Improving the Efficiency and Scalability of Multi-Drone Coverage Systems

A Decentralized Approach

Adit Shah

The Athenian School

Bryce Ferguson

Department of Electrical and Computer Engineering

University of California, Santa Barbara

RMP Research Symposium

August 1, 2019

UC SANTA BARBARA
Research Mentorship Program
Summer Sessions

THE ATHENIAN SCHOOL

Multi-Drone Coverage Systems

Multi-Drone Coverage Systems

Limited Resources

- Not enough drones to cover entire area
- Some areas are more important than others

Multi-Drone Coverage Systems

Limited Resources

- Not enough drones to cover entire area
- Some areas are more important than others

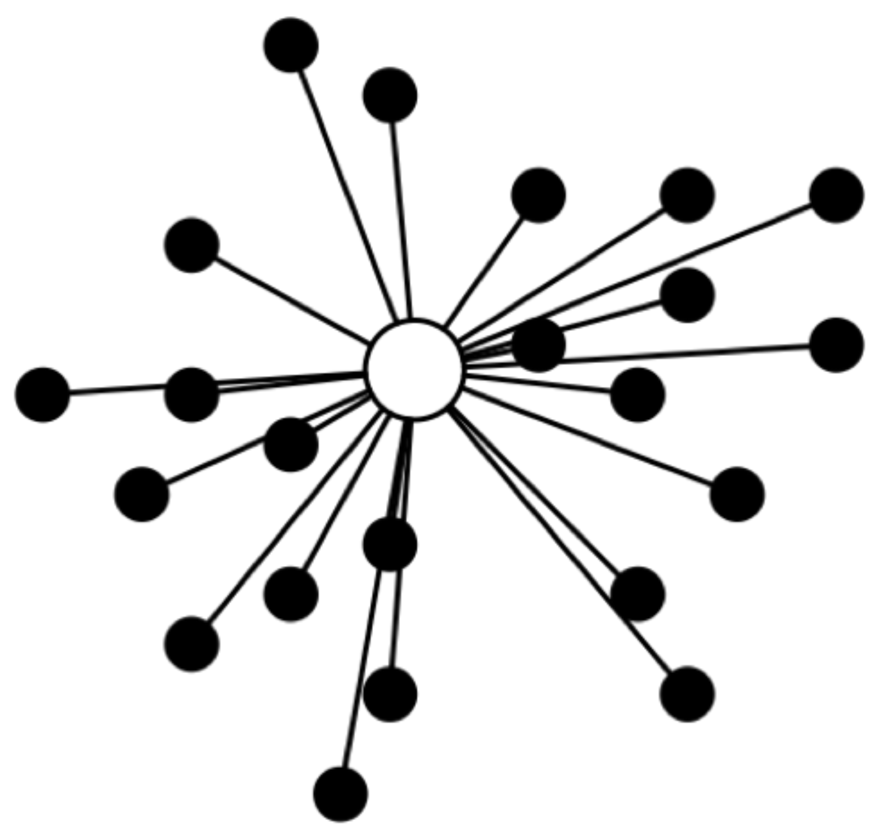
Goal: Distribute drones effectively based on the utility of each location

Centralized vs Decentralized Control

Centralized

One node decides optimal location for each drone, controls navigation

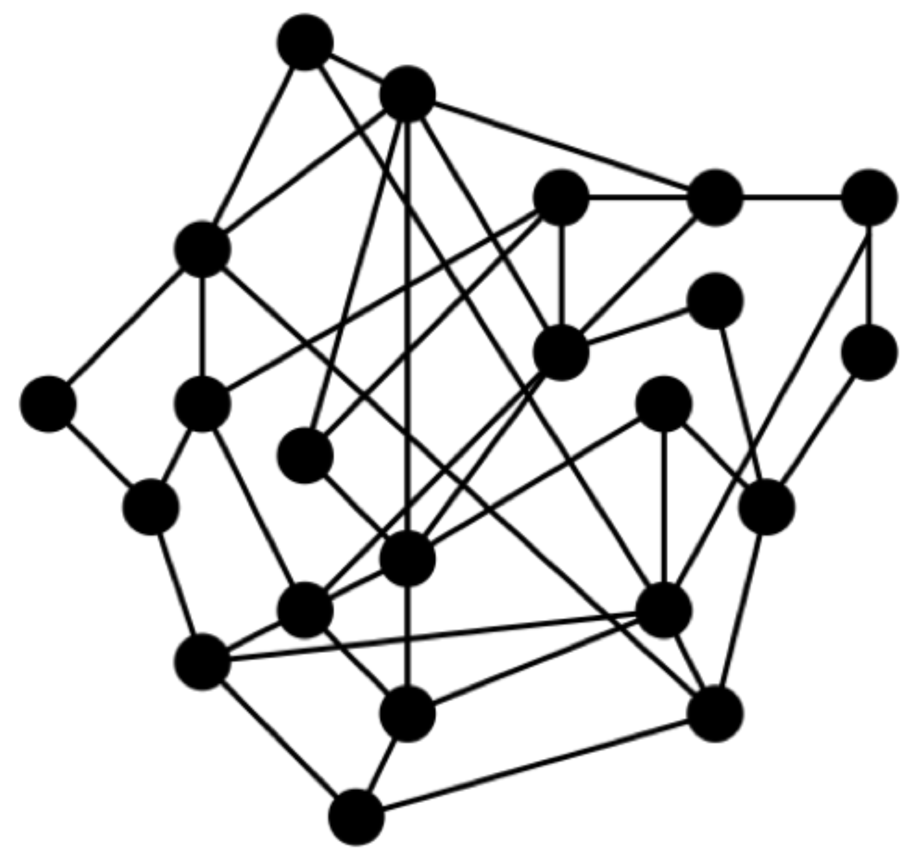
- More easily regulated
- Slower runtimes



