

Reorganization and Optimization in Structured P2P Systems

Rob Beverly

MIT LCS

`rbeverly@lcs.mit.edu`

October 29, 2003

Structured P2P Assumptions

- Structured P2P ala Chord
- Chord-like linear identifier space
- Service: Key to node mappings through DHT
- Consistent Hashing for Uniform Node Distribution
- Exponentially Distributed Node Lifetimes
 - High Rate of membership changes
- High Degree of Node Heterogeneity in Physical Net:
 - Bandwidth
 - Latency
 - Connectivity

Problem

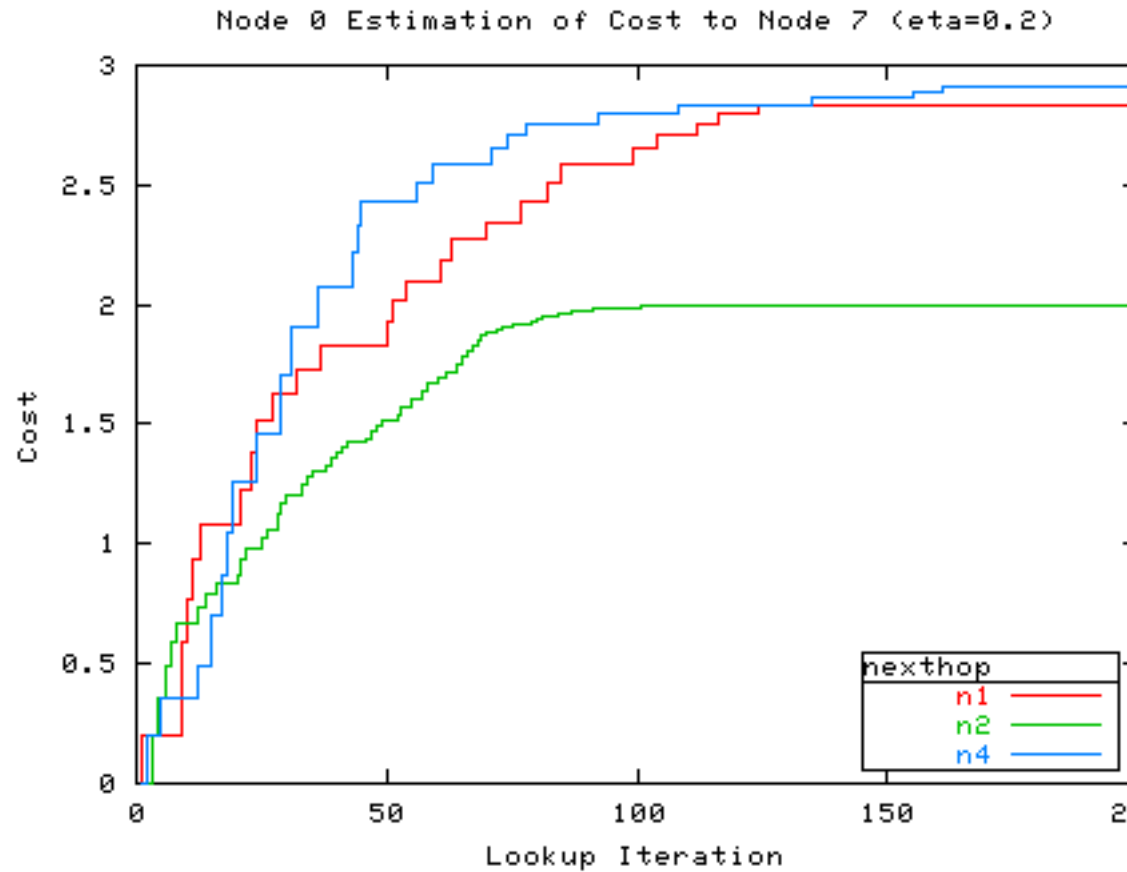
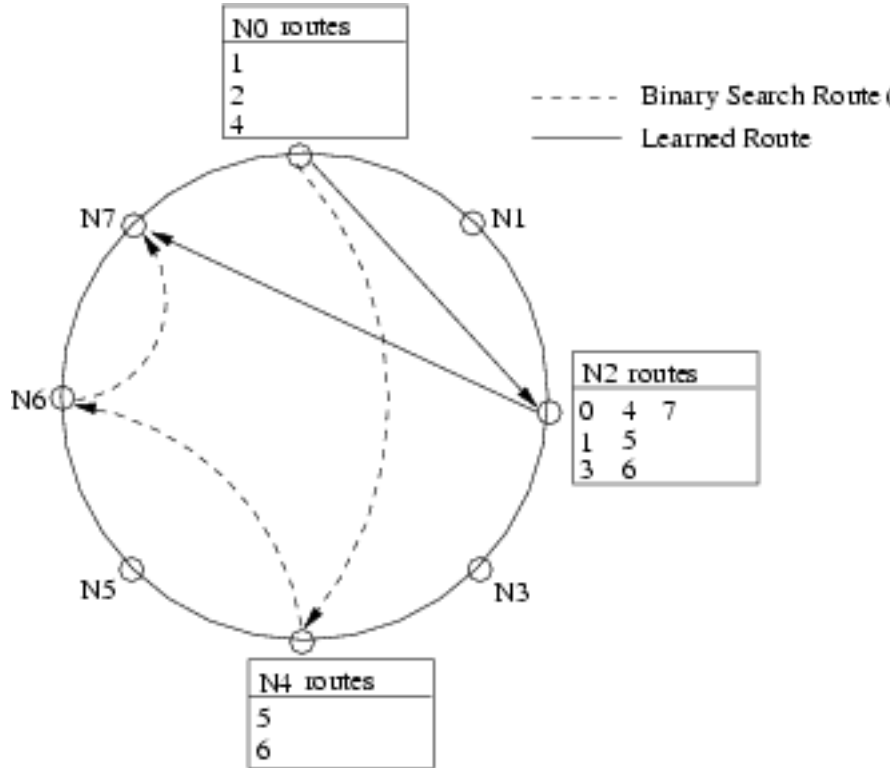
- Structured P2P system provide $\log(n)$ lookups with $\log(n)$ storage
- Generally ignore aforementioned heterogeneity
- E.g. path in physical topology can be poor
- Approach: Use Reinforcement Learning (RL)
- Reorganize by learning optimal routing policies in overlay
- Goal: demonstrate improved performance in the face of heterogeneity and membership changes
- Ideally show how long-lived nodes can stabilize system

