Towards Self-organizing Bureaucracies

Carlos Gershenson

Goal

- To explore the benefits of using self-organization to improve the efficiency and adaptability of bureaucracies.
- Application of General Methodology to Design and Control Self-organizing Systems
  - http://uk.arxiv.org/abs/nlin.AO/0603045
- Illustrate benefits with Random Agent Networs
  - Novel computational models

Bureaucracies

- Public or private sectors
  - e.g. tax collection systems, immigration services, military, educational/academic institutions
- No perfect bureaucracy
  - but can always improve
- Obstacles:
  - Rigidity, corruption, delays
- How to measure efficiency of a bureaucracy?
  - Related to the fulfillment of its goals.

Bureaucracies (2)

- Naïve to try to *optimize*
- Problem space constantly changing
- *Adaptation, anticipation & robustness* are required, self-organization as a method to achieve it.

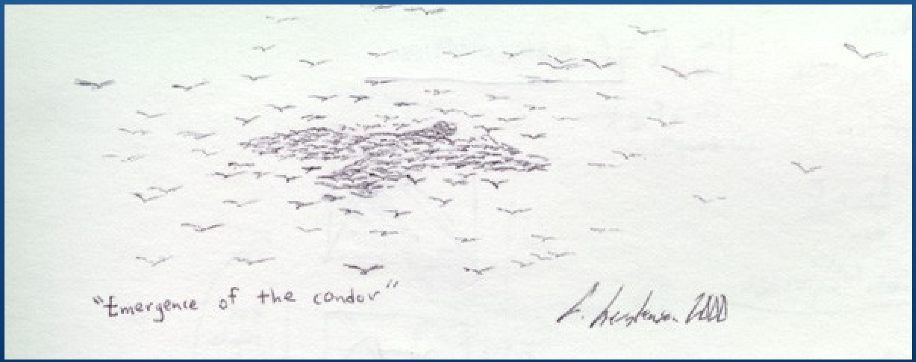
Previous work

- Cybernetics (Beer, 1966; Cybersyn;...)
- Distributed cognition (Hutchins, 1995;...)
- Organizational learning (March, 1991;...)
- Computational org. theory (Carley & Prietula, 1994)
- Agent Based Modeling (Epstein & Axtell, 1996;...)
- Complexity (Anderson et al., 1999; Lissack 1999;...)

Self-organization

•*A Notion*: a system *described* as self-organizing is one in which elements *interact* in order to achieve *dynamically* a global function or behaviour.

- not imposed, nor determined hierarchically
- achieved dynamically as elements interact
- interactions produce feedbacks that regulate the system



Designing S-O.S.

- Organizations as systems of information processing *agents* (Radner, 1993; Van Zandt, 1999; ...)
  - Individuals, deptartments, ministries, public, etc.
- Agents *act* to achieve *goals*
- "Satisfaction" of agents dependent of goals
- Different goals may lead to conflict
- Minimizing "friction" increases satisfaction of system (Helbing & Vicsek, 1999)

Designing S-O.S. (2)

- Synergy as negative friction
- *Mediators* (Heylighen, 2003) to constrain and promote behaviours: min friction & max synergy
  - How to do it? See Methodology...
- Need simulations
  - Cannot predict system, feedback with practice

