Comparison of server selection algorithms

Toshiyuki MIYACHI

toshi-m@jaist.ac.jp.

Japan Advanced Institute of Science and Technology

and

WIDE Project

Background

- There are many server-client model communications
  - With these communications, clients have to select its server
- There are some server selection algorithms
  - Best server selection
  - Uniform(Random) server selection
  - Reciprocal to cost server selection
- Best server selection is very popular but it is unstable
  - if there are 2 servers which cost is almost same from clients, these clients always choose one server
  - Then the cost to the server is changed, many clients choose the other server
- Using uniform server selection is very stable but it is not efficient

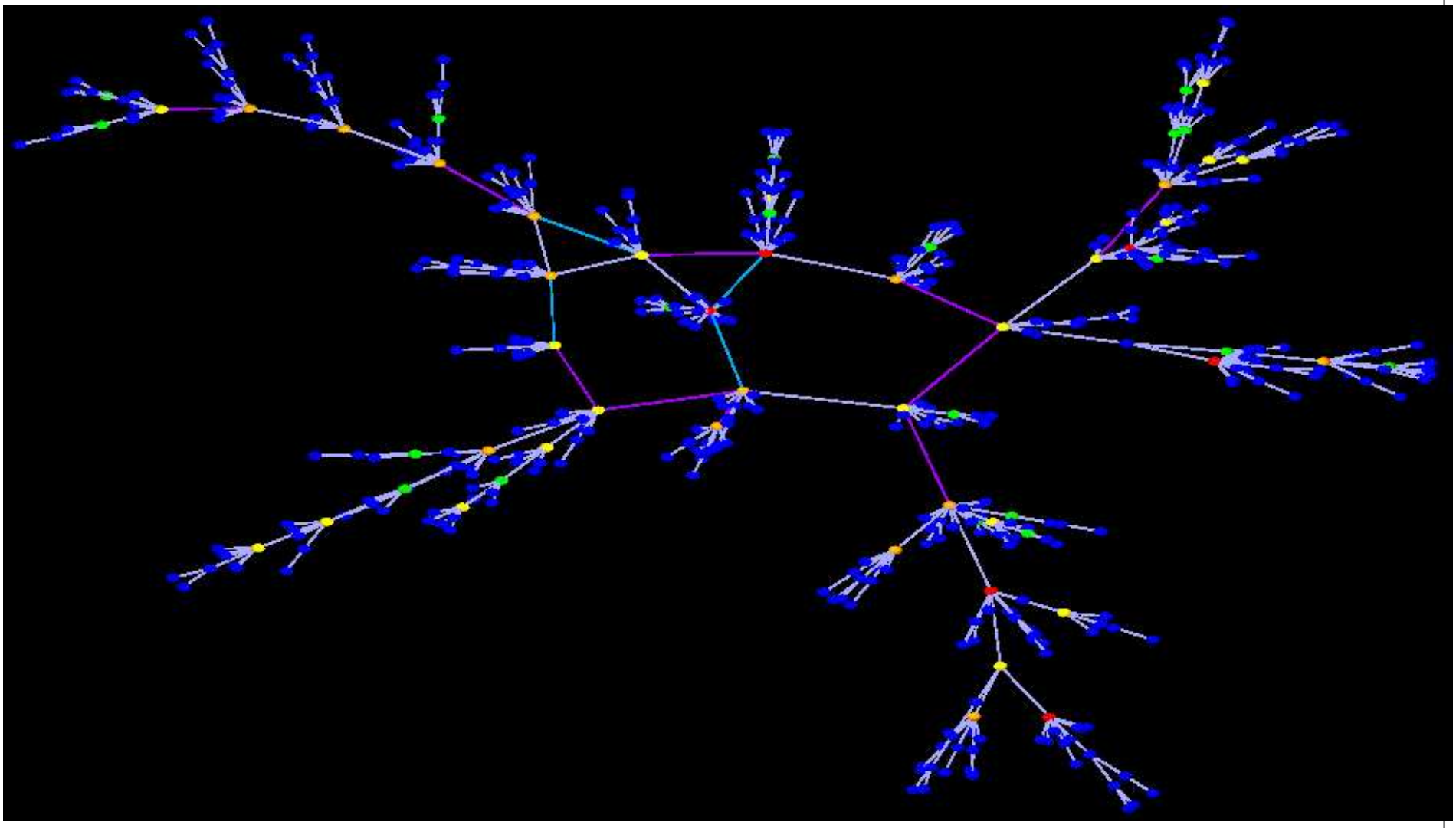
Motivations

- We want to know server and node behaviors with these algorithms when the cost to server from clients is changed
- And then we will consider better algorithms