Reinforcement Learning (RL)

- Greedy routing may lead to suboptimal global result
- Exploration vs. Exploitation
- Reward function
- Simulated Annealing
- Q-Learning:
 - $Q_x(y, dst) = Q_x(y, dst) + \eta(Q_y(z, dst) + cost - Q_x(y, dst))$
- Similar to Bellman-Ford: Asynchronous Dynamic Programming with Damping

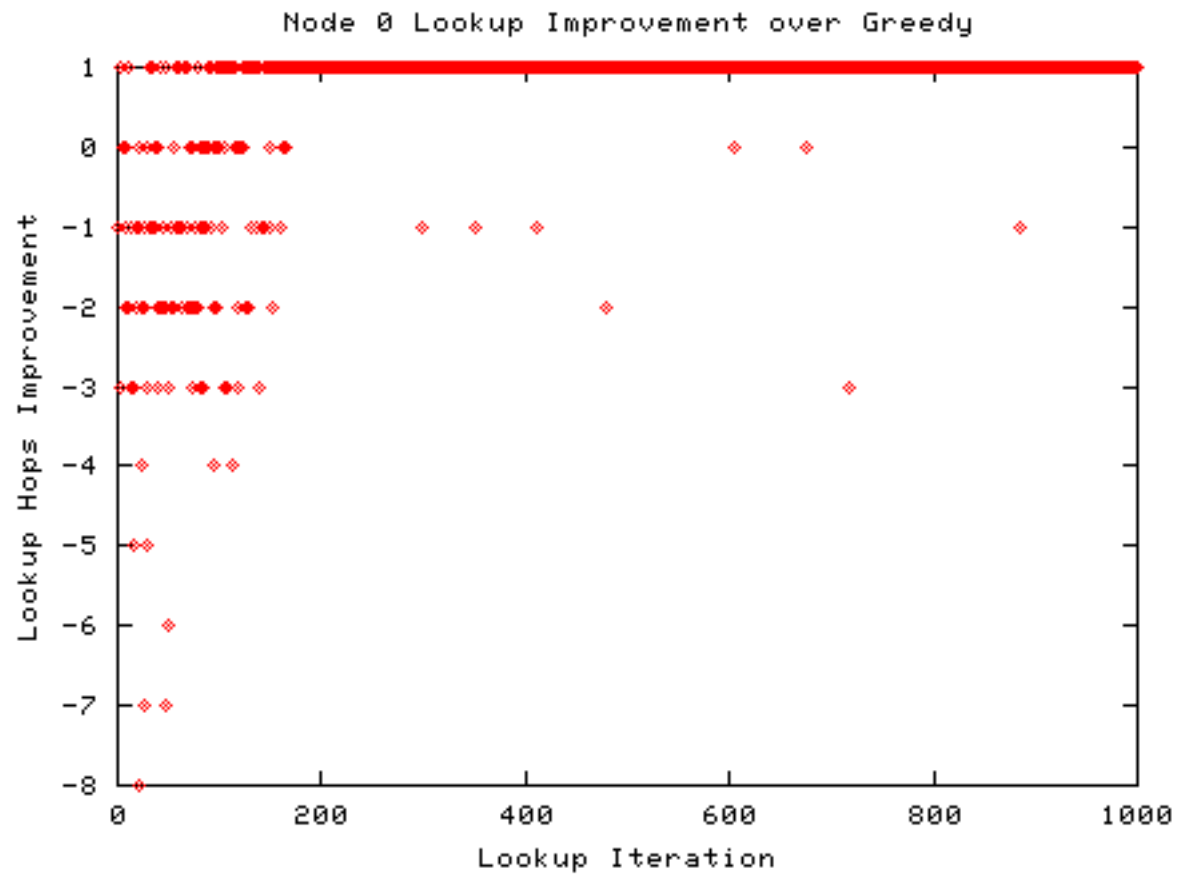
Intuitive Example

- Example Eight Node Network



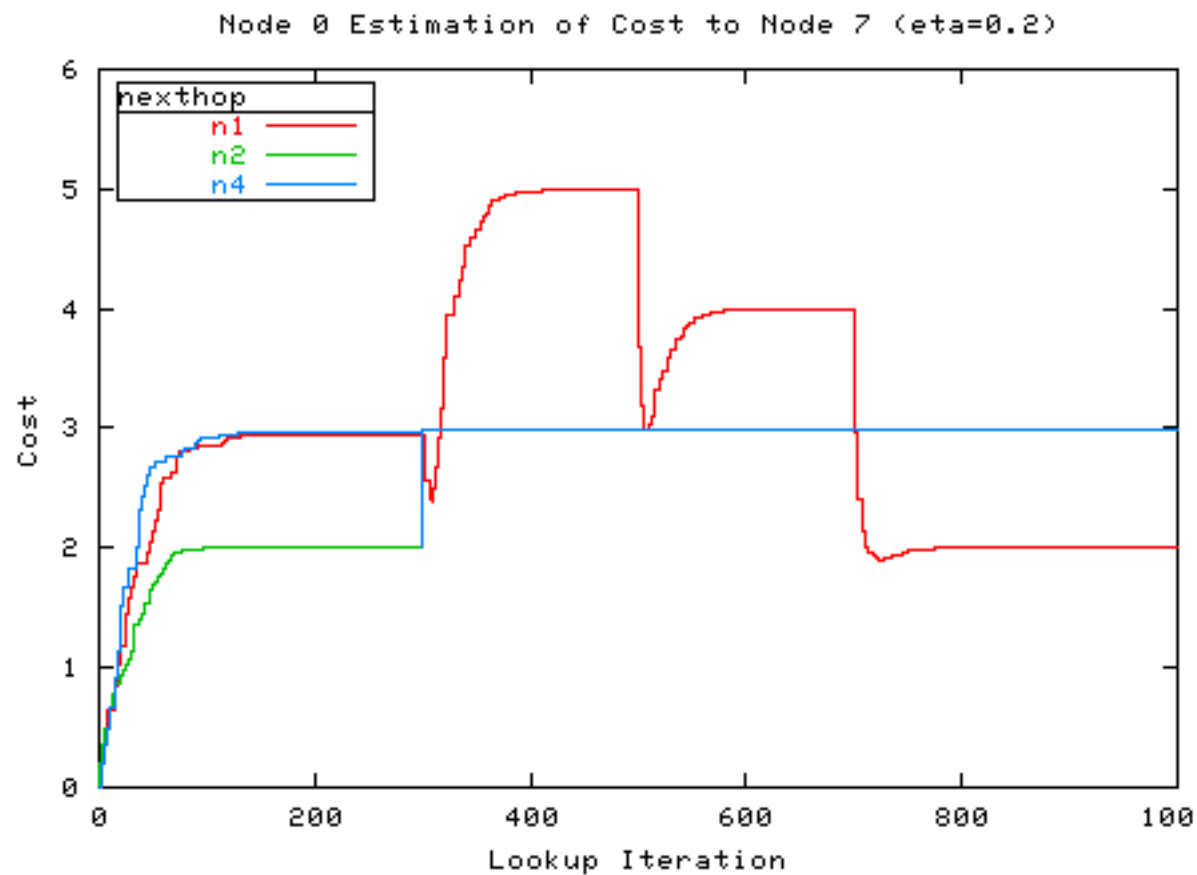
Comparison

- Q-Learning improvement vs. greedy:



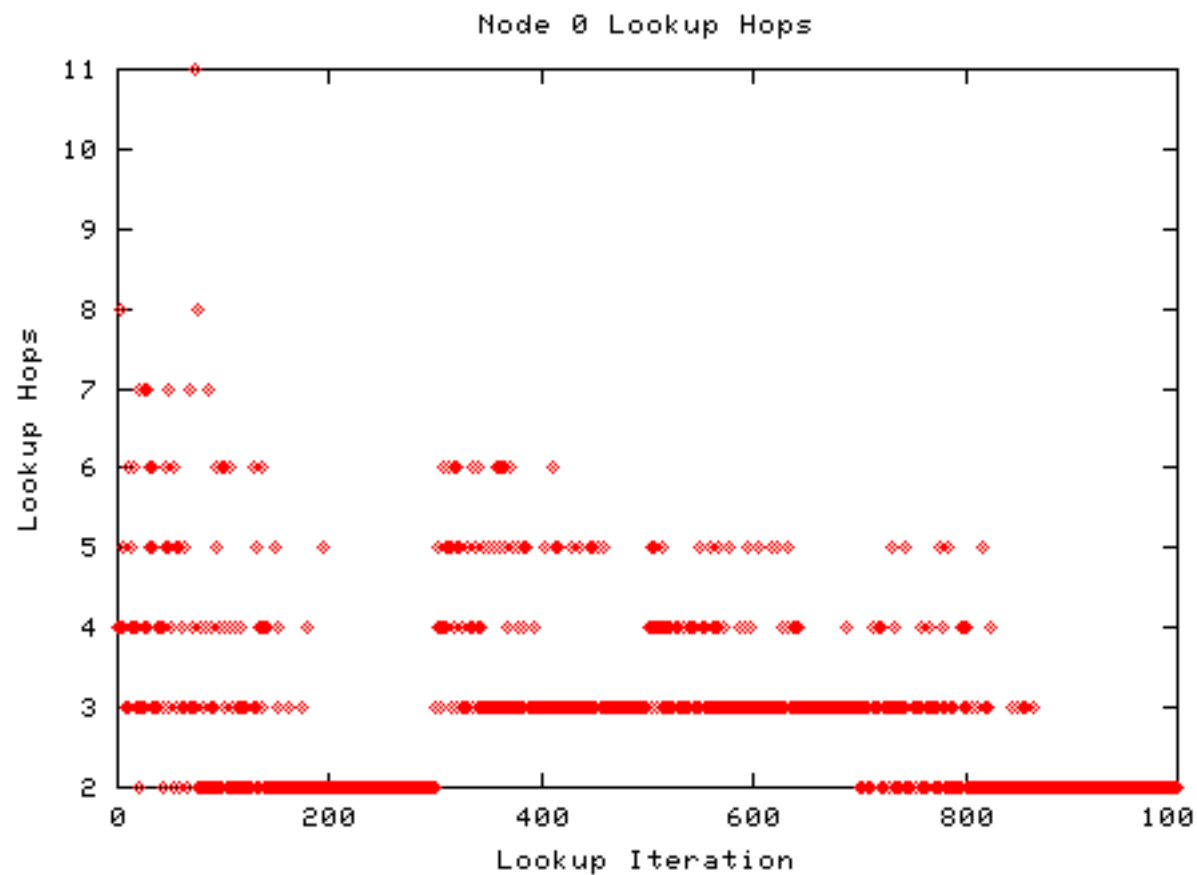
Reaction to Membership Changes

- Simulate events at iteration 300, 500, 700:



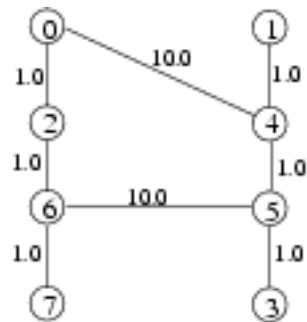
Reaction to Membership Changes

- Simulate events at iteration 300, 500, 700:



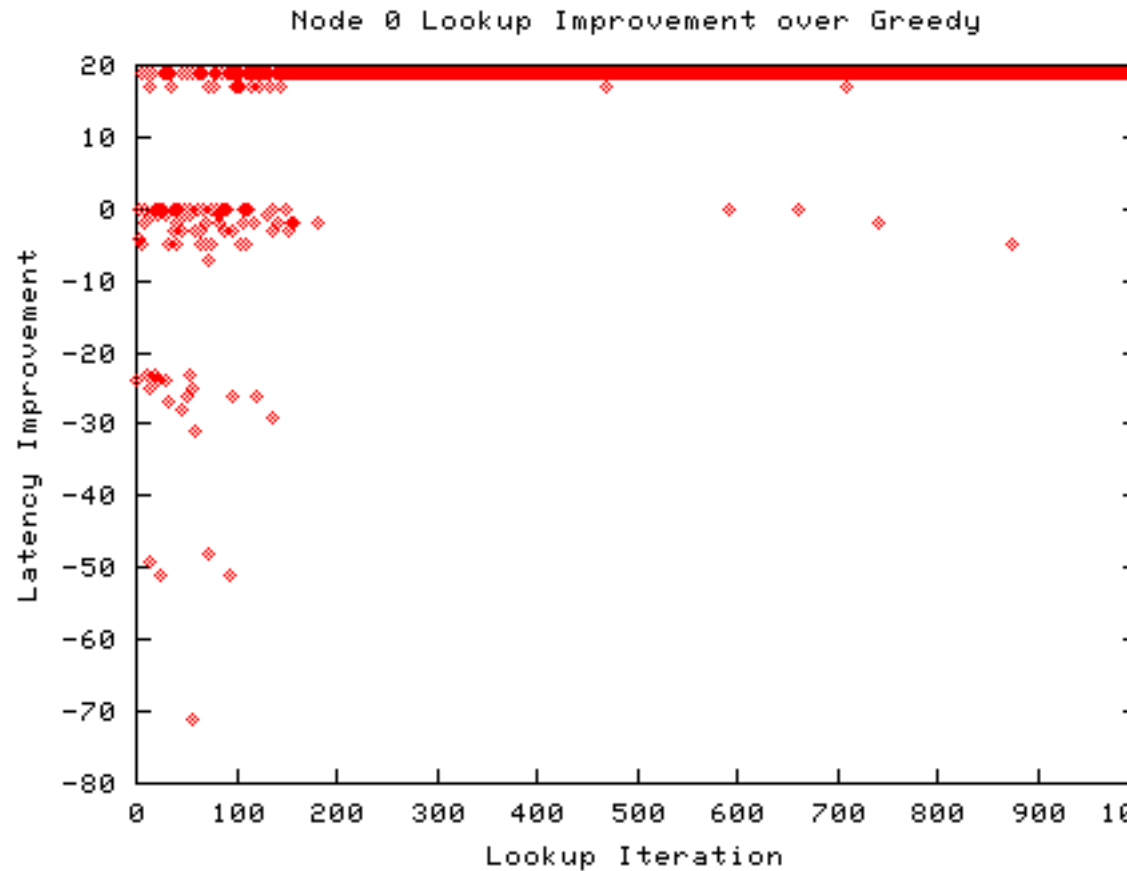
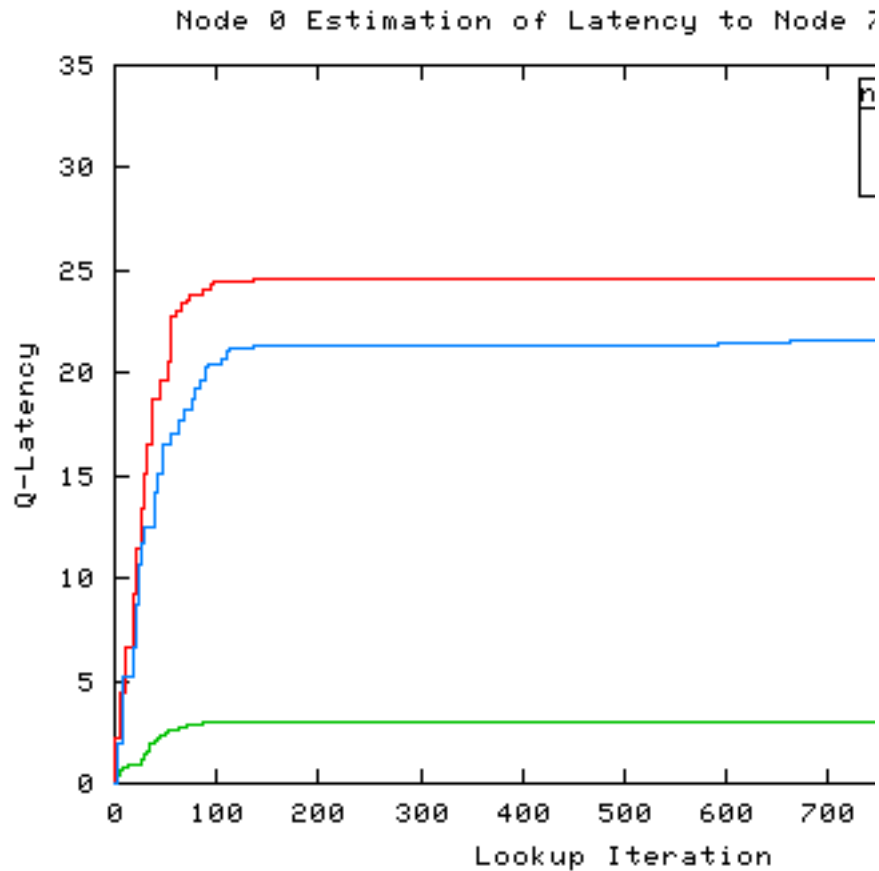
Q-learning of Underlying Topology

- Assume nodes are distributed across two continents
- For eight-node example:



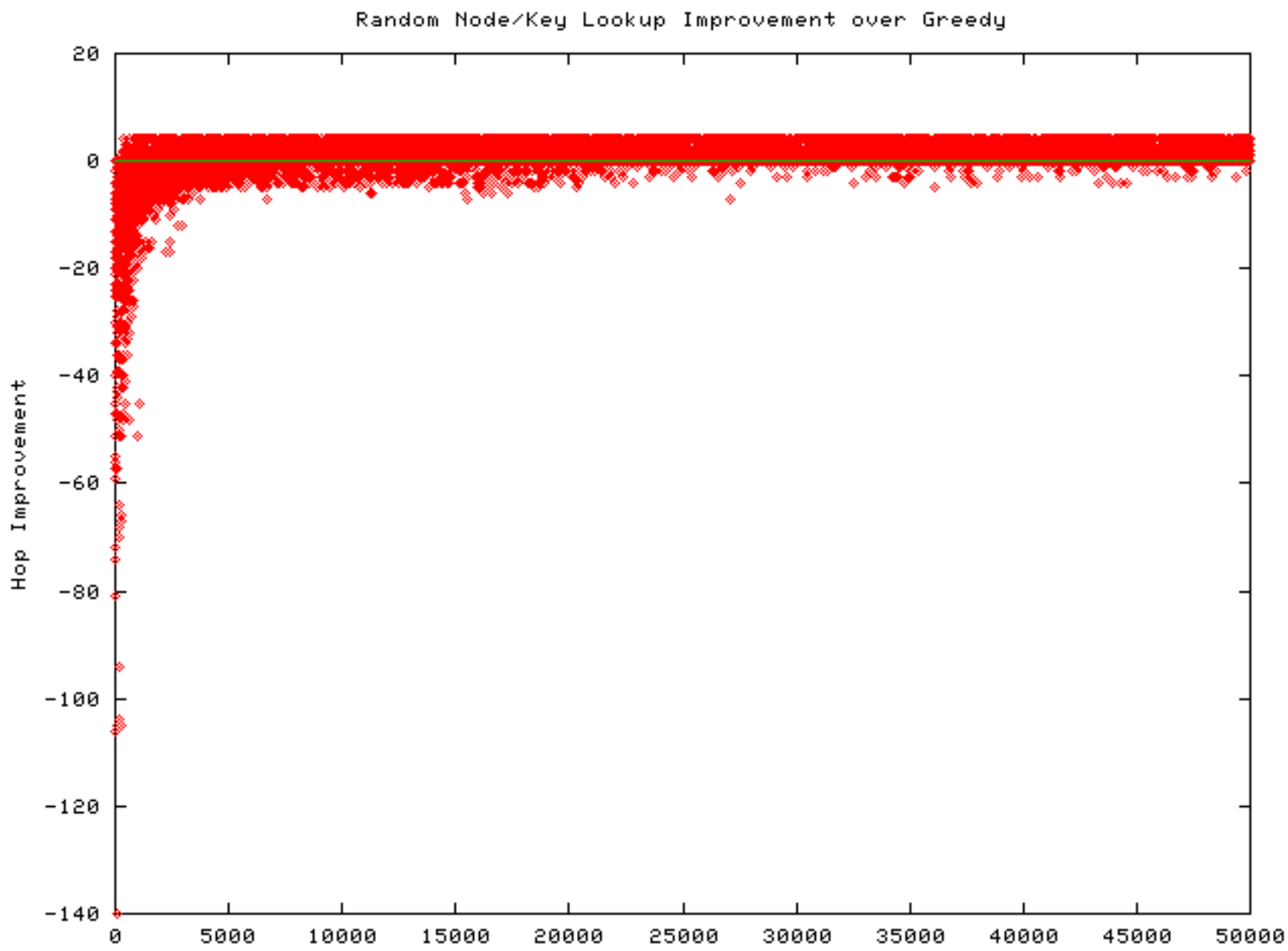
Q-learning of Underlying Topology

- Assume nodes are distributed across two continents



Stability under Randomness

- Still small keyspace: 2^6
- Each node maintains successor and a Kleinberg pointer to another node
- $n = 20$, additional 10 "landmark" long-lived nodes
- What is the stability of this system under random lookups?



Scaling Q-Learning

- Q-learning state requirements scale as $n(routes/node)$.
- Segmentation Strategy
- Identifying independent nodes
- Others

