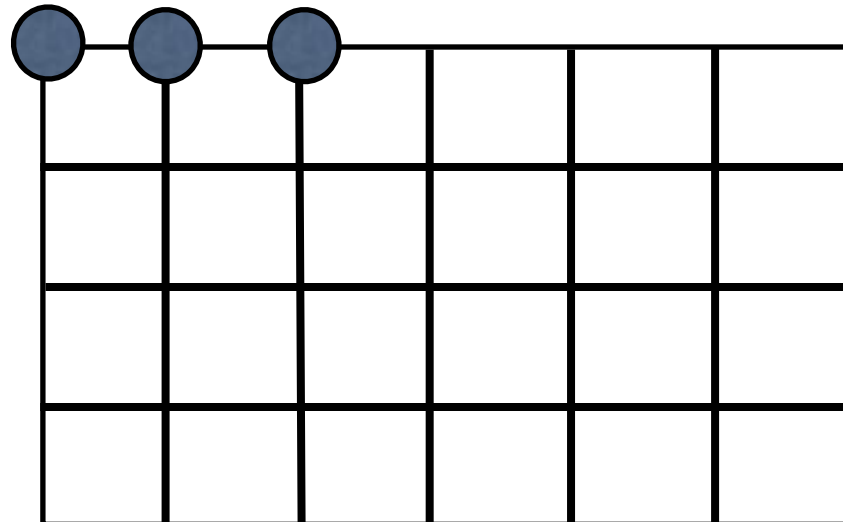


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid



Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

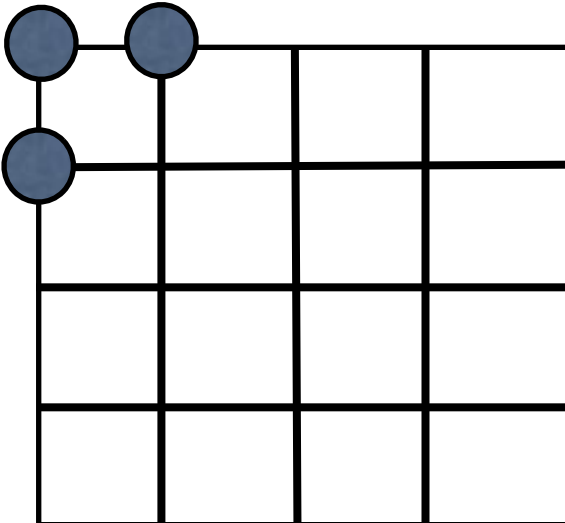
There is **exactly** one robot at a corner of the grid

Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

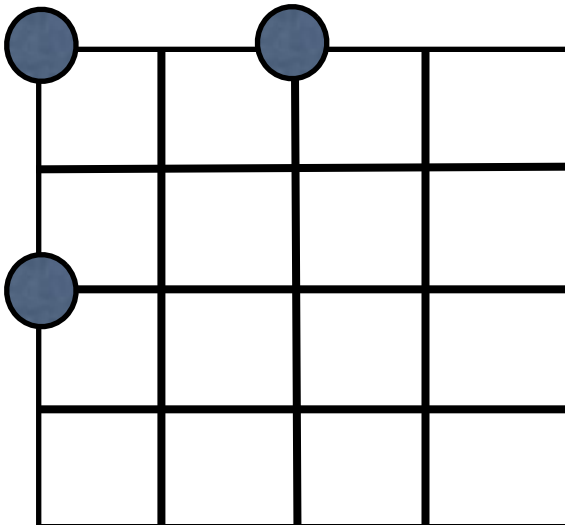


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

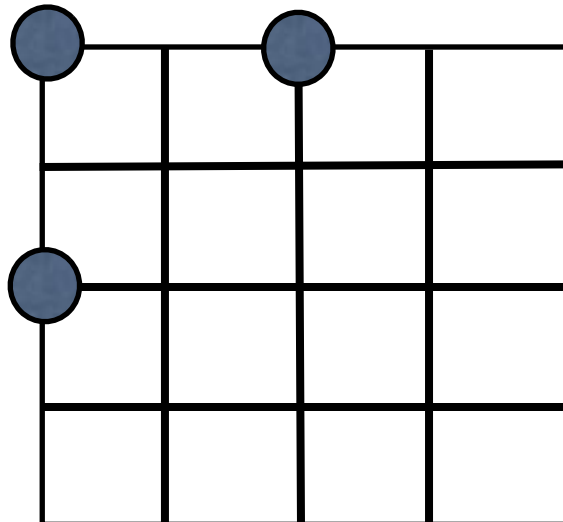
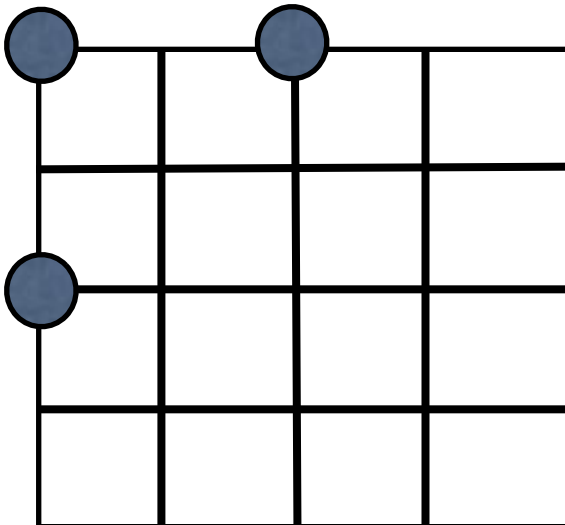


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

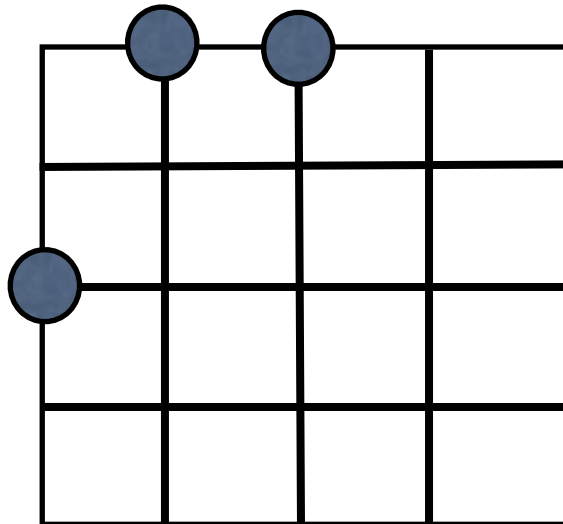
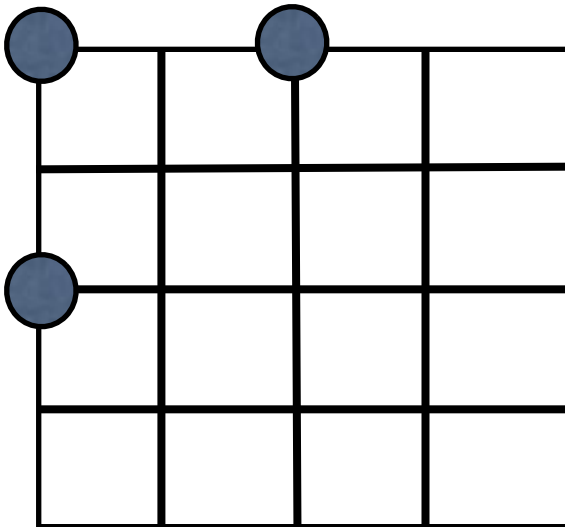


