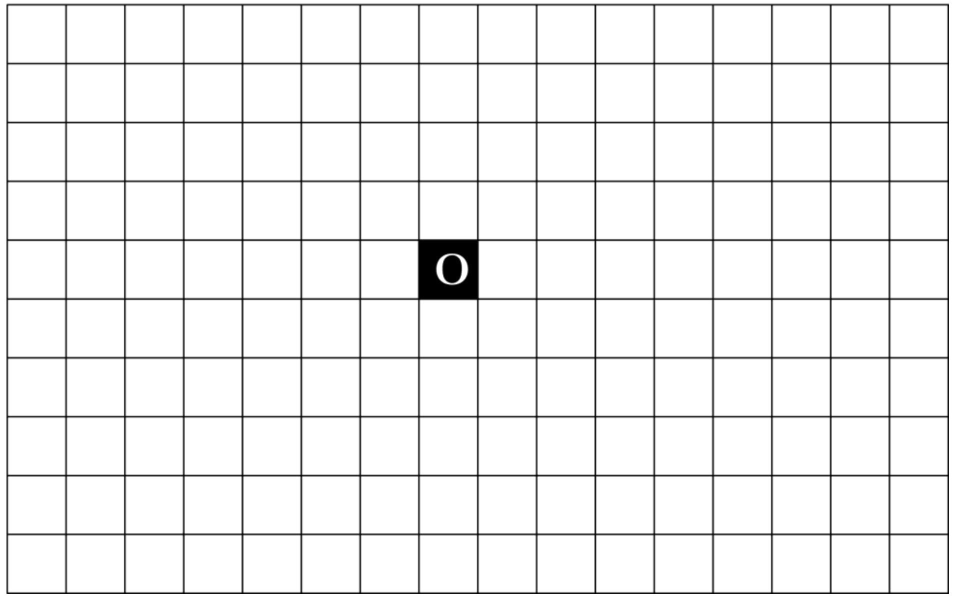
How Many Ants Does It Take to Find the Food?

Jara Uitto

Ants Nearby Treasure Search

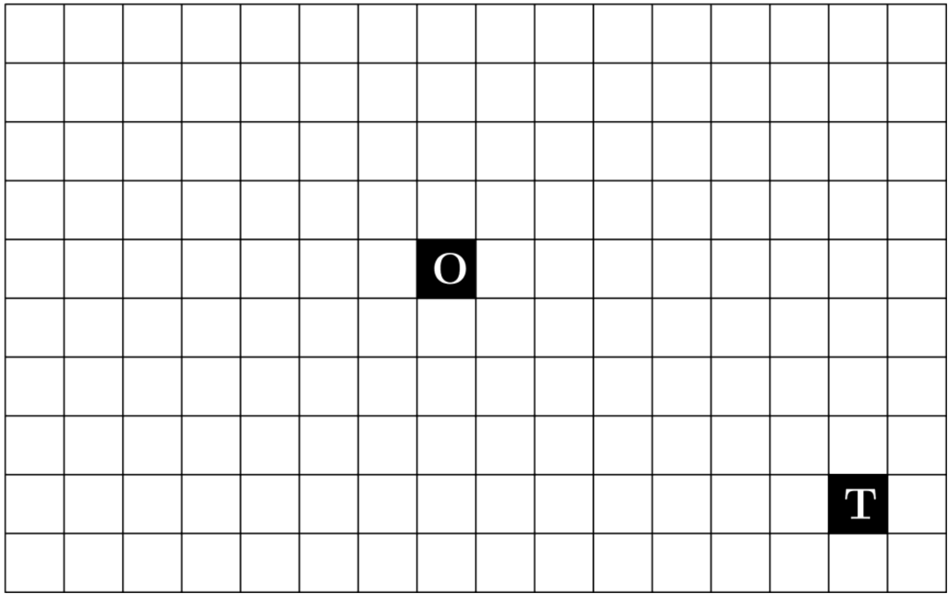
- Introduced by Feinerman, Korman, Lotker and Sereni [PODC 2012].
- n mobile agents, controlled by Turing machines, search for a treasure.
- Communication not allowed.

Model



- Infinite integer grid.
- Each ant initially located in the origin.

Model



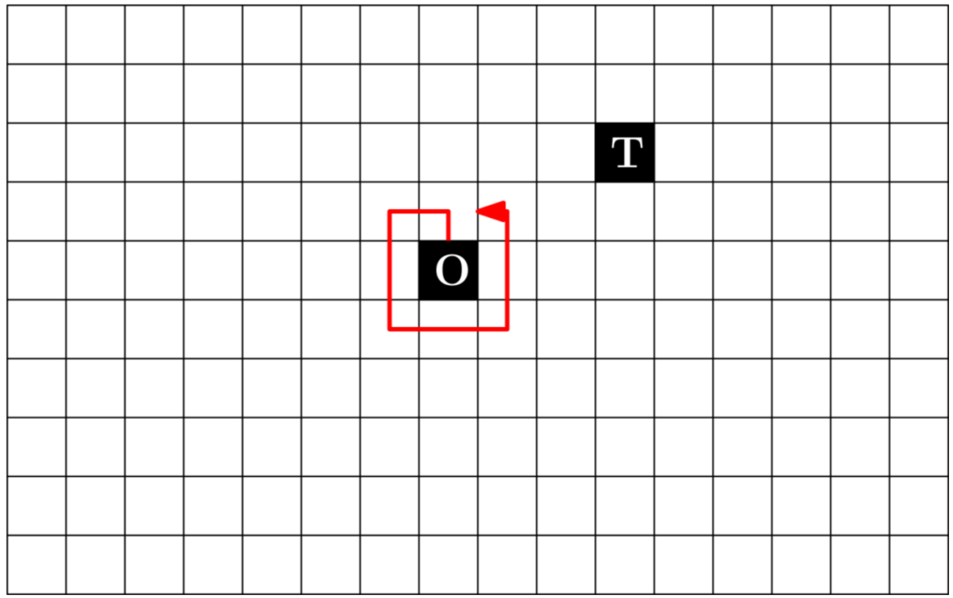
- Adversarially hidden treasure/food.
- (Manhattan) distance to treasure is D.

Ants Nearby Treasure Search

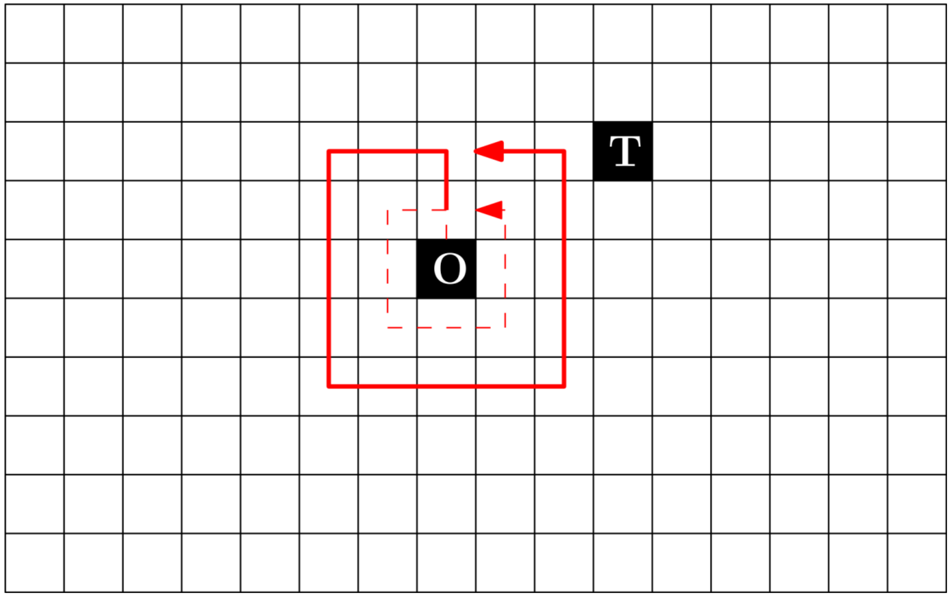
- How many rounds until the treasure is found?

- We study the number of ants needed to find the treasure at all.

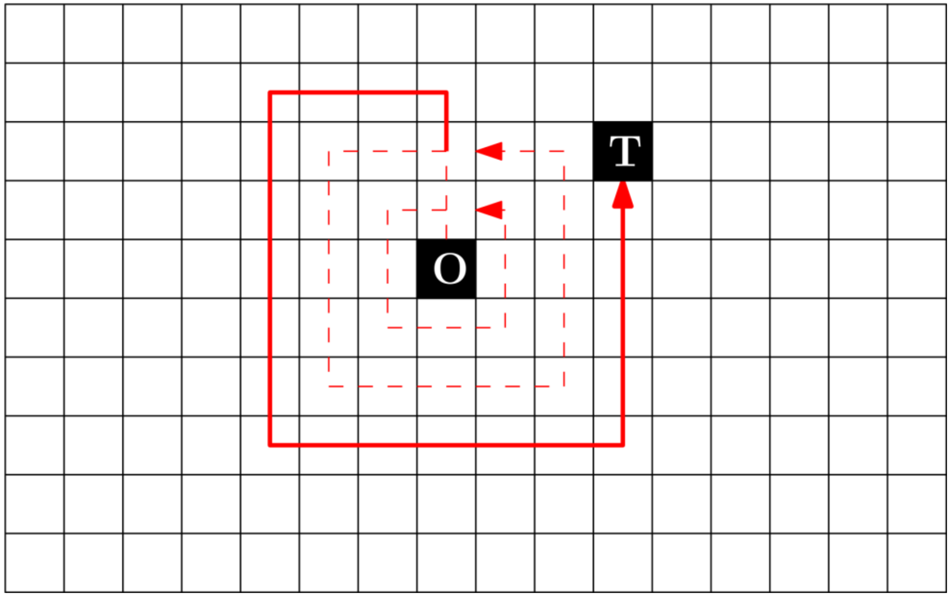
Model



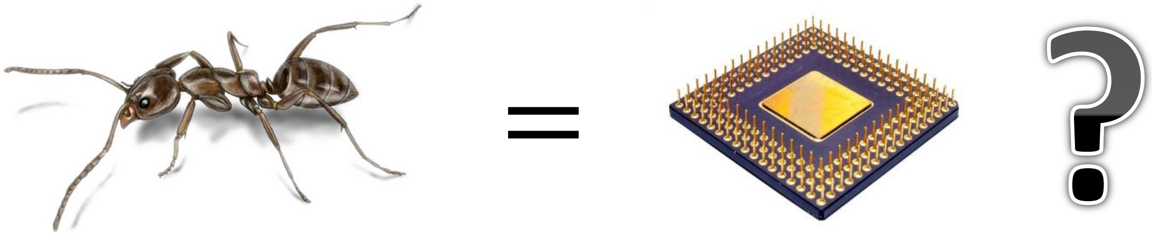
Model



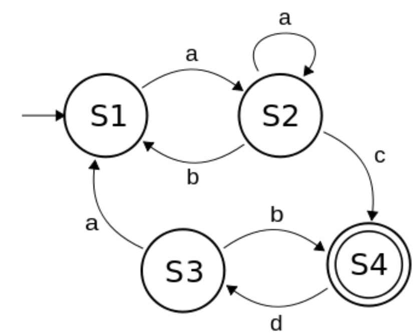
Model



- One Turing Machine is enough. No communication needed.



Model



- Ants are controlled by (randomized) finite state machines.
- Communicate by sensing the states of nearby ants.
- Run-time studied by Emek, Langner, Uitto and Wattenhofer [ICALP2014].

Model

- Synchrony vs. Asynchrony

- A deterministic protocol?

Model

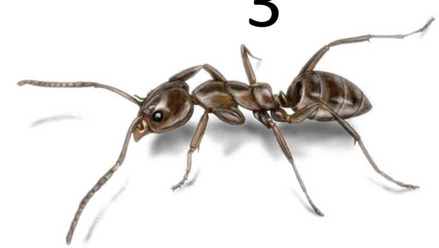
- Individual algorithm for each ant.

- An algorithm works correctly if the ants find the treasure in expected finite time.