In this presentation I'll talk about the simulation of these existing server selection

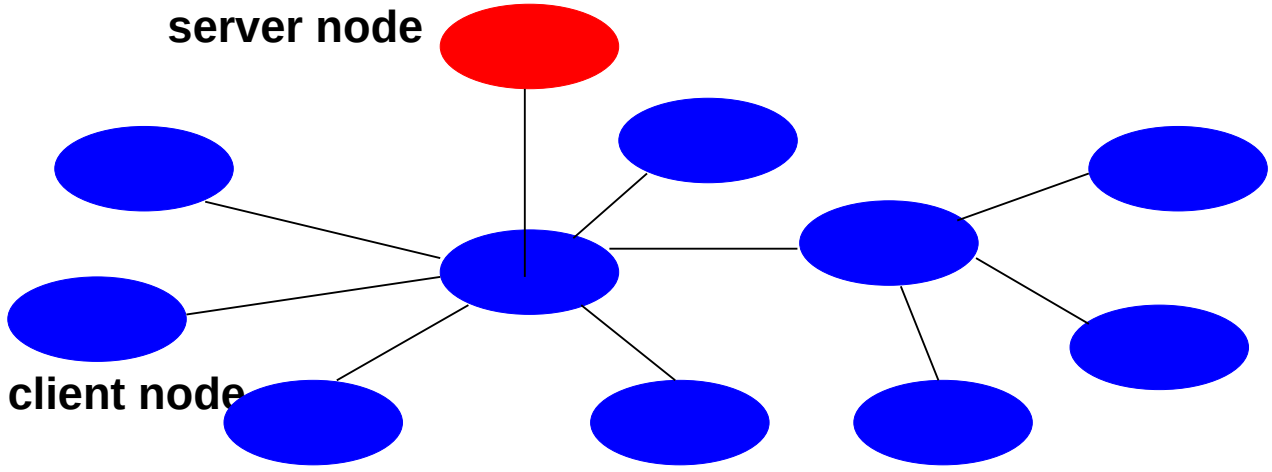
The simulation topology

There are 510 nodes and 60 servers



Rules of making topology(1/2)

- Selecting a node as a parent and connecting a new node to the parent
  - The parent node is selected at the probability of proportion to the number of its edges
  - The edge cost is 10
- Placing a server
  - On the node with the most a lot of edges
  - Every 10 nodes

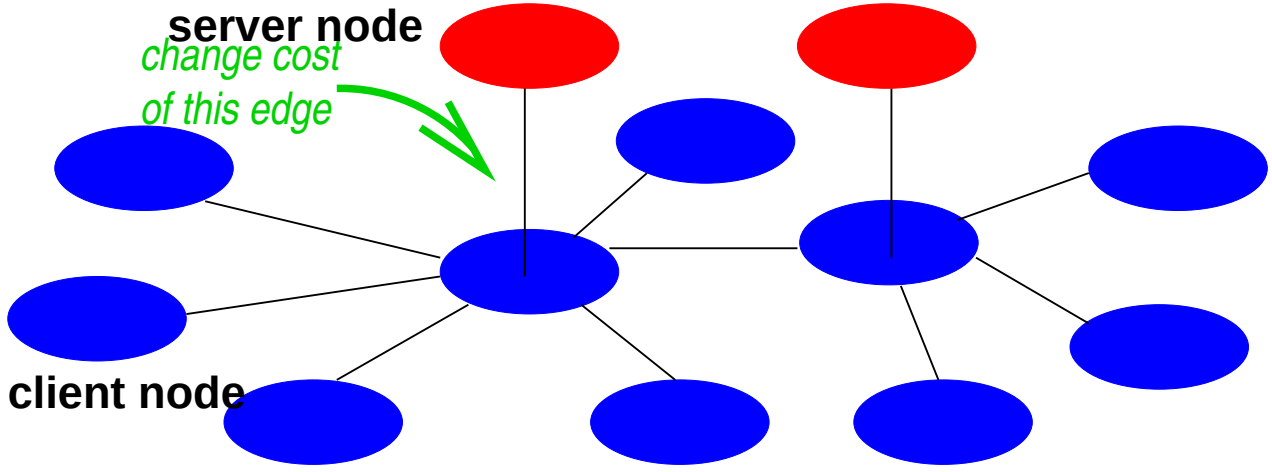


Rules of making topology(2/2)