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

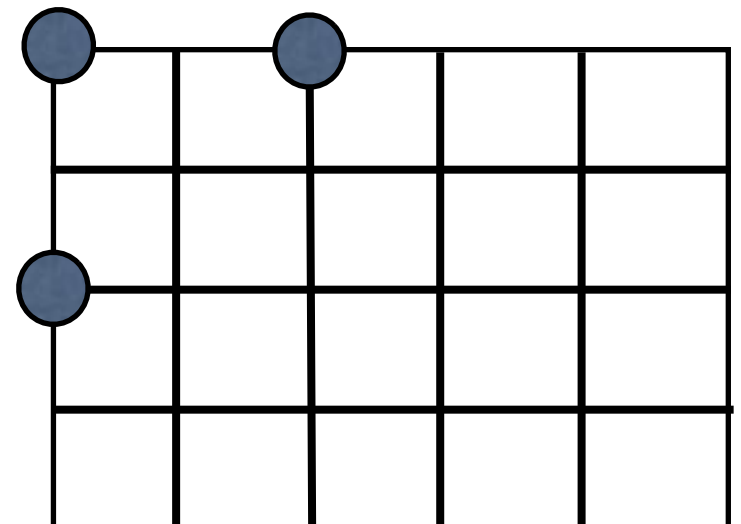
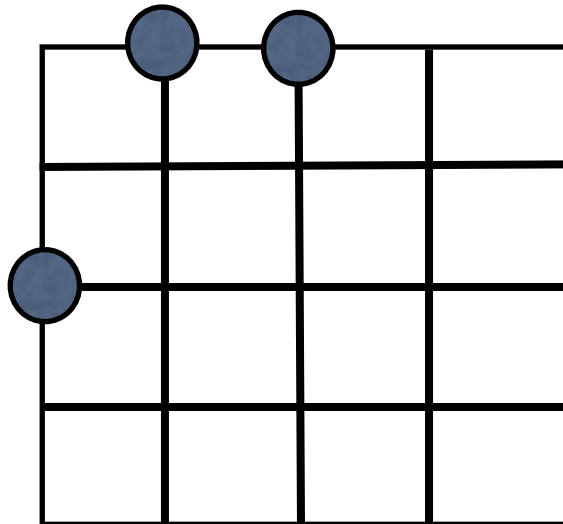
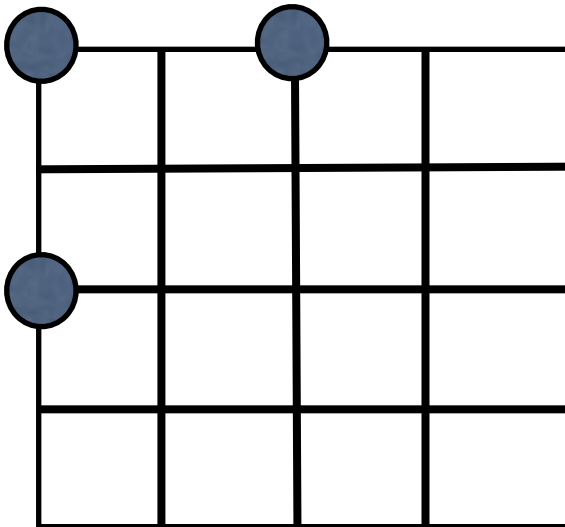


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid

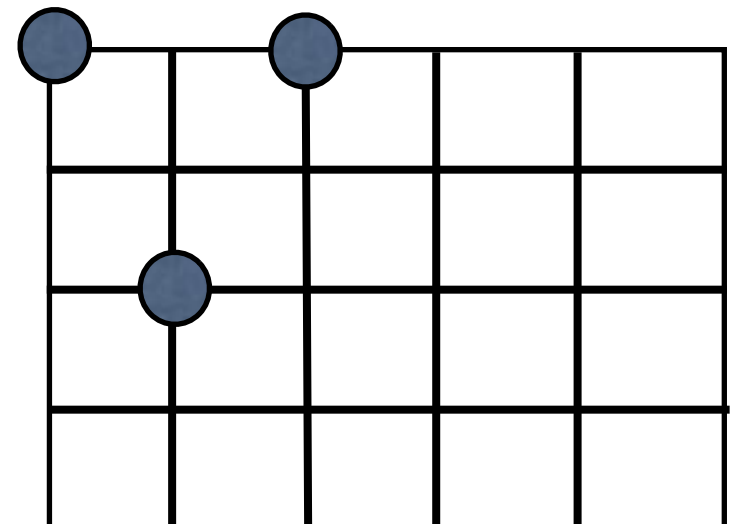
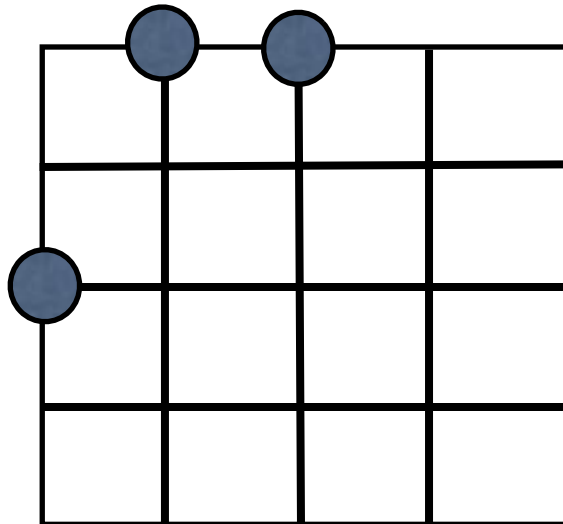
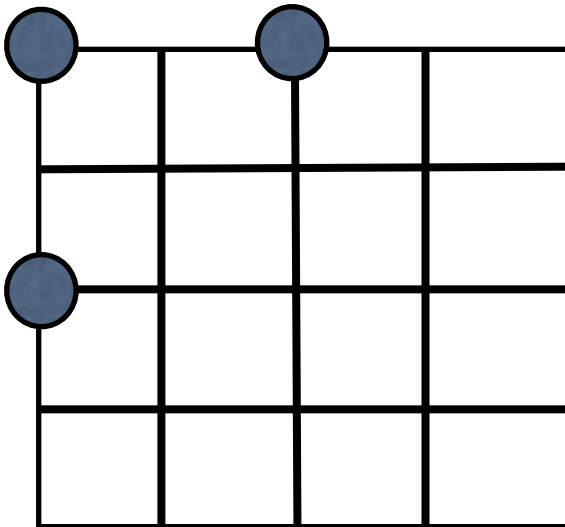


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Leader

There is **exactly** one robot at a corner of the grid



Algorithm

Algorithm

□ Phase I: Set-Up

Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Choice

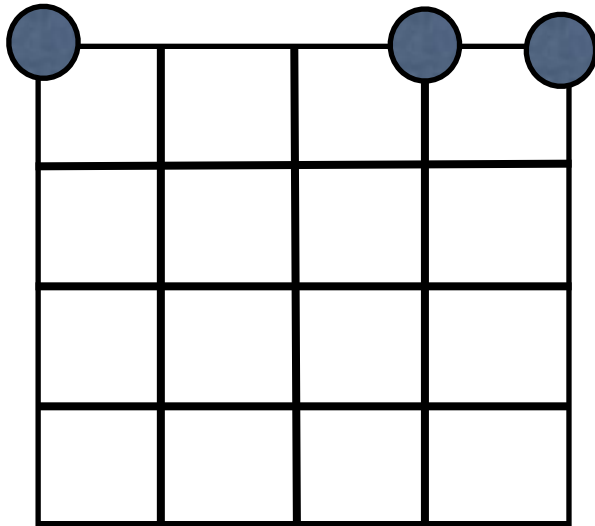
There are **at least two** robots at a corner of the grid