Decentralized

Each drone decides its own optimal location and navigation strategy

- More efficient and scalable
- Doesn't always find global optima



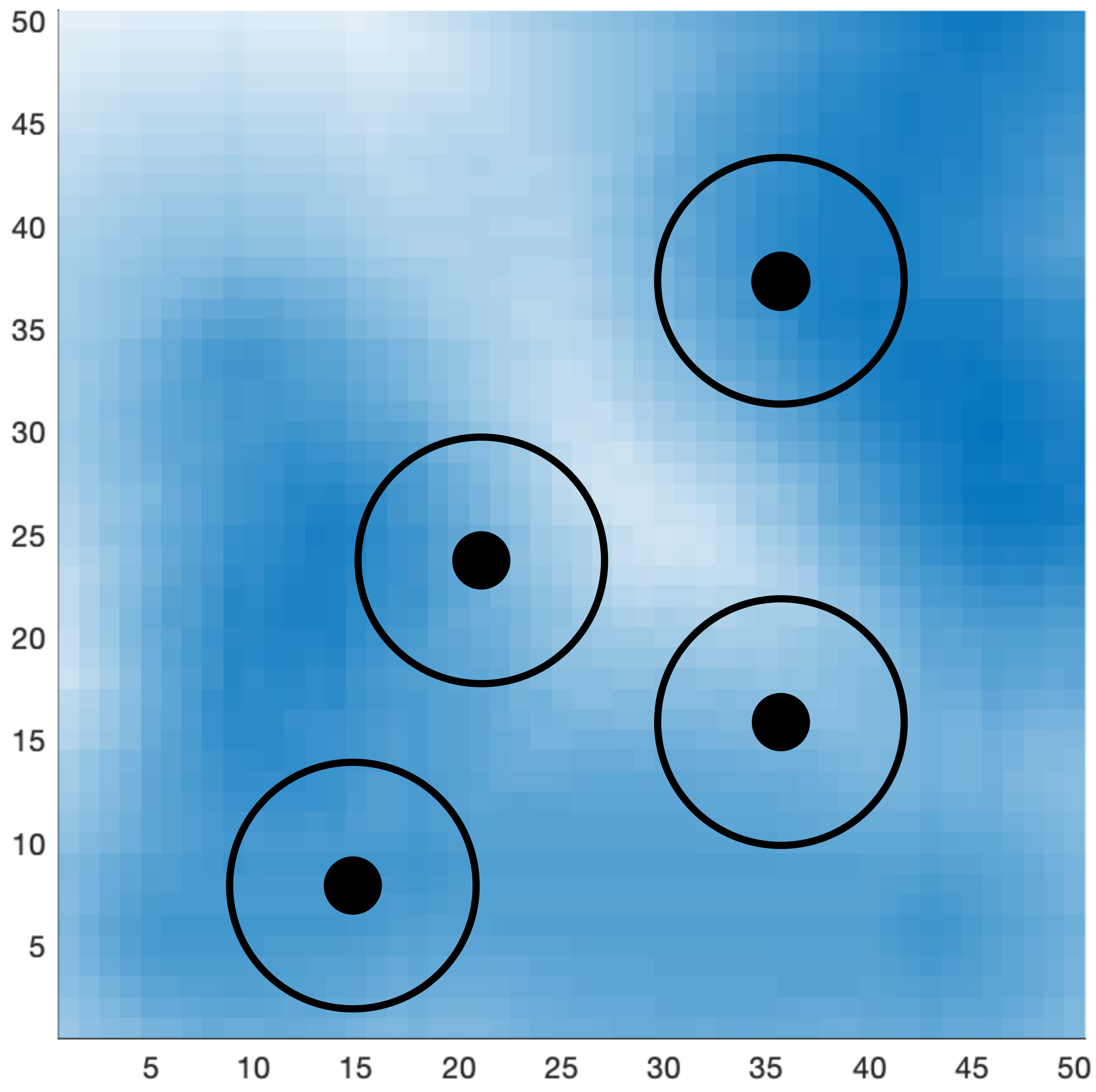
Enhancing the Decentralized Algorithm

Develop multi-drone simulation

Implement and compare algorithms

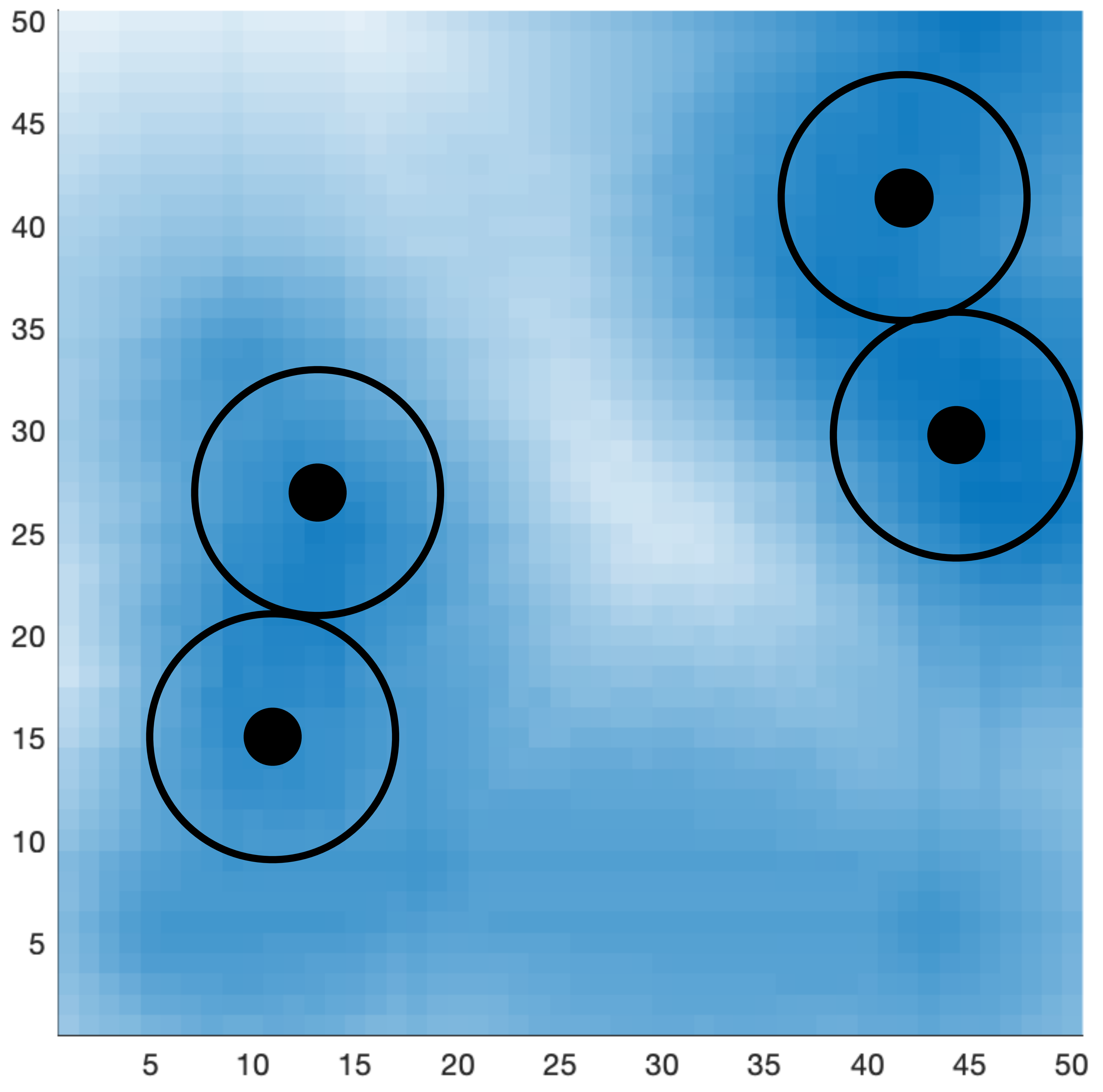
Develop and verify enhancements

Modeling the System



- Map of a region to be covered, where each cell has a utility (importance) value
- Each agent starts at either a random or pre-determined position
- The circle around each agent represents the sensing or coverage radius
- An agent is chosen at random to make a move within its movement radius
- Ultimate goal: maximize the sum of values of the covered cells

Modeling the System



- Map of a region to be covered, where each cell has a utility (importance) value
- Each agent starts at either a random or pre-determined position
- The circle around each agent represents the sensing or coverage radius
- An agent is chosen at random to make a move within its movement radius
- Ultimate goal: maximize the sum of values of the covered cells

Decentralized Algorithms

Greedy Algorithm
(Baseline approach)

Drones try solely to maximize their own coverage

Log-Linear Learning (LLL)

Introduces randomness

Drones occasionally take a suboptimal decision

LLL with Automatic Tau Generation

Eliminates need for manual tuning of constants

Decentralized Algorithms

Greedy Algorithm
(Baseline approach)

Drones try solely to maximize their own coverage

- Agents' only goal is to maximize their utility
- Moves are only made if the new location has a higher utility value:

$$a_i(t+1) = \arg \max_{a_i \in A_i(t)} U_i(a_i, a_{-i})$$

- Can get stuck at local optima

Log-Linear Learning (LLL)

Introduces randomness

Drones occasionally take a suboptimal decision

LLL with Automatic Tau Generation

Eliminates need for manual tuning of constants

Decentralized Algorithms

Log-Linear Learning (LLL)

Introduces randomness

Drones occasionally take a suboptimal decision

- Actions chosen based on a probability distribution based on utilities of each action:

$$p_i^{a_i}(t) = \frac{e^{\frac{1}{\tau}U_i(a_i, a_{-i}(t-1))}}{\sum_{\bar{a}_i \in A_i} e^{\frac{1}{\tau}U_i(\bar{a}_i, a_{-i}(t-1))}}$$

Greedy Algorithm
(Baseline approach)

Drones try solely to maximize their own coverage

LLL with Automatic Tau Generation

Eliminates need for manual tuning of constants

Decentralized Algorithms

Log-Linear Learning (LLL)

Introduces randomness

Drones occasionally take a suboptimal decision

- Value τ acts as a randomness or "exploration" constant
    - As $\tau \to 0$: Acts as a greedy system
    - As $\tau \to \infty$: Actions chosen uniformly at random
- At each time-step, τ decays by 0.997

Greedy Algorithm
(Baseline approach)

Drones try solely to maximize their own coverage

LLL with Automatic Tau Generation

Eliminates need for manual tuning of constants

Decentralized Algorithms

LLL with Automatic Tau Generation

Eliminates need for manual tuning of constants

- In LLL, starting τ selected manually – can be automated
- Correlates with average map value, agent radius, starting configuration
- For random starting position: $\tau = \frac{1}{2}\pi R^2 \cdot \overline{M}$
- For starting in one corner: $\tau = \pi R^2 \cdot \overline{M}$

Greedy Algorithm
(Baseline approach)

Drones try solely to maximize their own coverage

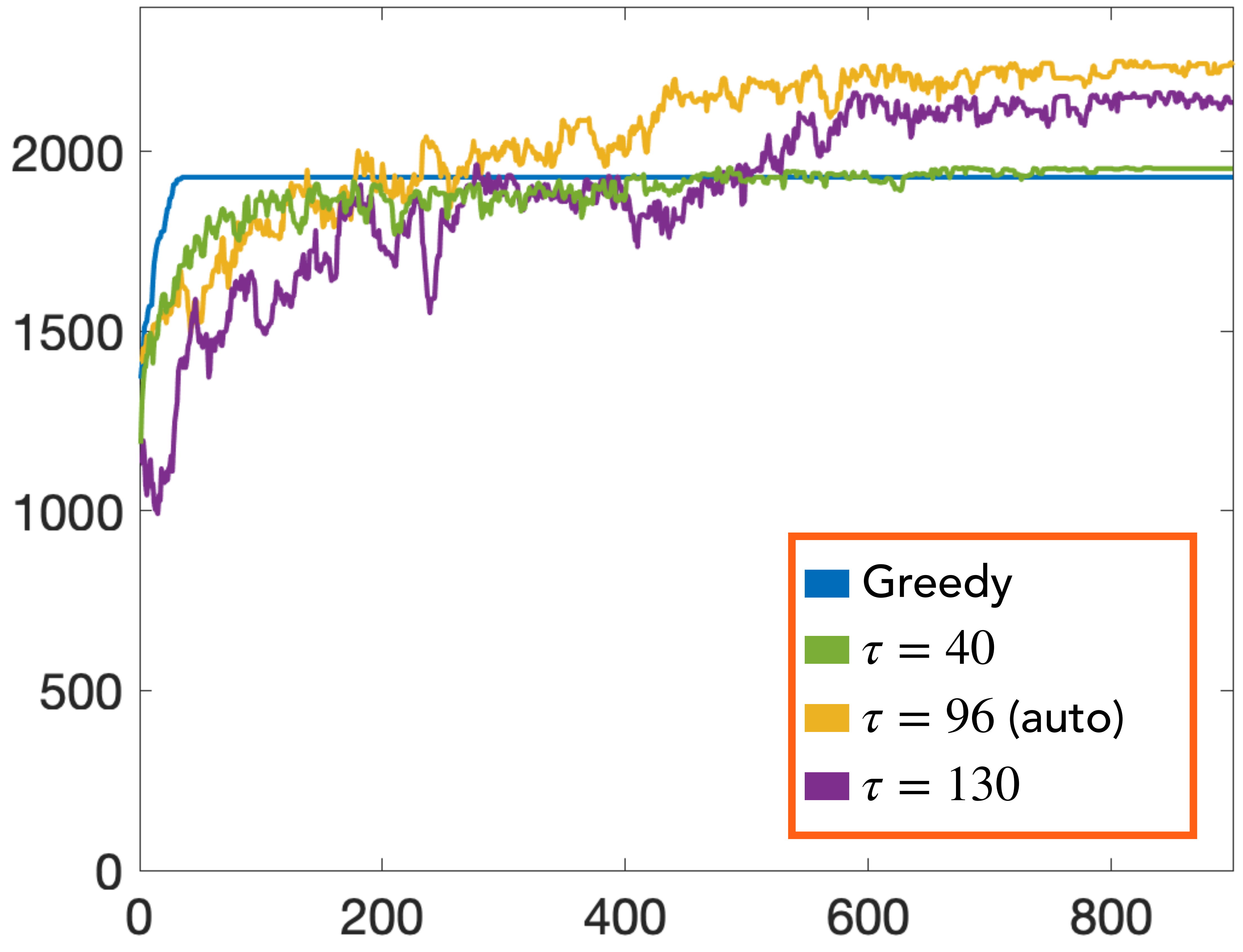
Log-Linear Learning (LLL)

Introduces randomness

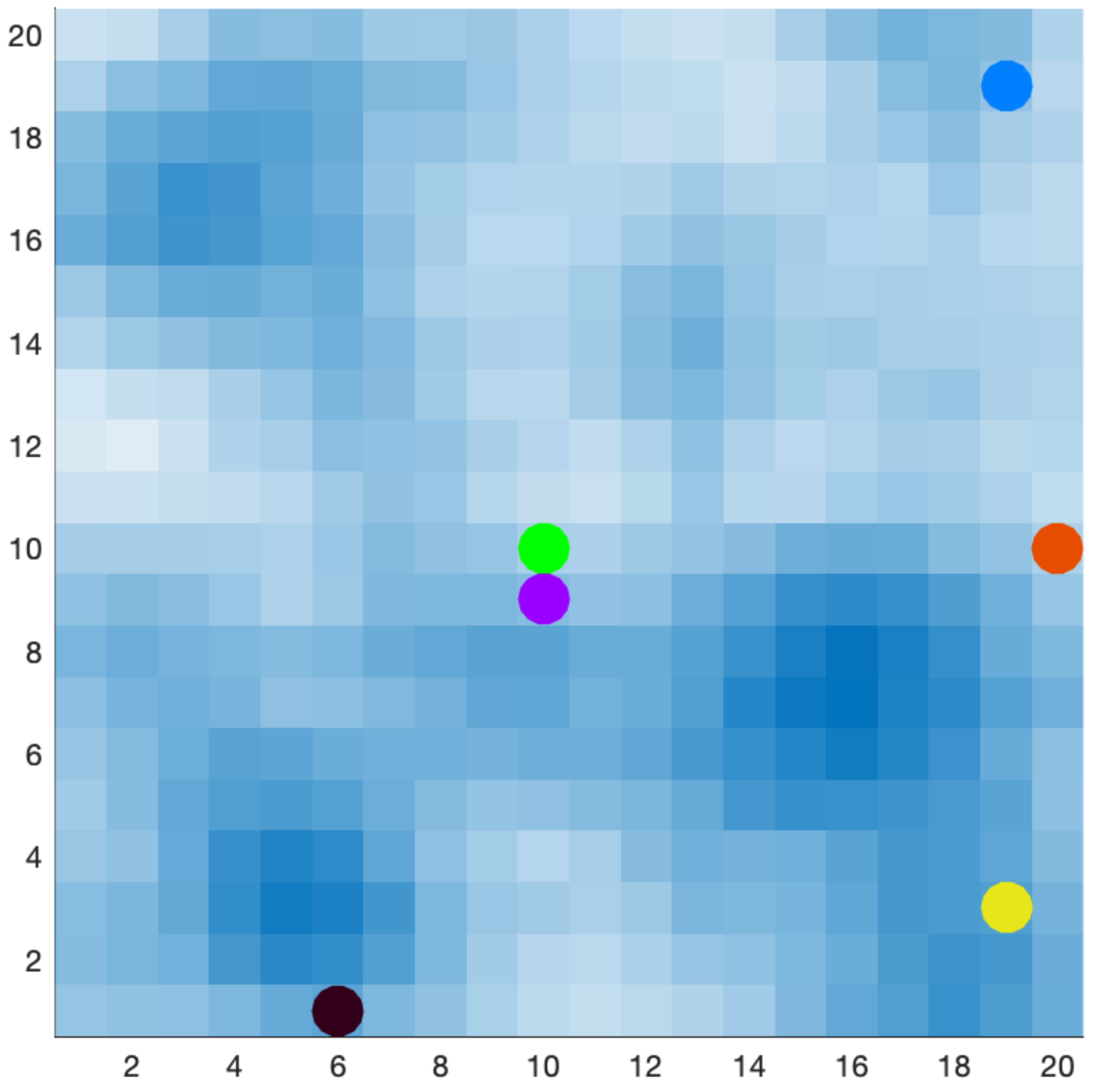
Drones occasionally take a suboptimal decision