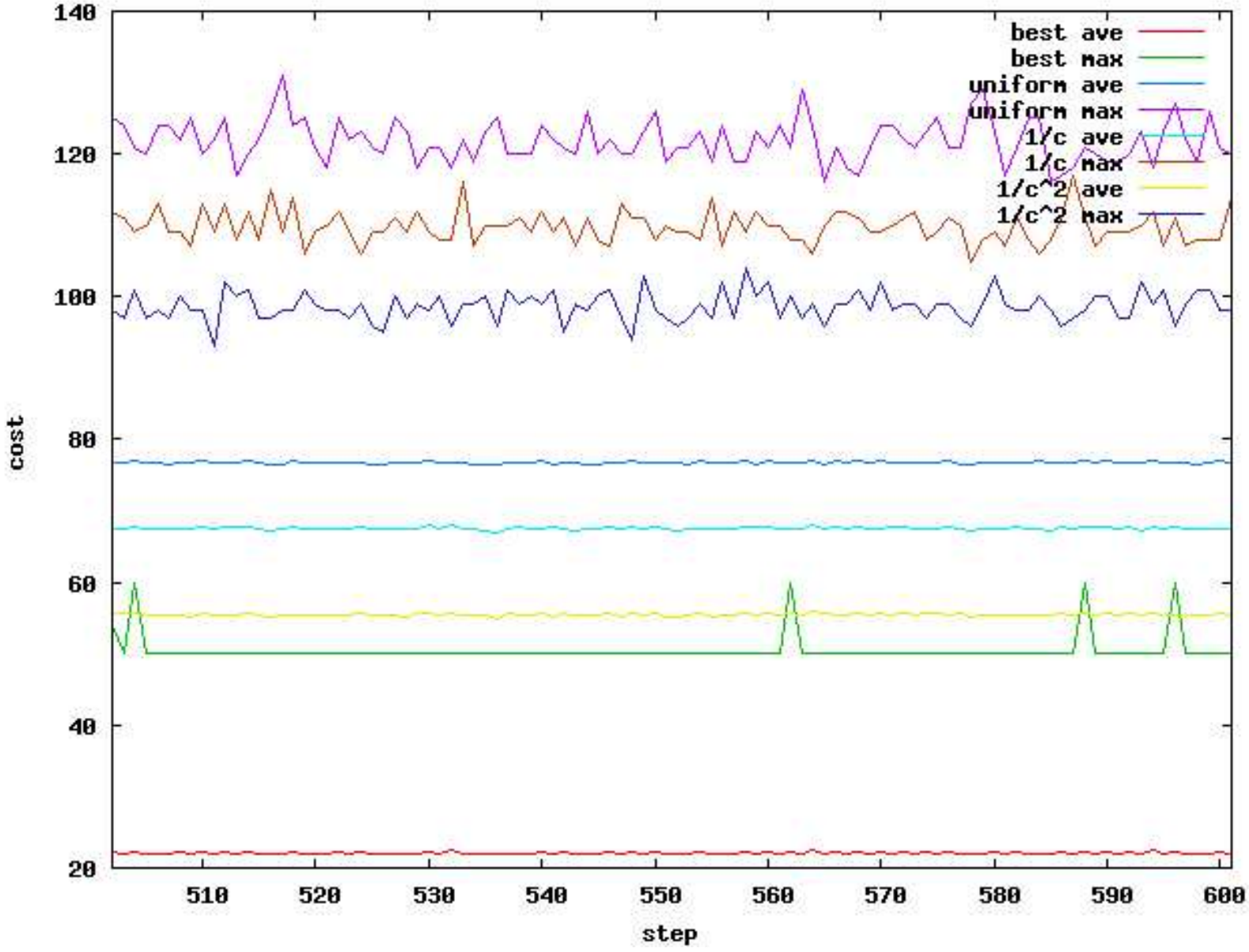
- A server splits when its clients number become over 20
  - (In making topology phase, clients chooses its server using best server selection)
- Connecting servers
  - 2 servers are selected at the probability of proportion to the number of its edges
  - Every 100 nodes