Algorithm

□ Phase I: Set-Up

✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid

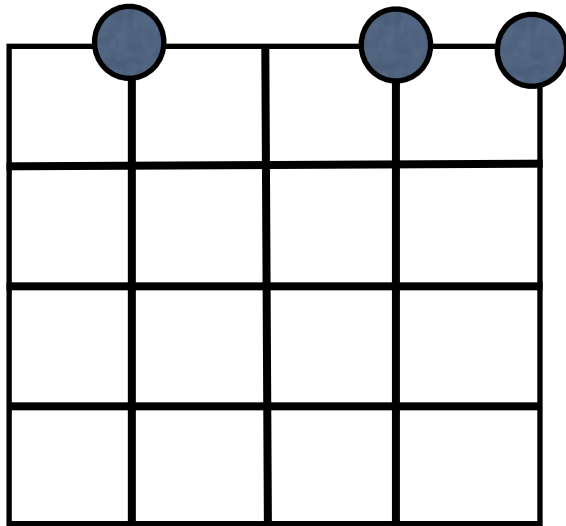


Algorithm

□ Phase I: Set-Up

✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid

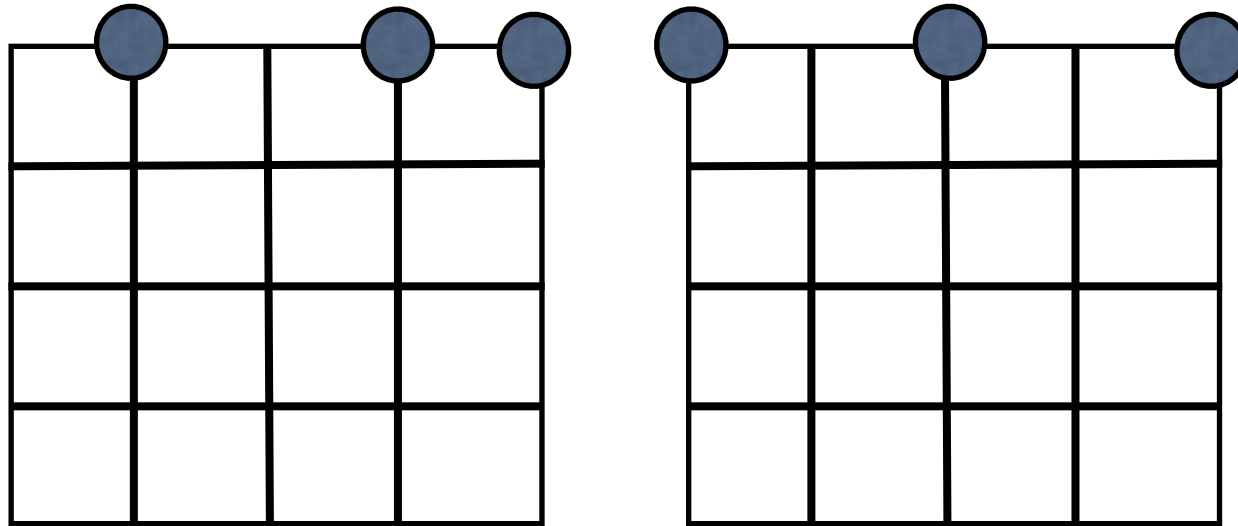


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid

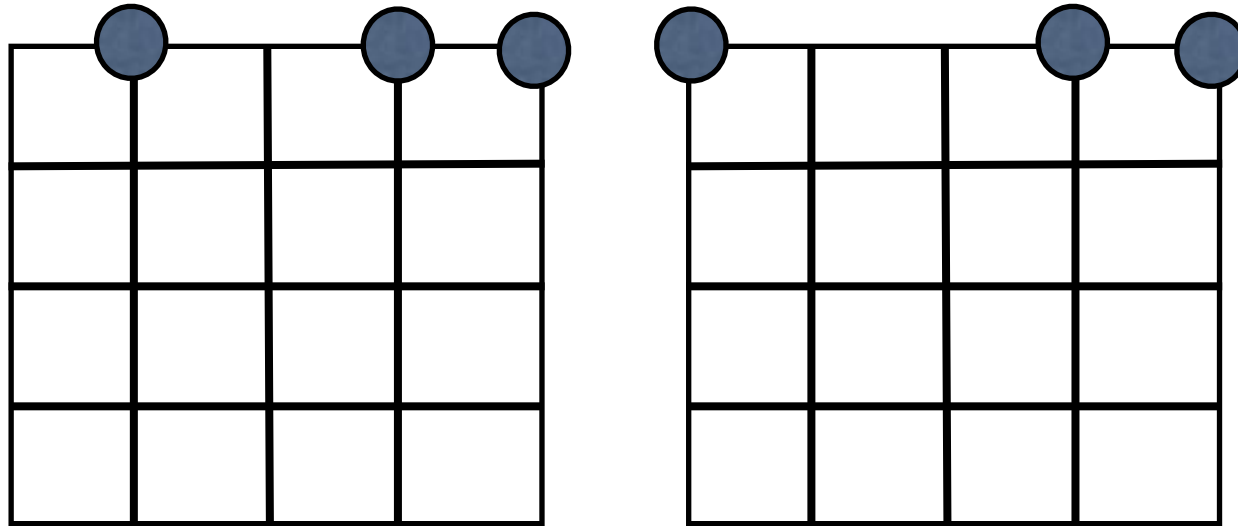


Algorithm

□ Phase I: Set-Up

✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid



Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Choice

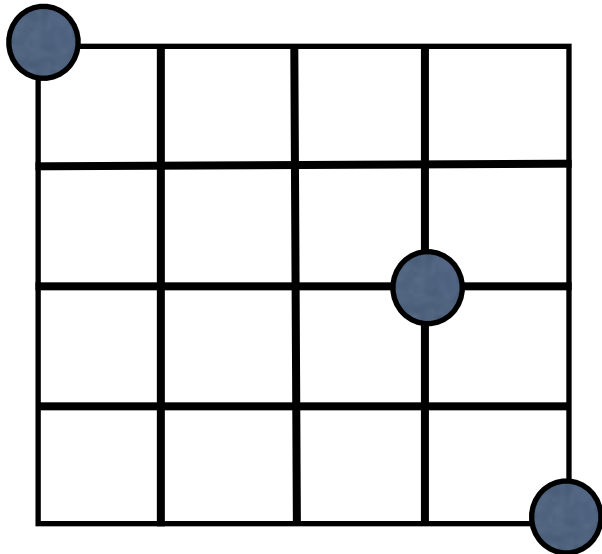
There are **at least two** robots at a corner of the grid