1

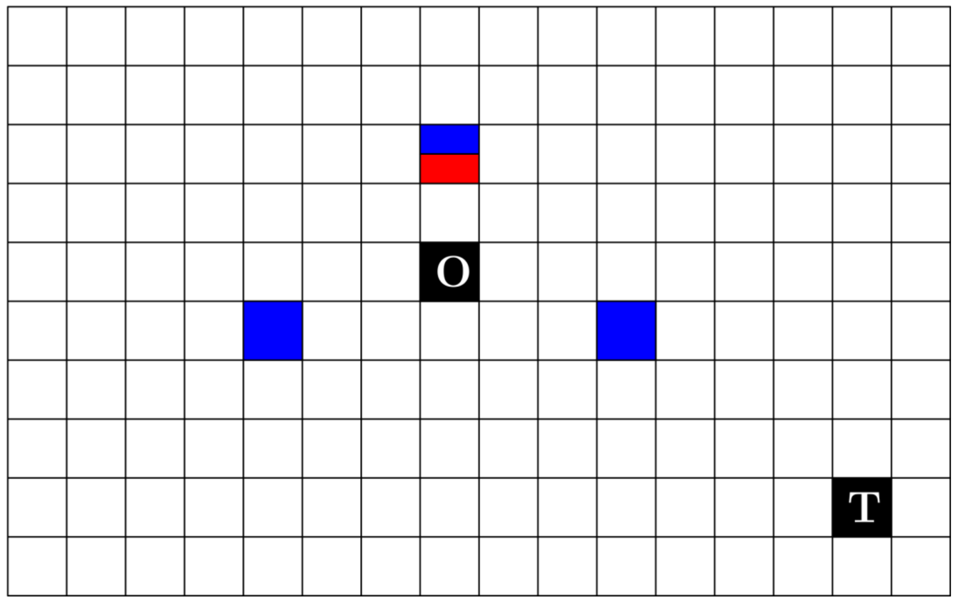
2

3

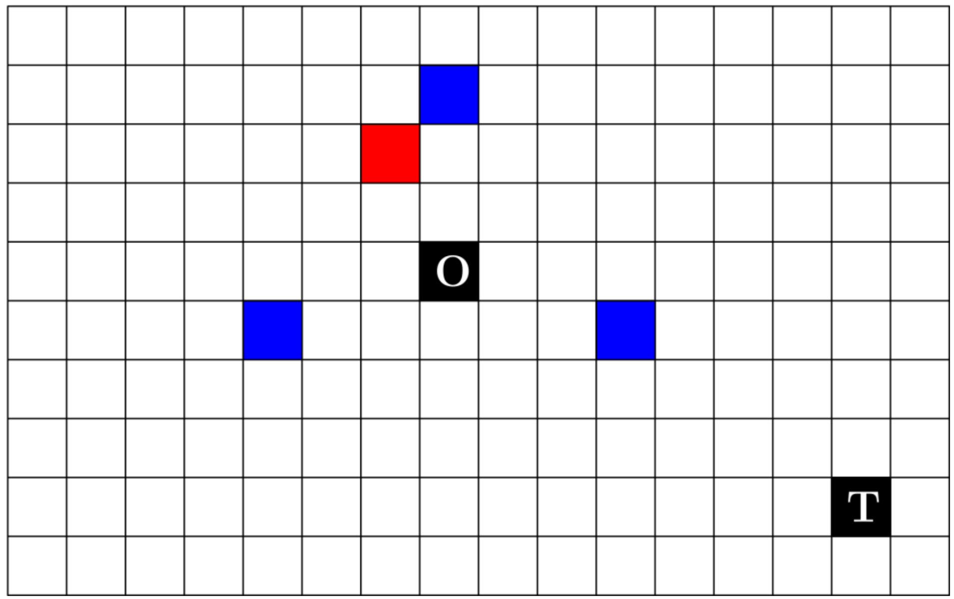


Deterministic + Asynchronous

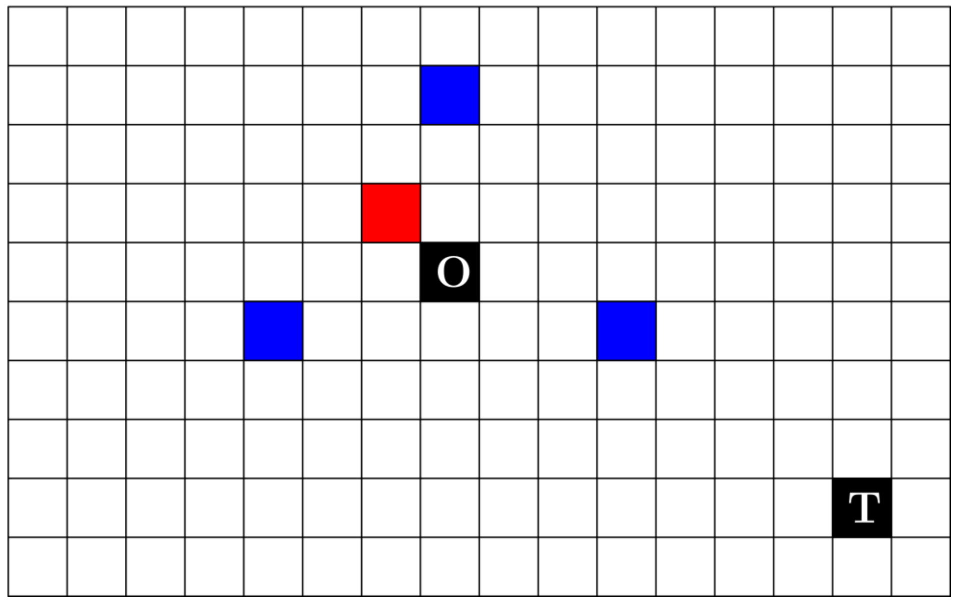
Triangle Search



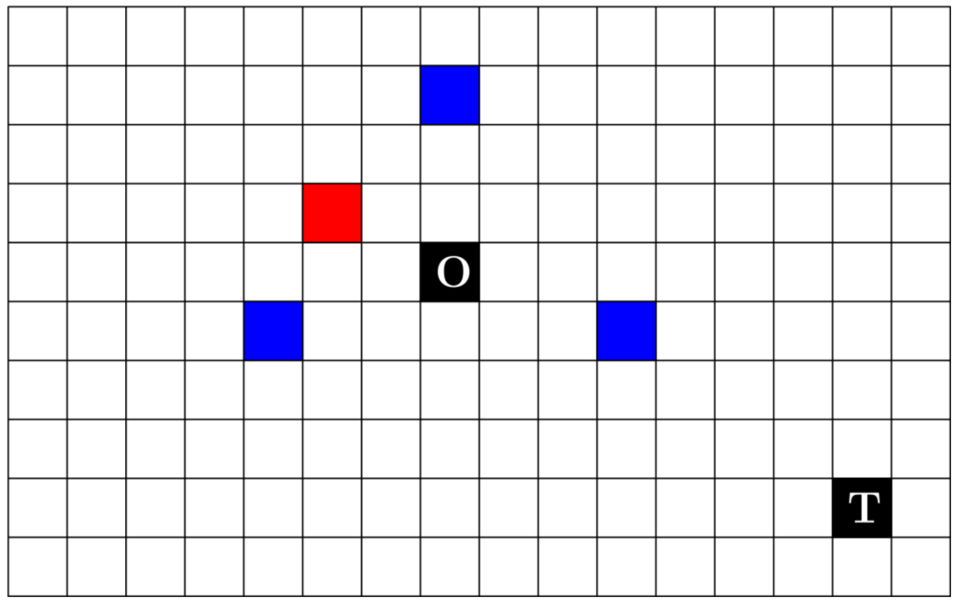
Triangle Search



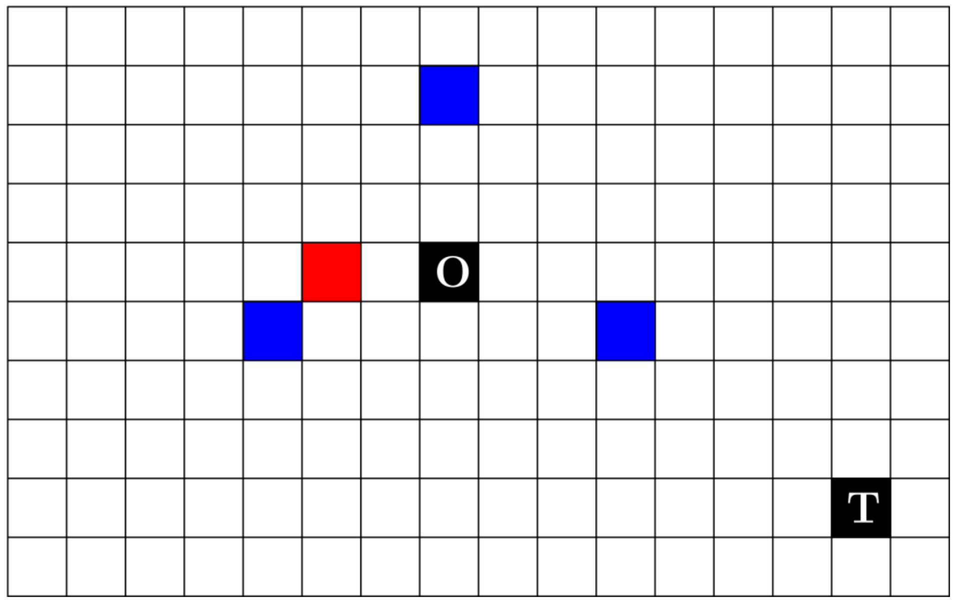
Triangle Search



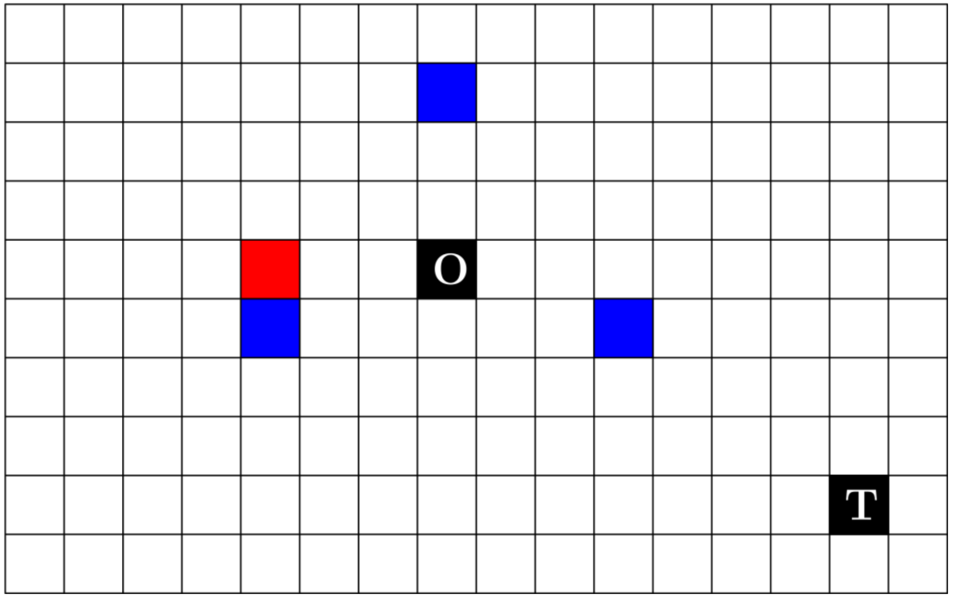
Triangle Search



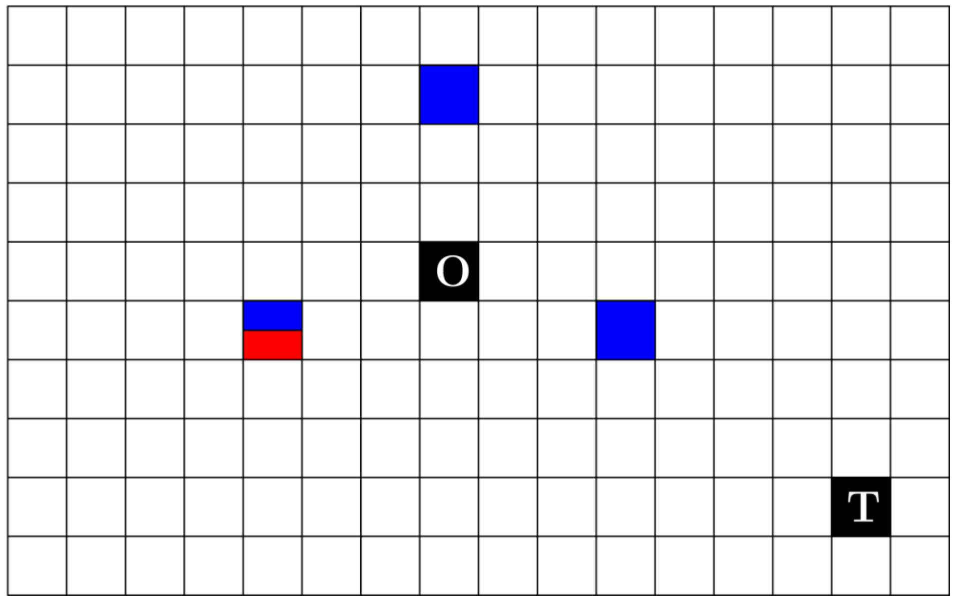
Triangle Search



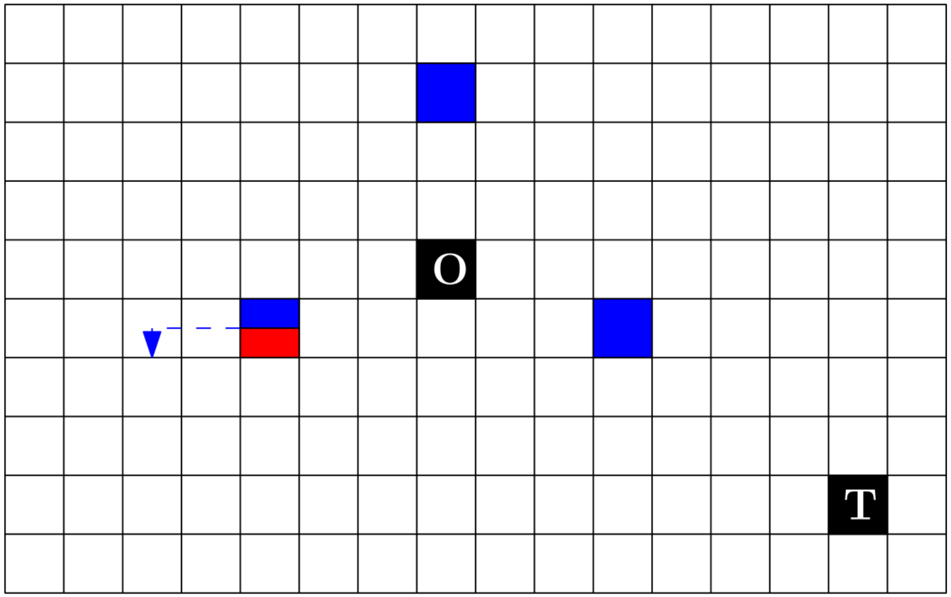
Triangle Search



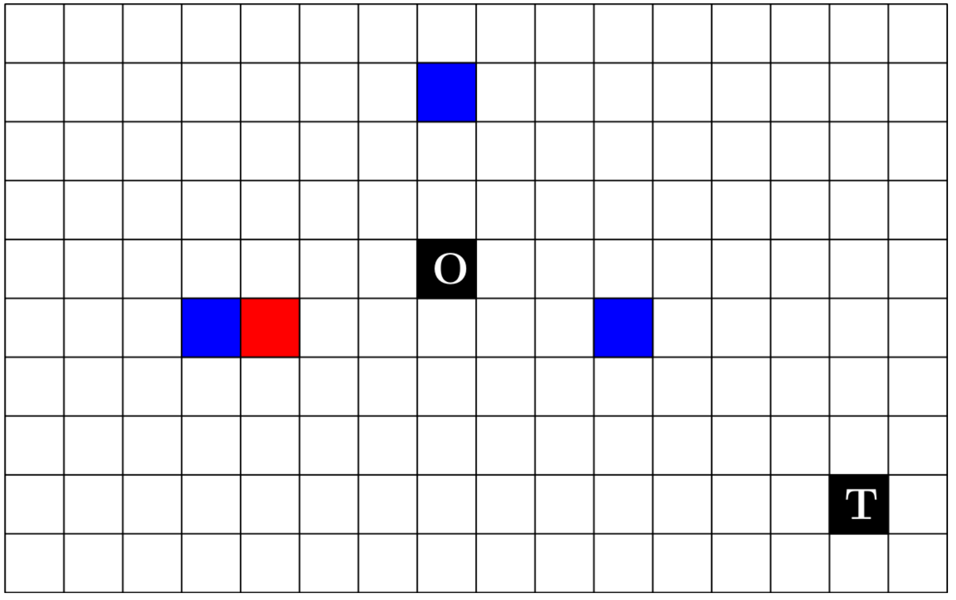
Triangle Search



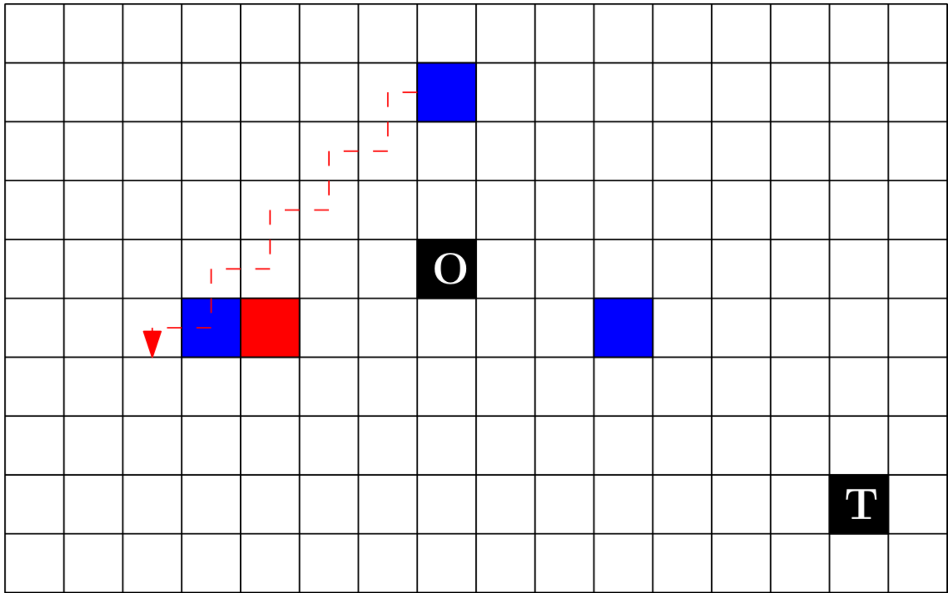
Triangle Search



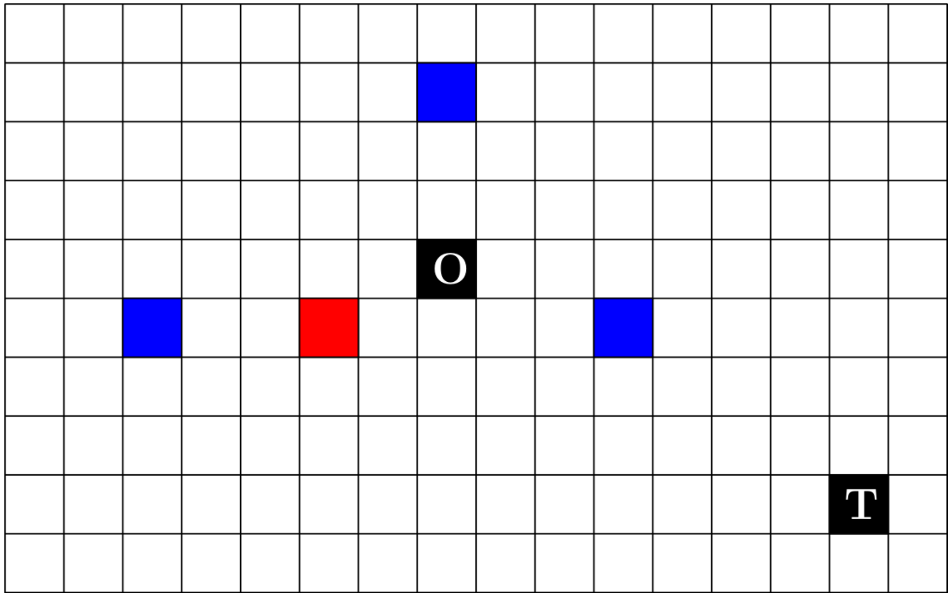
Triangle Search



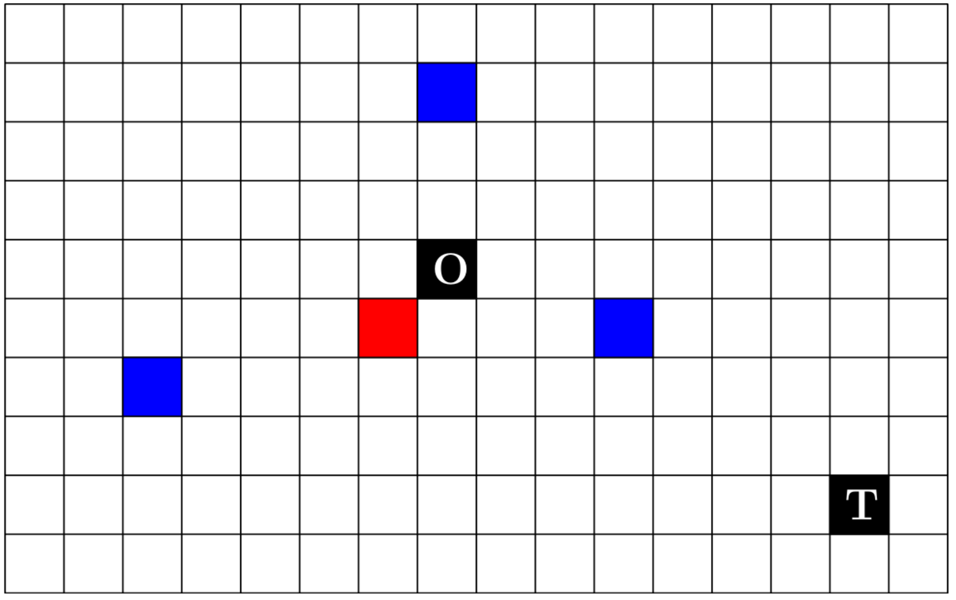
Triangle Search



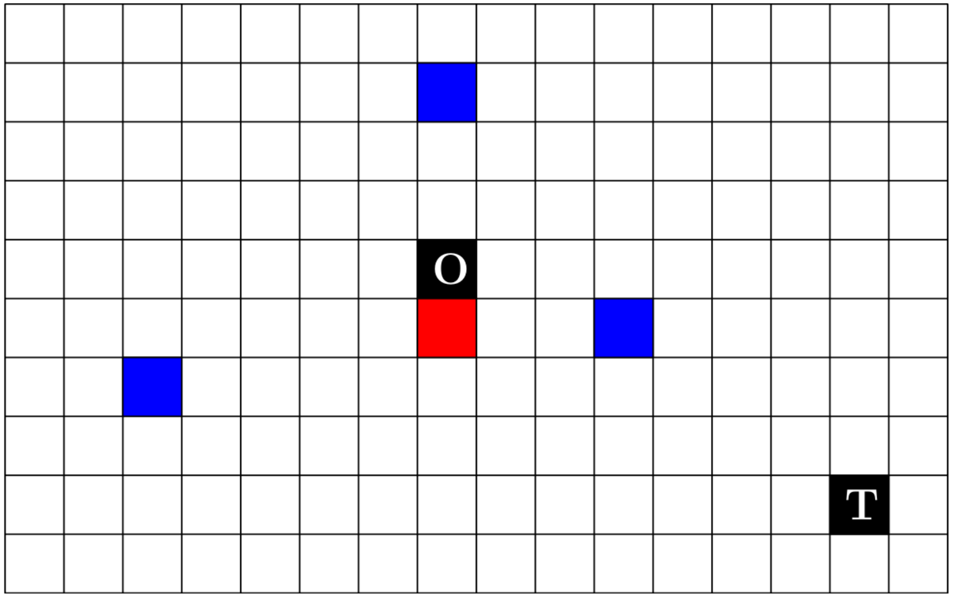
Triangle Search



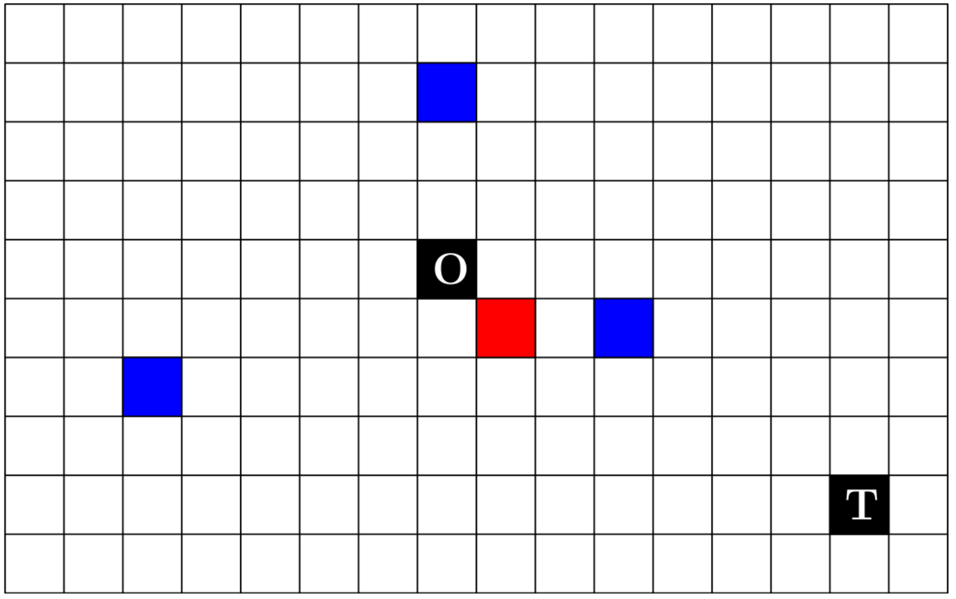
