

# Graph-Based Ant Colony Simulation

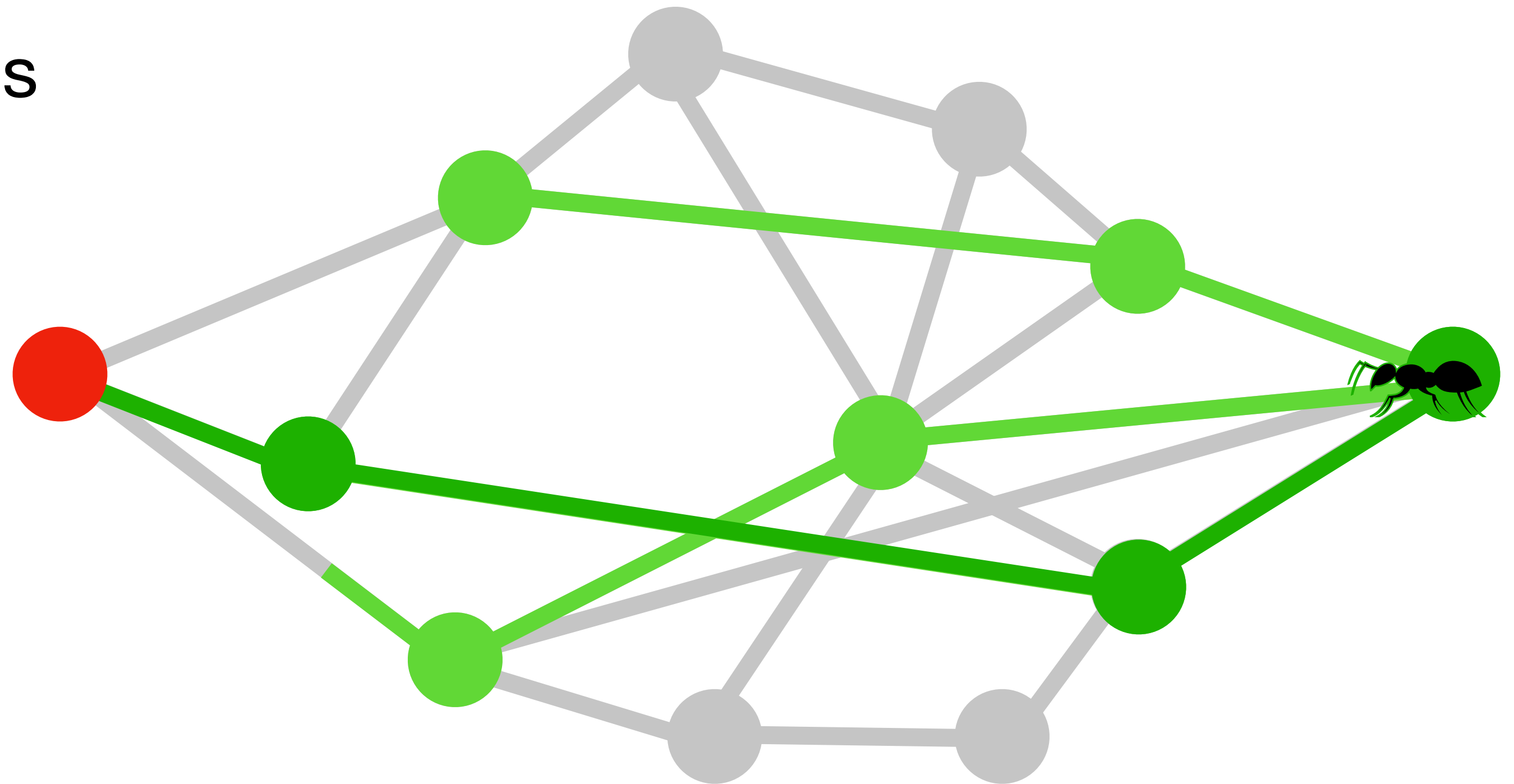
Implemented in JavaScript

朱劭璿、吳孟庭、陳居庭、林竝逸, Jan 4, 2021

# Recap

## Ant Colony Optimization