Algorithm

□ Phase I: Set-Up

✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid

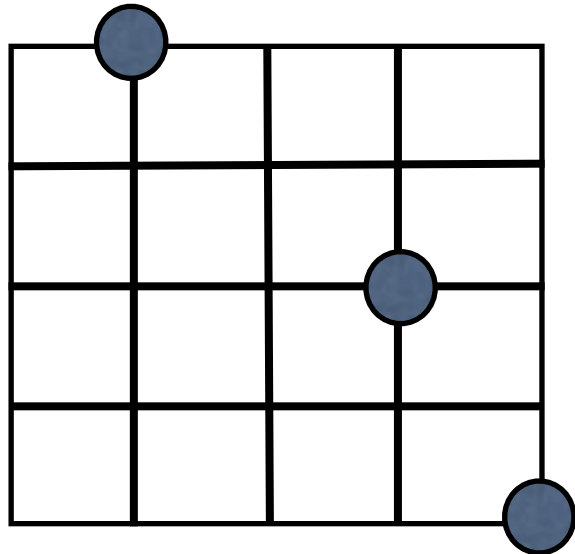


Algorithm

□ Phase I: Set-Up

✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid

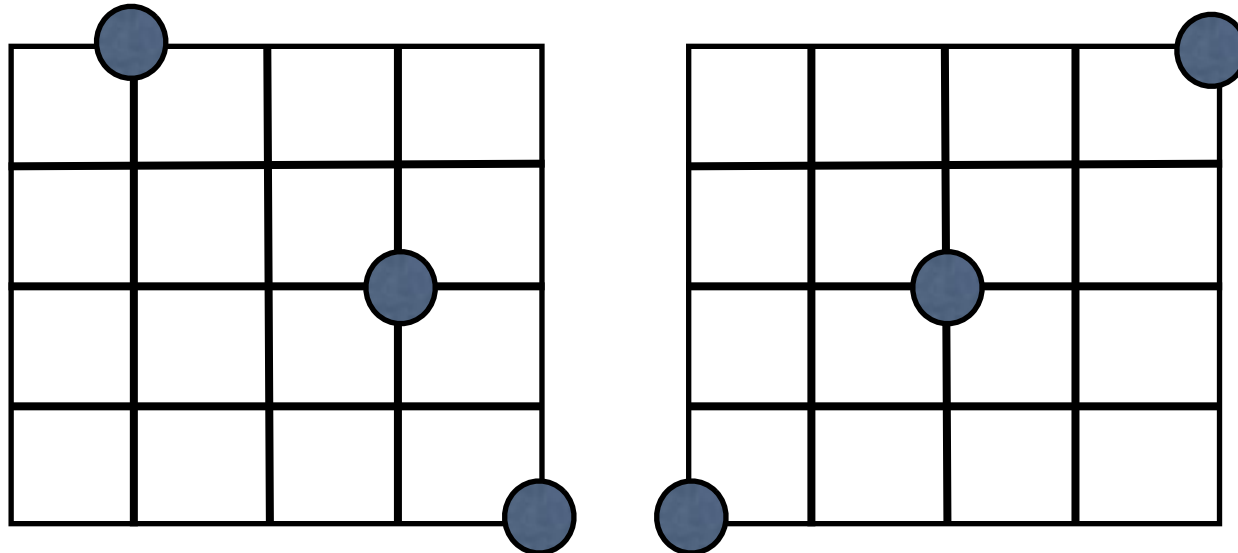


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid

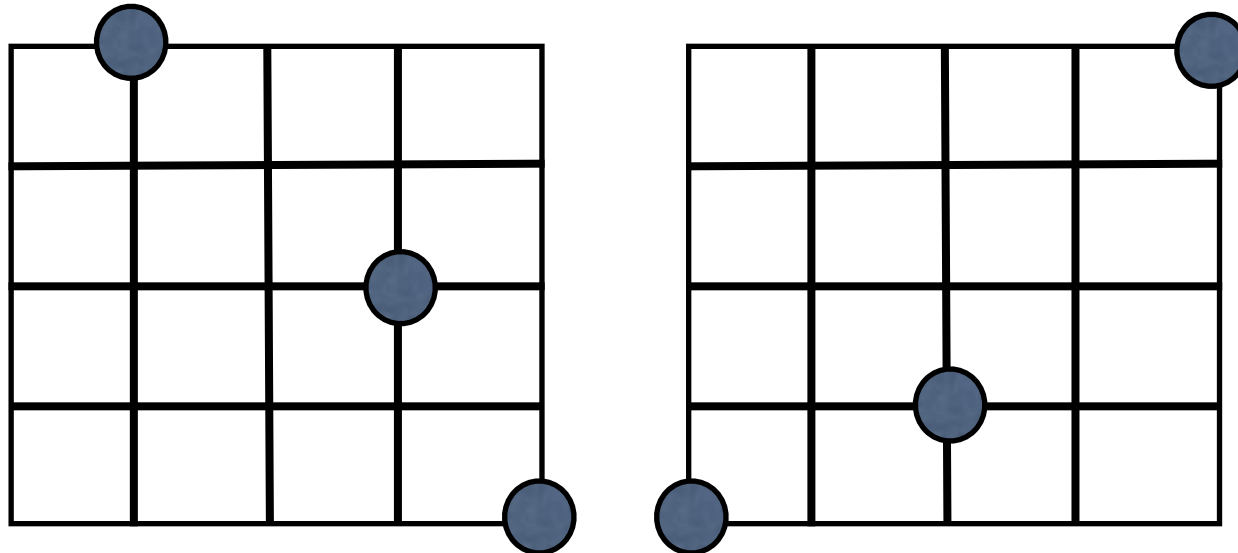


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid



Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Choice

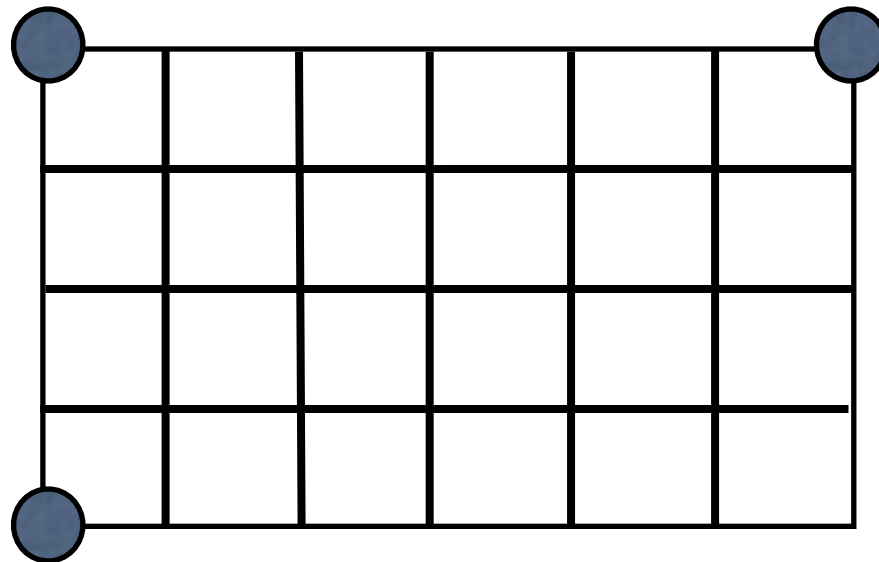
There are **at least two** robots at a corner of the grid