Triangle Search



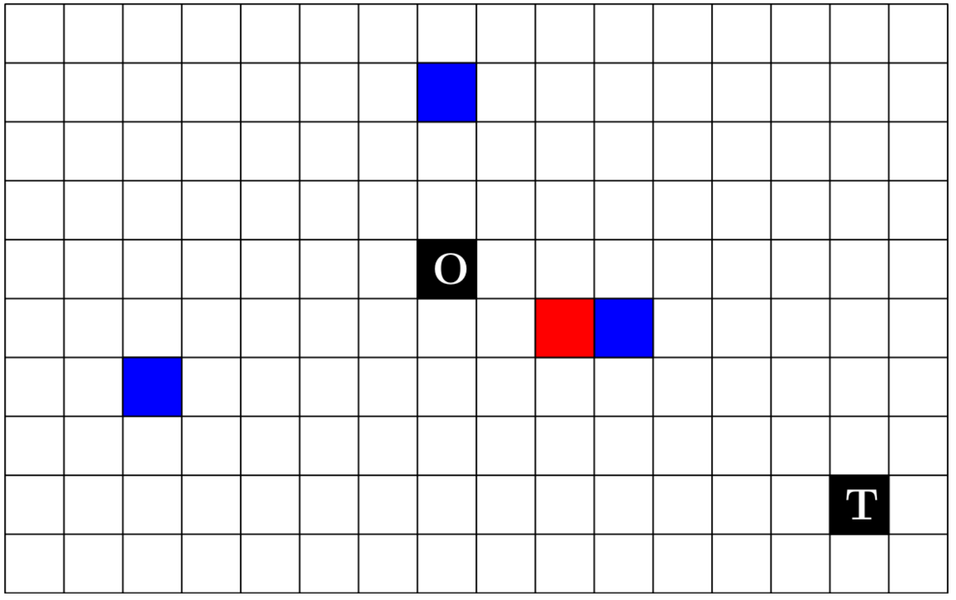
Triangle Search



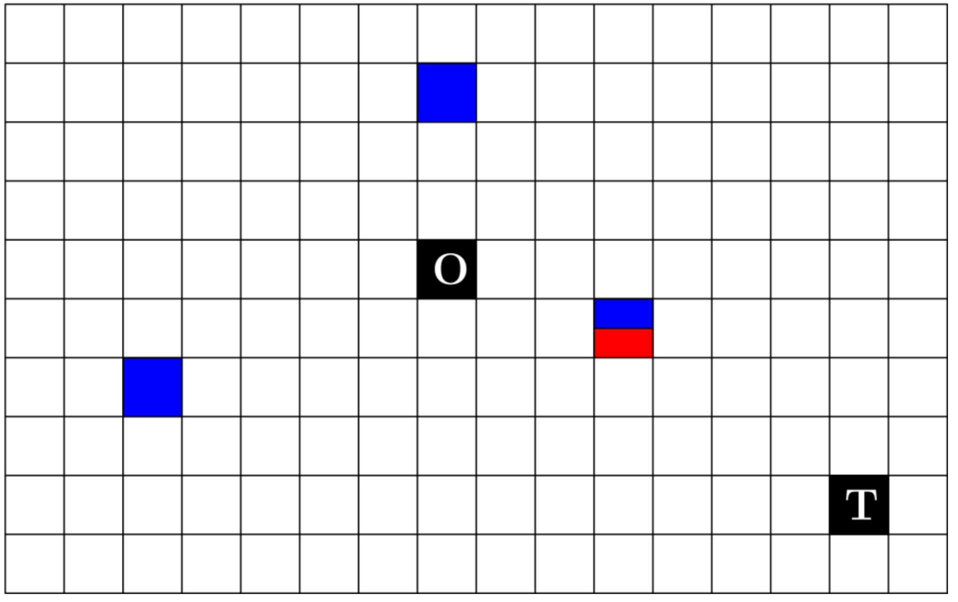
Triangle Search



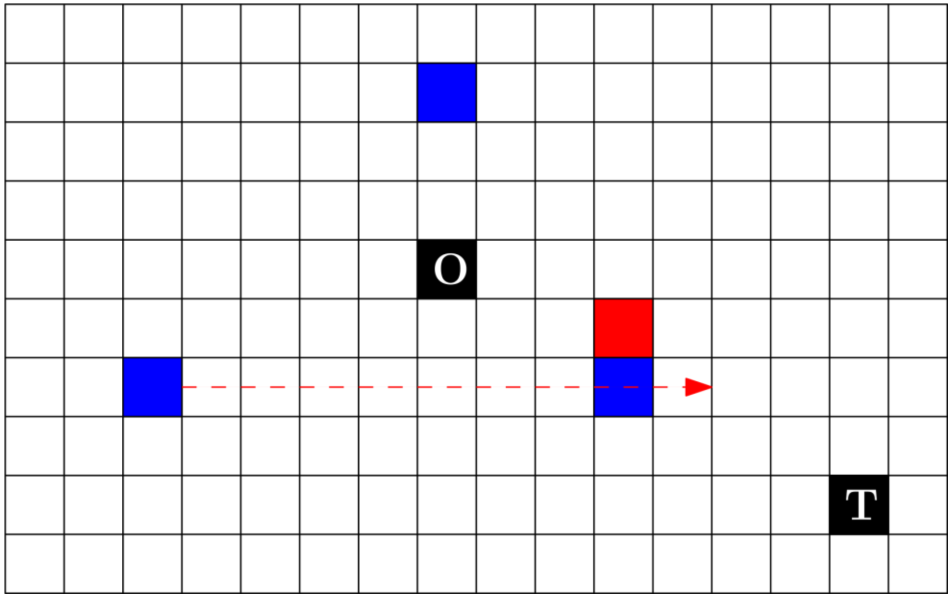
Triangle Search



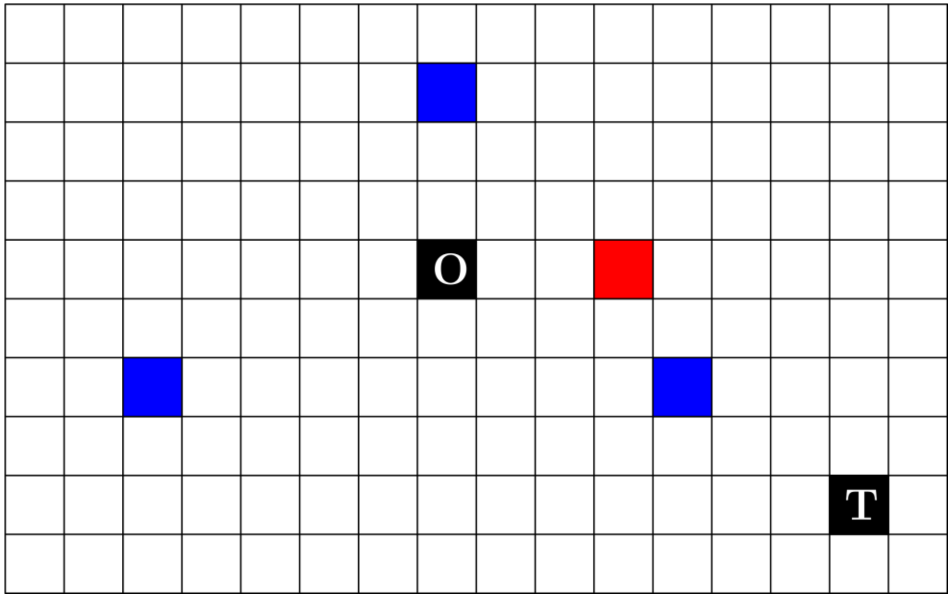
Triangle Search



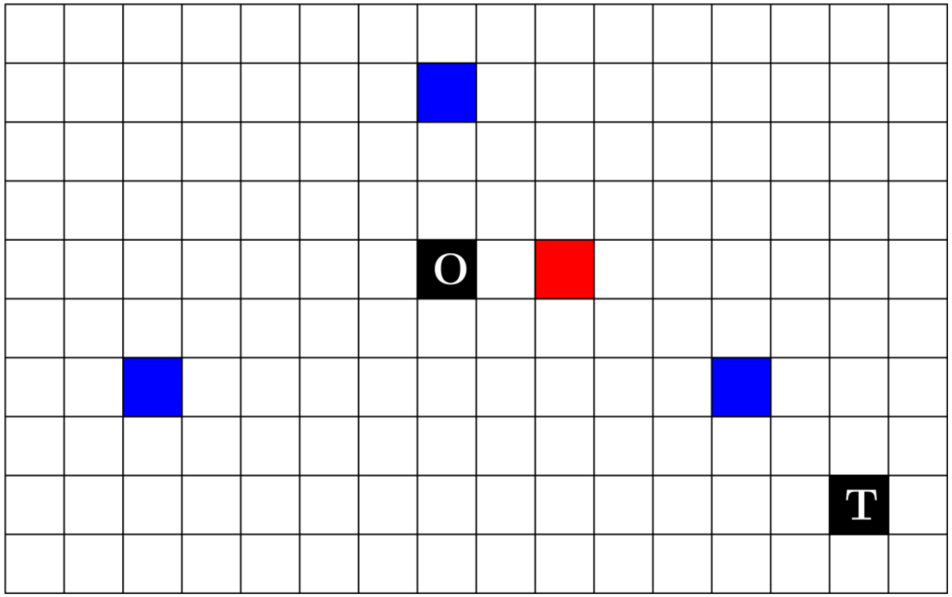
Triangle Search



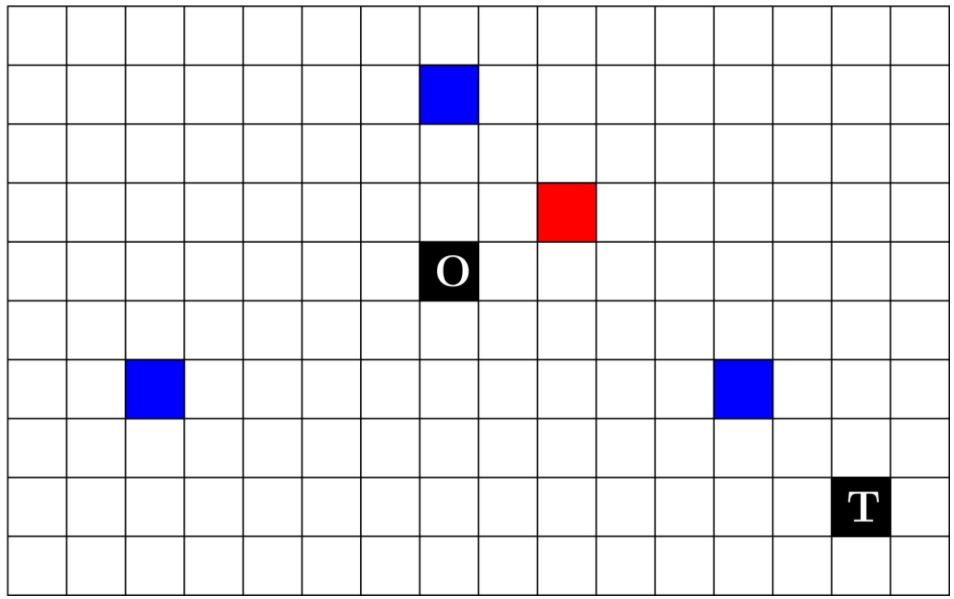
Triangle Search



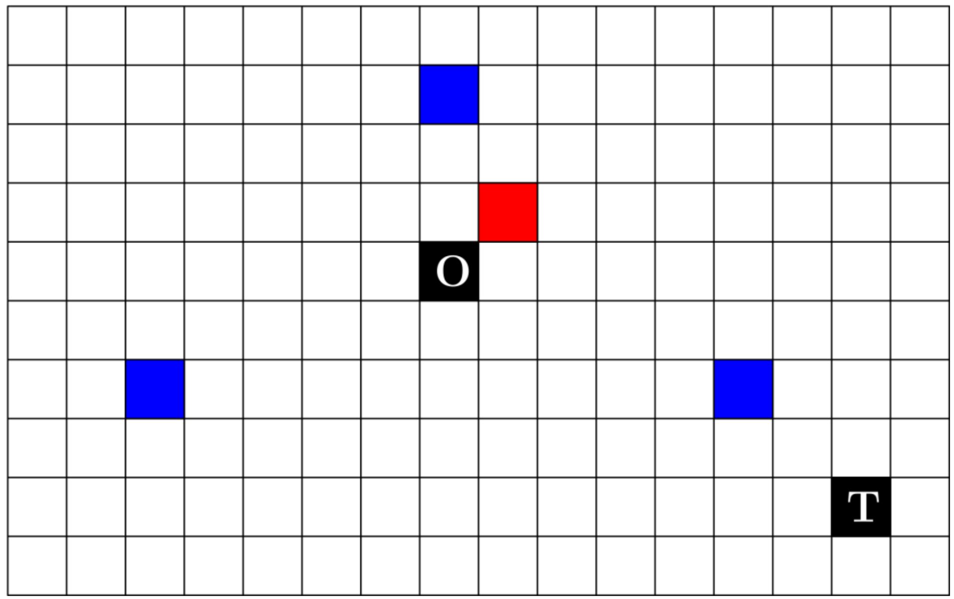
Triangle Search



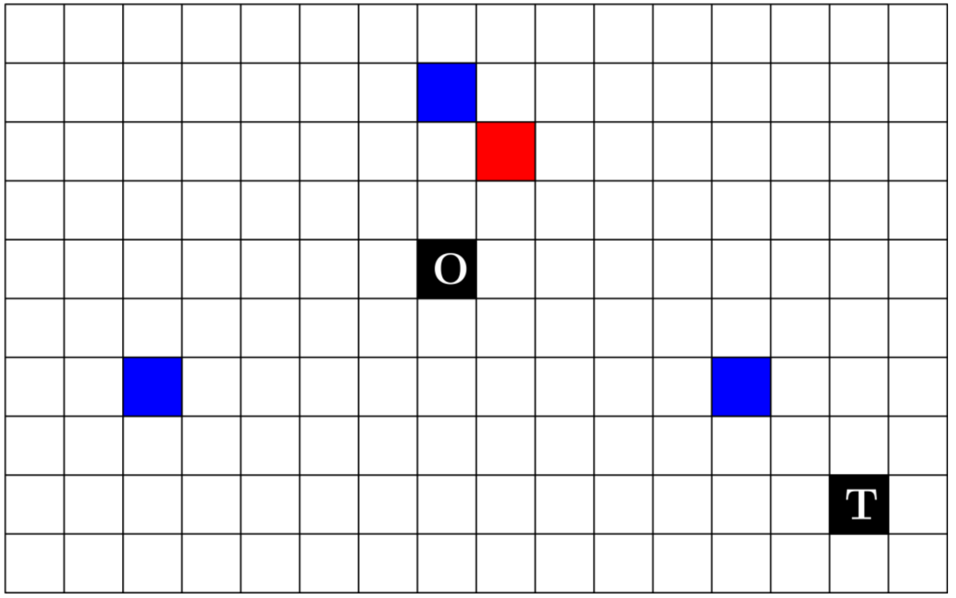
Triangle Search



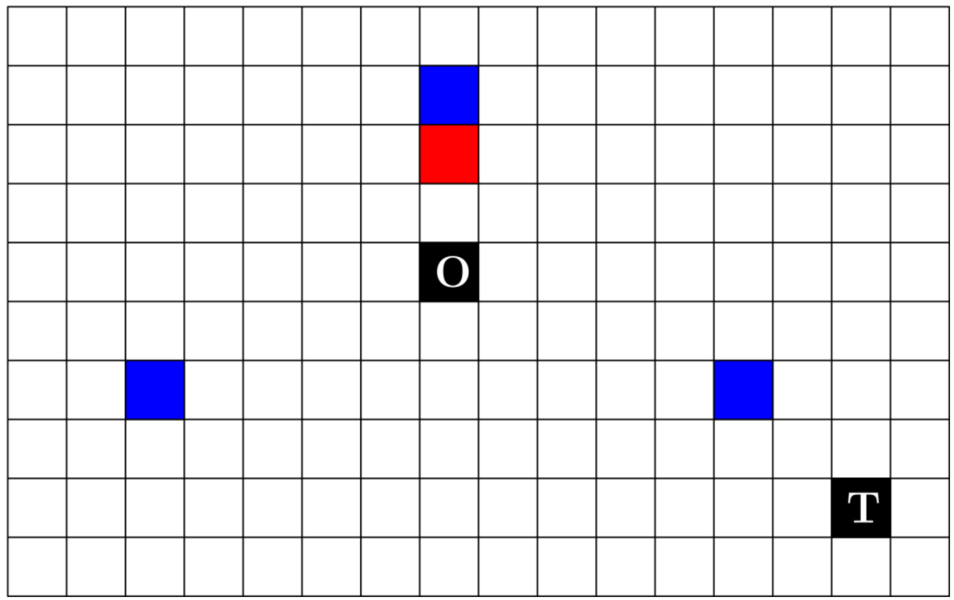
Triangle Search



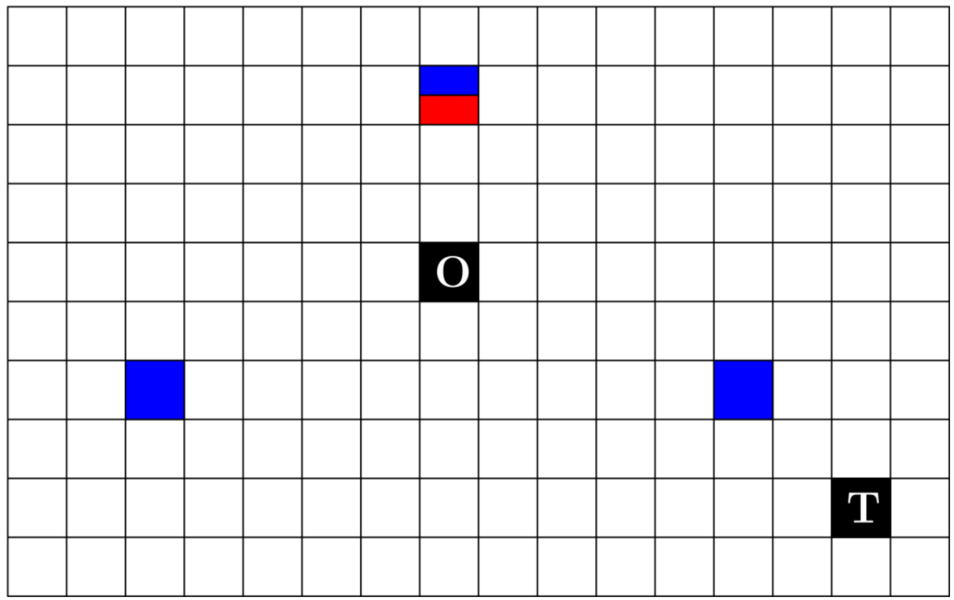
Triangle Search



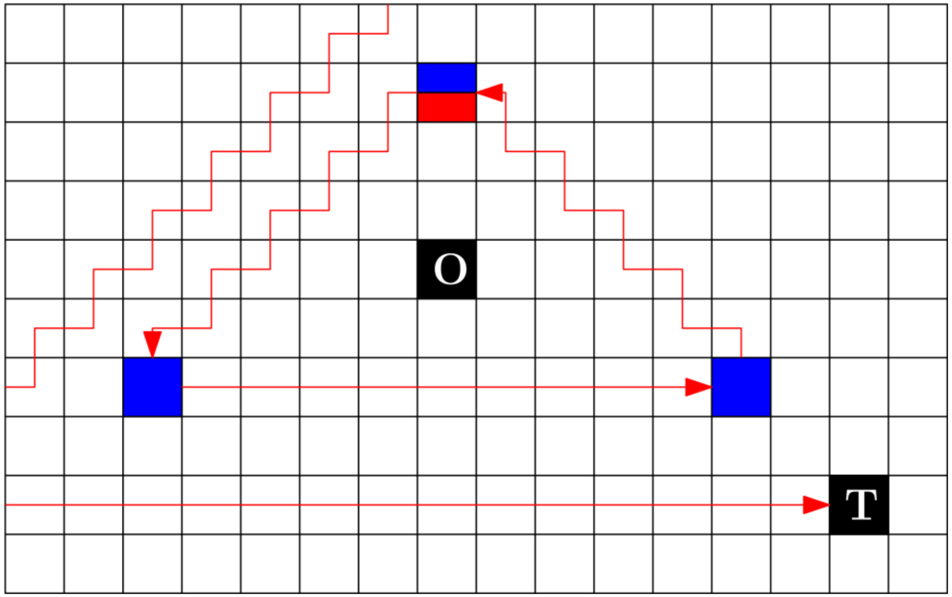
Triangle Search



Triangle Search



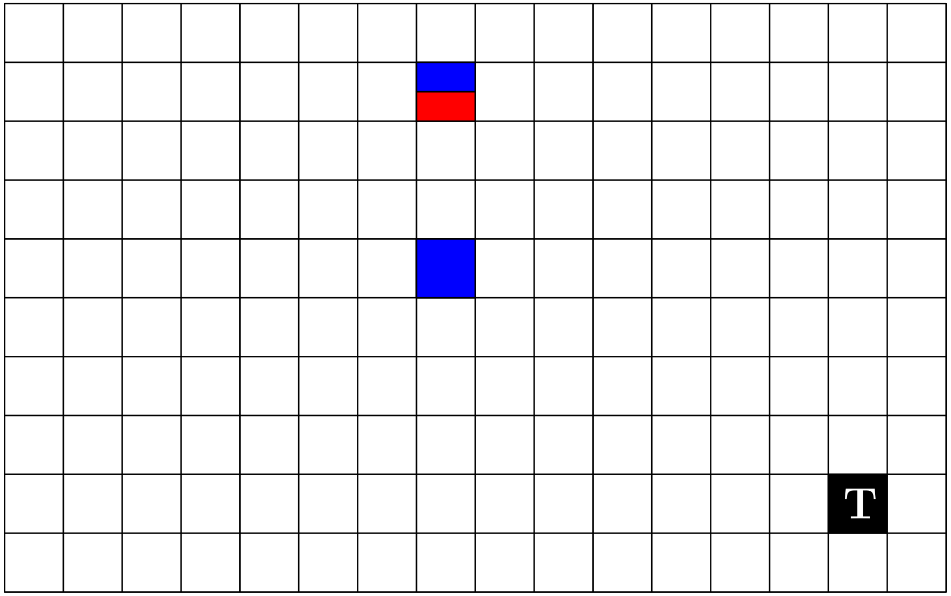
Triangle Search



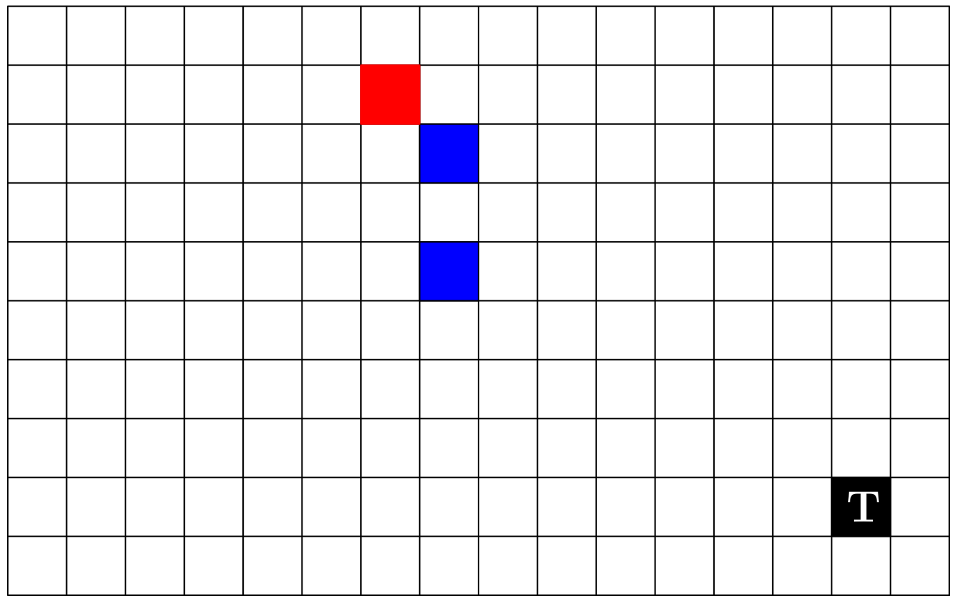
Synchronization?