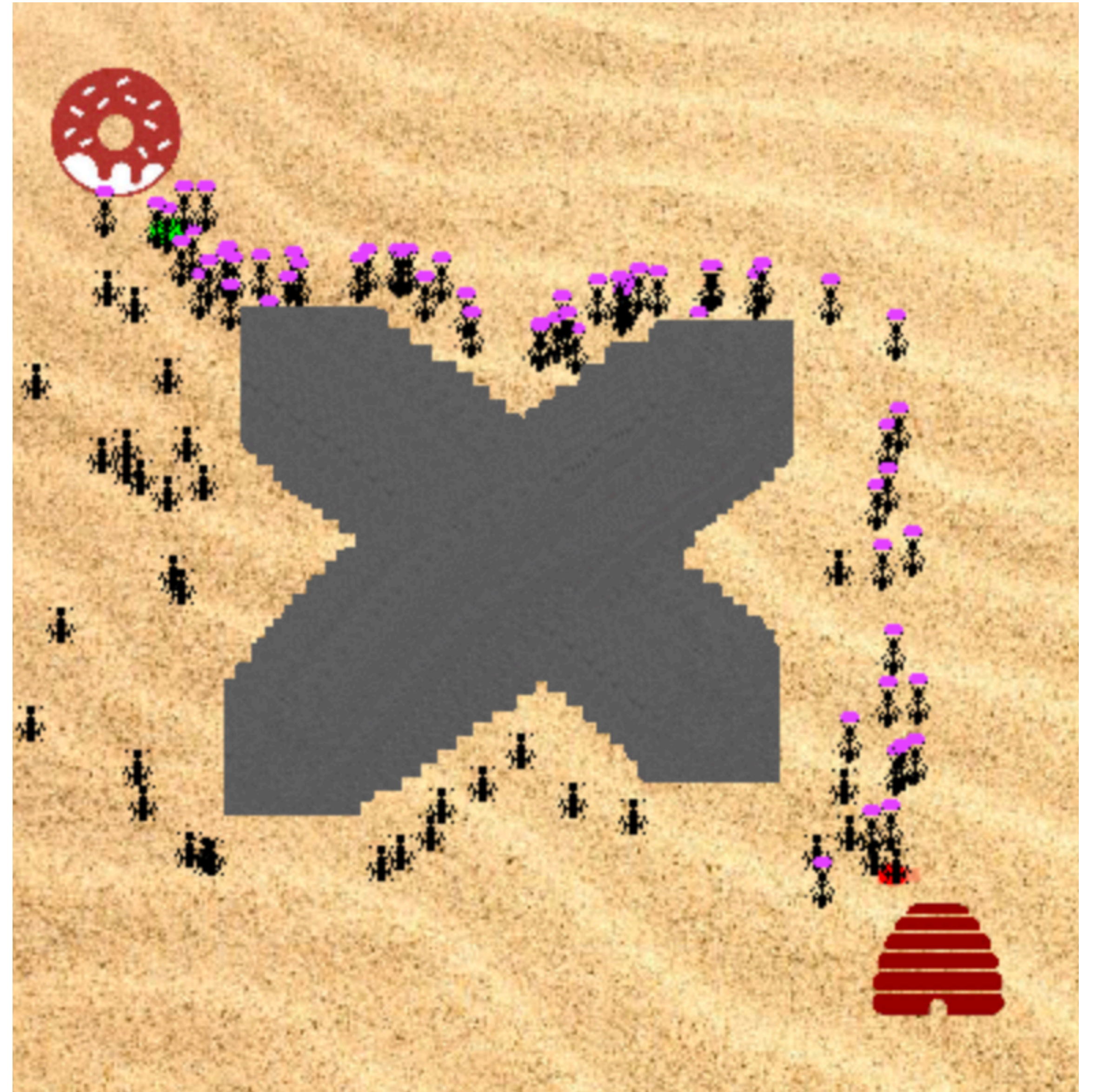
- Ant Colony Optimization helps you find paths between vertices
- We discussed back in midterms
- We even build a program!