Algorithm

□ Phase I: Set-Up

✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid

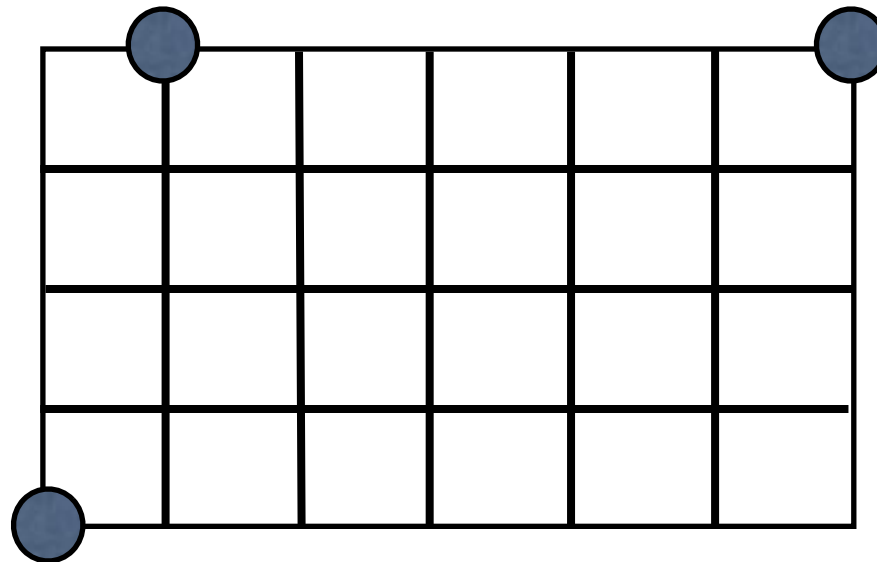


Algorithm

□ Phase I: Set-Up

✓ Configuration of type Choice

There are **at least two** robots at a corner of the grid



Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Undefined

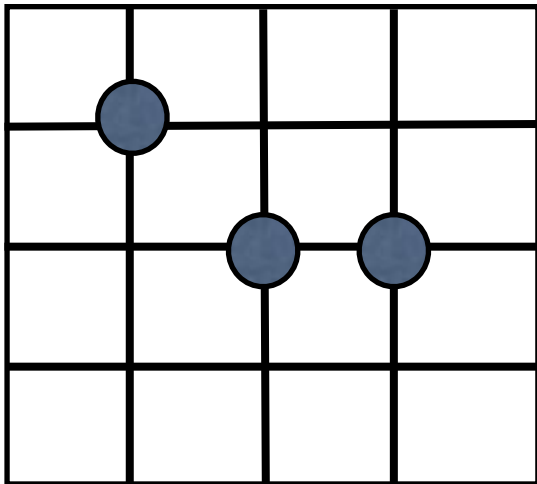
All the corners of the grid are free

Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Undefined

All the corners of the grid are free

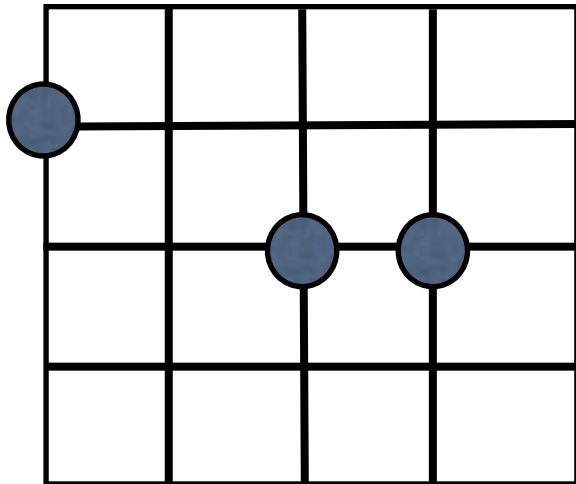


Algorithm

□ Phase I: Set-Up

✓ Configuration of type Undefined

All the corners of the grid are free

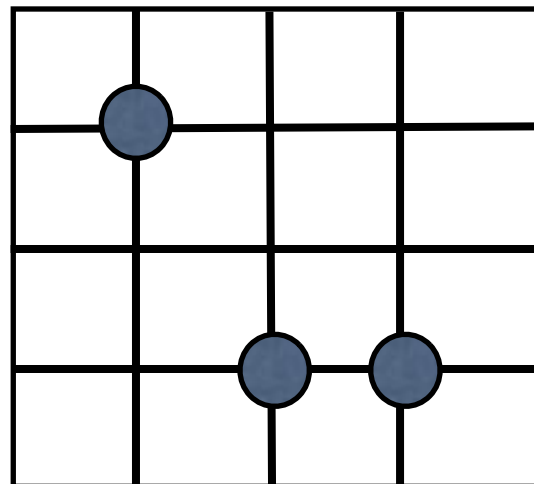
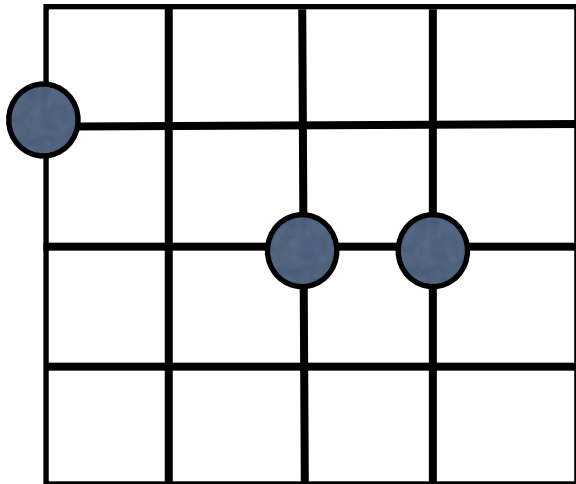


Algorithm

□ Phase I: Set-Up

✓ Configuration of type Undefined

All the corners of the grid are free

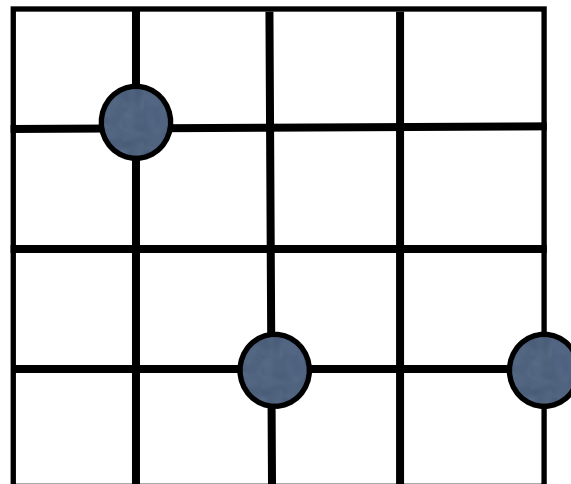
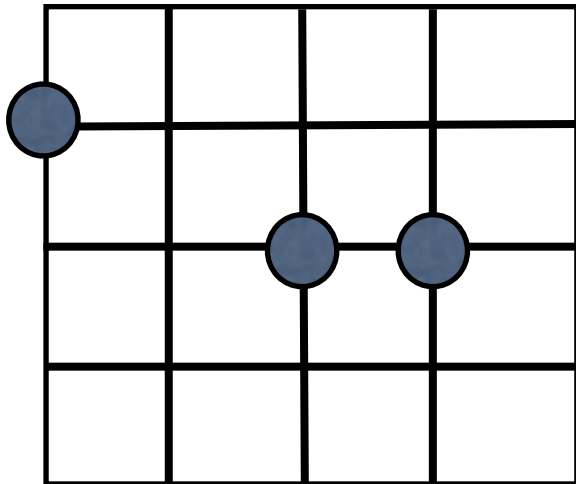


Algorithm

□ Phase I: Set-Up

- ✓ Configuration of type Undefined

All the corners of the grid are free

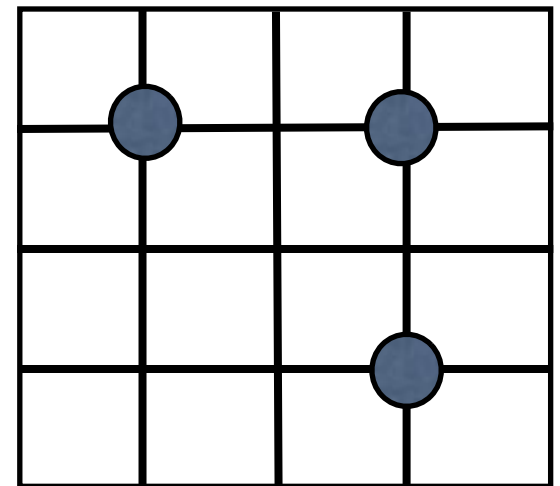
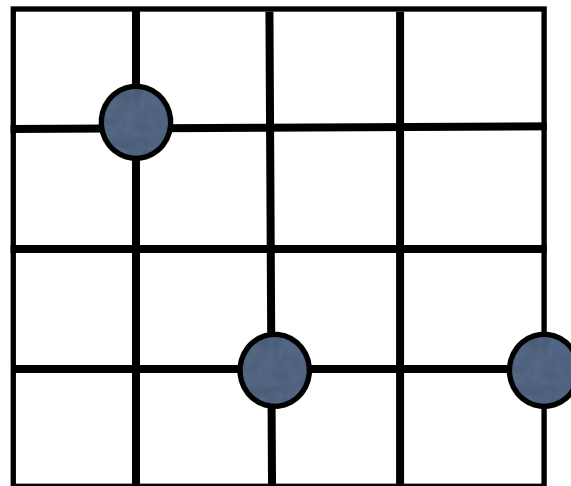
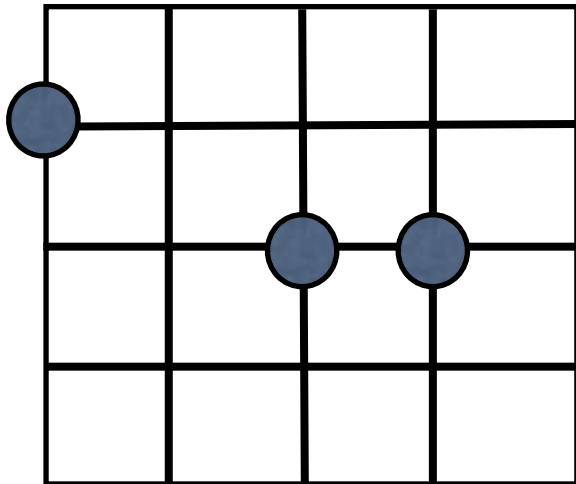