Algorithm Comparison

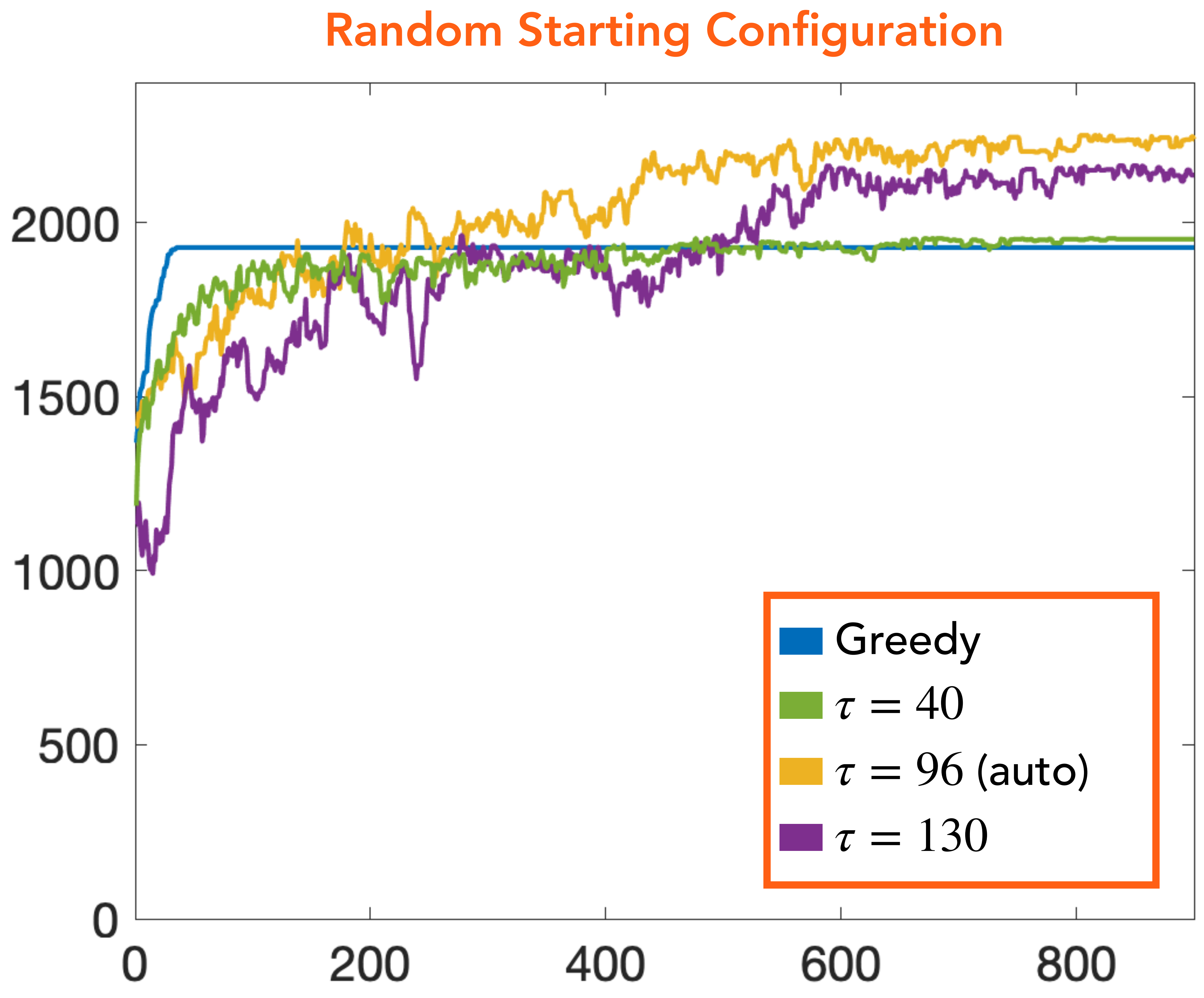
Random Starting Configuration

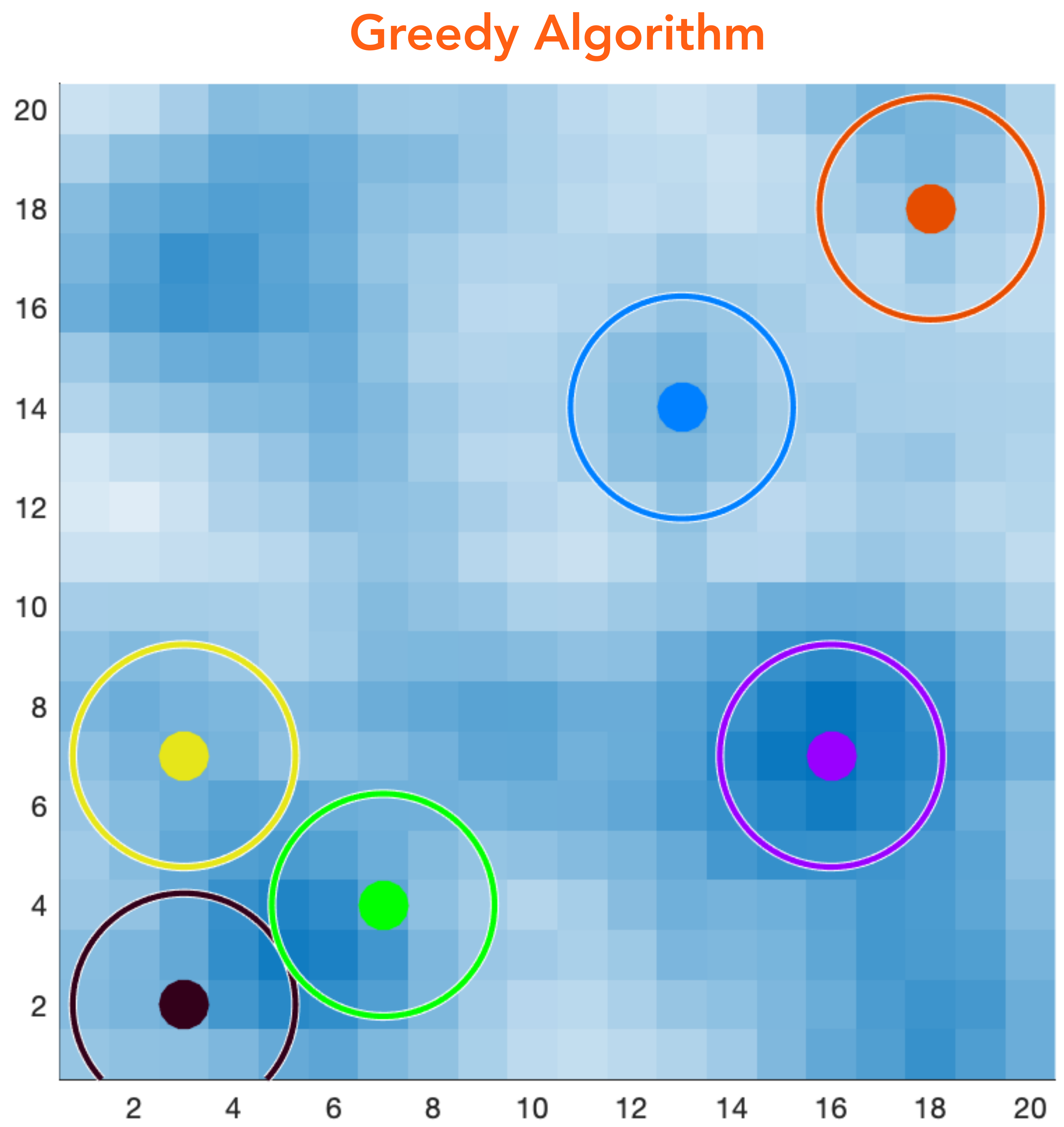


Initial State

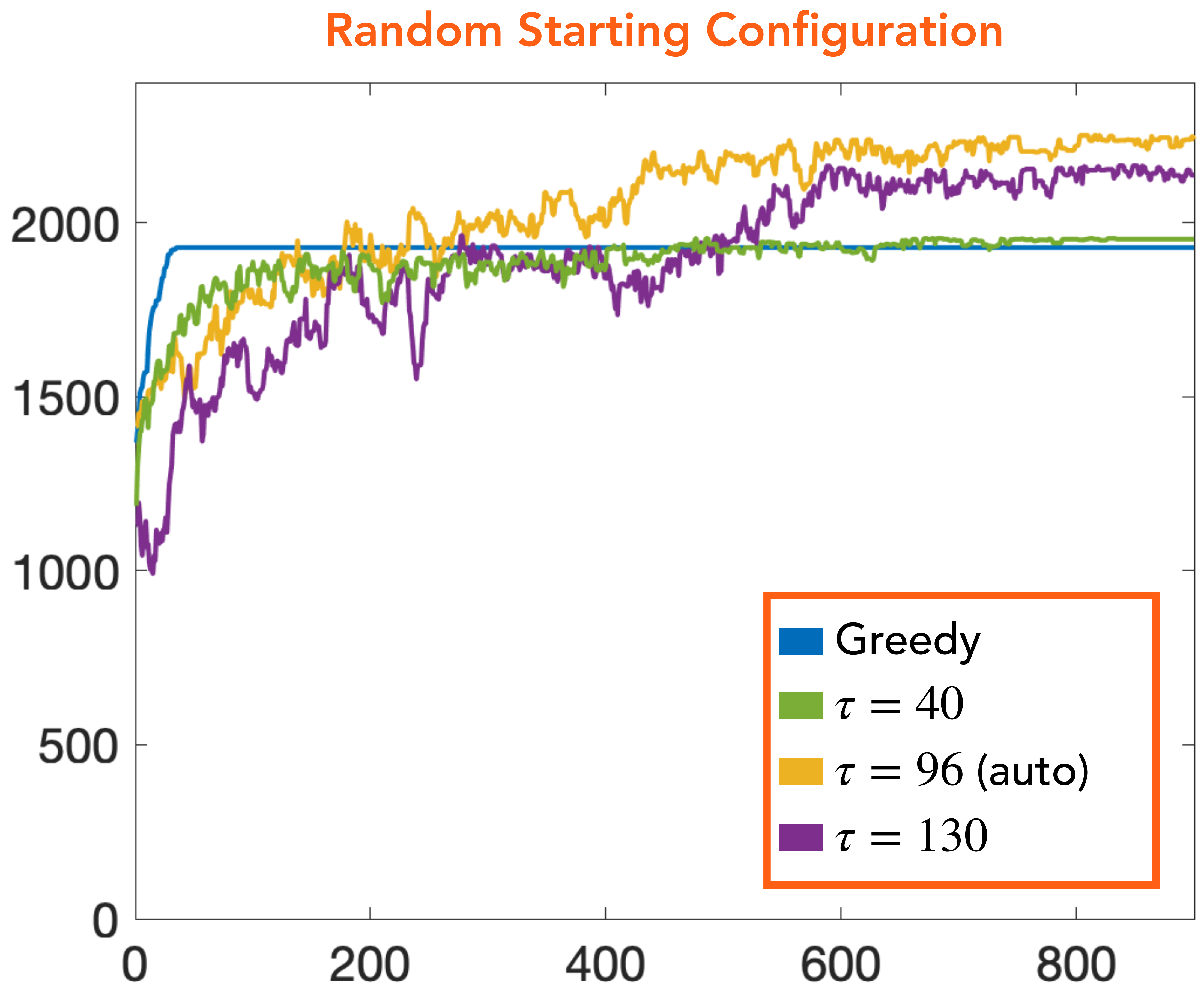