- Can we perform better if the ants have a common sense of time?

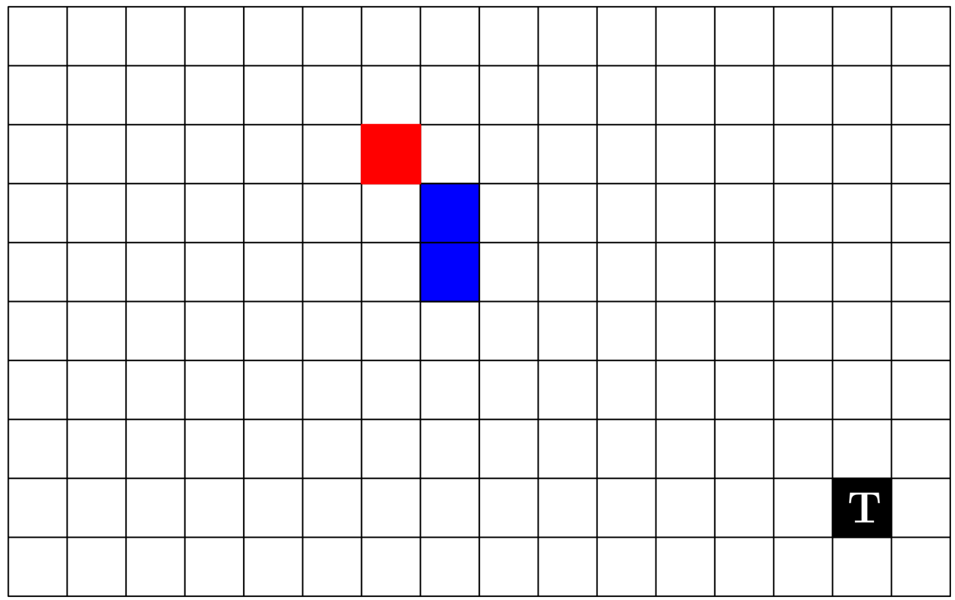
Rectangle Search



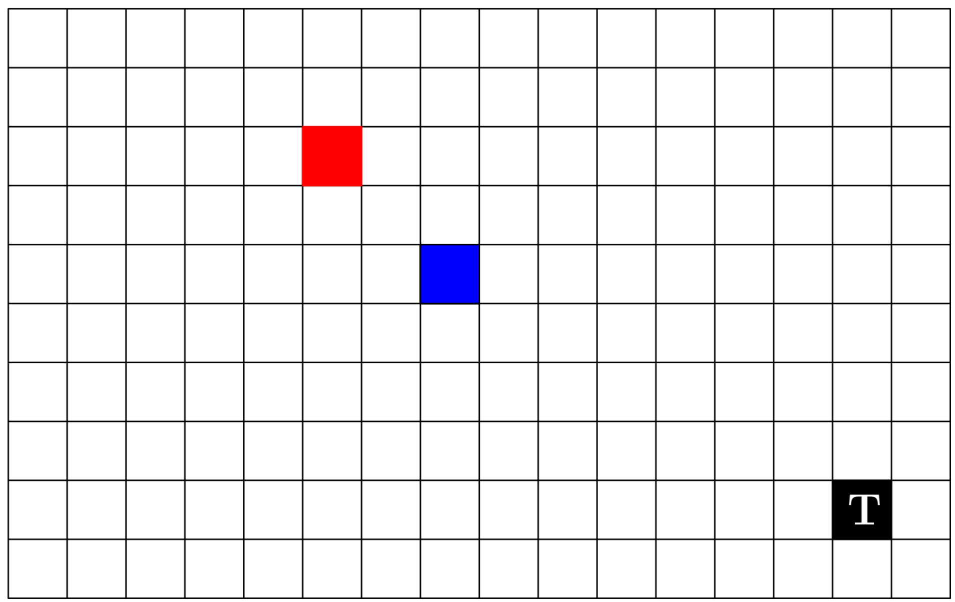
Rectangle Search



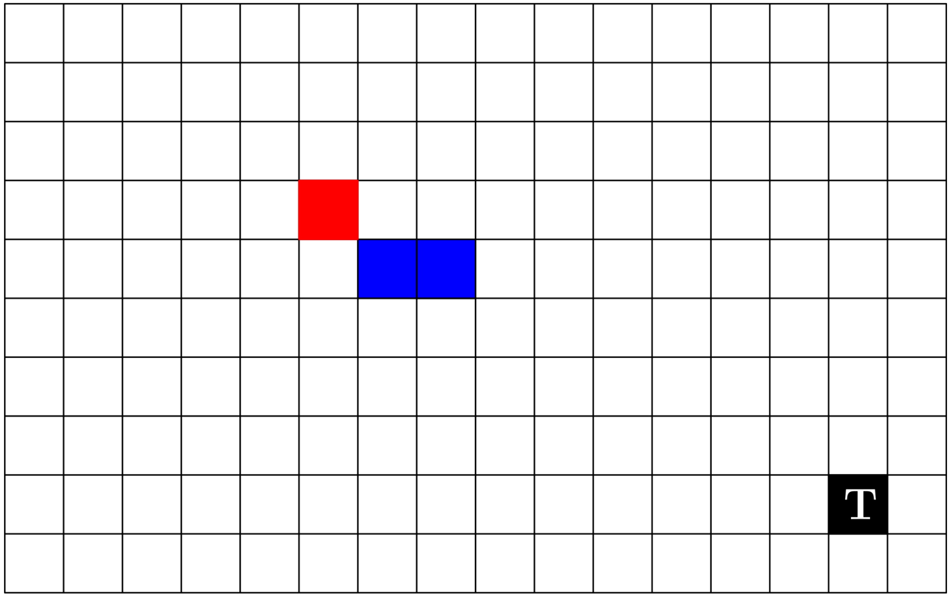
Rectangle Search



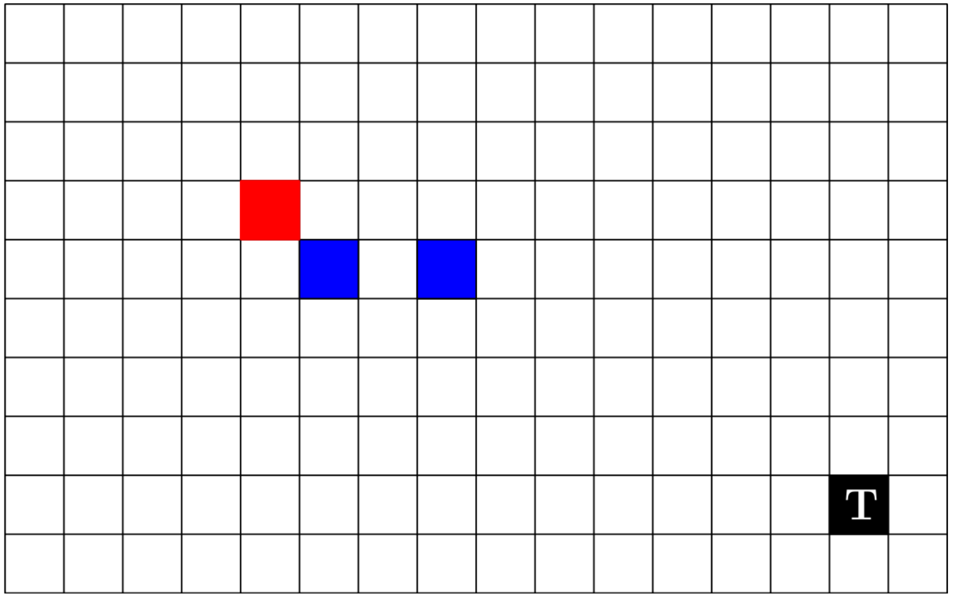
Rectangle Search



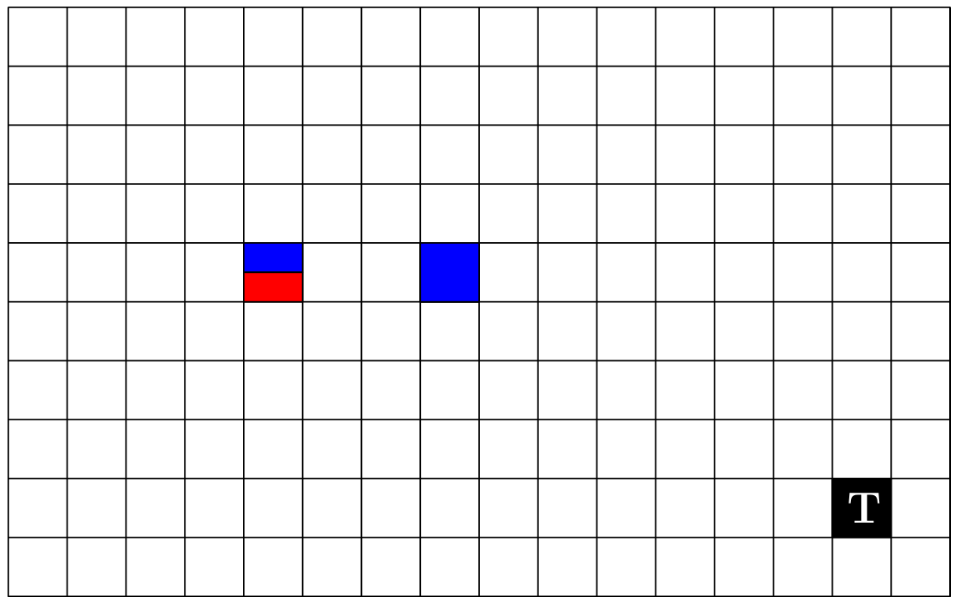
Rectangle Search



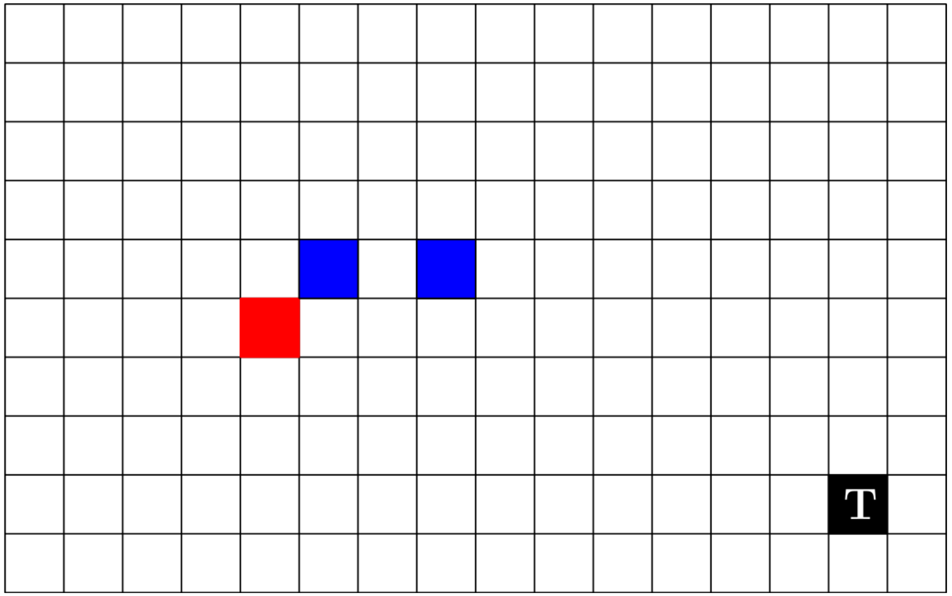
Rectangle Search



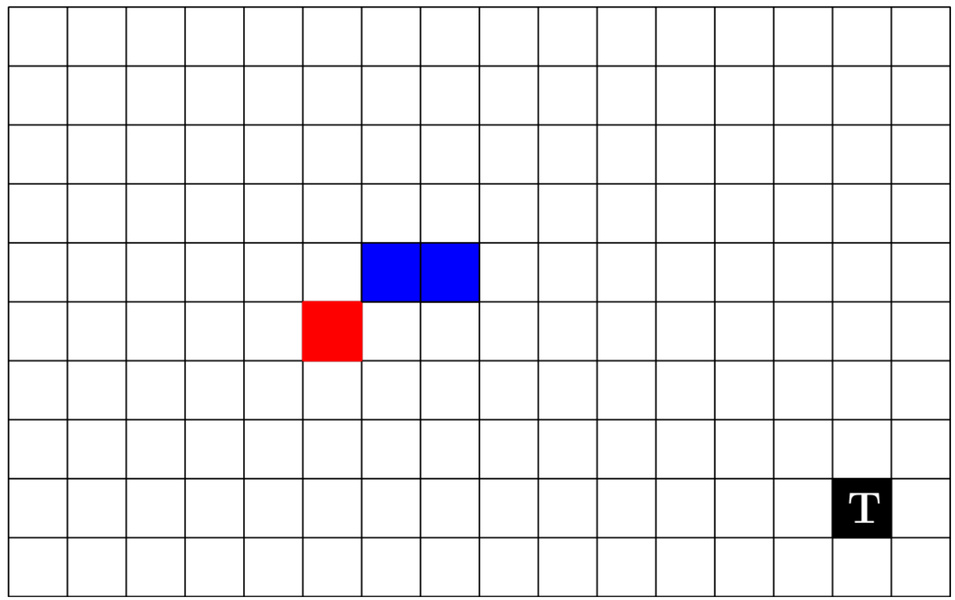
Rectangle Search



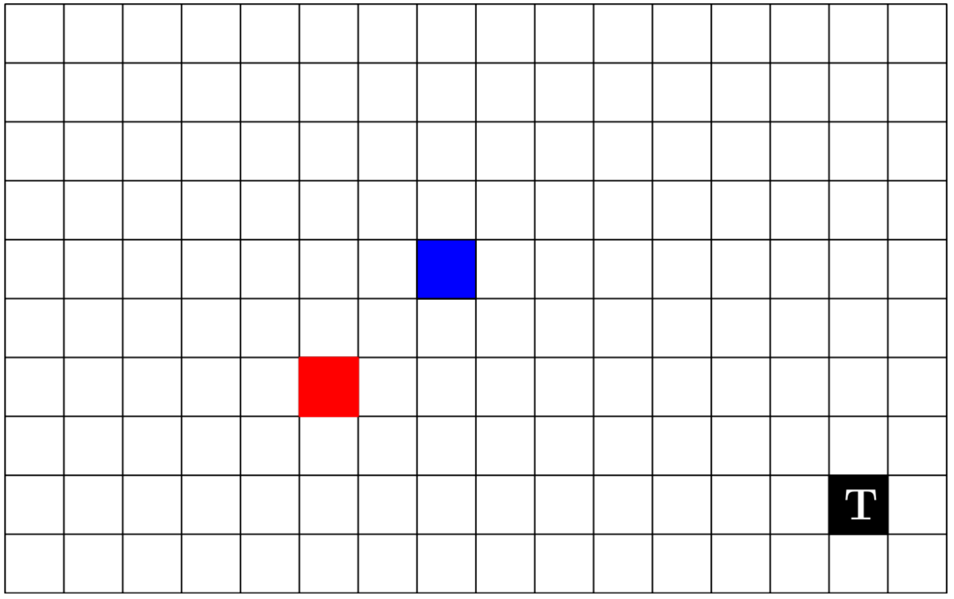
Rectangle Search



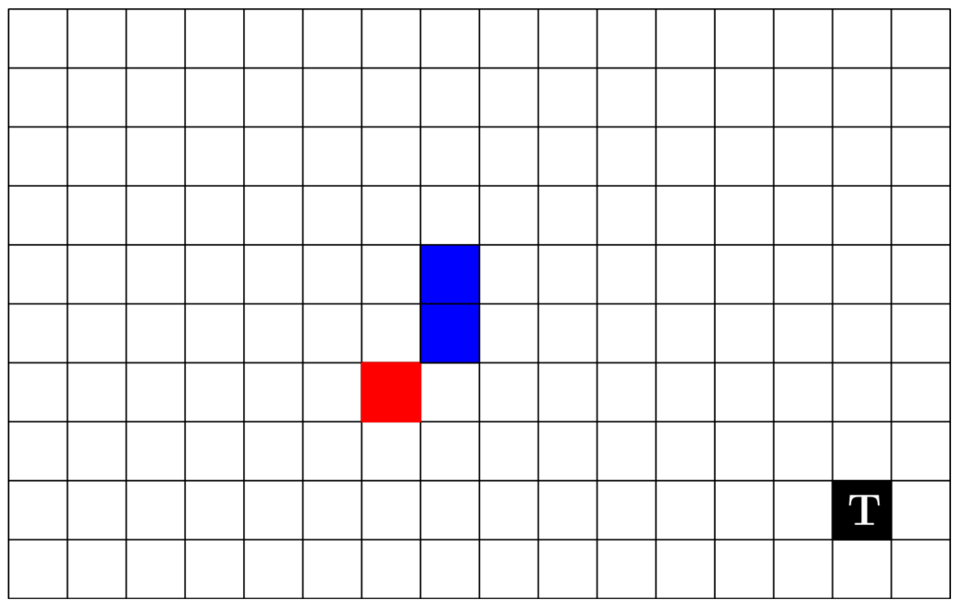
Rectangle Search



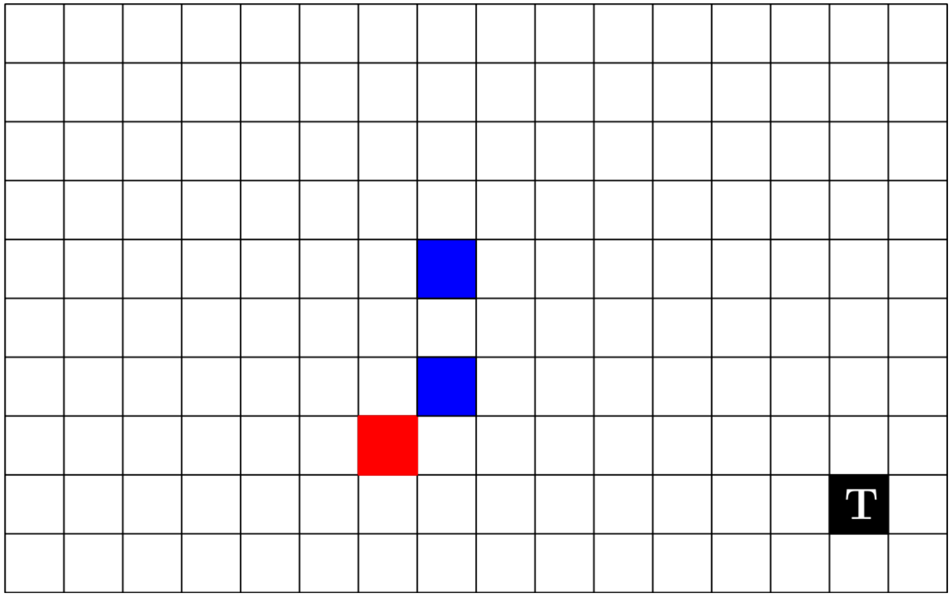