# Back in midterms

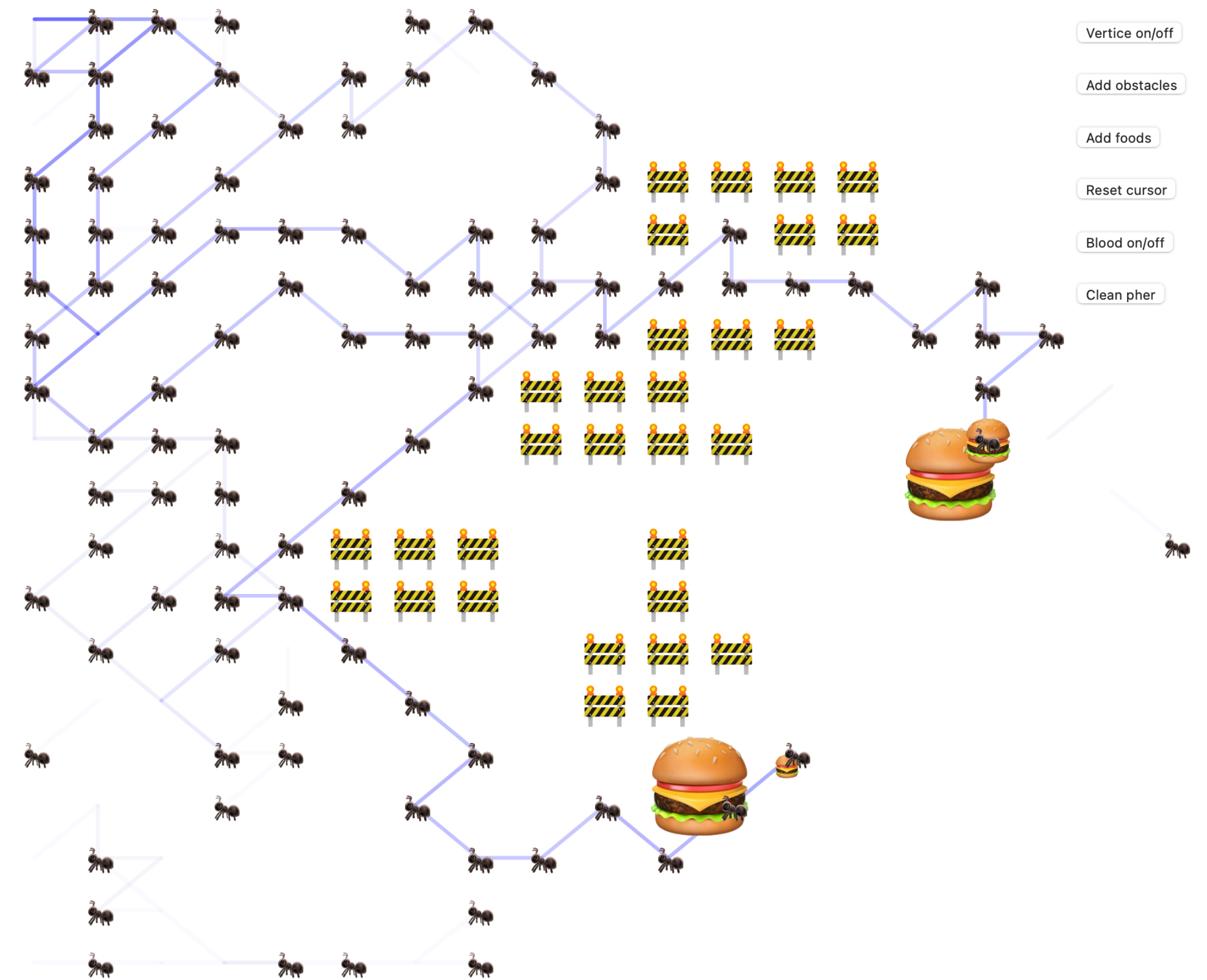
- We used dual pheromone to help the ants find home
- We placed food and obstacles
- We wrote the program in pygame