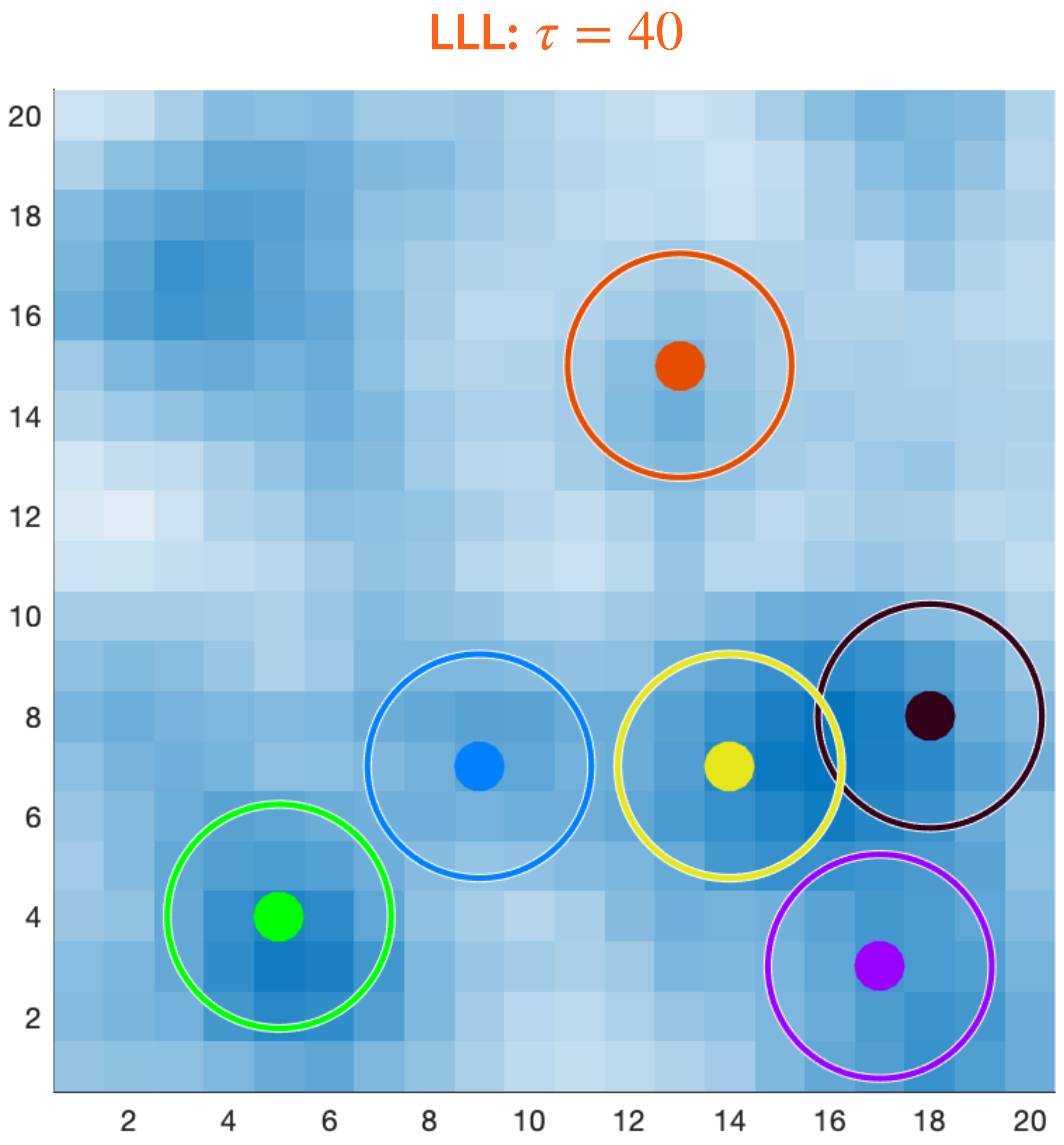
Algorithm Comparison



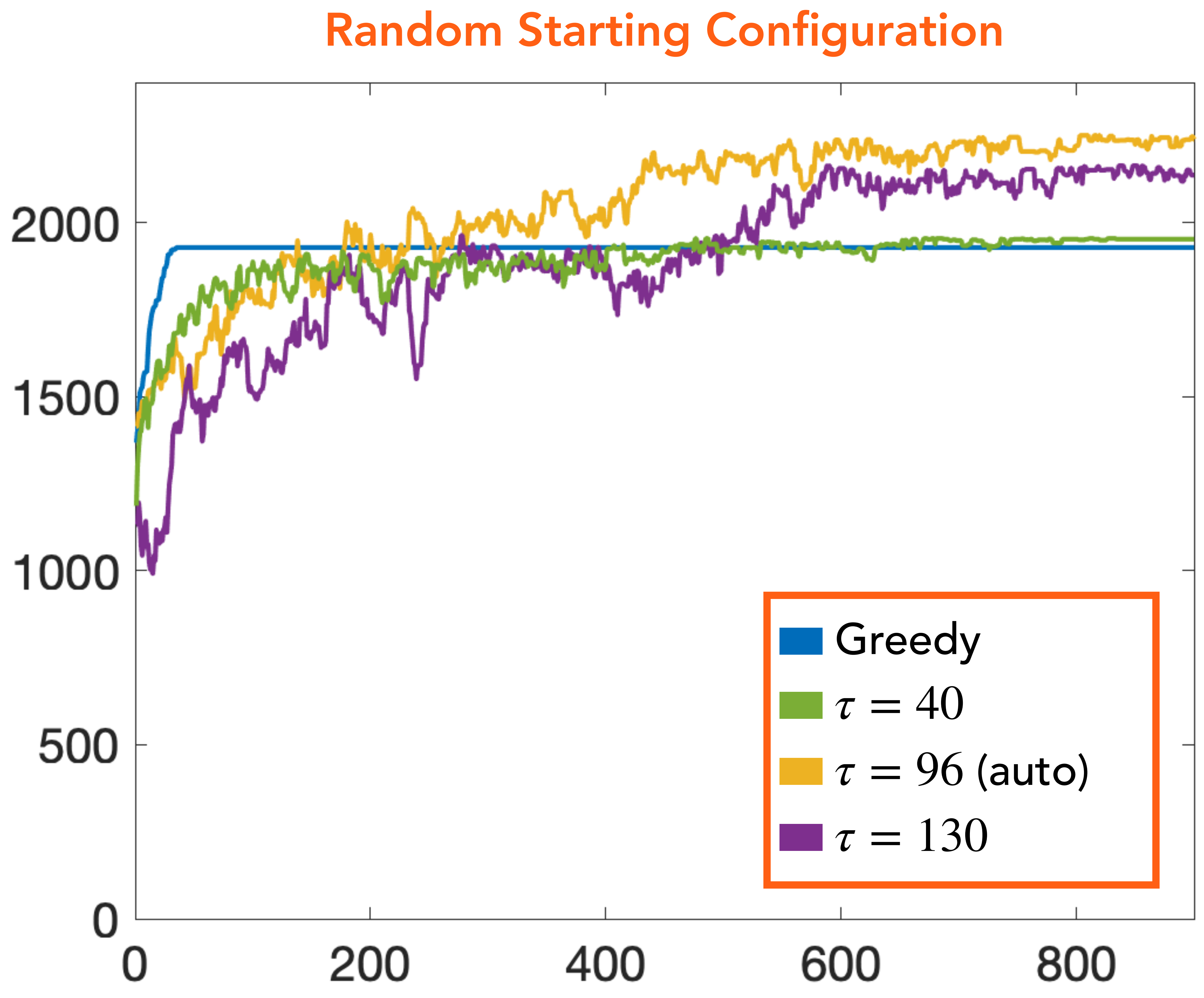


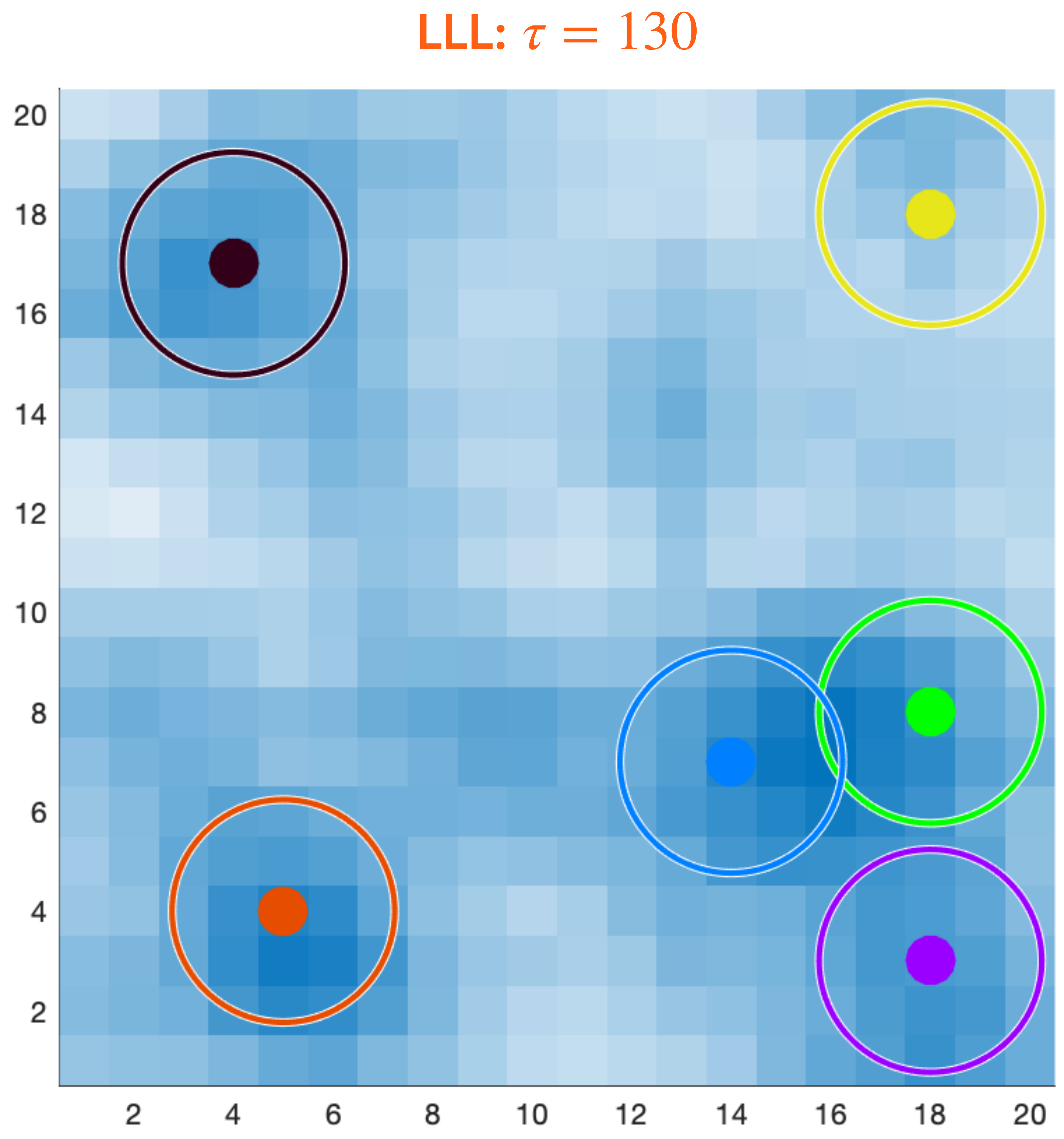
Algorithm Comparison





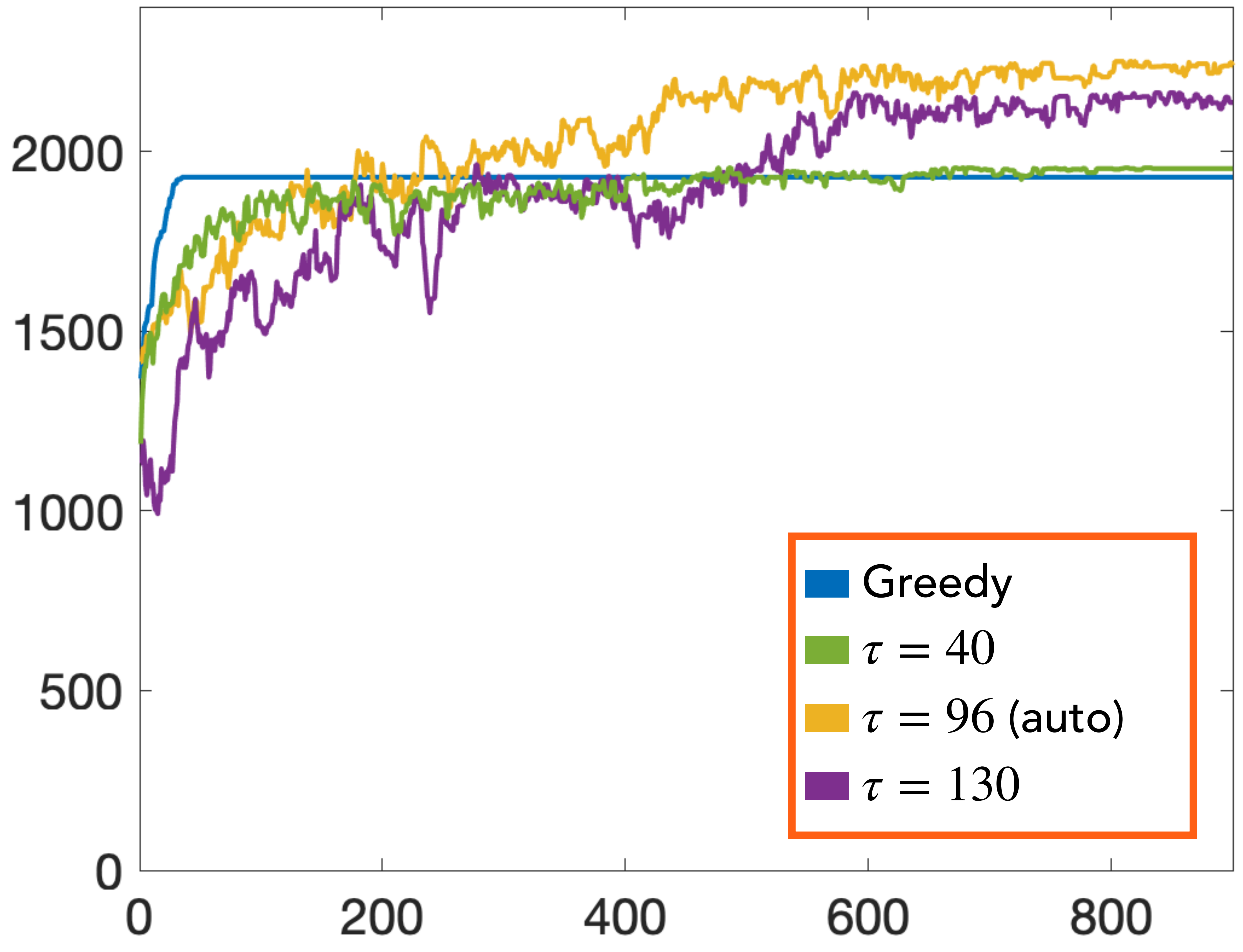