Rectangle Search



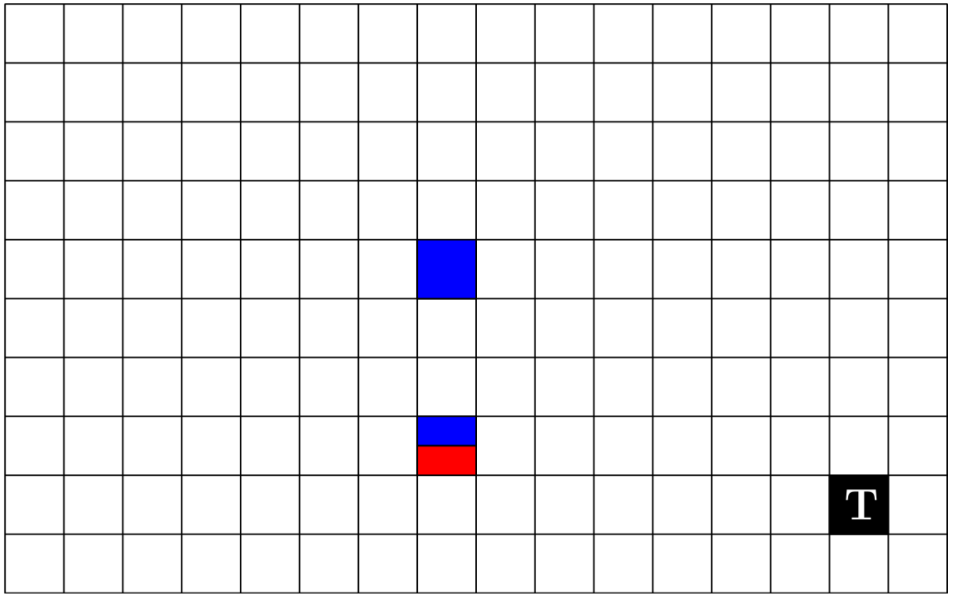
Rectangle Search



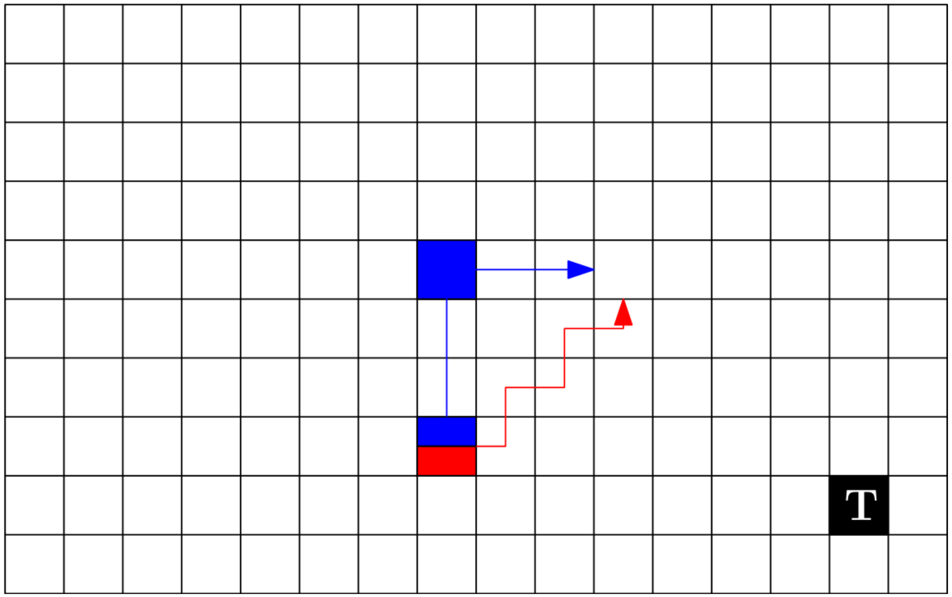
Rectangle Search



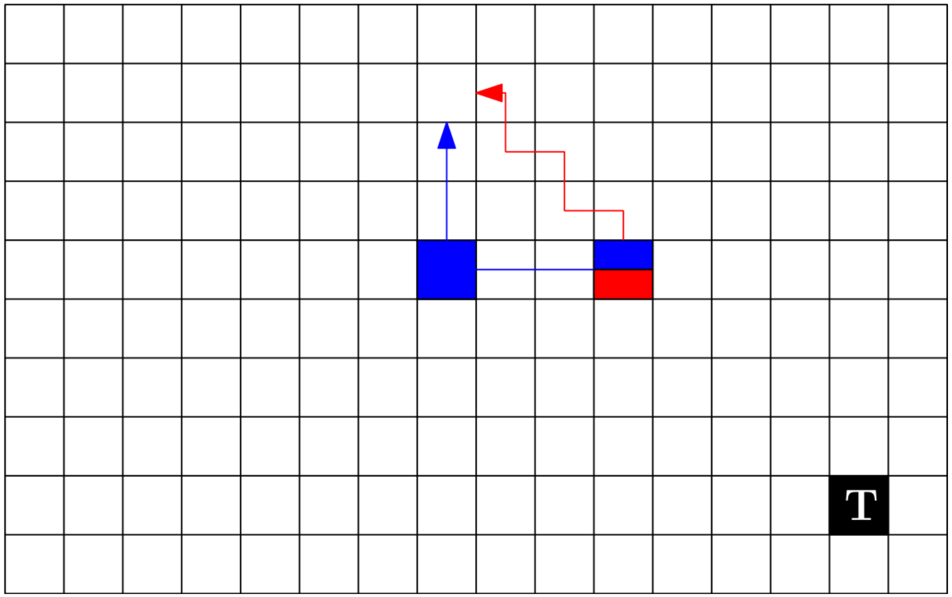
Rectangle Search



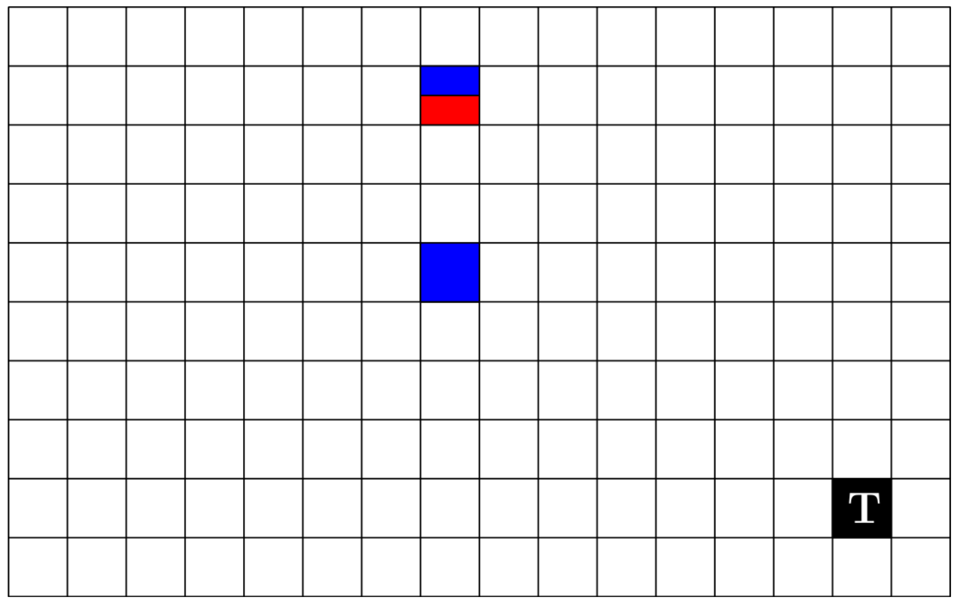
Rectangle Search



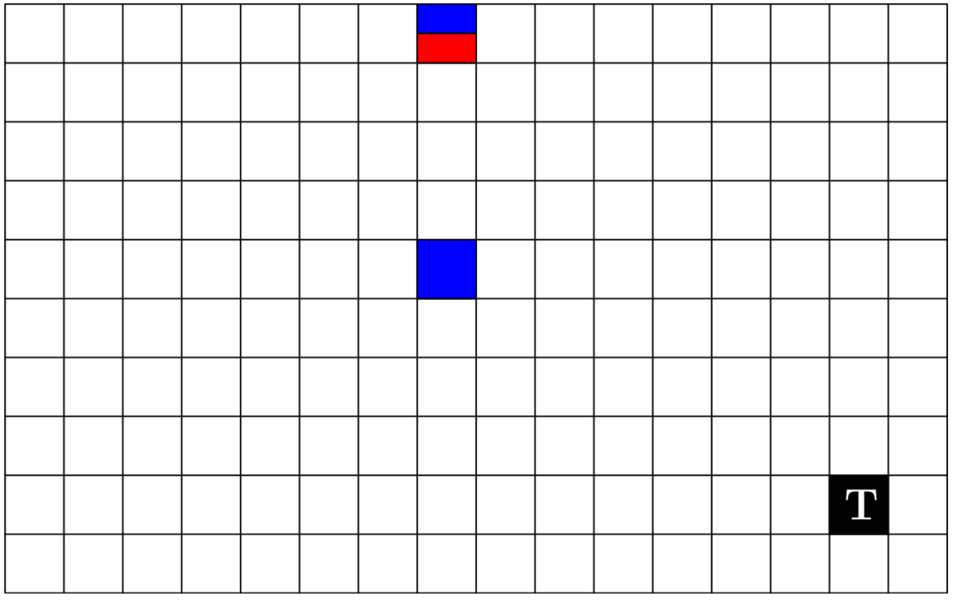
Rectangle Search



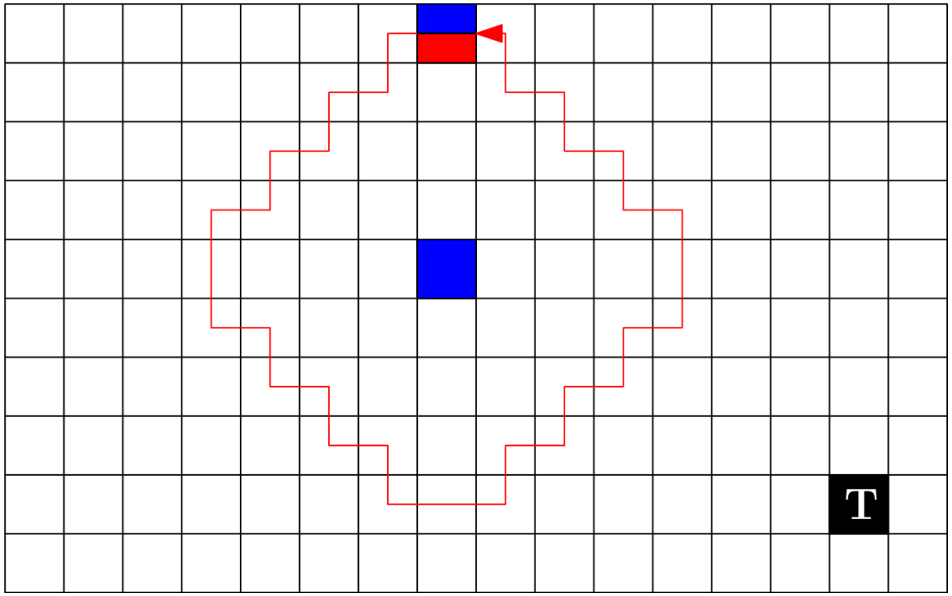
Rectangle Search



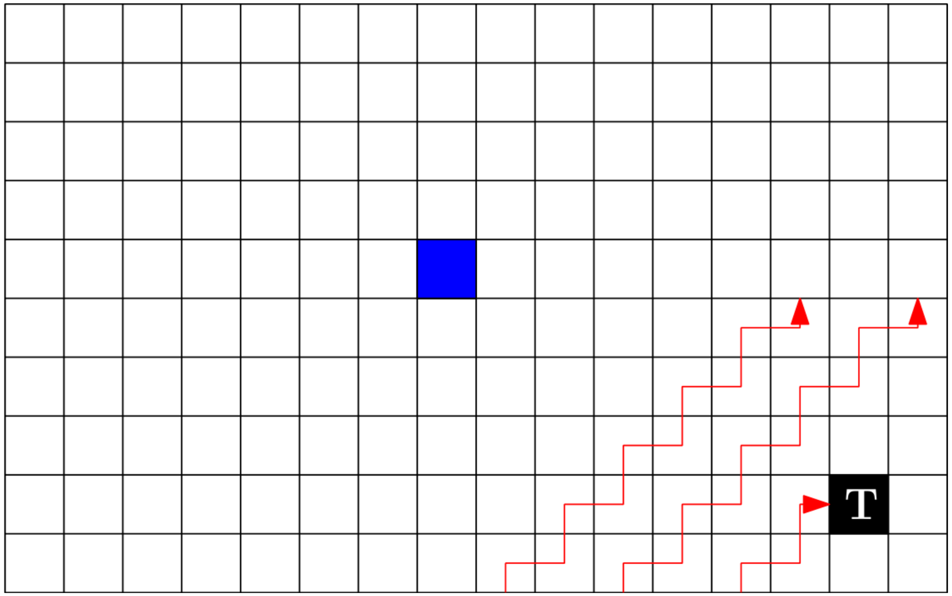
Rectangle Search



Rectangle Search



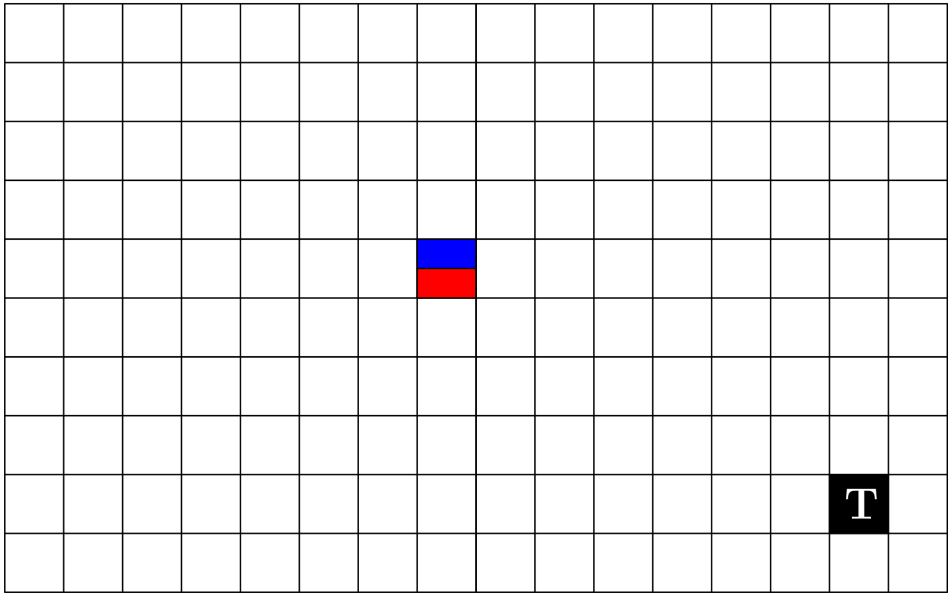
Rectangle Search



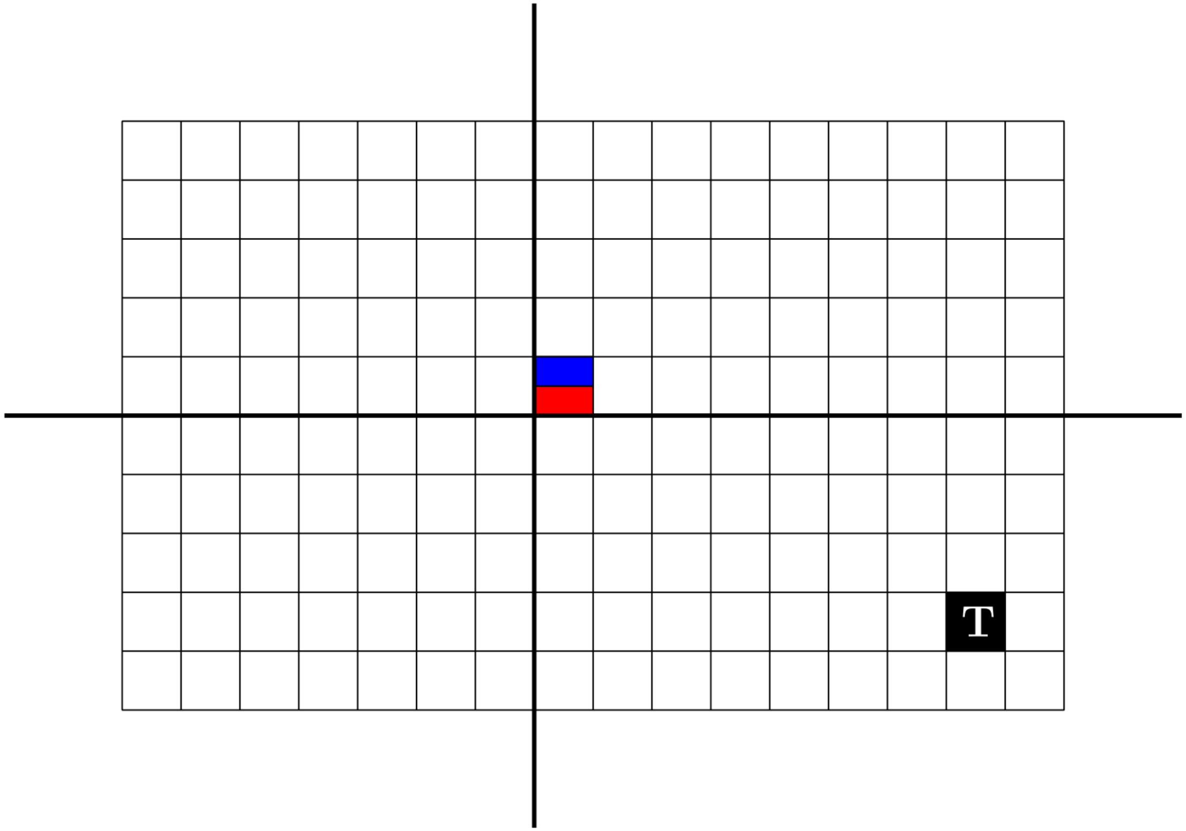
Randomization

- How about random coin tosses?