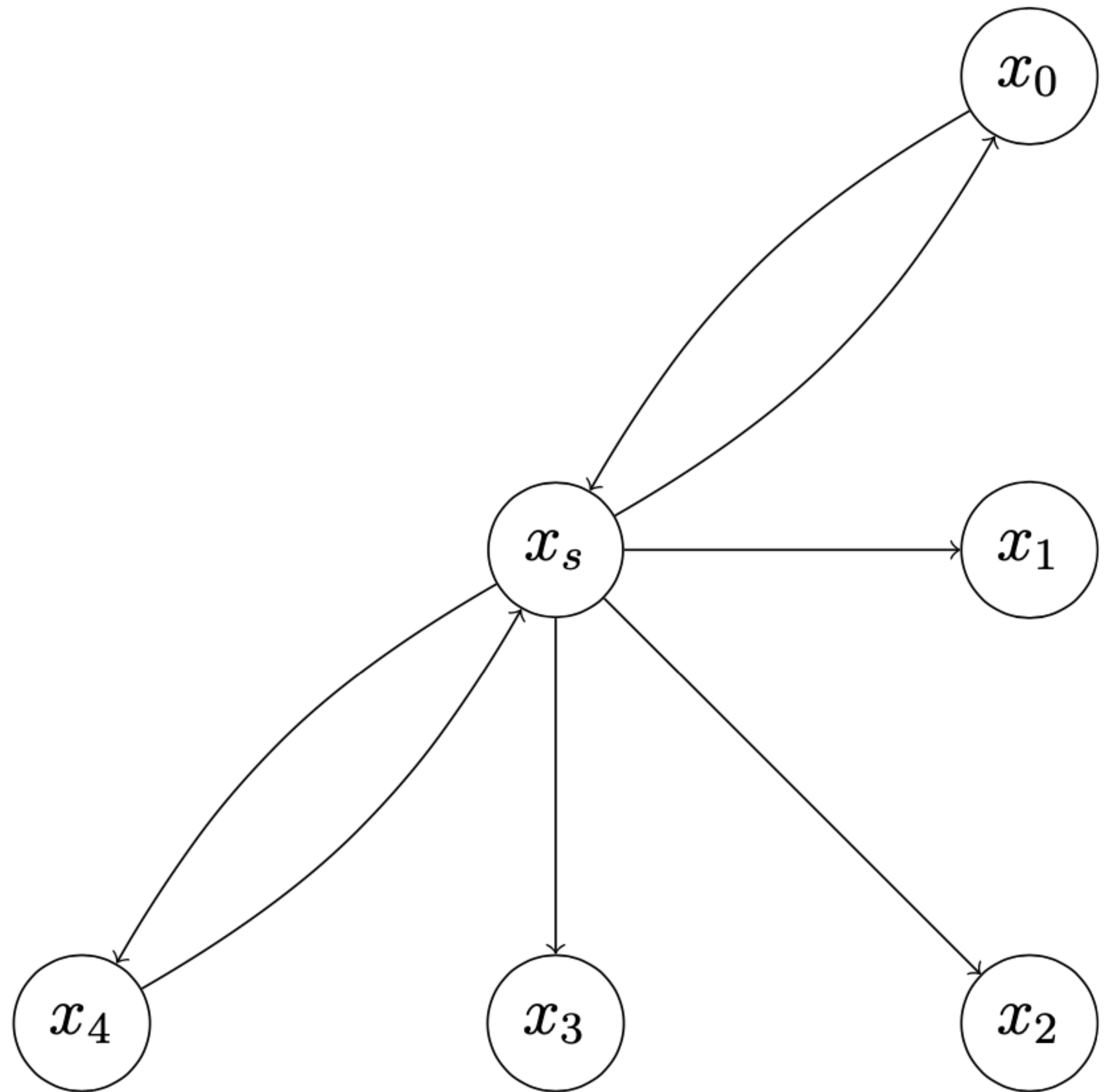


# Now

- No dual pheromone
- We dynamic food and obstacles
- We wrote the program in JavaScript / p5js