Algorithm

□ Phase I: Set-Up

✓ Configuration of type Undefined

All the corners of the grid are free

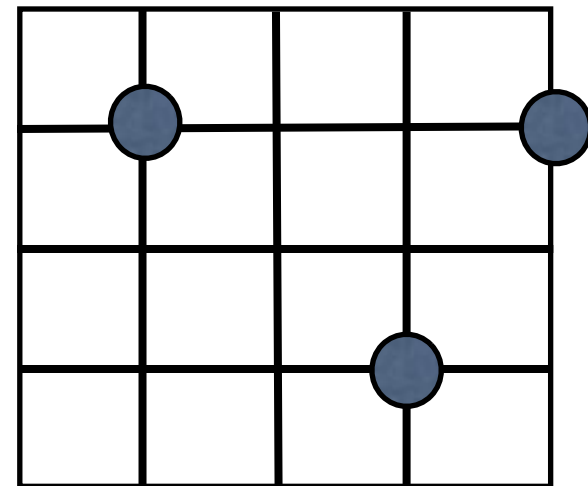
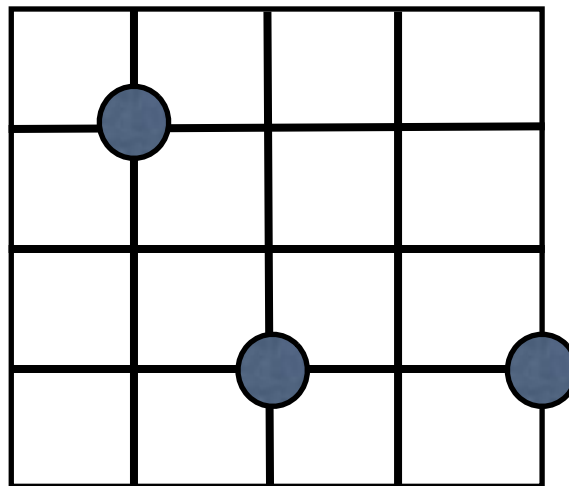
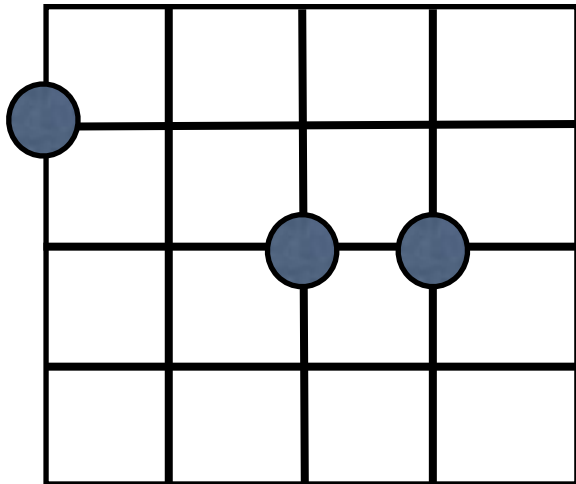


Algorithm

□ Phase I: Set-Up

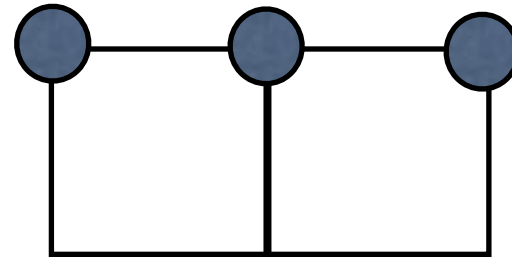
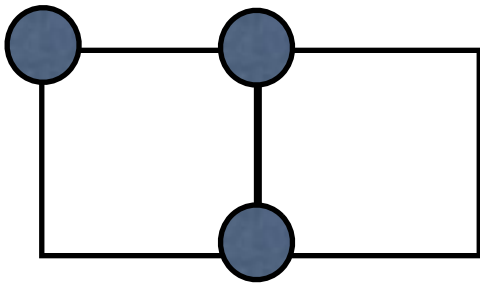
✓ Configuration of type Undefined

All the corners of the grid are free



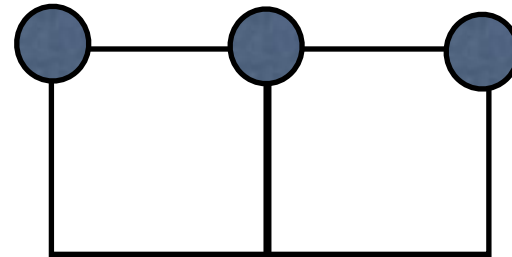
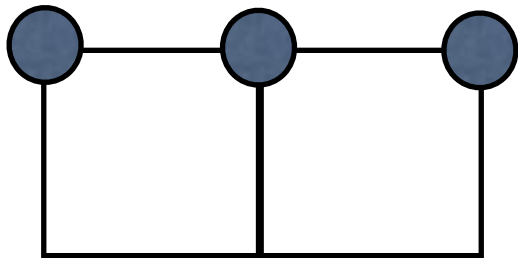
Algorithm

□ Special grid: (3,2) grids



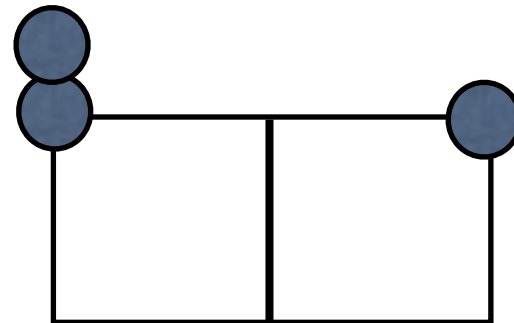
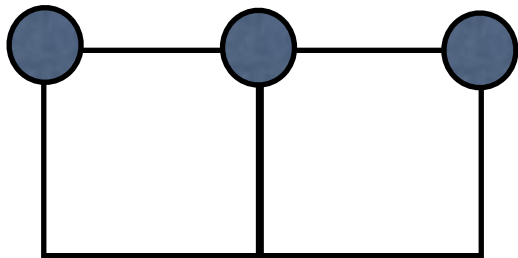
Algorithm