Geometric Search

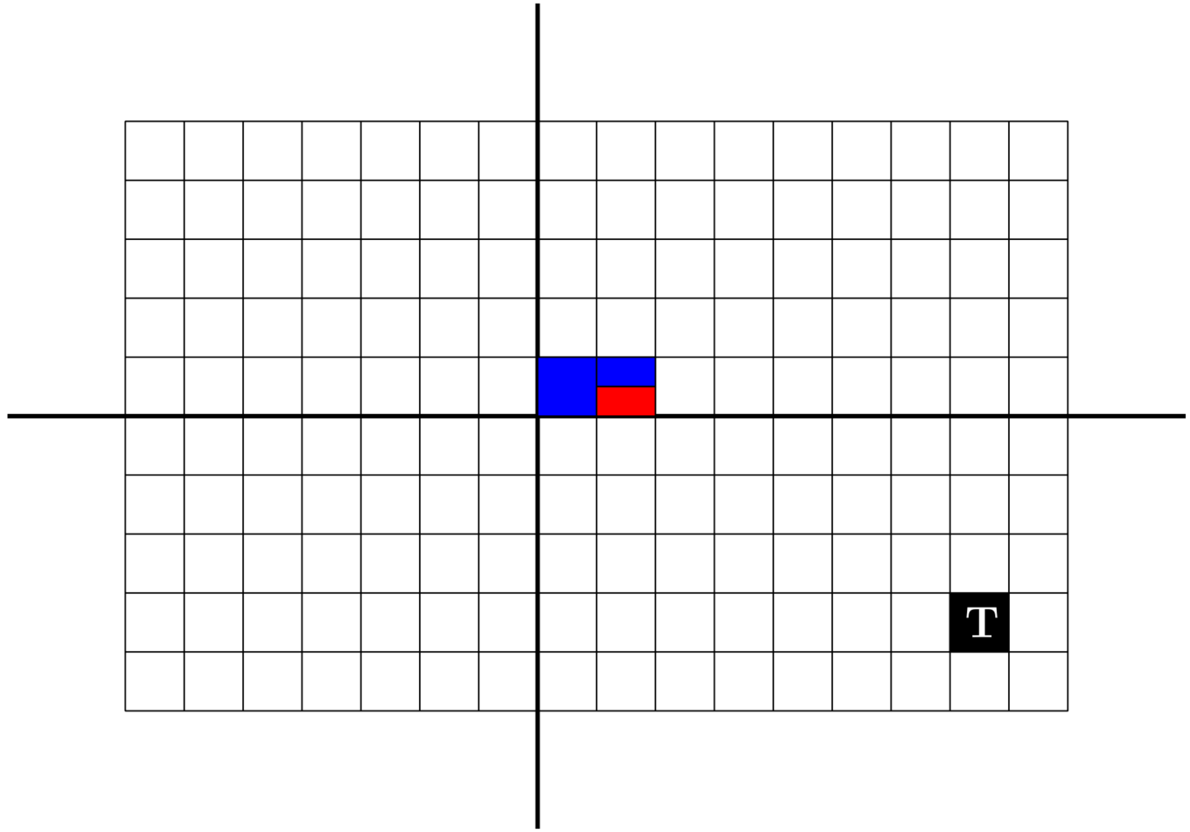


Geometric Search



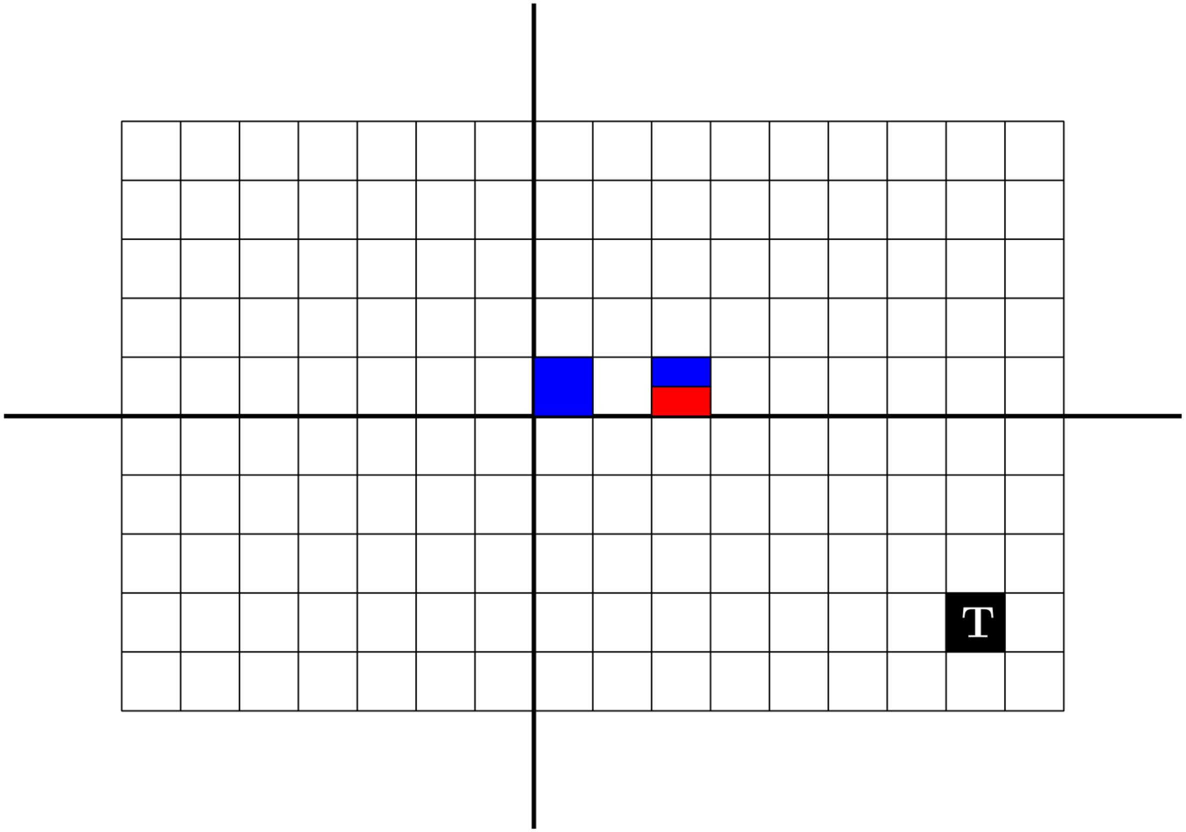
NE

Geometric Search



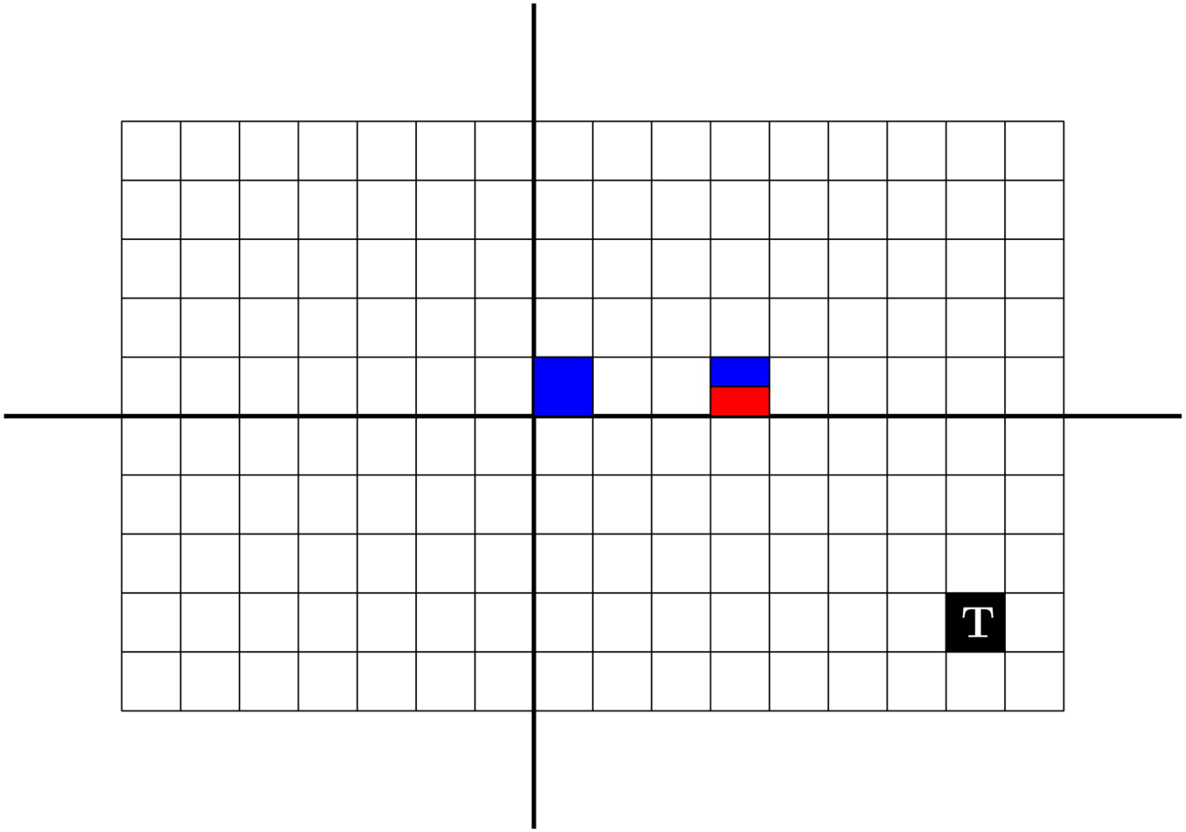
NE 1

Geometric Search



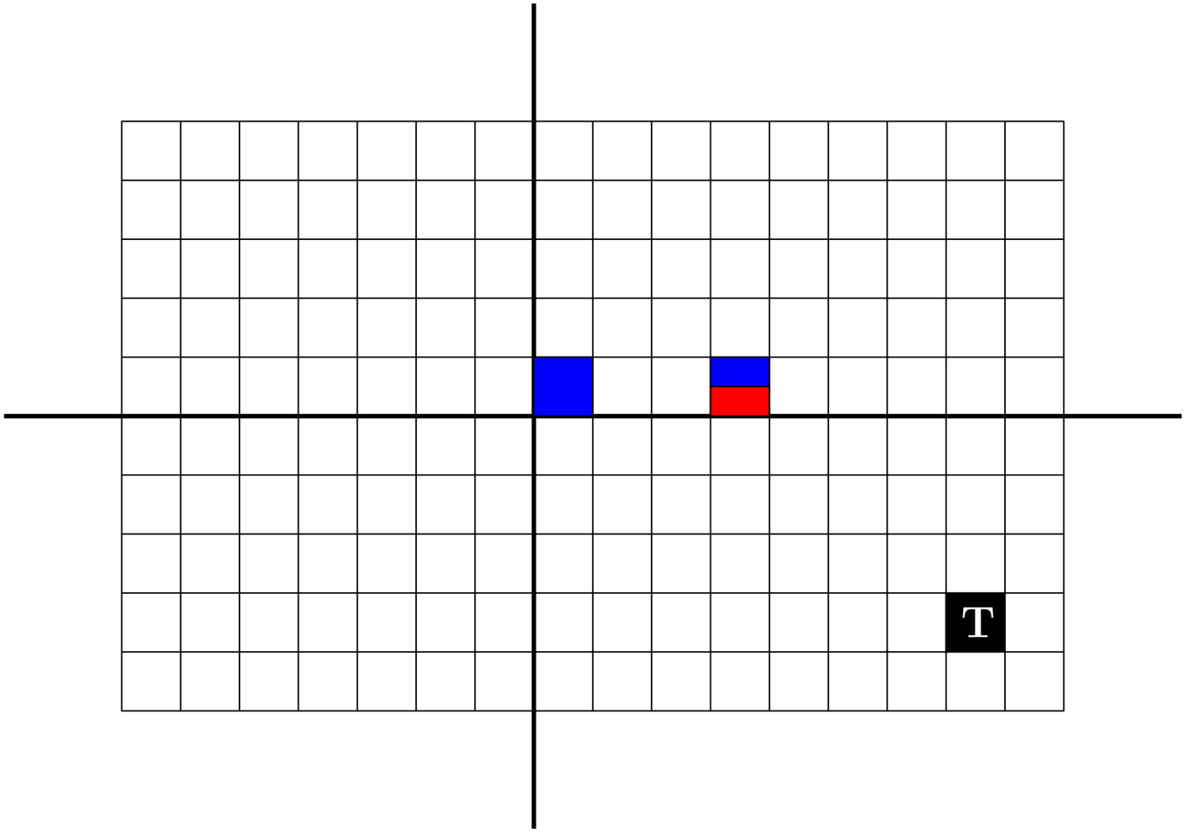
NE 11

Geometric Search



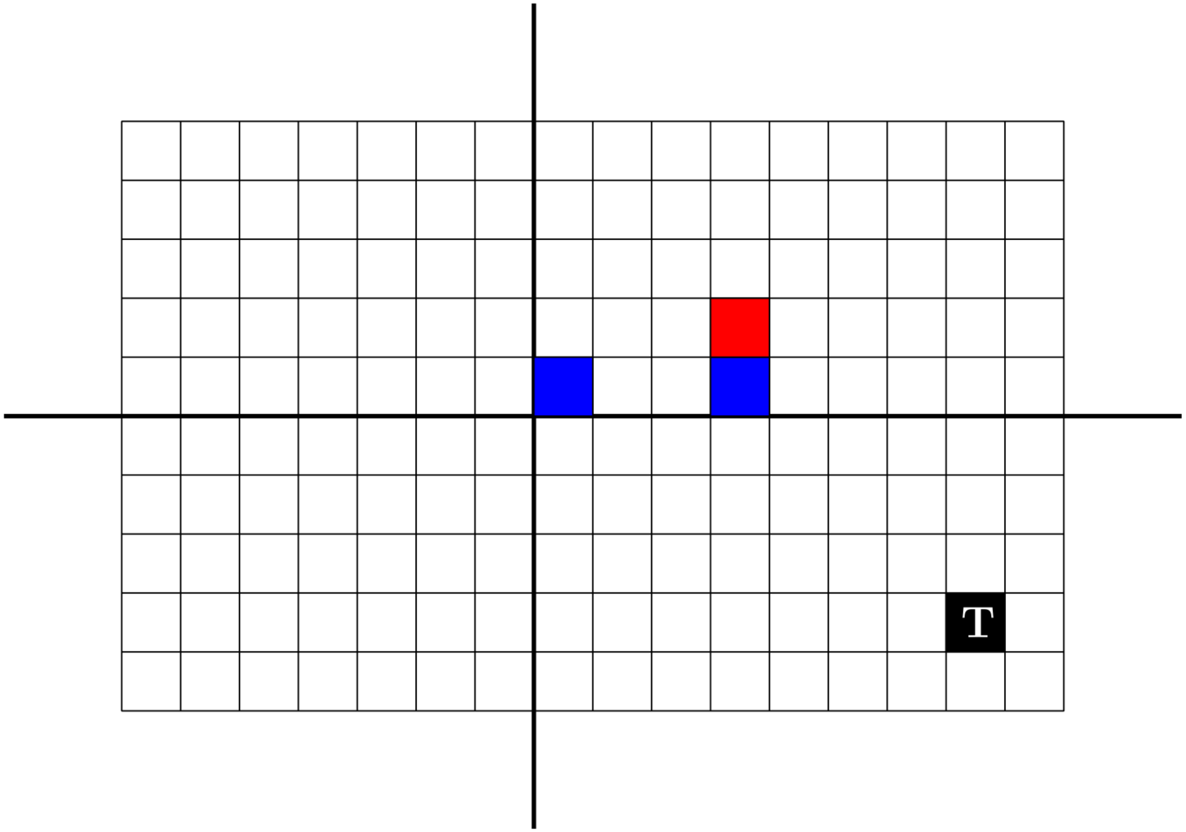
NE 111

Geometric Search



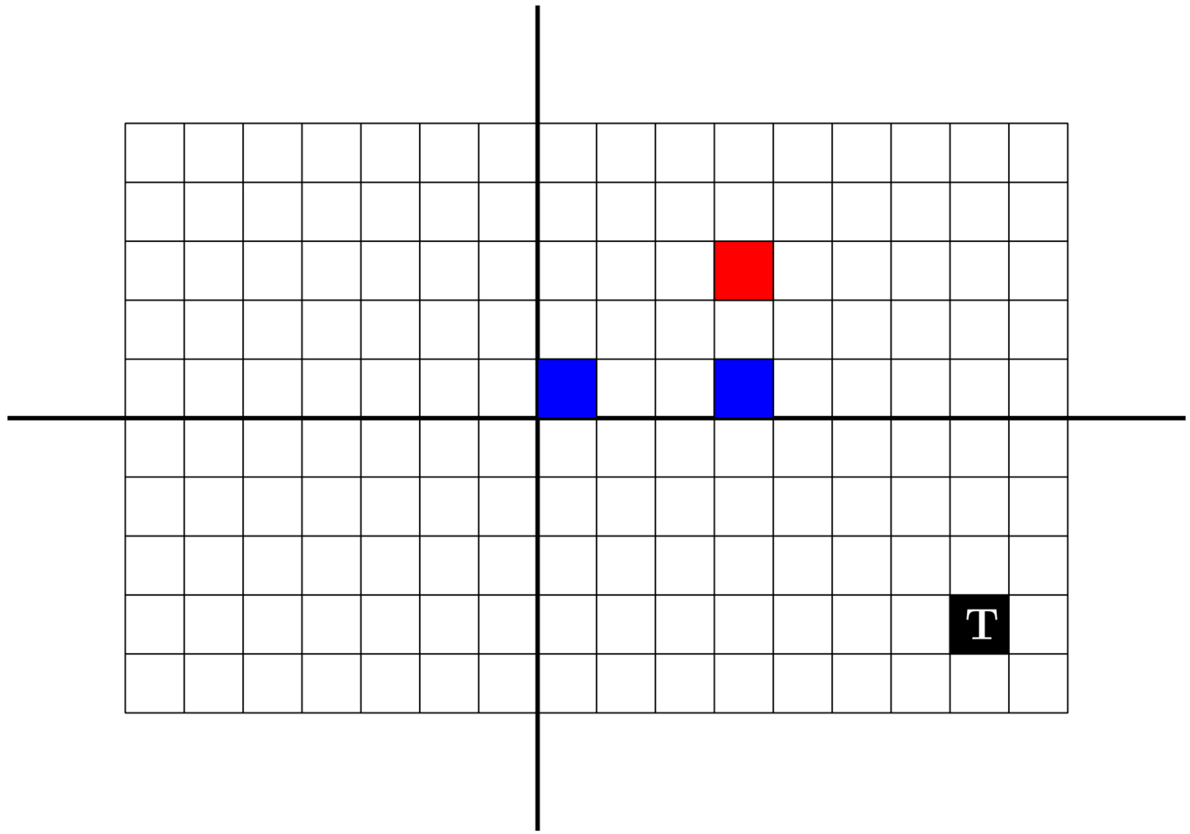
NE 1110

Geometric Search



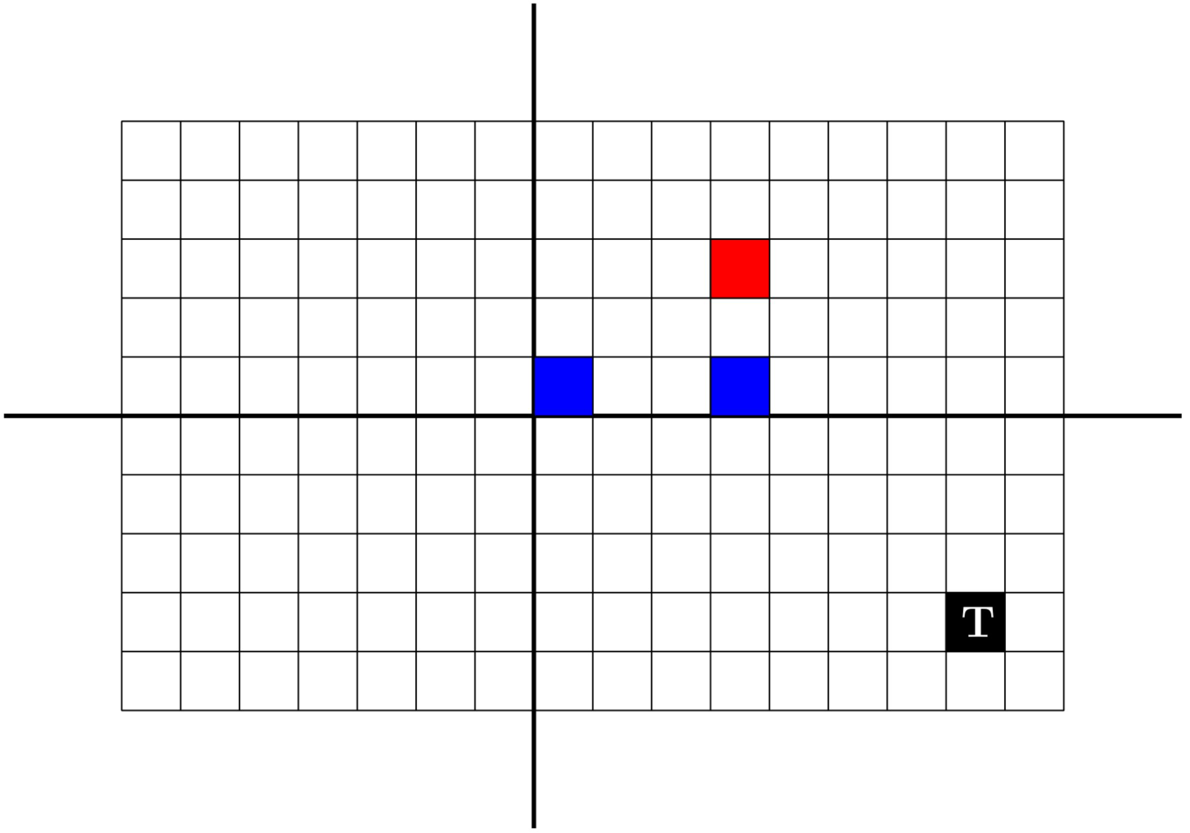
NE 11101

Geometric Search



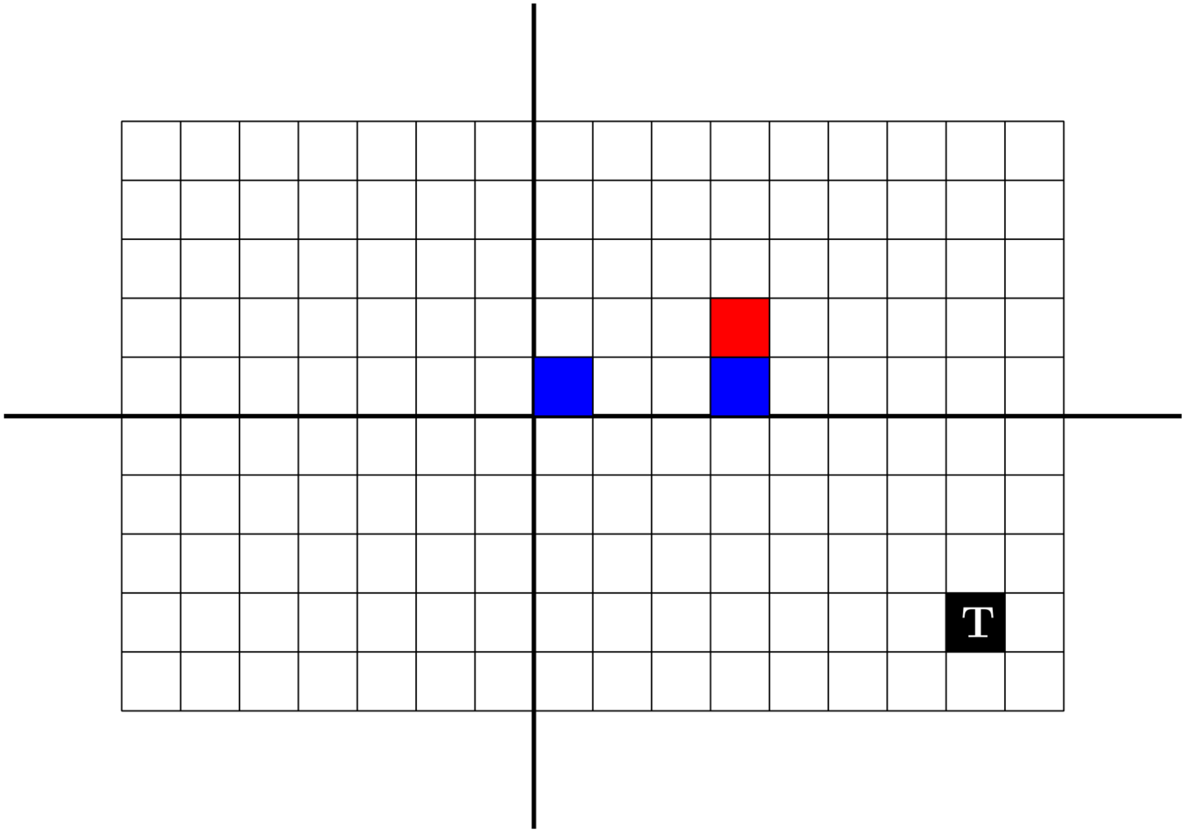
NE 111011

Geometric Search



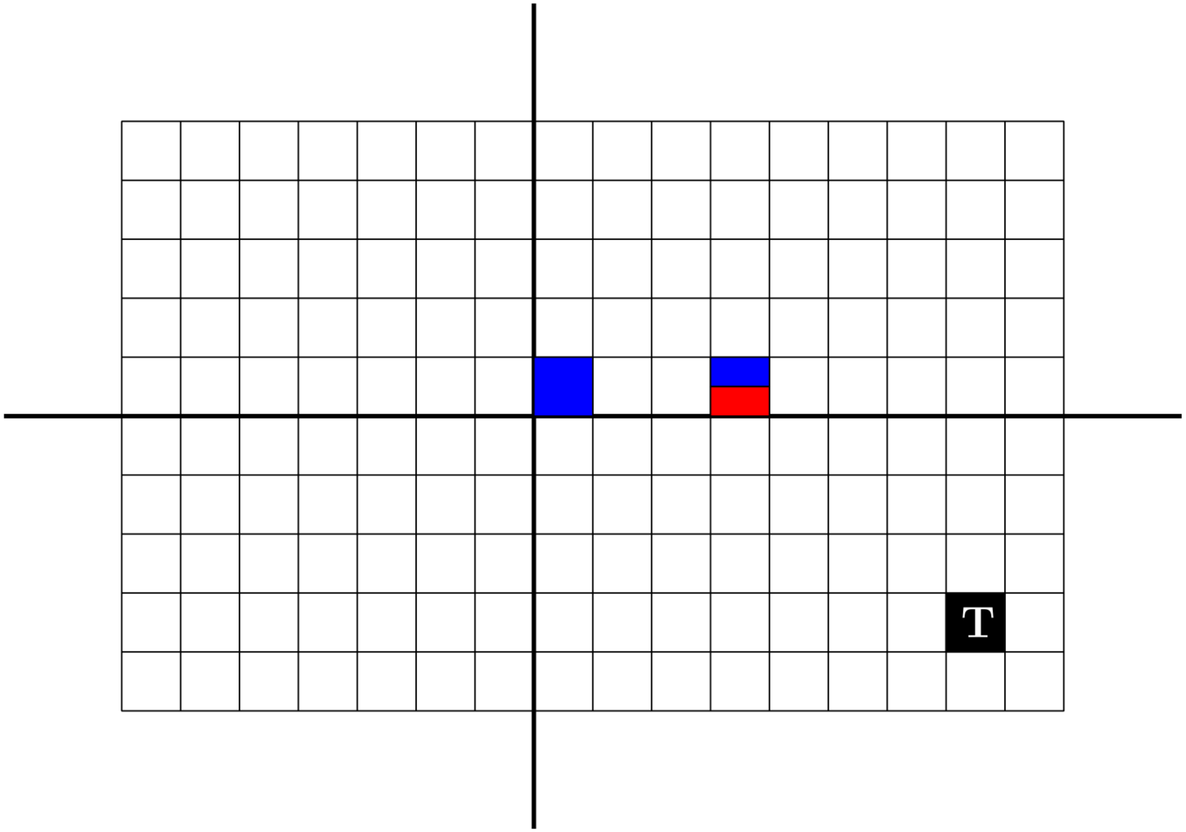
NE 1110110

Geometric Search



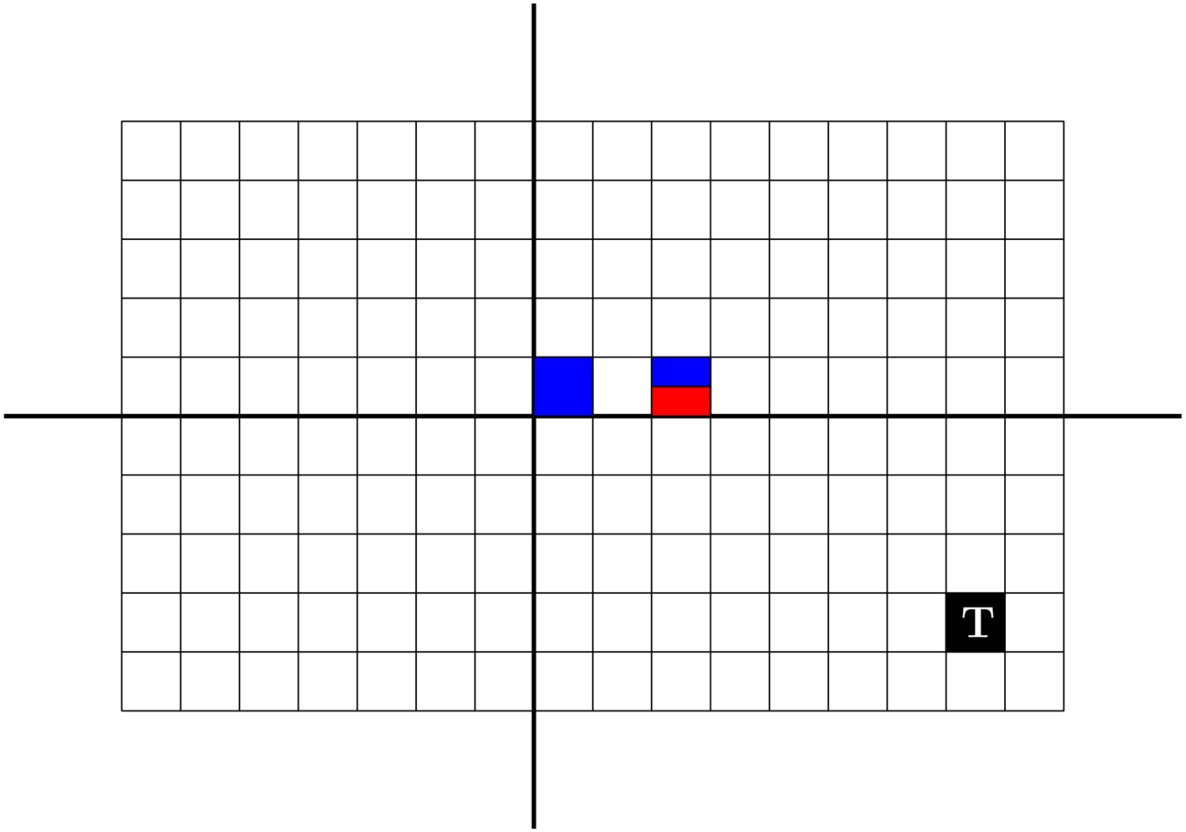
NE 1110110

Geometric Search



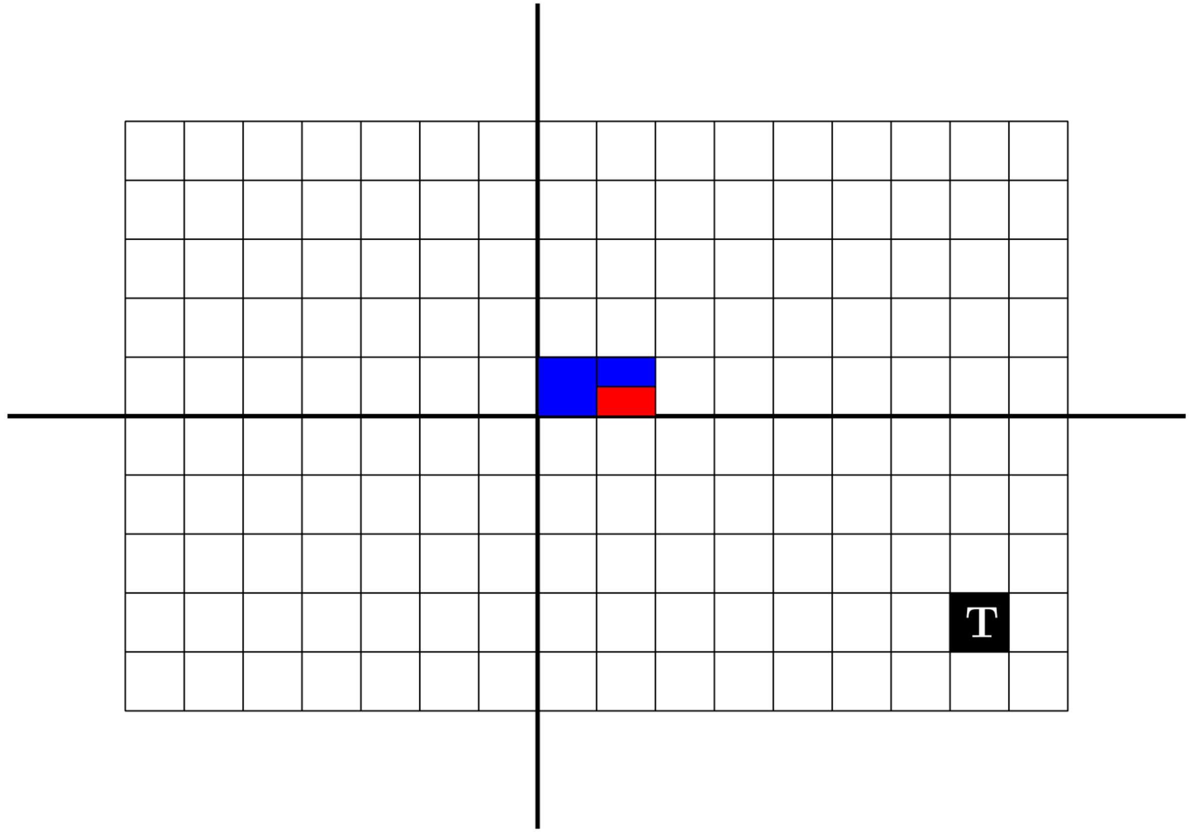
NE 1110110

Geometric Search



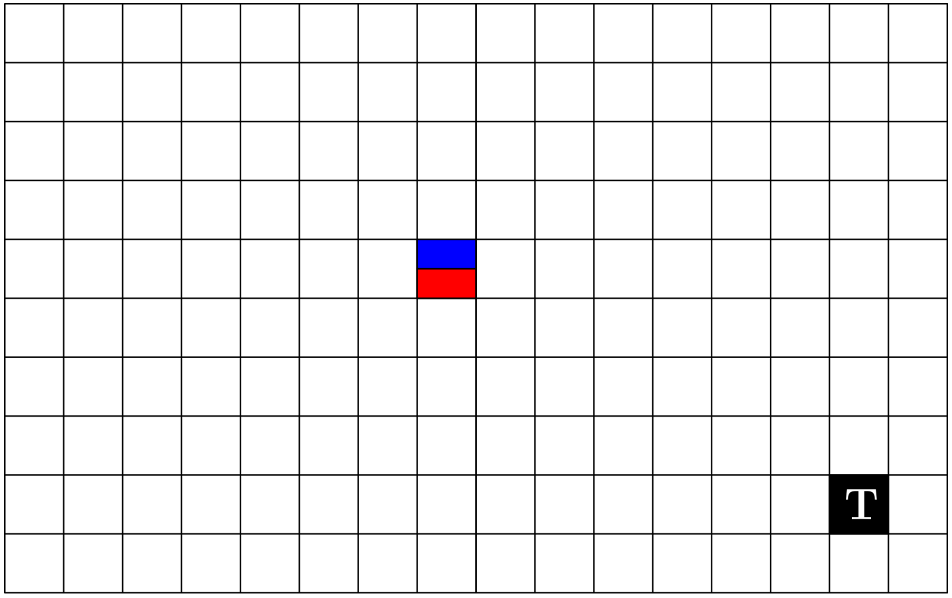
NE 1110110

Geometric Search



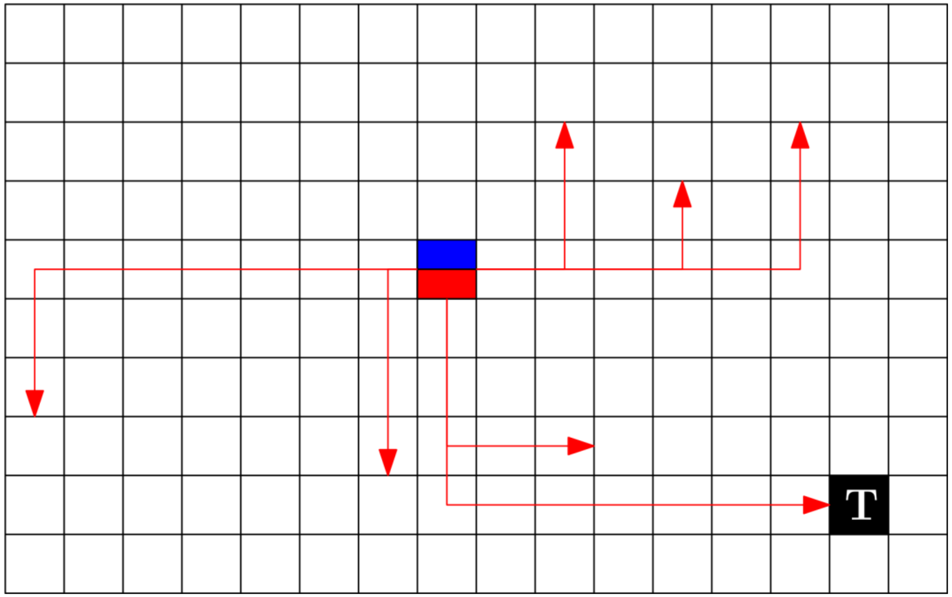