The simulation

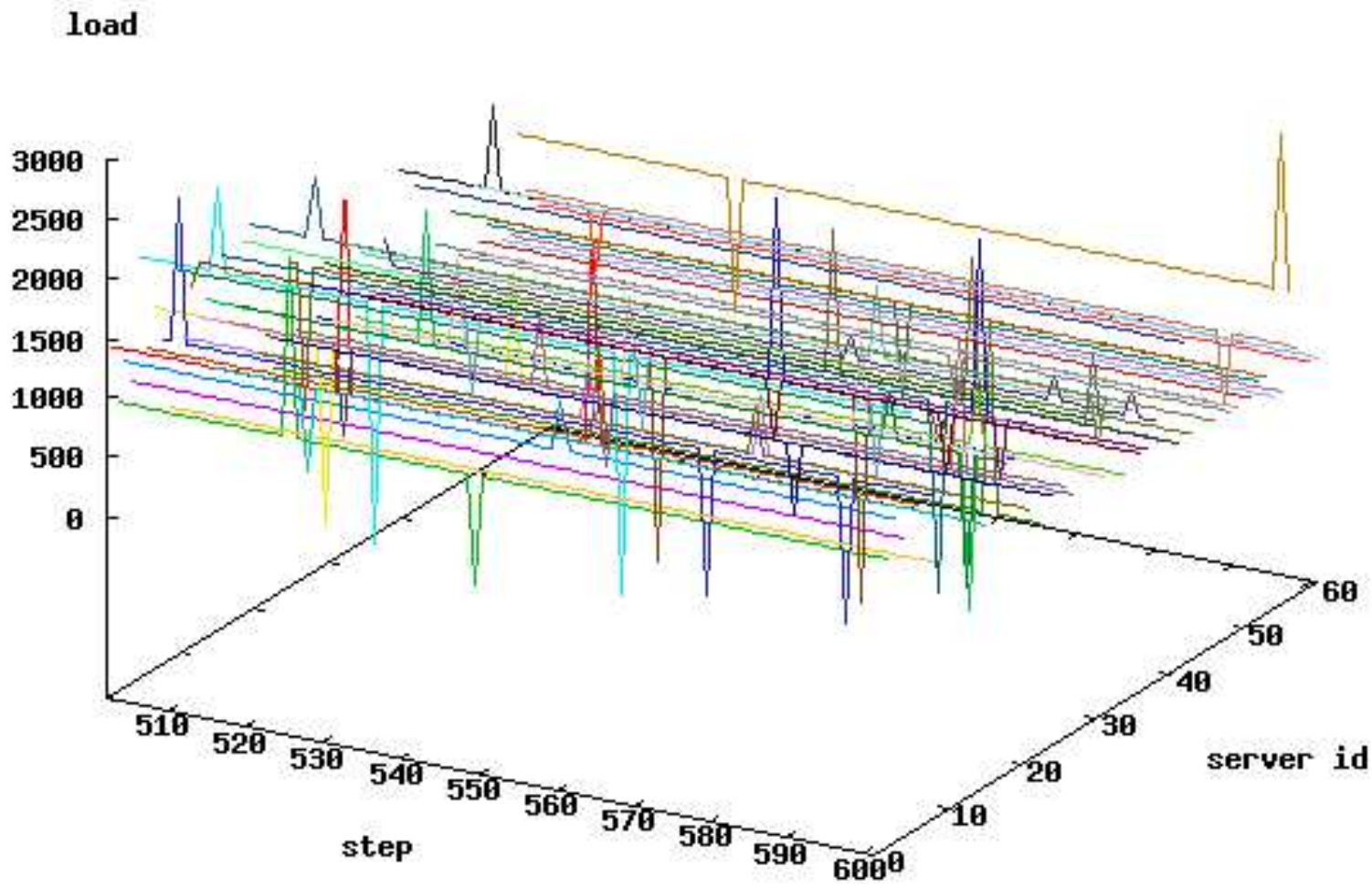
- Picking up one server at random each step
- Changing the cost between the server and its binded nodes
  - The cost is chosen at random between 1-40
- All nodes send 100 queries to its server each step
- Then restore the cost to the initial condition