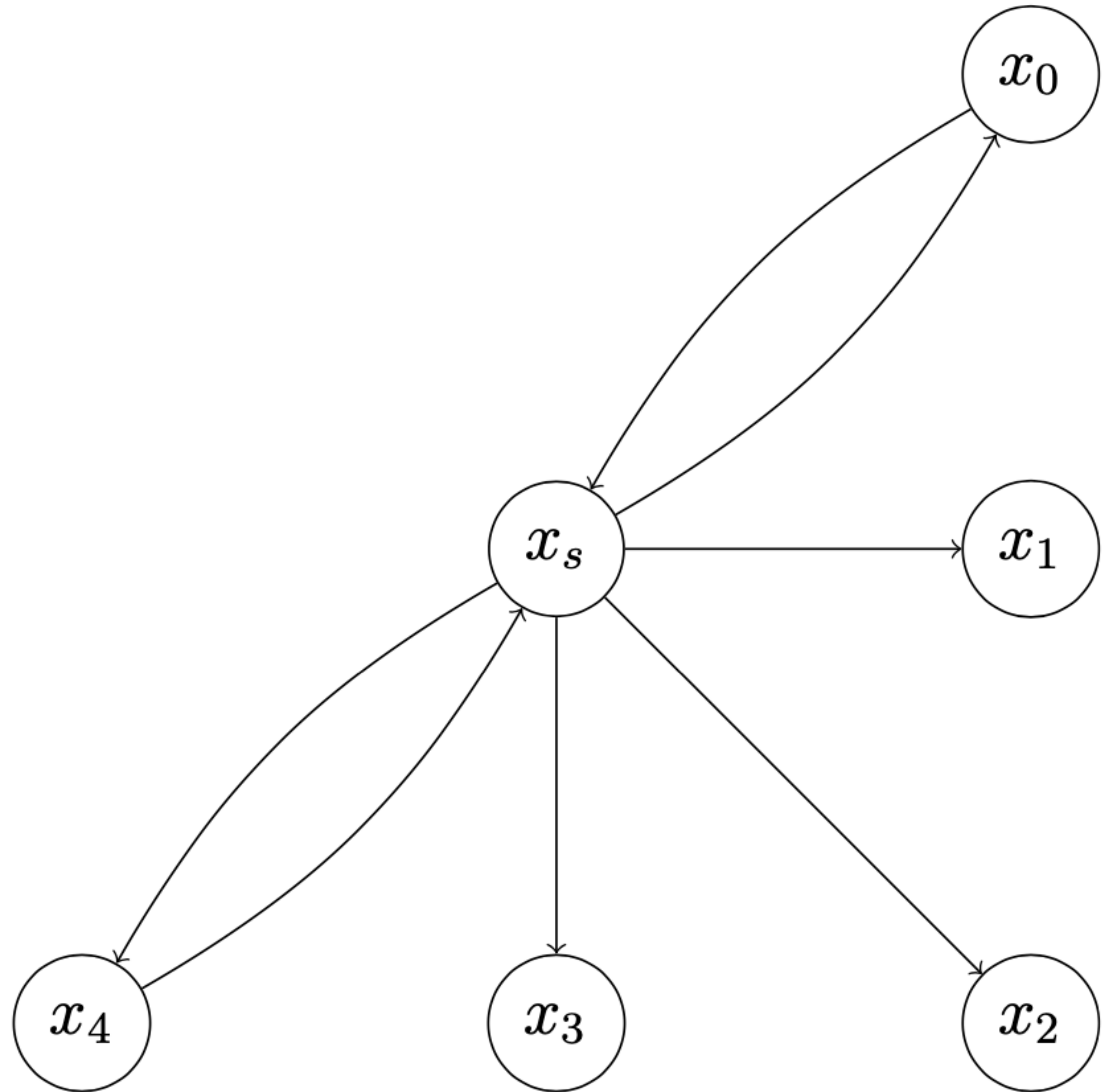


# DAG



# DAG

- The directed acyclic graph
- Prevent infinity cycles
- Explore the entire world