NE 1110110

Geometric Search



NE 1110110

Geometric Search



Run-Time

- For every search i, we have a probability of at least $\mathrm{A_i} = \frac{1}{4} \cdot 2^{-(D+1)}$ to find the treasure.

- Let B_i be the event that the treasure is not found during any search $j < i$.

Run-Time

- Let T be the total time required.
- $E[T] \ \leq \sum_{i=1}^{\infty} P(A_{i+1} \cdot B_i) \ (O(i) + O(D))$.
- $P(A_{i+1} \cdot B_i) \leq 2^{-(D+3)} \cdot \left(1 - 2^{-(D+3)}\right)^i$.
- $E[T] \ \leq 2^{-(D+3)} \sum_{i=1}^{\infty} \left(1 - 2^{-(D+3)}\right)^i (O(i) + O(D)) = O(2^D)$.

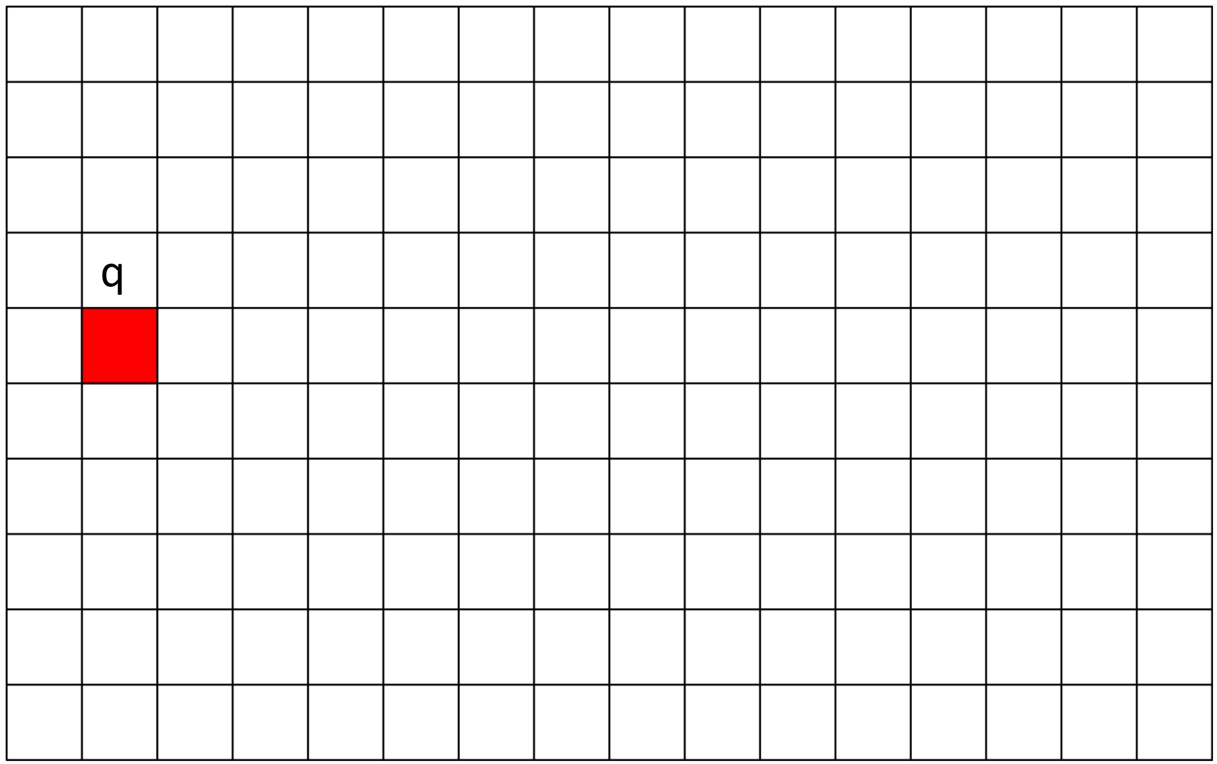
Lower Bounds?

- Can we do better? In the deterministic and synchronous case, the answer is no.
- Let us start with showing that one ant is not enough.

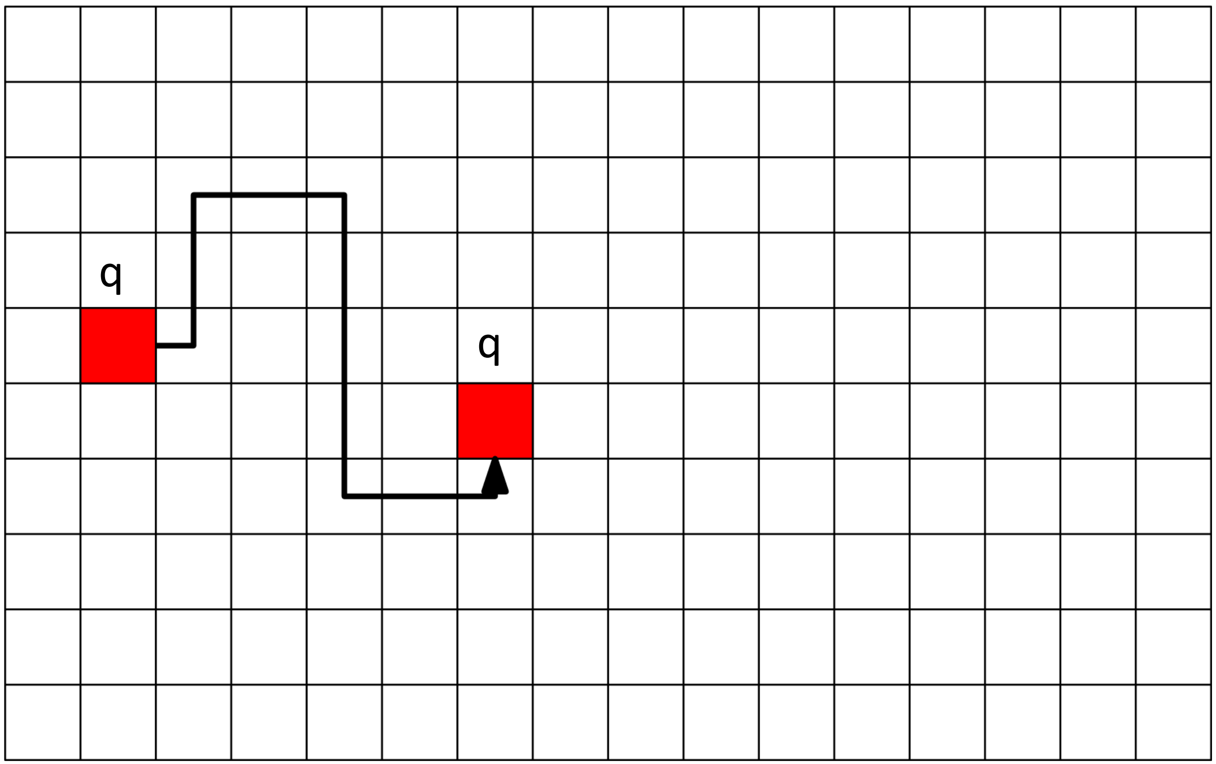
One Ant

- A finite state machine repeats its behavior.