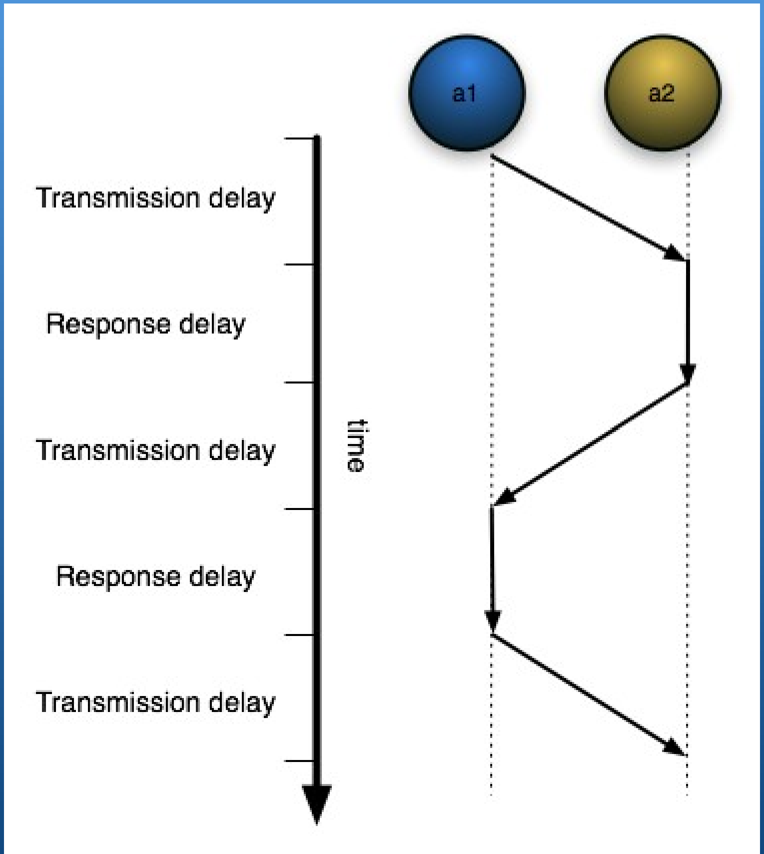
A Self-organizing Bureaucracy

- Elements are expected to *dynamically* and *autonomously* solve a problem or perform a function at the system level

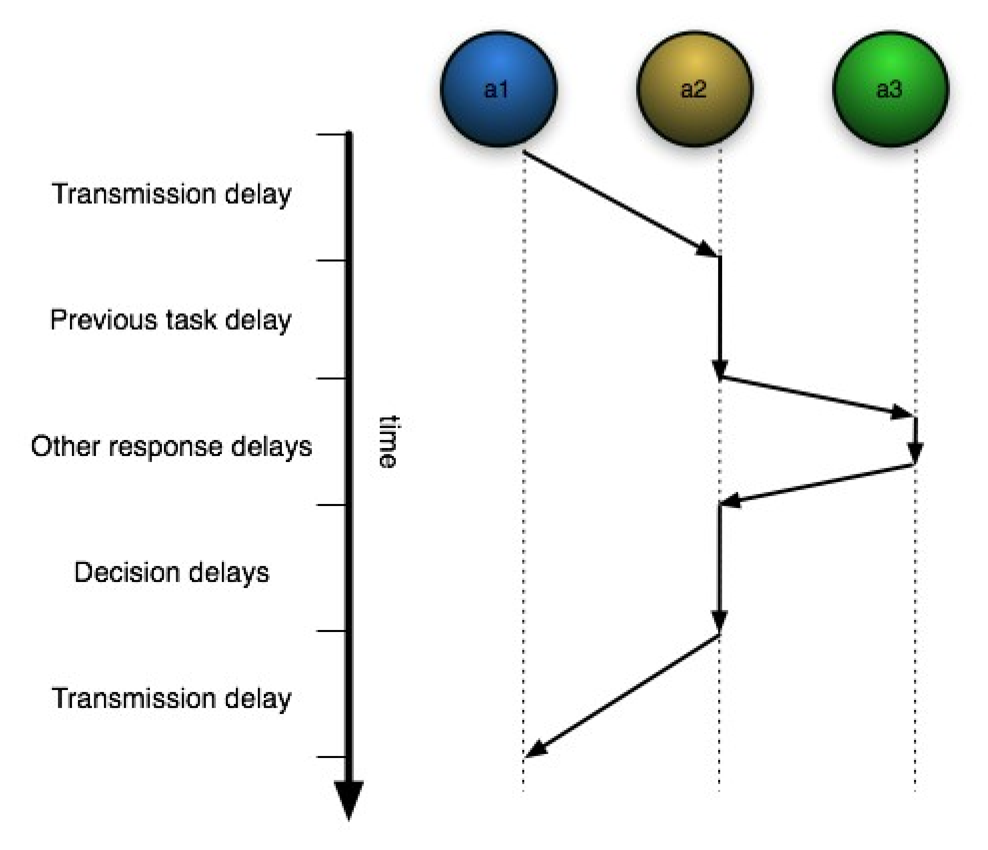
The Role of Communication

- Synchronous
  - Verbal, phone, video, IRC
  - Quick, but needs coordination of agents
- Asynchronous
  - Post, telegraph, telex, fax, IM
  - Delayed, but no coordination
  - Technology has reduced delays...