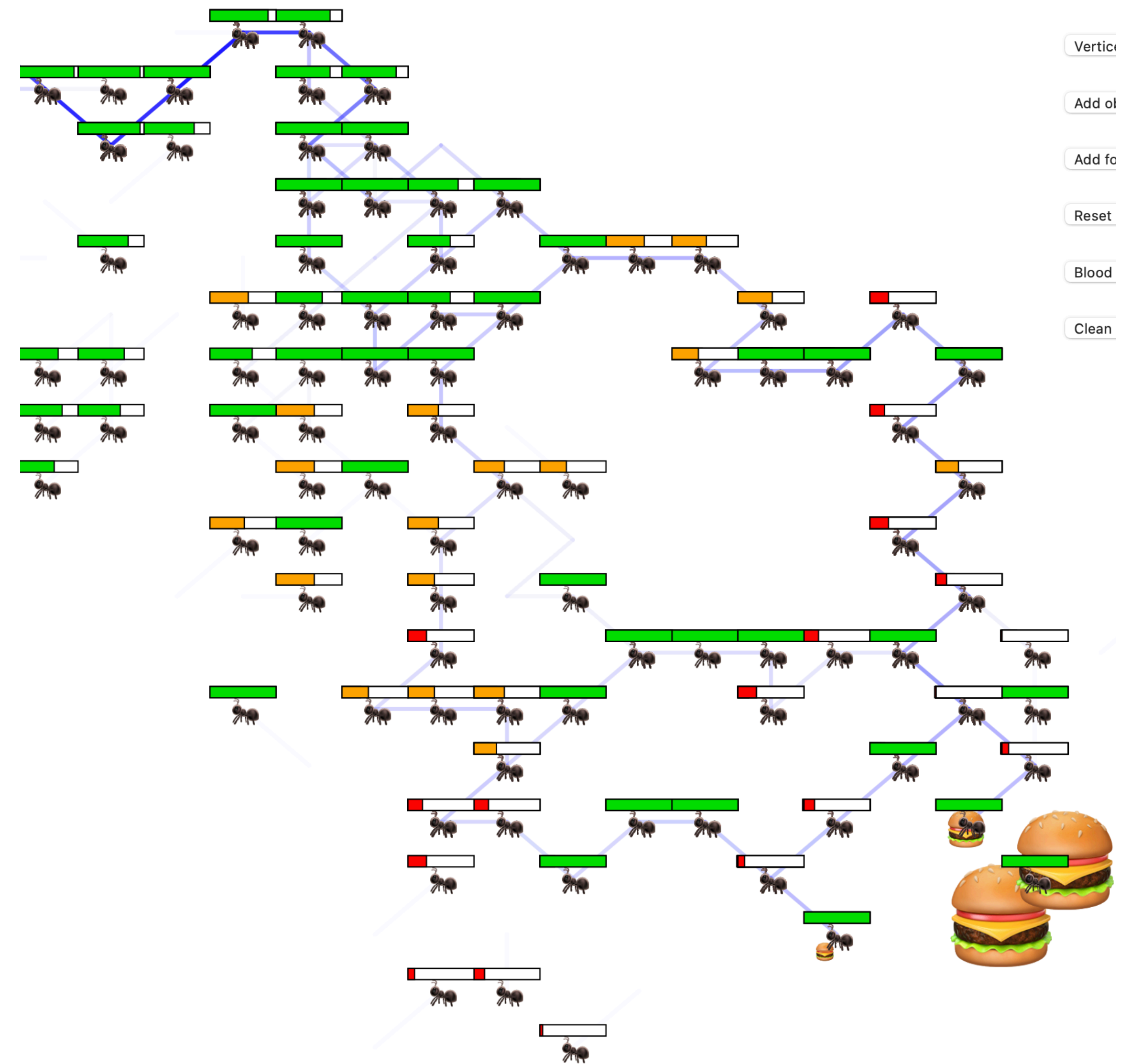


# Pheromone Affinity

- From vertex  $i$  to vertex  $j$
- Probability  $p_{ij}$  of visiting  $j$  is given by
$$p_{ij} = \begin{cases} p_{pher} \frac{\tau_{ij}}{\sum_{k \in \mathbf{Adj}(i)} \tau_{ik}} + (1 - p_{pher}) \frac{1}{\|\mathbf{Adj}(i)\|} & \text{if } j \in \mathbf{Adj}(i), \\ 0 & \text{otherwise,} \end{cases}$$
- Where  $\tau_{ij}$  is the pheromone on the edge and  $p_{pher}$  is the probability of affected by pheromone