Average cost and maximum cost

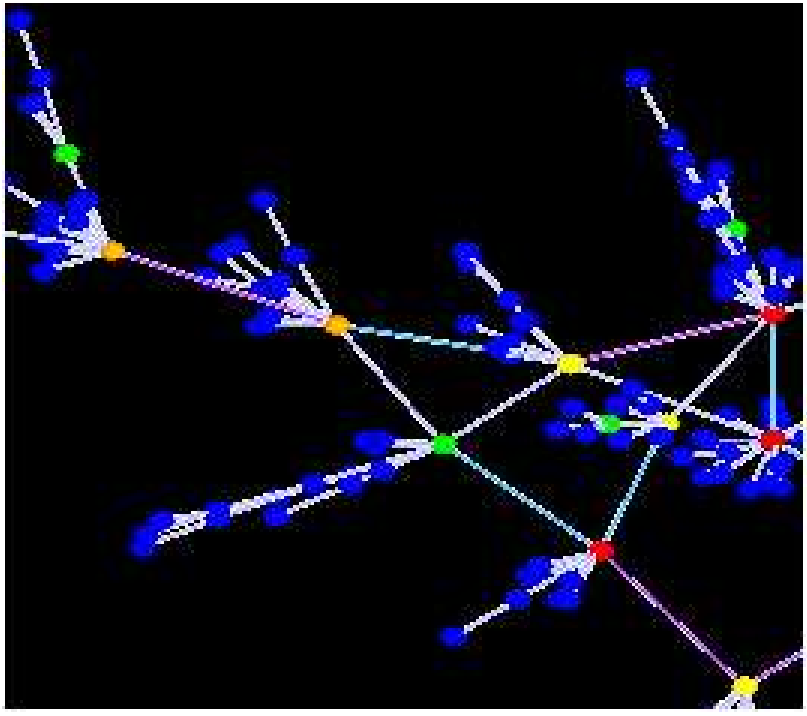


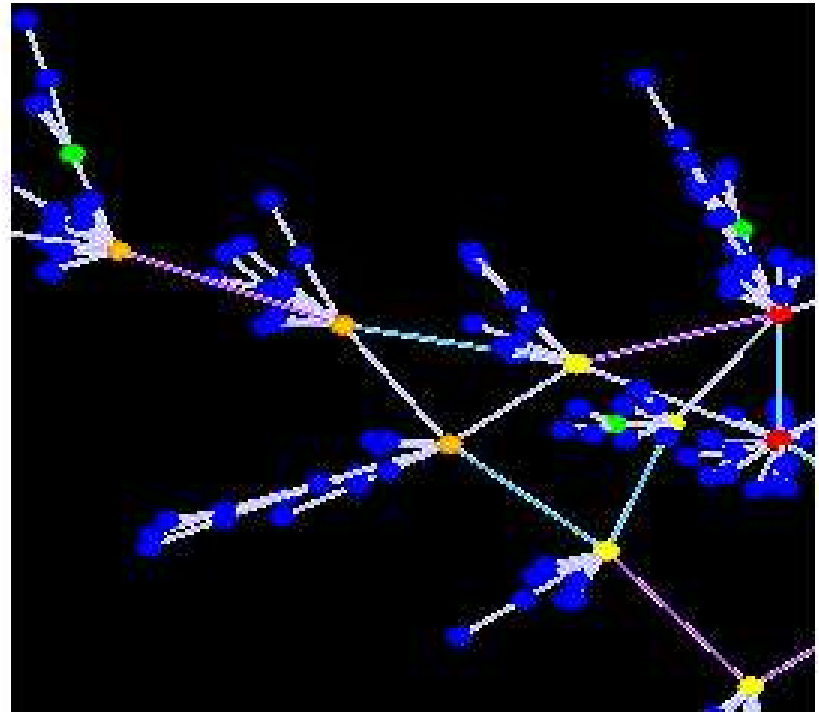
Server load: Best selection(1/2)



Server load: Best selection(2/2)