□ Special grid: (3,2) grids



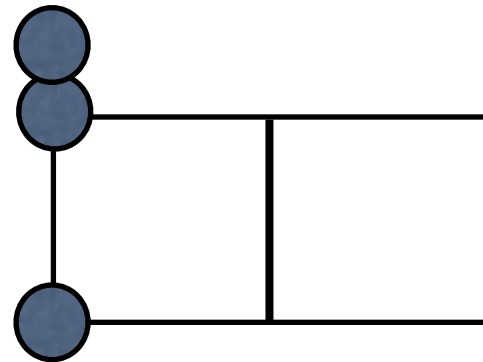
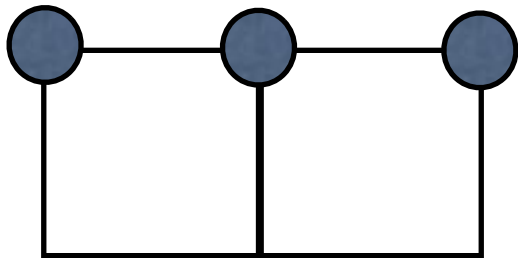
Algorithm

□ Special grid: (3,2) grids



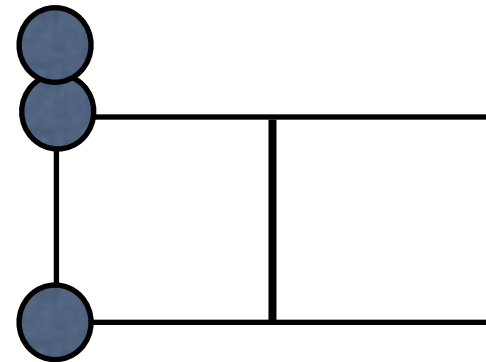
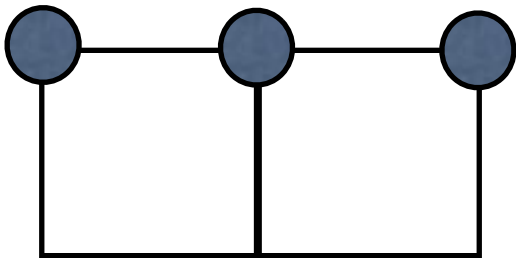
Algorithm

□ Special grid: (3,2) grids



Algorithm

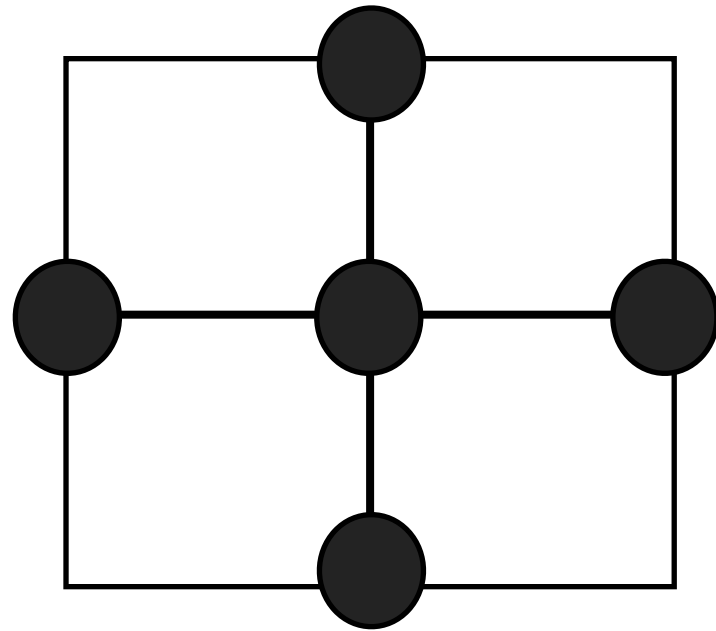
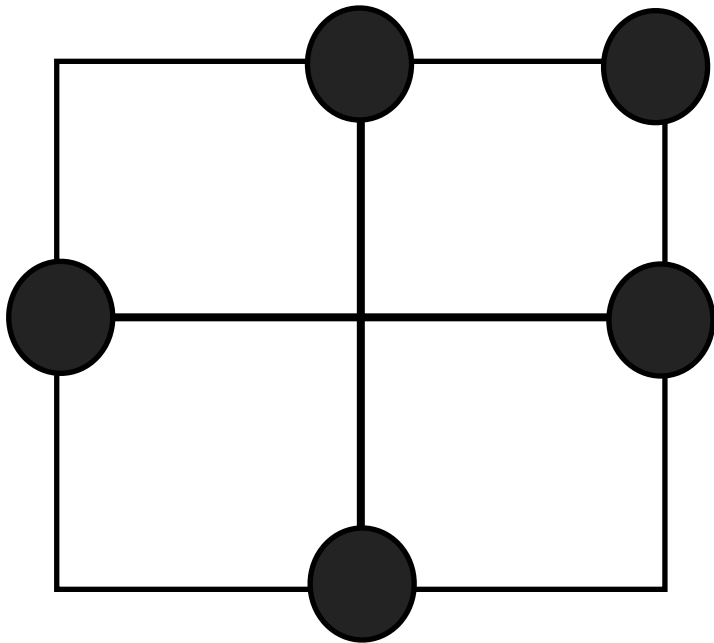
□ Special grid: (3,2) grids



Final Configuration

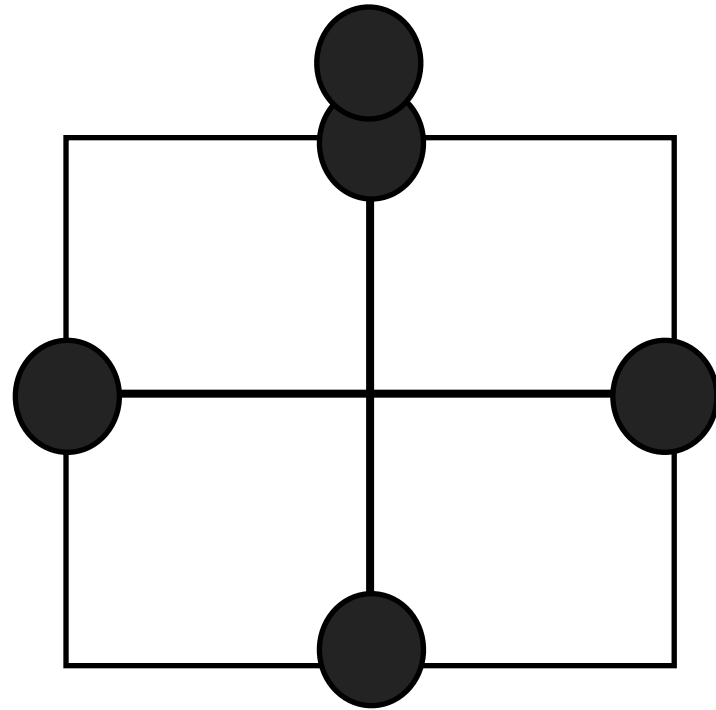
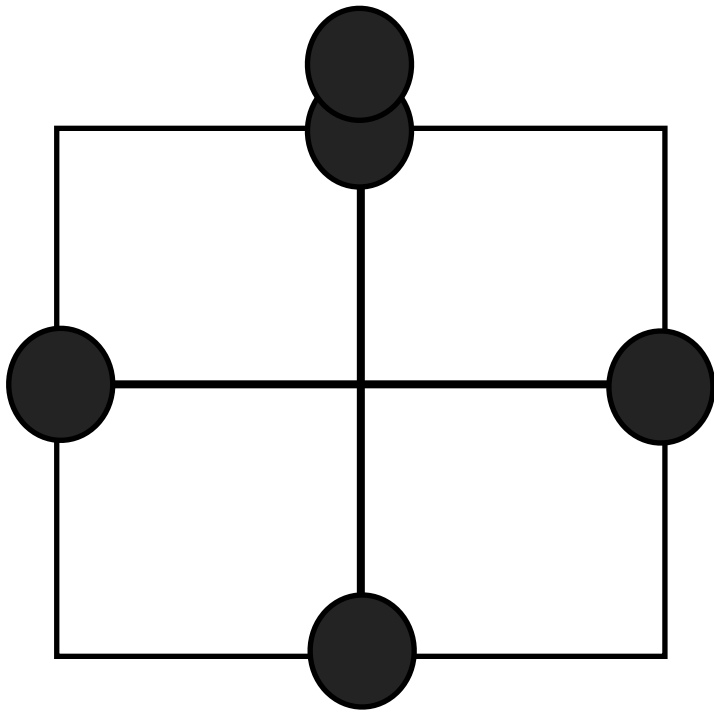
Special case