# Food

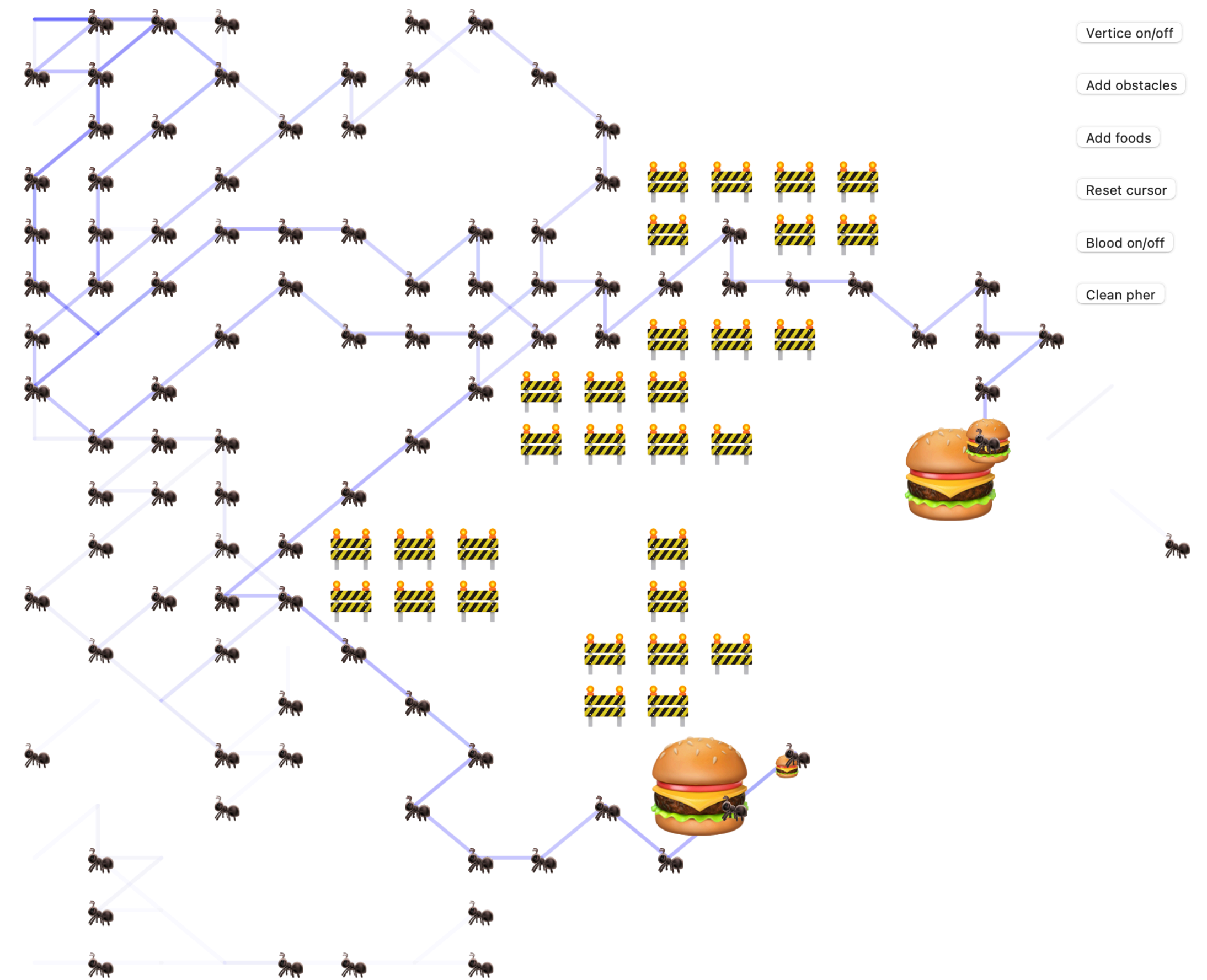
- Can be placed dynamically
- Refill the blood
- Blood can be regarded as the longest path allowed
- If an ant died, a new ant is summoned immediately





# Obstacles

- Can be placed dynamically on vertices.
- Sadly the ants can cross the obstacles when they head back
- Otherwise they don't know how to get home
- In the traditional ACO, the obstacles are fixed