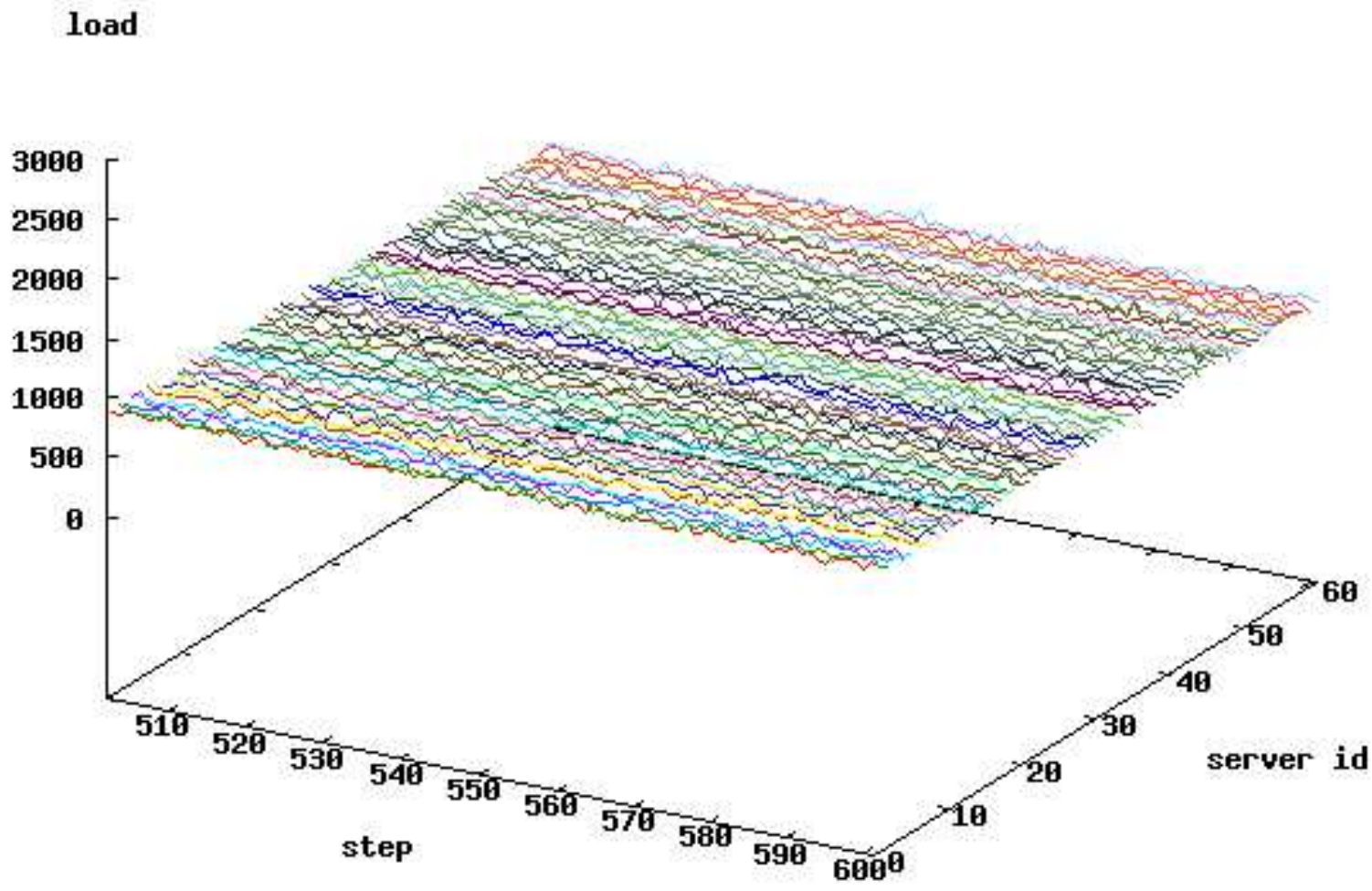
- An edge from a server to its binded cost is changed 29 $\rightarrow$ 10
- Another server's load is highly increased
- The server's load is reduced