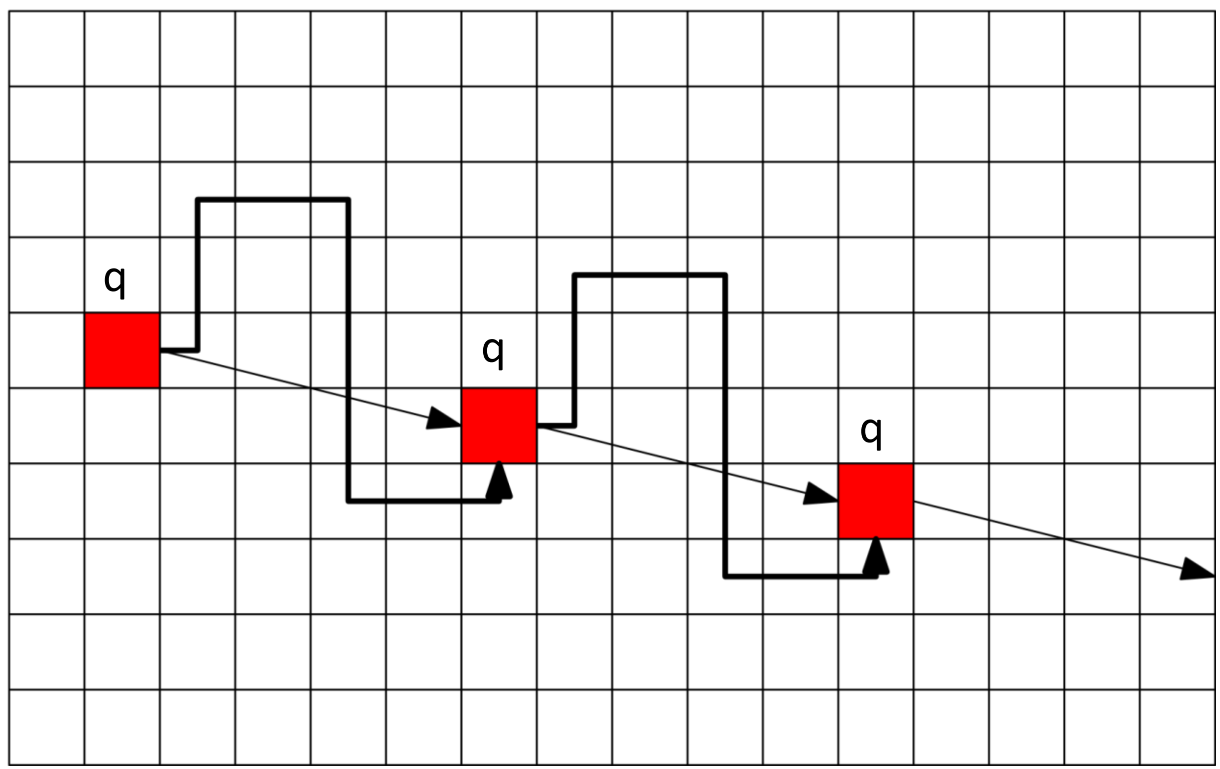
One Ant



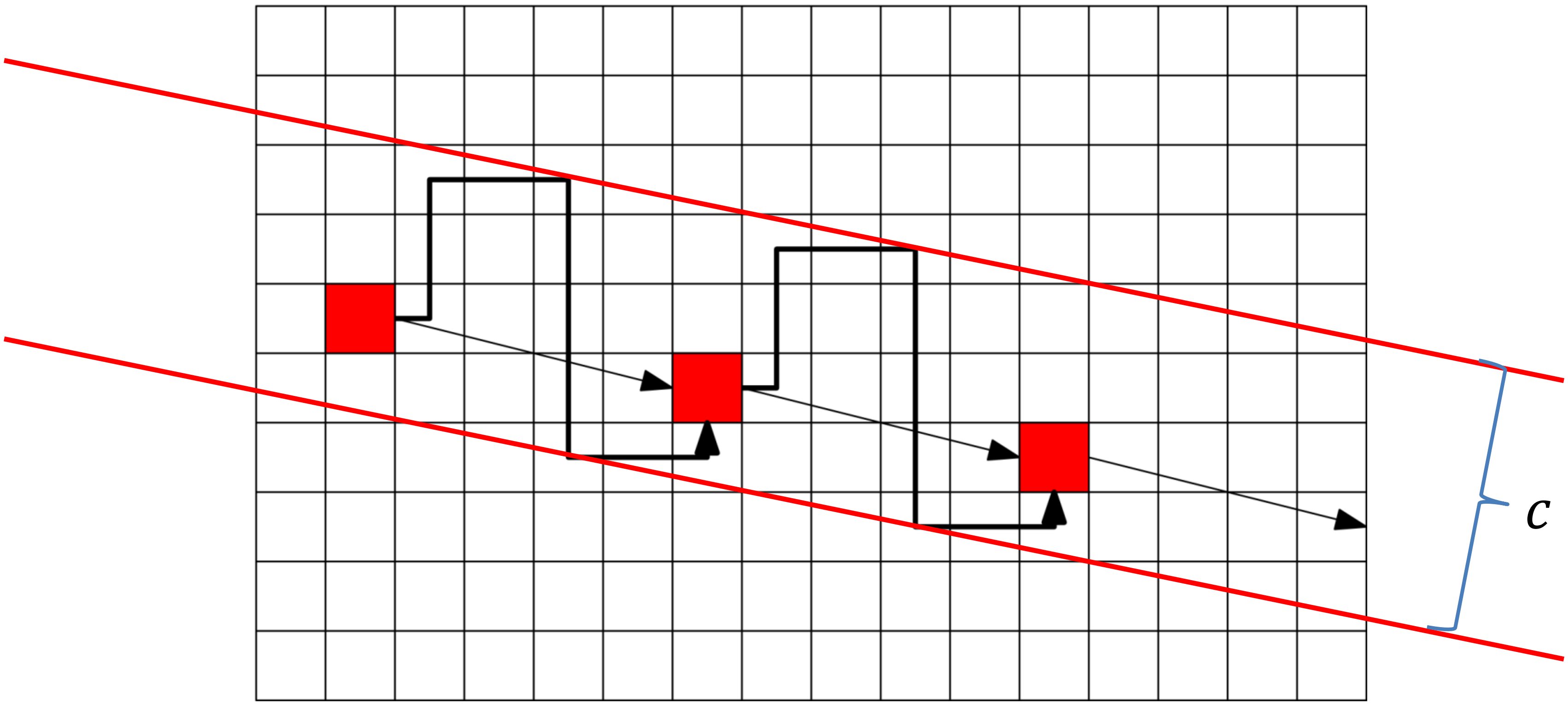
One Ant



One Ant



One Ant



A band of constant width

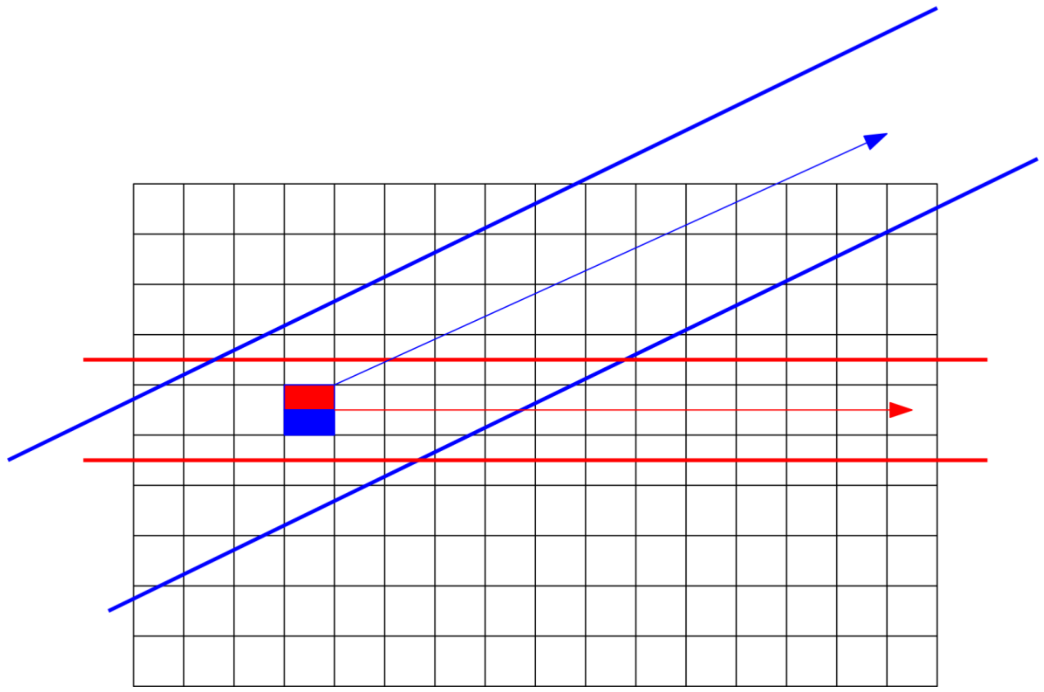
One Ant

- One ant can only discover a band of constant width.

- How about two ants?

Two Ants

- Let t be the time of the last meeting.



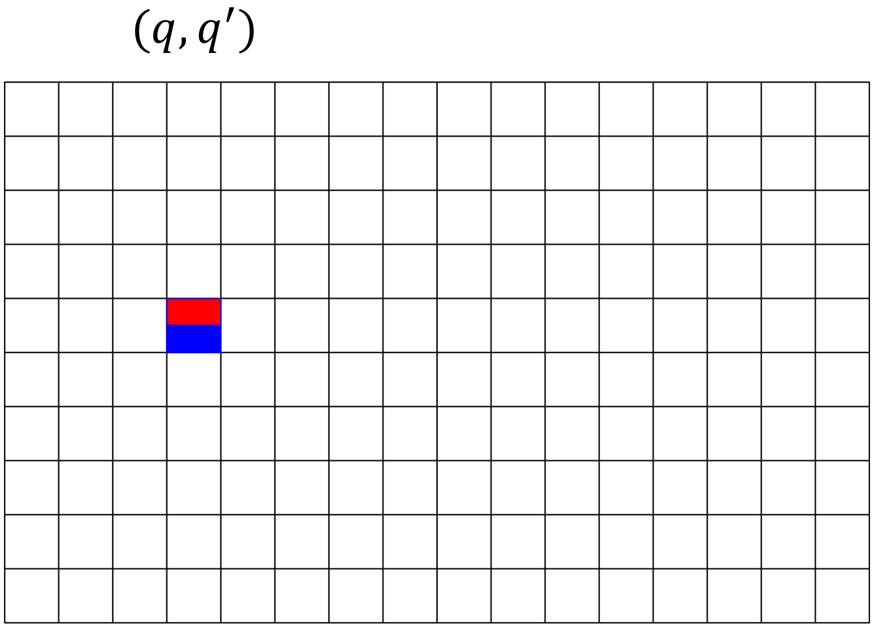
- Both agents (alone) discover a band after t.

Two Ants

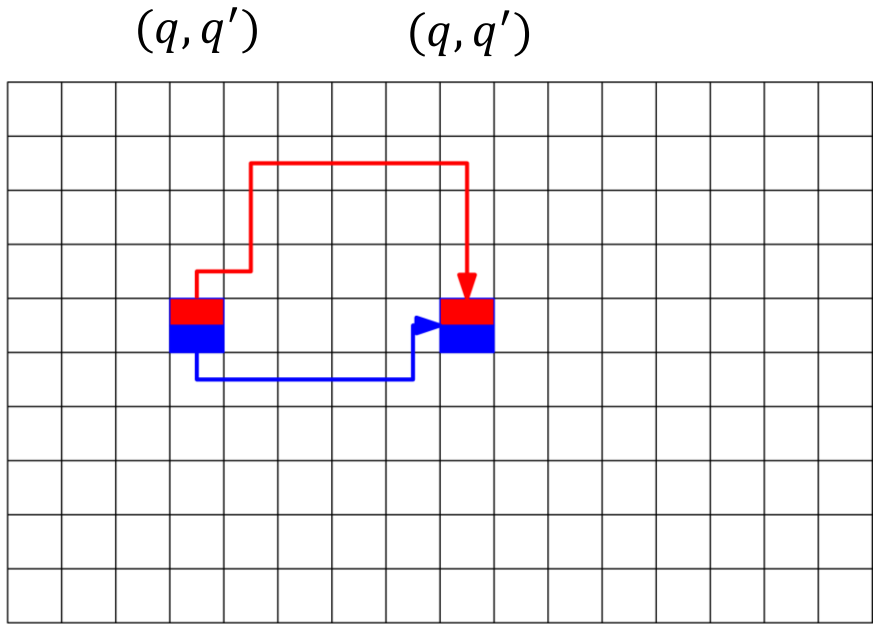
- Lemma: The ants meet infinitely often in some pair of states (q, q').

- Observation: the time between two such meetings is bounded by a constant.

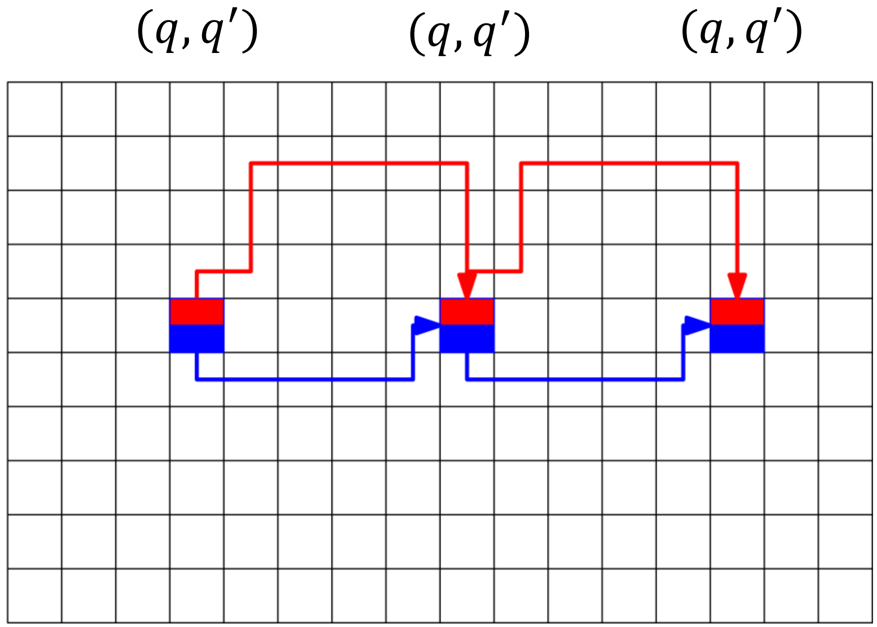
Two Ants



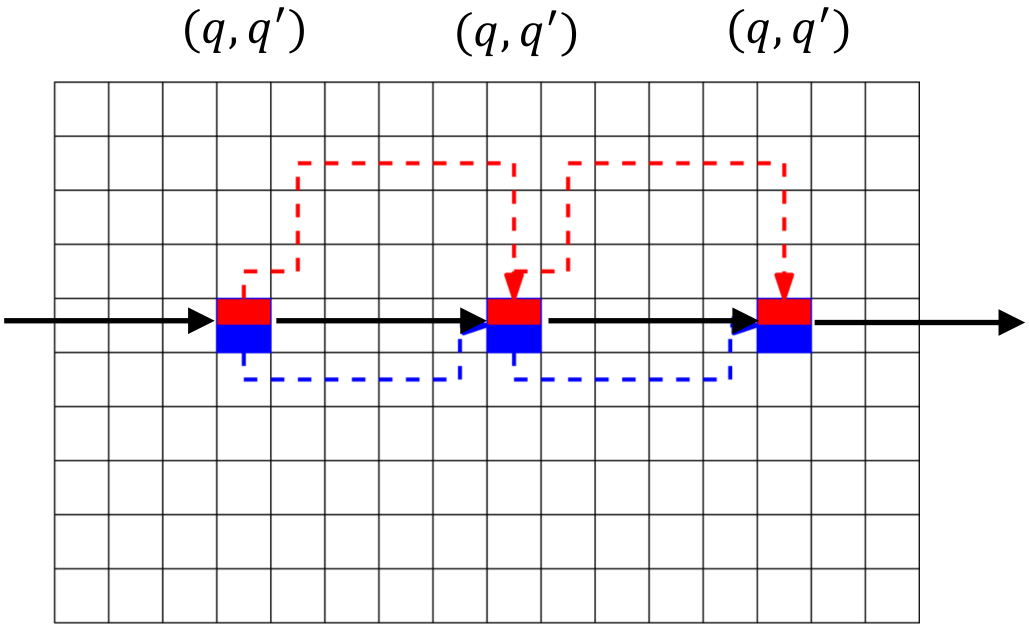
Two Ants



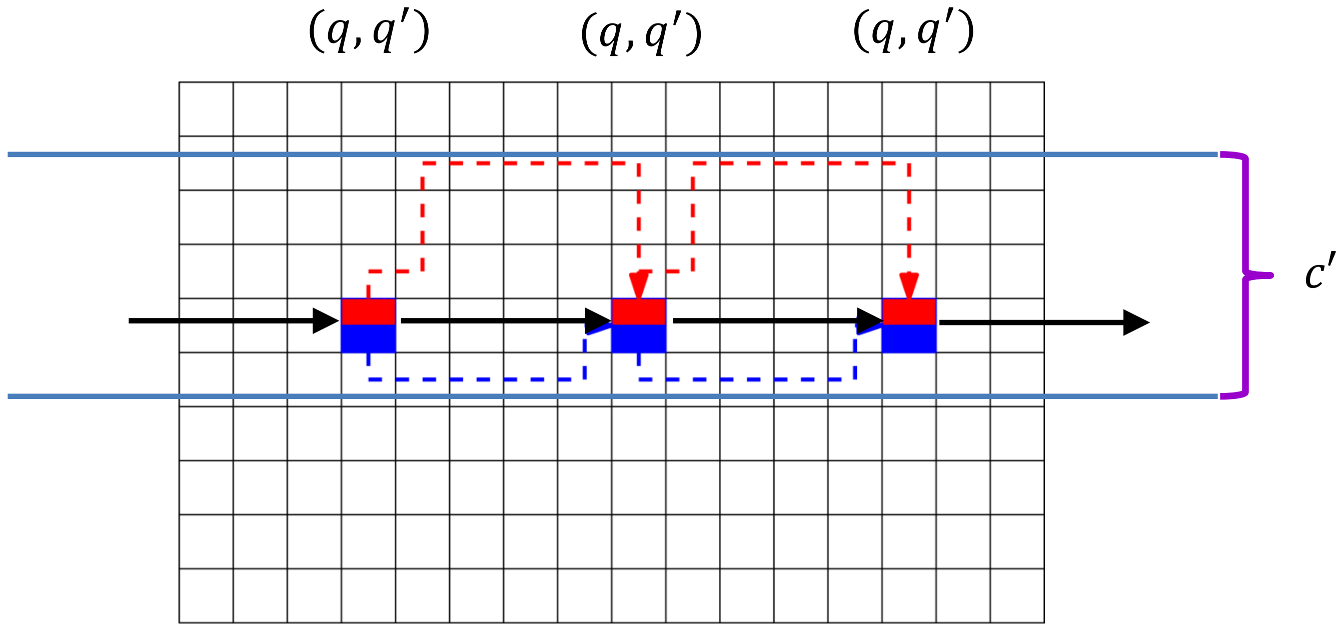
Two Ants



Two Ants



Two Ants



- Two deterministic ants can only discover a band of constant width.
- Two deterministic ants cannot find the food.

Conclusion

Problem	FA sync det	FA sync rand	FA async det	FA async rand
One agent		×		×
Two agents	×	?	×	?
Three agents	✓	✓	?	✓
Four agents			✓	

- Three asynchronous ants?
- Two randomized ants?

Conclusion

Problem	PDA			
	sync		async	
	det	rand	det	rand
One agent	×	✓	×	✓
Two agents	✓		✓	
Three agents				
Four agents				

Questions?

Thanks to my co-authors
Yuval Emek, Tobias Langner, David Stolz and Roger Wattenhofer