□ (3,3)-grids with 5 robots



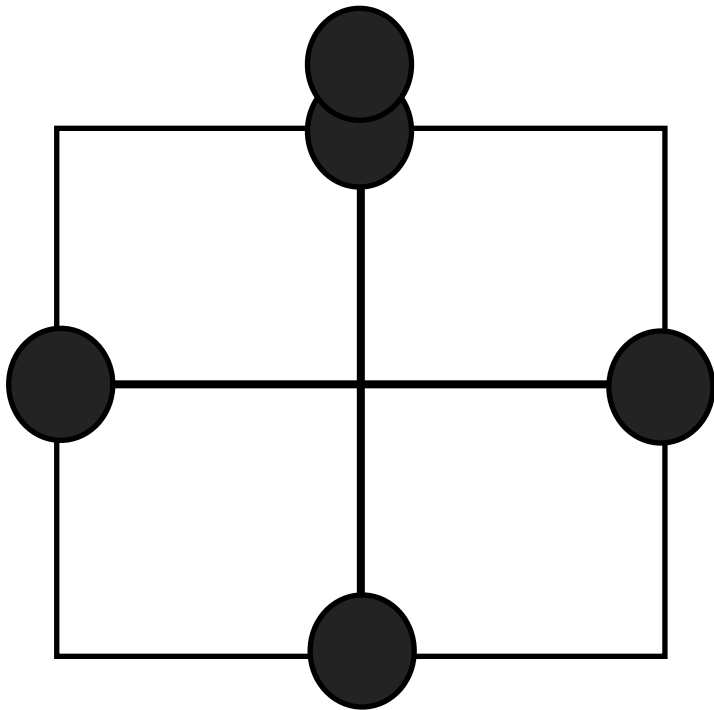
Special case

□ (3,3)-grids with 5 robots



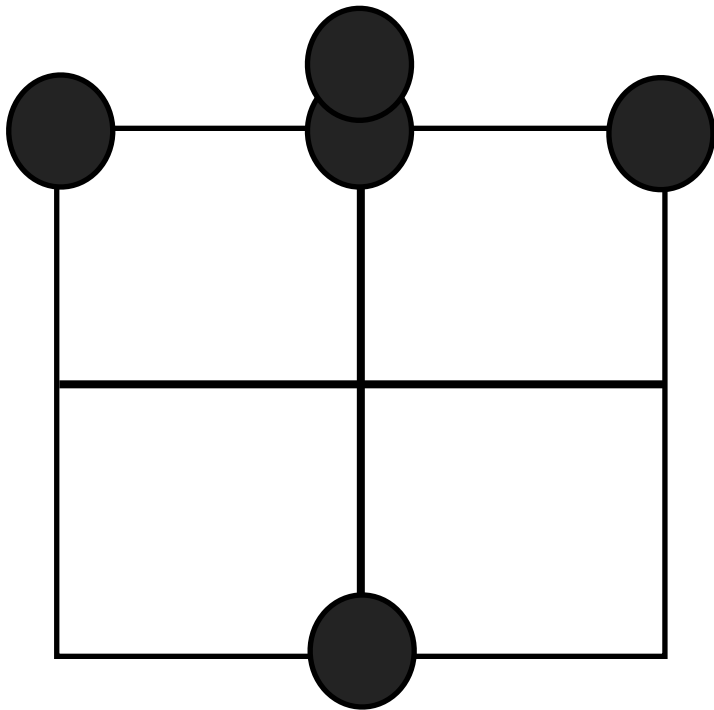
Special case

□ (3,3)-grids with 5 robots



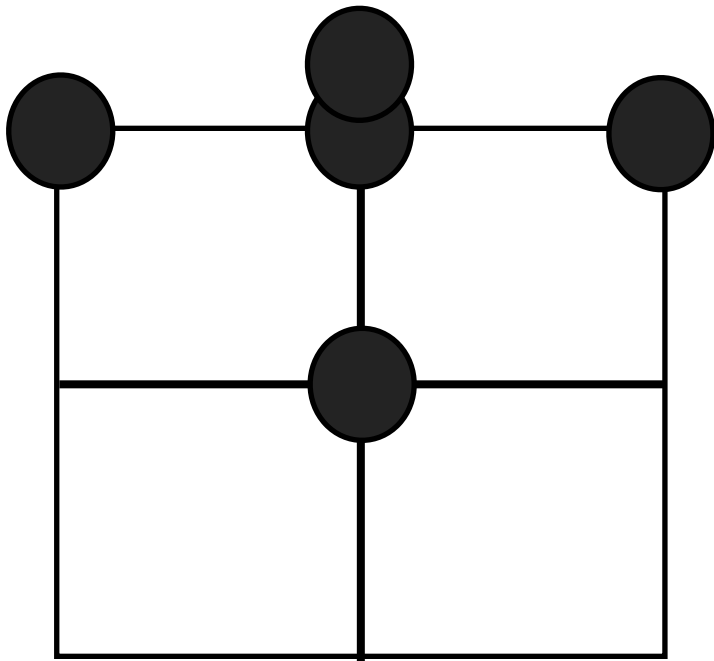
Special case

□ (3,3)-grids with 5 robots



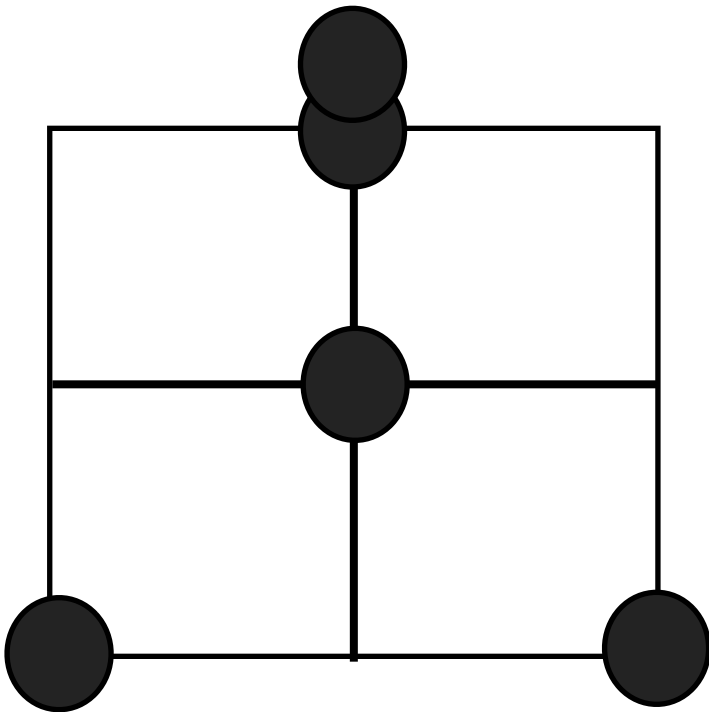
Special case

□ (3,3)-grids with 5 robots



Special case

□ (3,3)-grids with 5 robots



Final Configuration

Conclusion

- ✓ 5 robots are necessary and sufficient to explore $(3,3)$ -grids
- ✓ 4 robots are necessary and sufficient to explore $(2,2)$ -grids
- ✓ 3 robots are necessary and sufficient to explore (i,j) -grids such that $j > 3$