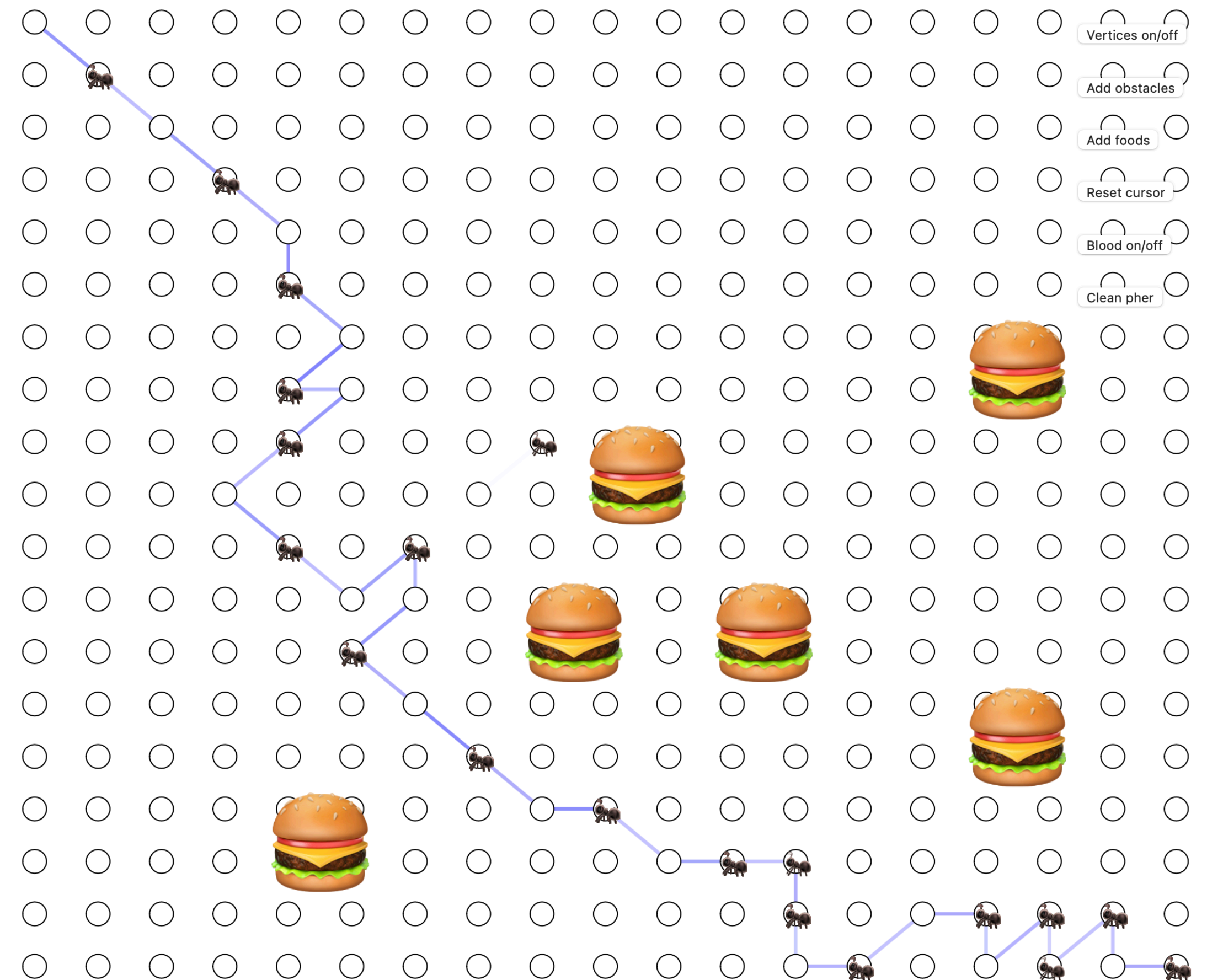


# When does an ants stop exploring

- No reachable vertex anymore
  - World boundary
  - Obstacles
- Died
  - blood  $\rightarrow$  zero
- Found food

# Local Optima

- The world is limited
- Stuck on nest-to-boundary path
- Produce pheromone as soon as they leave nest
- Most likely to happen when the food near the boundary was eaten





# Let's Play



<https://bit.ly/3oaIYAo>