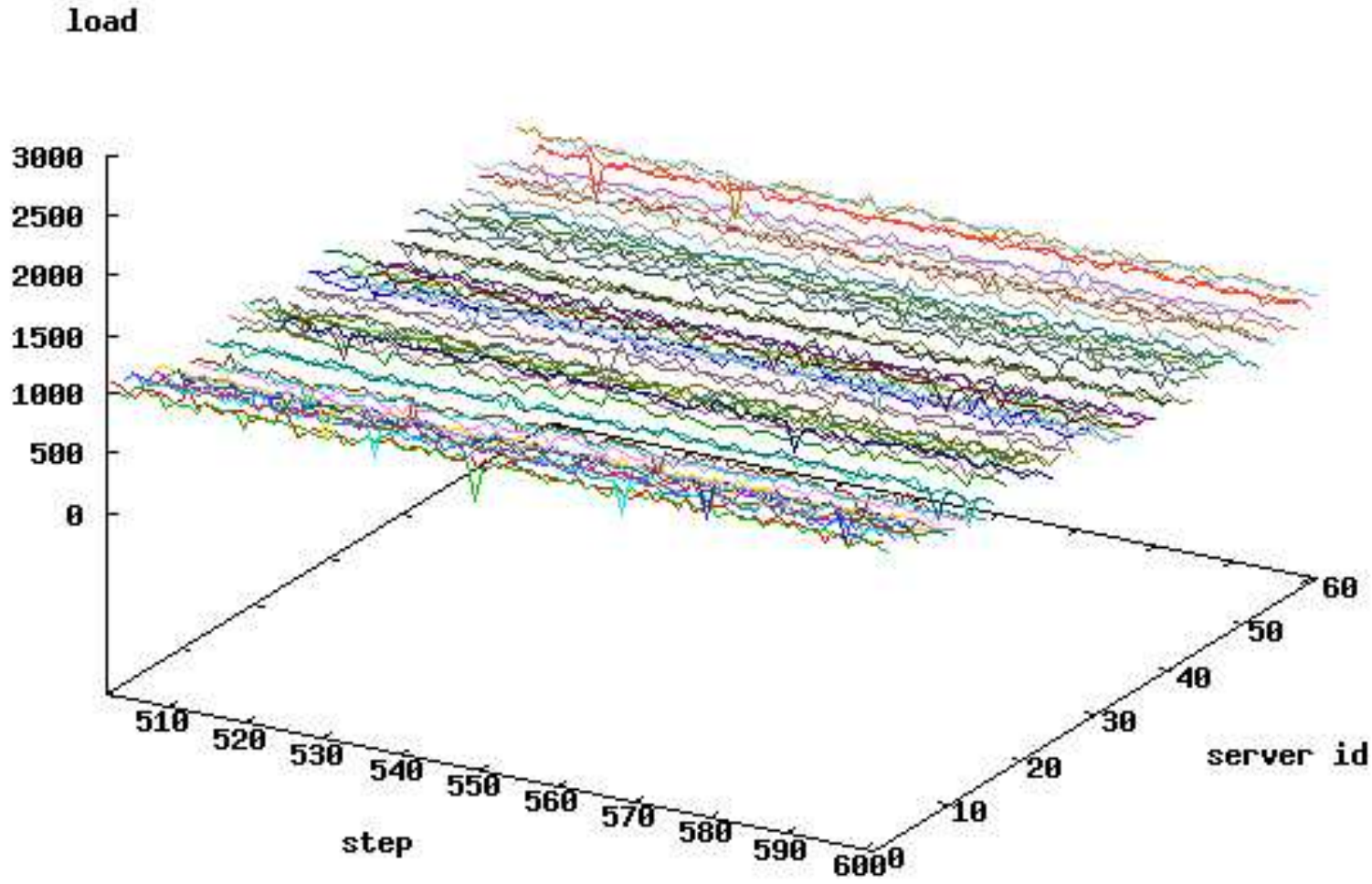


green:0-599 yellow:600-1099 orange:1000-1599 red:1600-

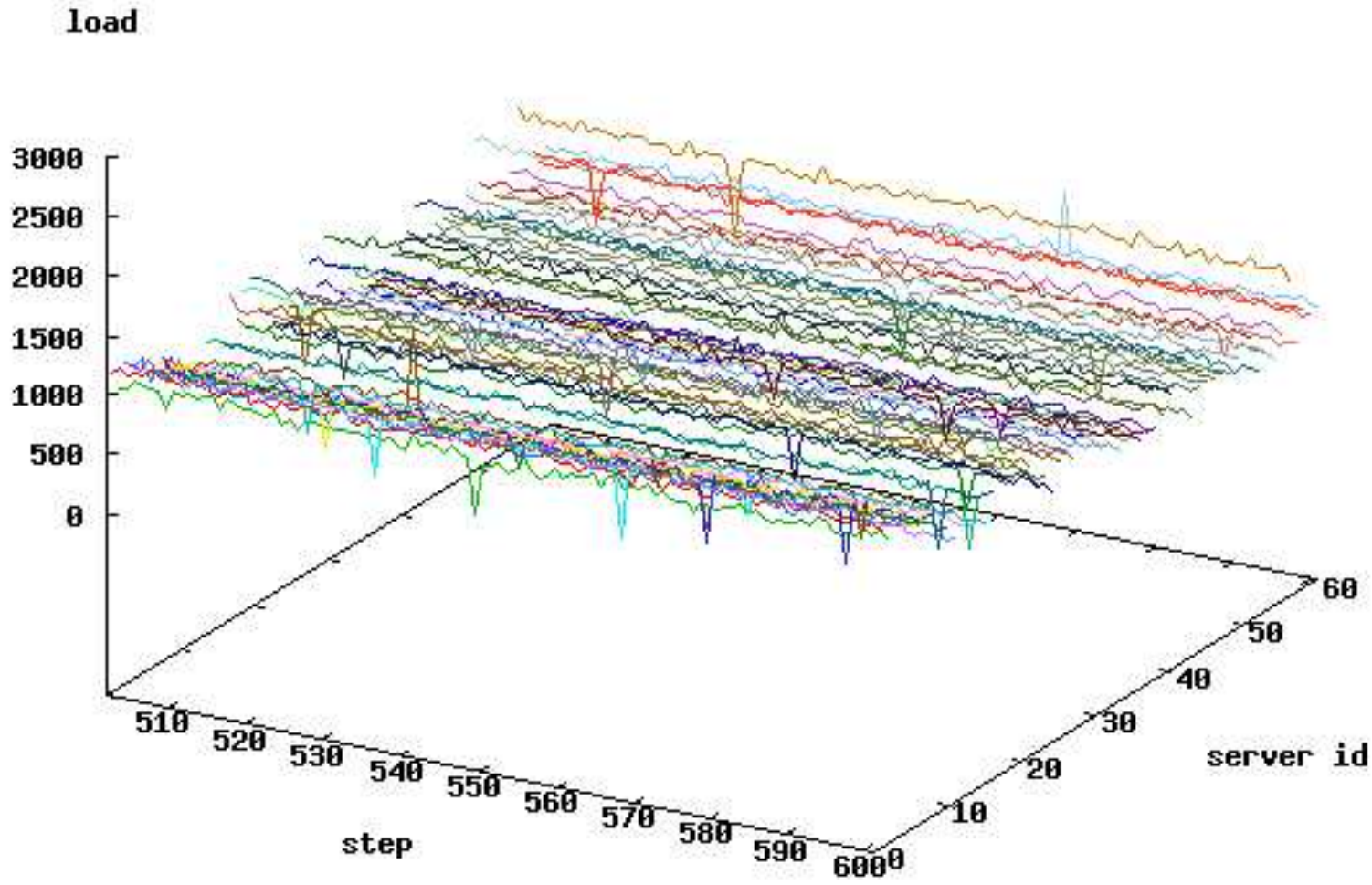
Server load: Uniform selection



Server load: Reciprocal selection(c)



Server load: Reciprocal selection(c^2)

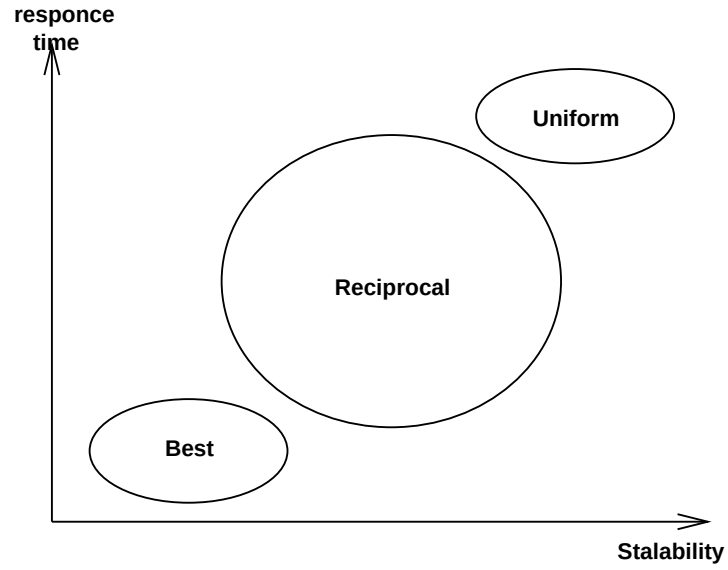


Server load: Reciprocal selections

- There are no server which have many clients
- Not so big fluctuation, if any edge cost is changed
- Between best server selection and uniform server selection

Comparison of server selection algorism

- Using best server selection, when cost to big server is changed, the influence is very big
- Using uniform server selection, the server load is very stable, but its efficiency is bad
- Using kind of reciprocal server selection, the performance is middle of these algorithms



Future work

- More simulations
  - Simulation results are influenced by the topology, we consider the relationship between topologies and simulation results
  - If server cost is proportional to its load, I will get another type of simulation result
- New server selection algorism
  - Categorize topologies and try suitable server selection algorithms for these topologies

Conclusion

- Measured server loads and costs from a client to its server
- Simple visualization of server load
- Using reciprocal server selection, server load and cost from client to server is not so much fluctuated
- Consider relationships between topologies and server selection algorism
- Try to find better server selection algorism for each type of topologies