Algorithm Comparison



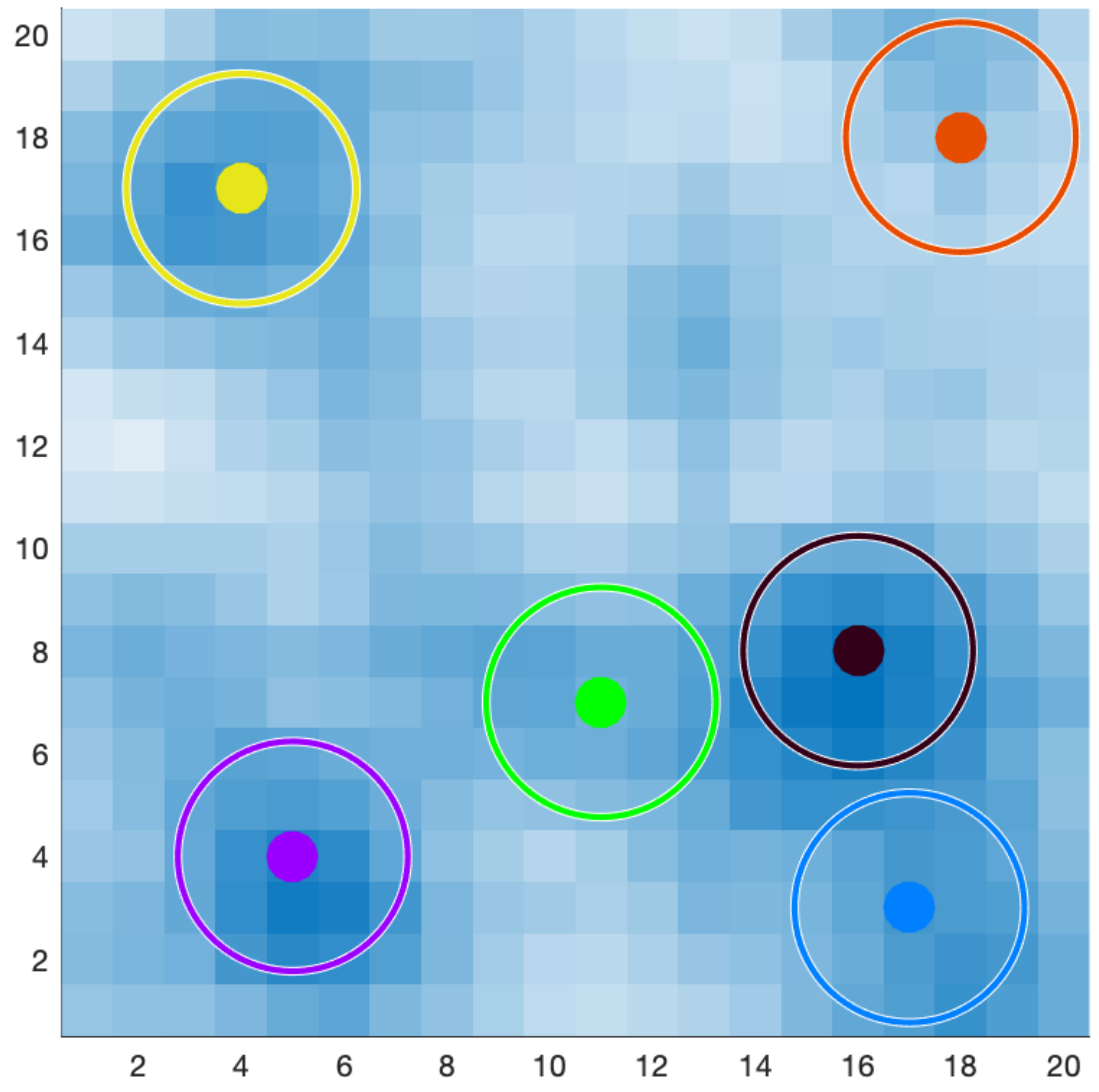


Algorithm Comparison

Random Starting Configuration

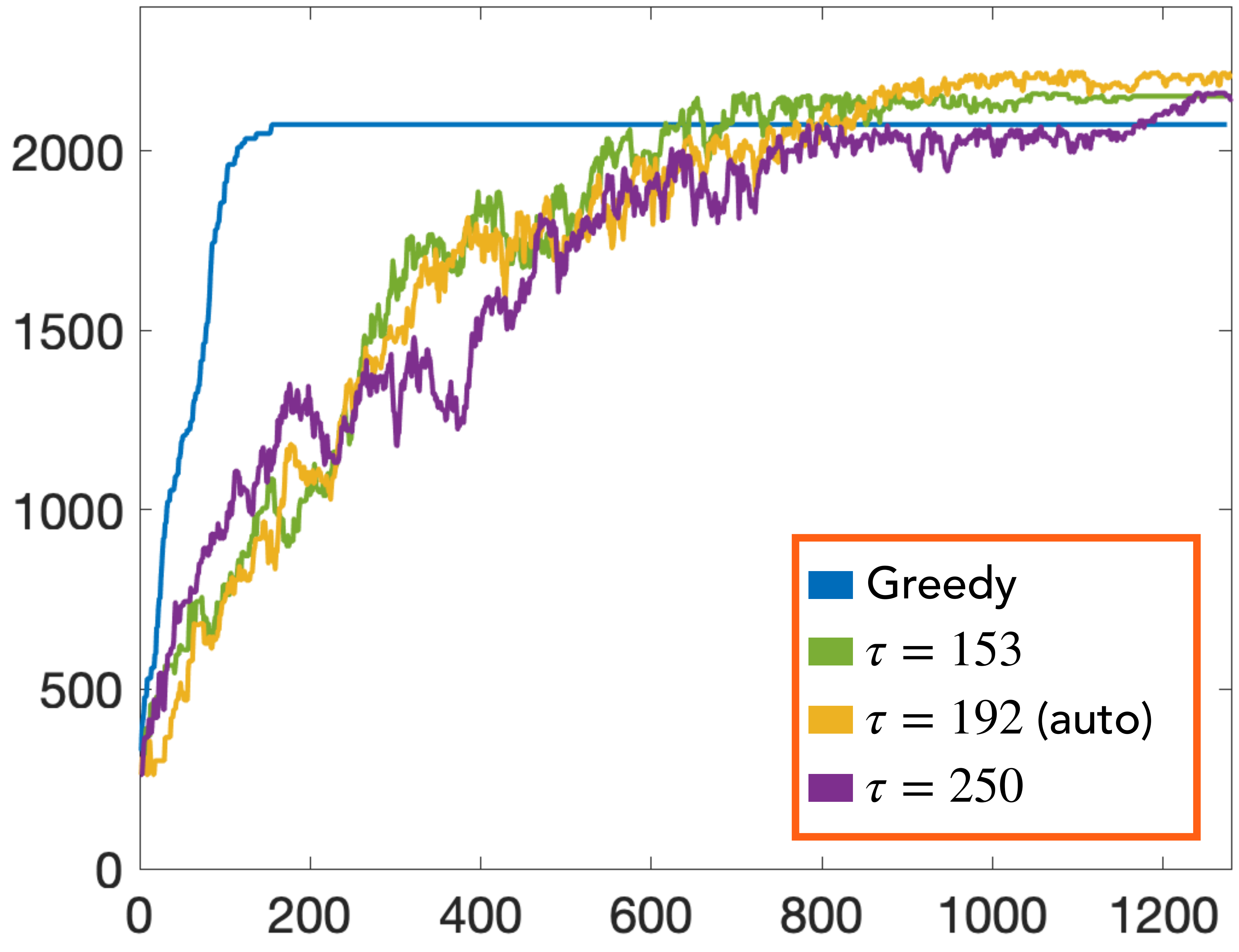


LLL: $\tau = 96$ (auto)

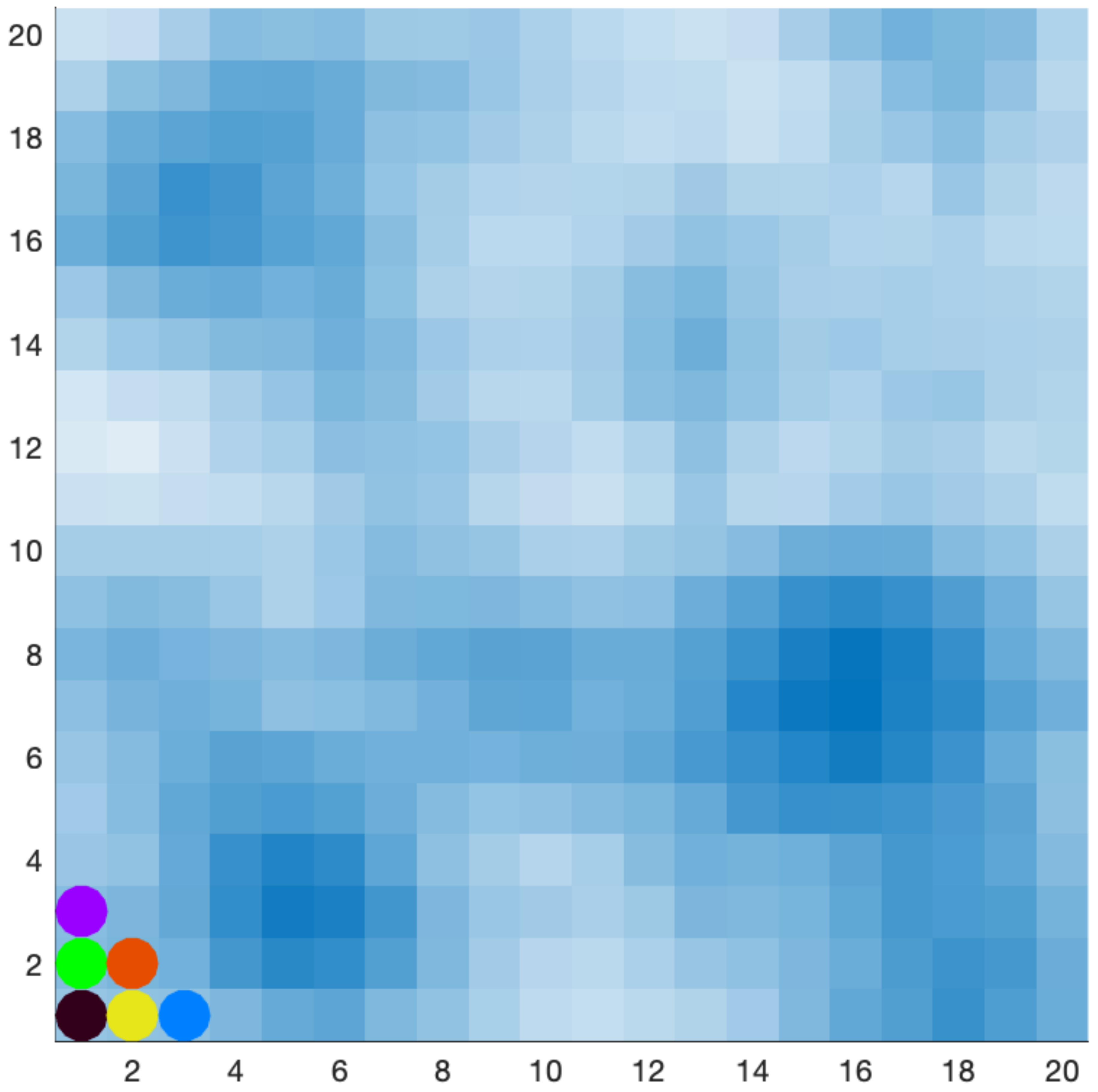


Algorithm Comparison

Corner Starting Configuration

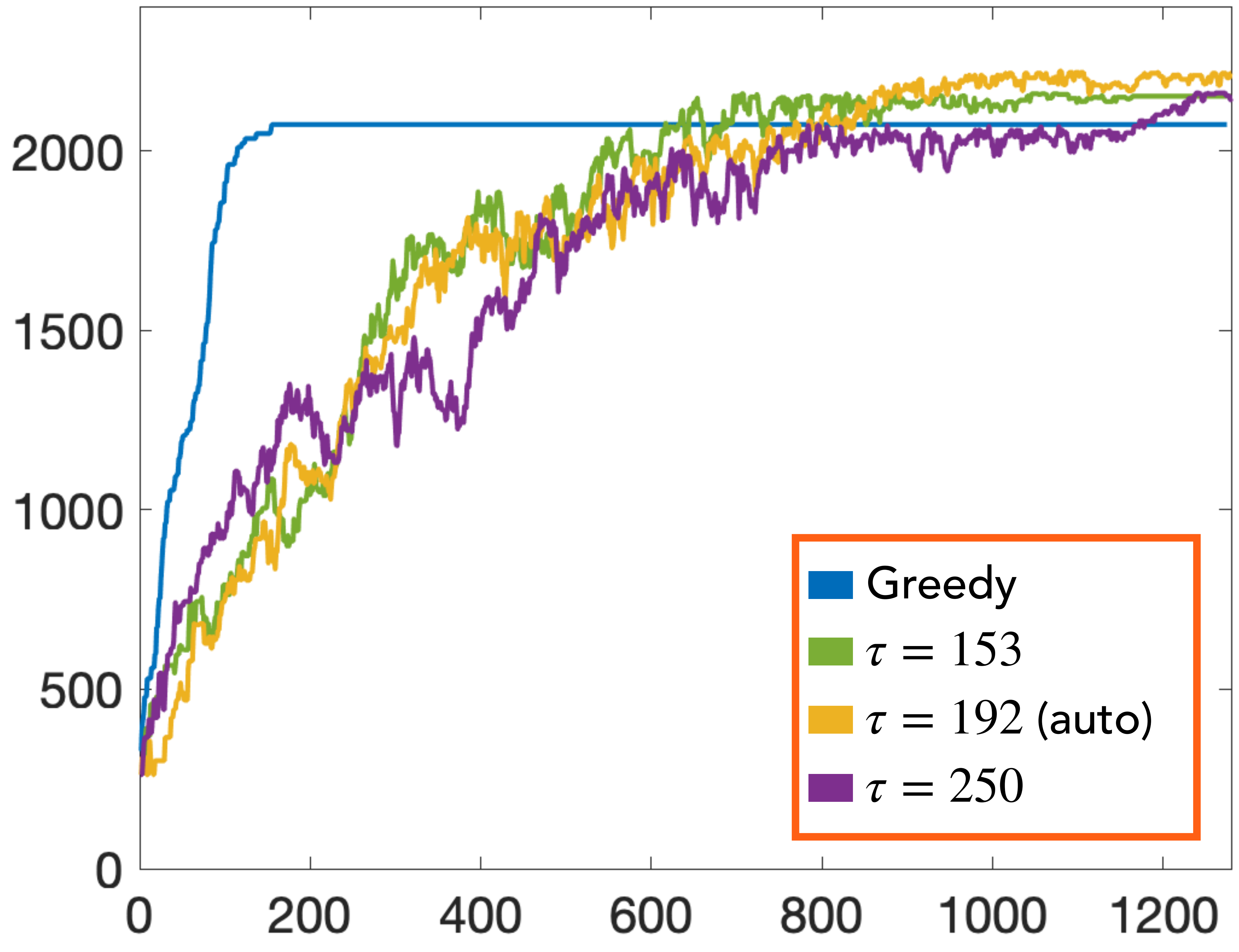


Initial State

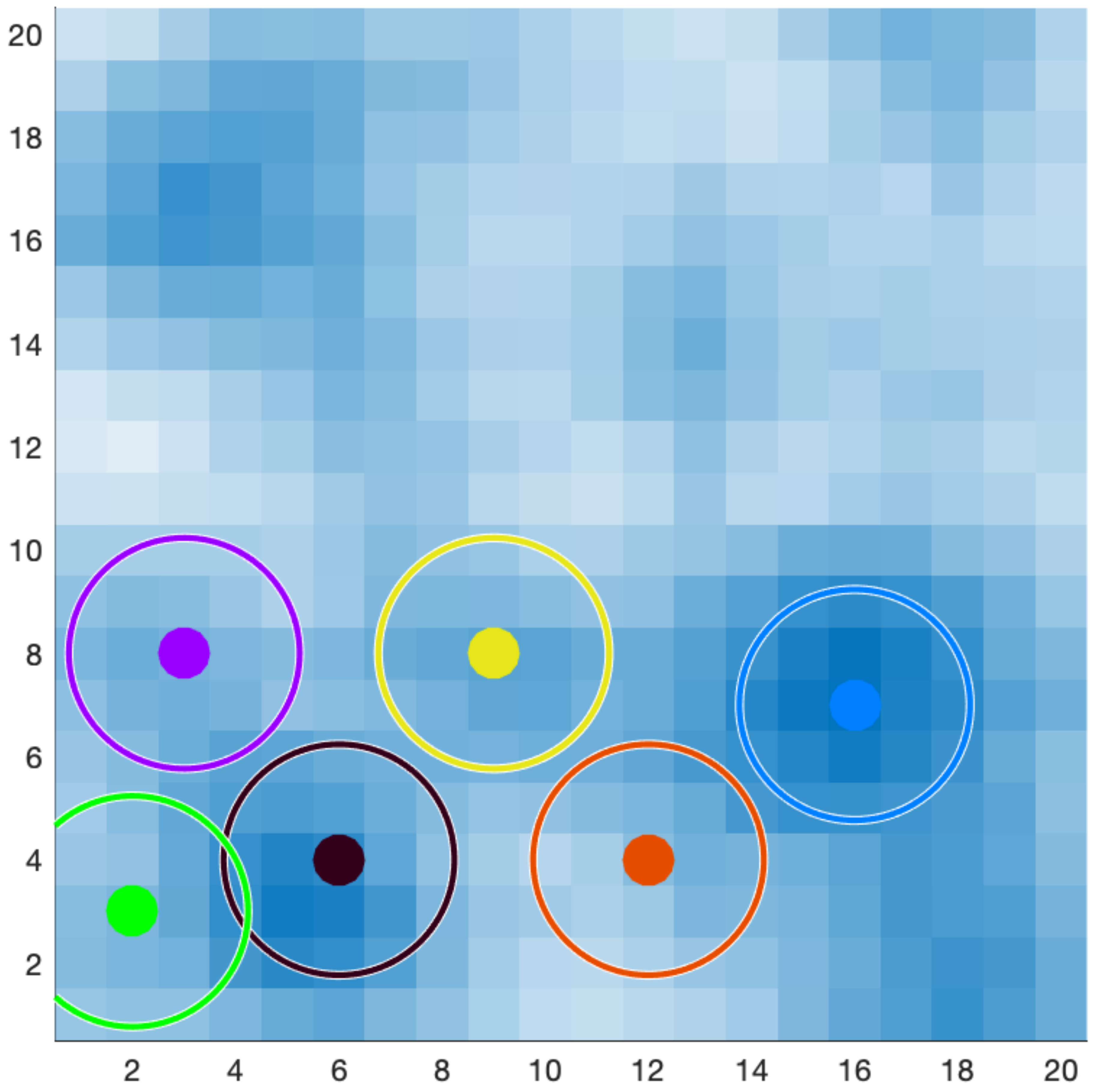


Algorithm Comparison

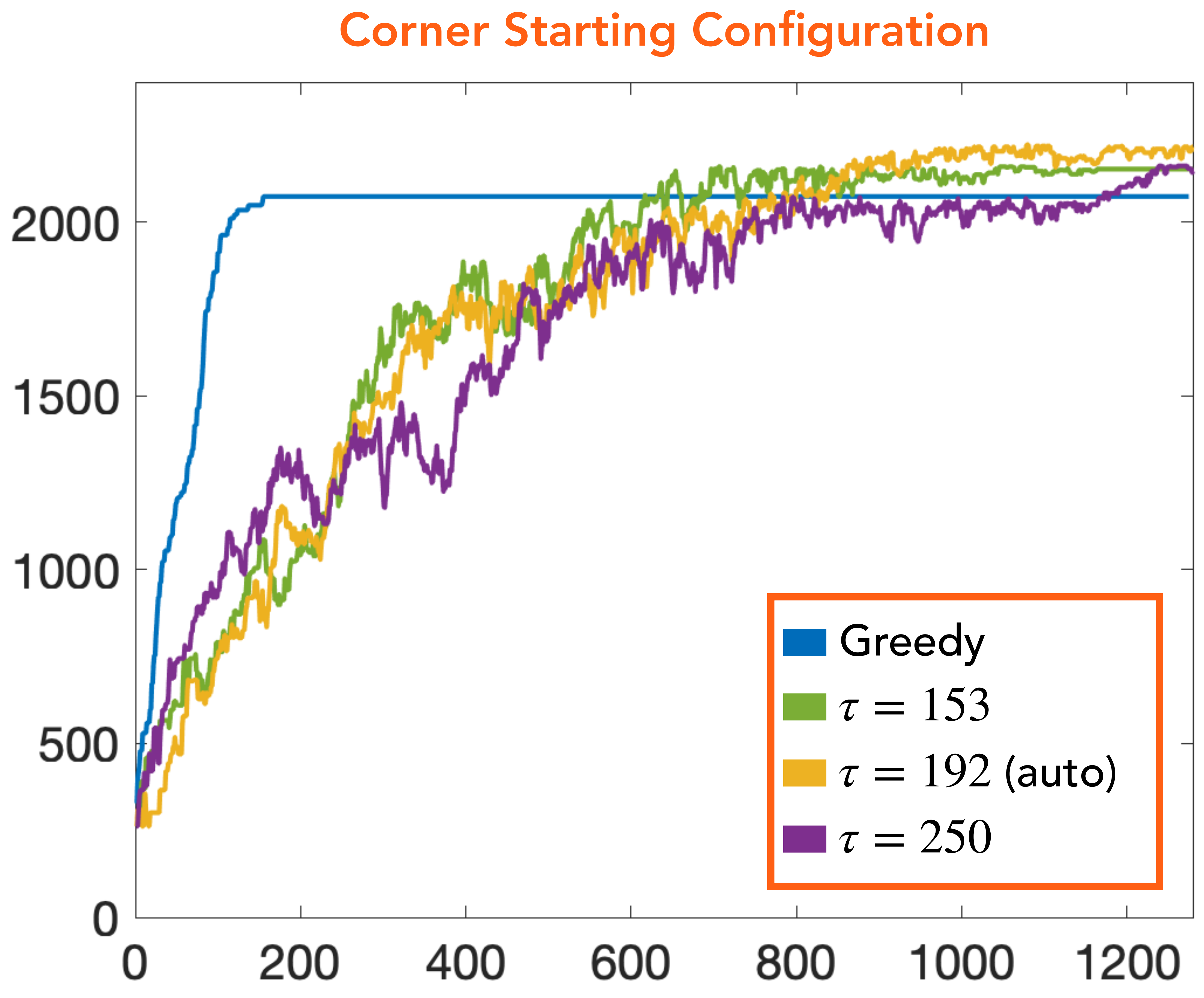
Corner Starting Configuration

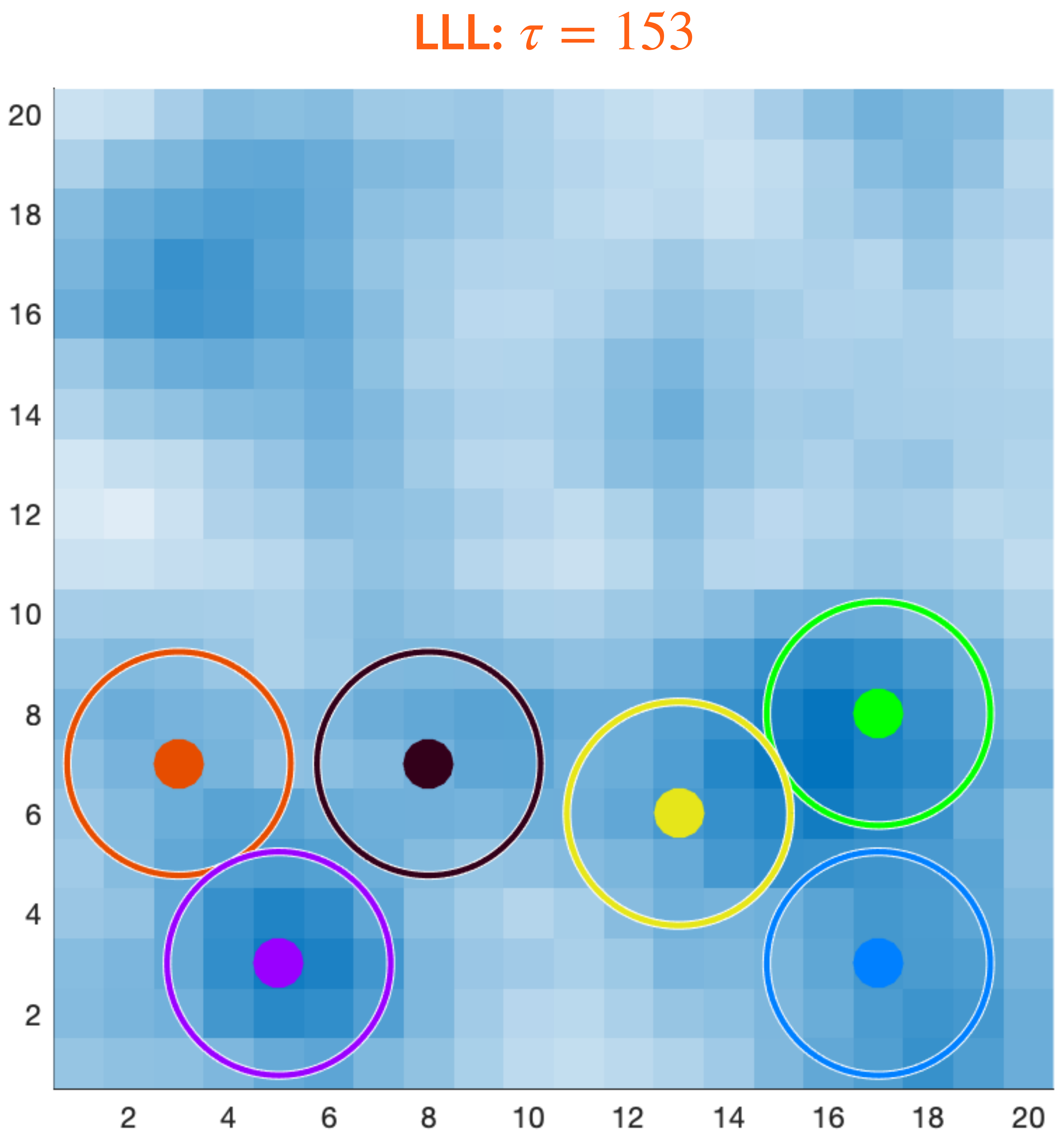


Greedy Algorithm

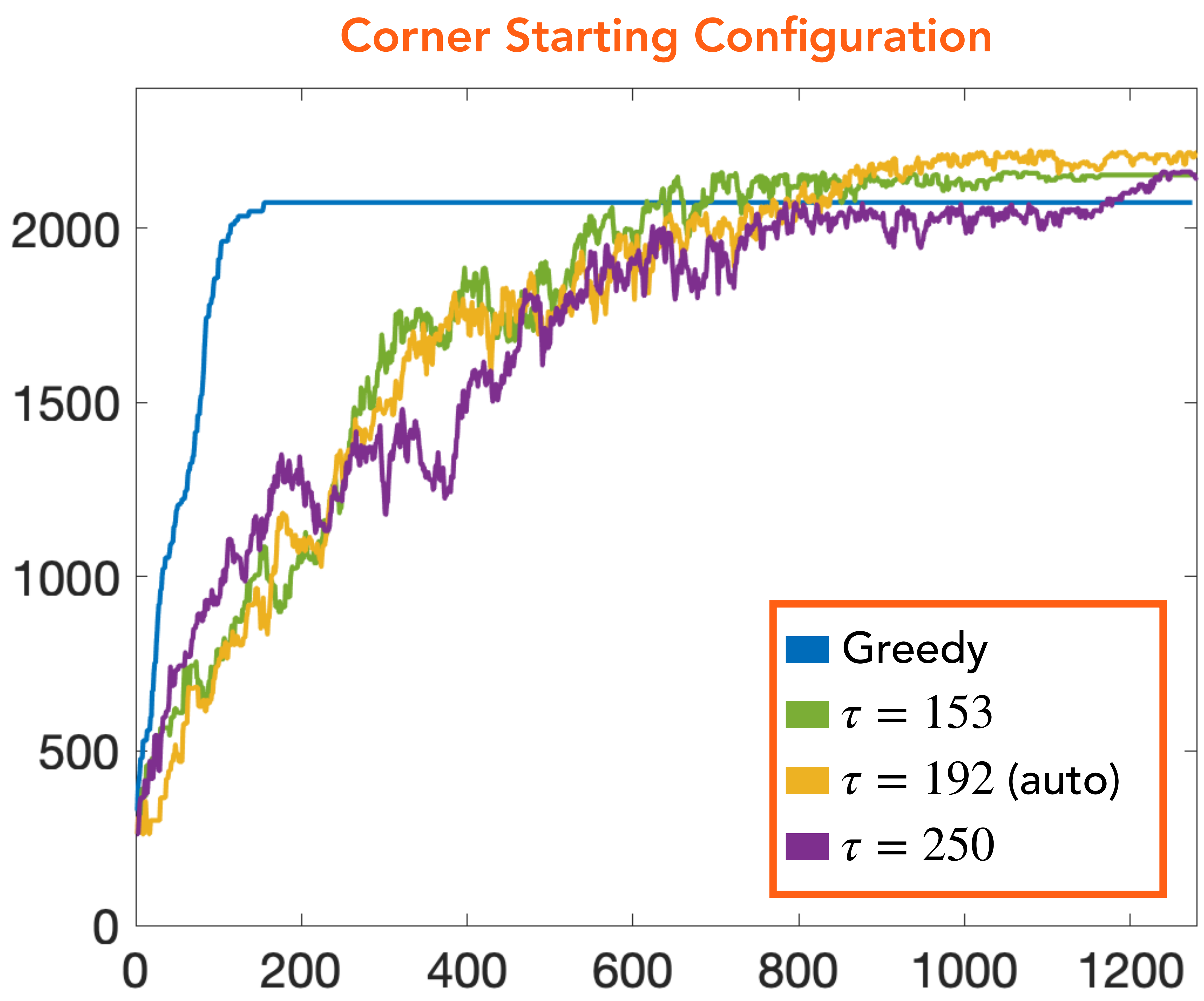


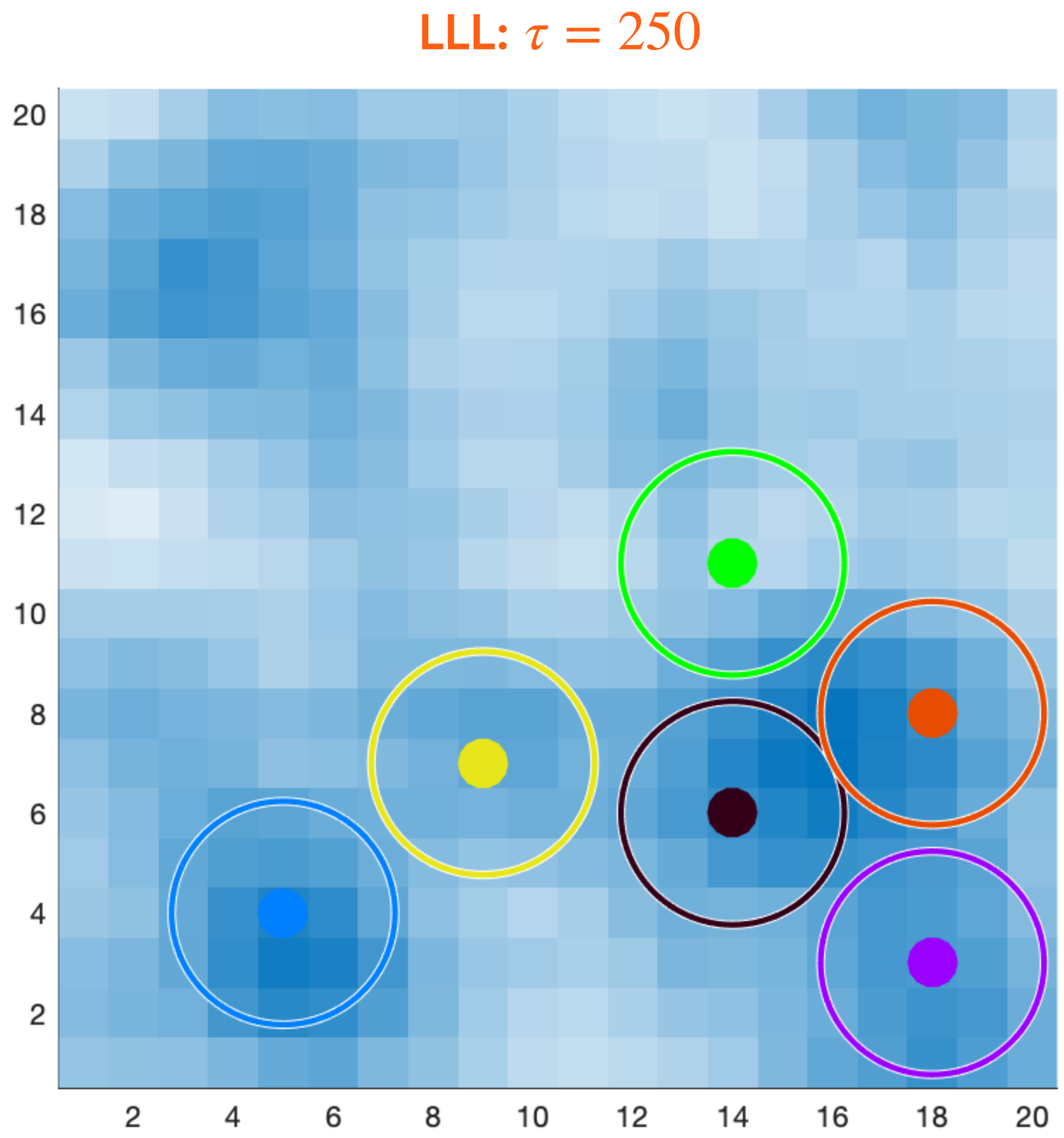
Algorithm Comparison



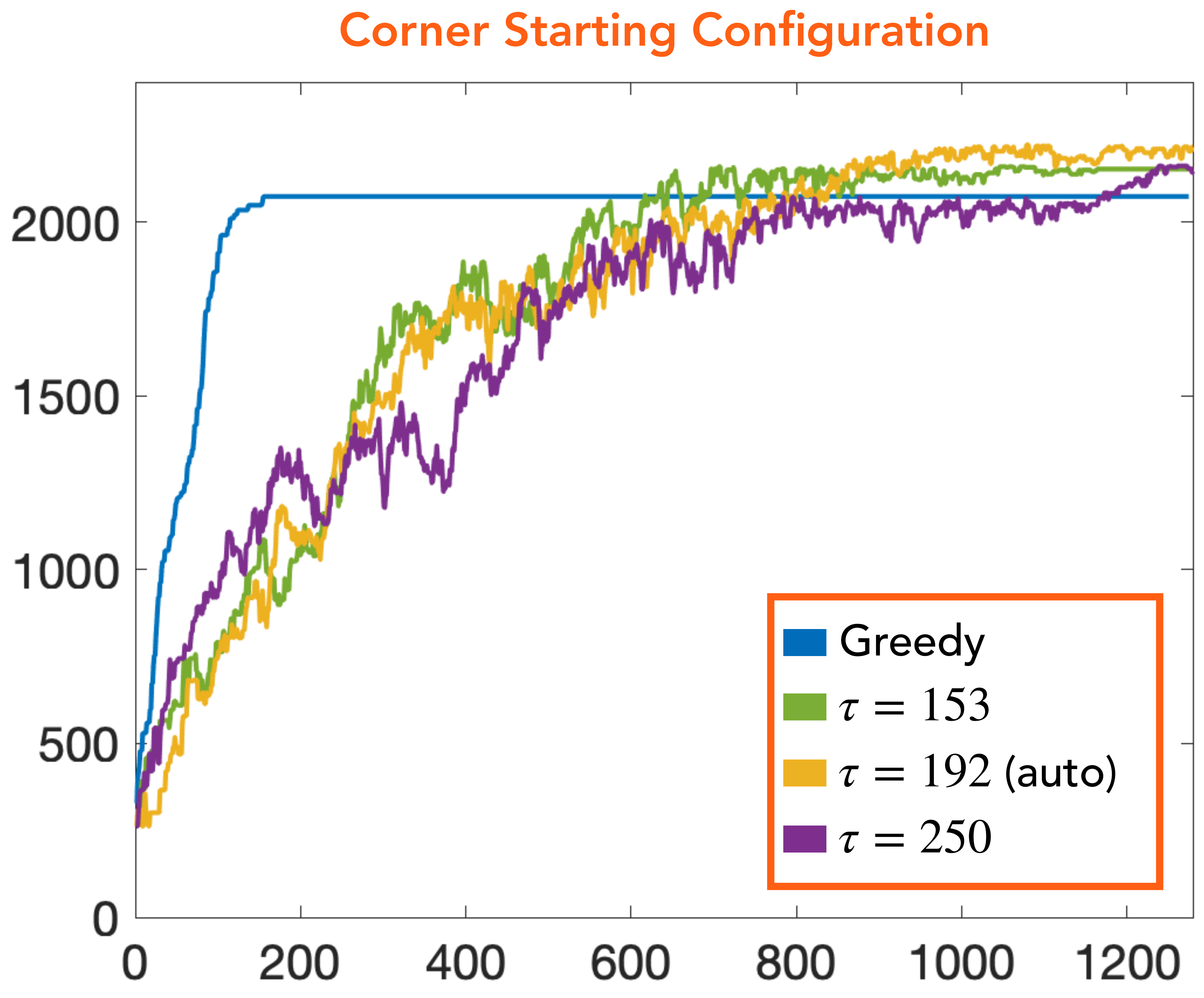


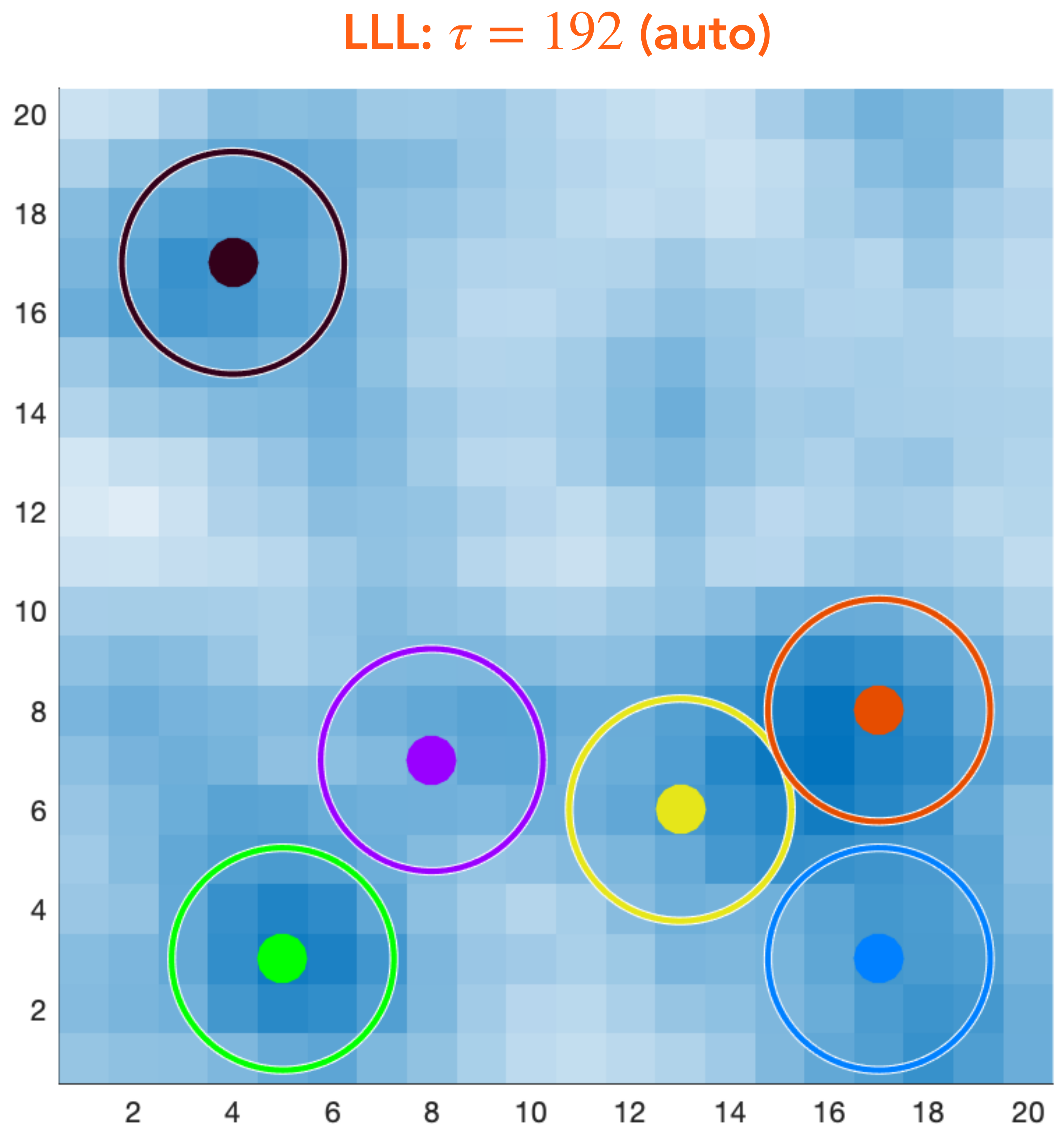
Algorithm Comparison





Algorithm Comparison





Algorithm Comparison

Greedy Algorithm
(Baseline approach)

Drones try solely to maximize their own coverage

Log-Linear Learning (LLL)

Introduces randomness

Drones occasionally take a suboptimal decision

LLL with Automatic Tau Generation

Eliminates need for manual tuning of constants

Algorithm Comparison

Greedy Algorithm
(Baseline approach)

Drones try solely to maximize their own coverage

Log-Linear Learning (LLL)

Introduces randomness

Drones occasionally take a suboptimal decision

LLL with Automatic Tau Generation

Eliminates need for manual tuning of constants

Summary

- Simulated multi-agent coverage system
- Implemented and compared greedy, log-linear learning, and automatic τ generation algorithms
- Demonstrated that LLL with automatic τ generation has the best performance

Future Work

- Test algorithms on maps with different characteristics
- Additional starting configurations
- Analyze effect of other variables on utility
  - Sensing radius, movement radius, number of agents, distance from other agents, and adding obstacles, limited communication
  - Take into account for τ generation
- Verify algorithms on physical multi-drone systems

Acknowledgements

- Bryce Ferguson (Mentor)
- Jason Marden (PI)
- A S M Iftekhar (TA)
- Dr. Michael Hughes (RMP)
- Dr. Lina Kim (RMP)
- Lab Peers: Siddharth Ganesan, Nischal Sinha