Delays as Friction

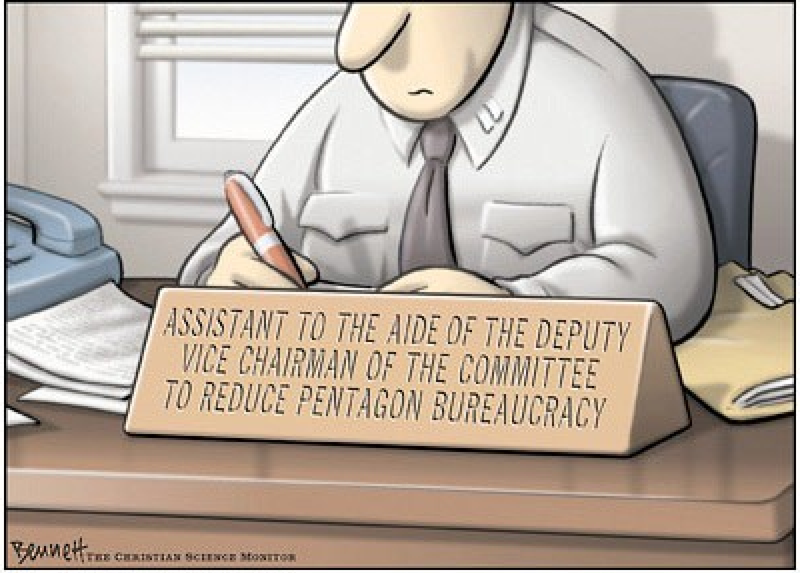


Response delay



- E-media reduce transmission delays
- But also their logs can be analysed to restructure SOBs:
  - Logs show efficiency, workload, and visualization of agents and their interactions

Decision delays



- Technology also reduces them
- E-decision-makers
- Negotiation, trust, reputation facilitate coordination
- E-government
- Computer-aided decision-making
- "Cognitive Stigmergy" (Ricci et al., 2006)

The Role of Sensors

- Public as environment of bureaucracies
- Need good sensors to make good decisions
  - Complex sensors “digest” relevant information
- Public participation slow and difficult
  - e.g. polls

Public satisfaction as efficiency

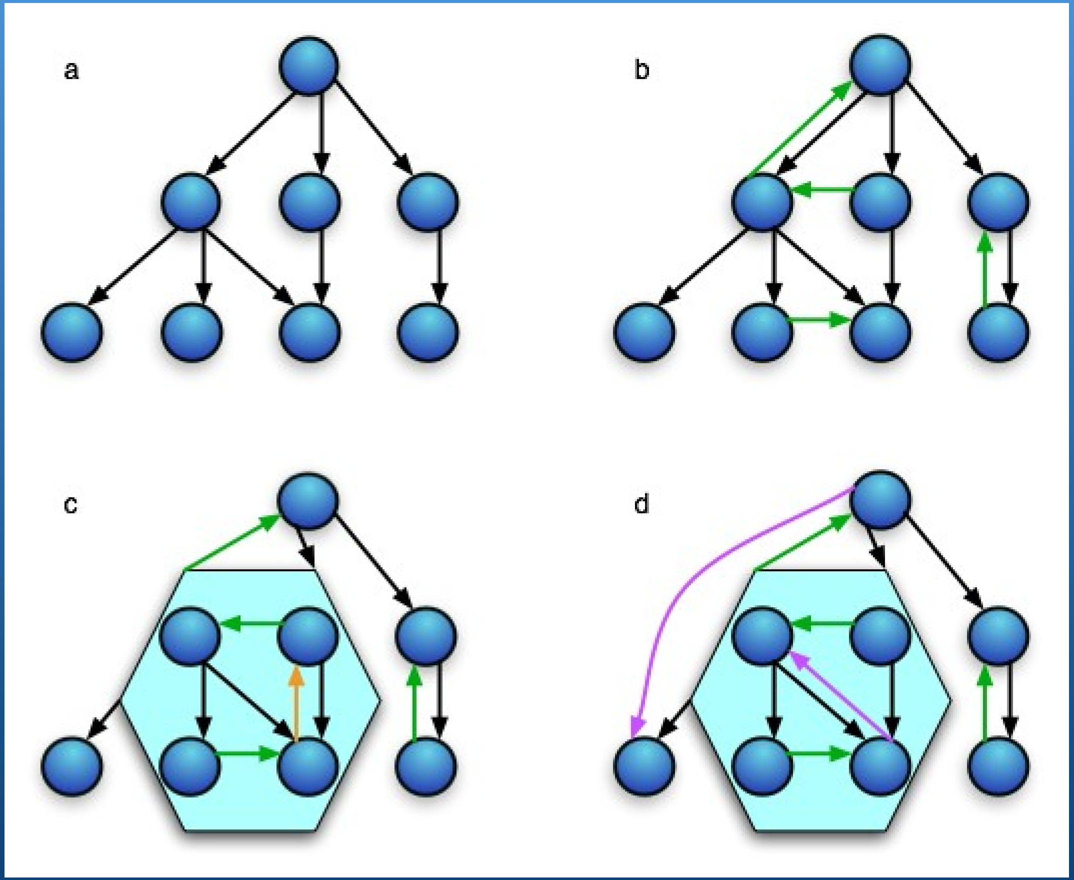
- Low satisfaction = friction
- How to measure without public participation?
  - Public attention delay
    - Waiting delay + processing delay
  - Frequency of interaction
- Public and bureaucracy will be satisfied if delays and interactions are minimized
- Automatically detect bottlenecks

The Role of Hierarchies

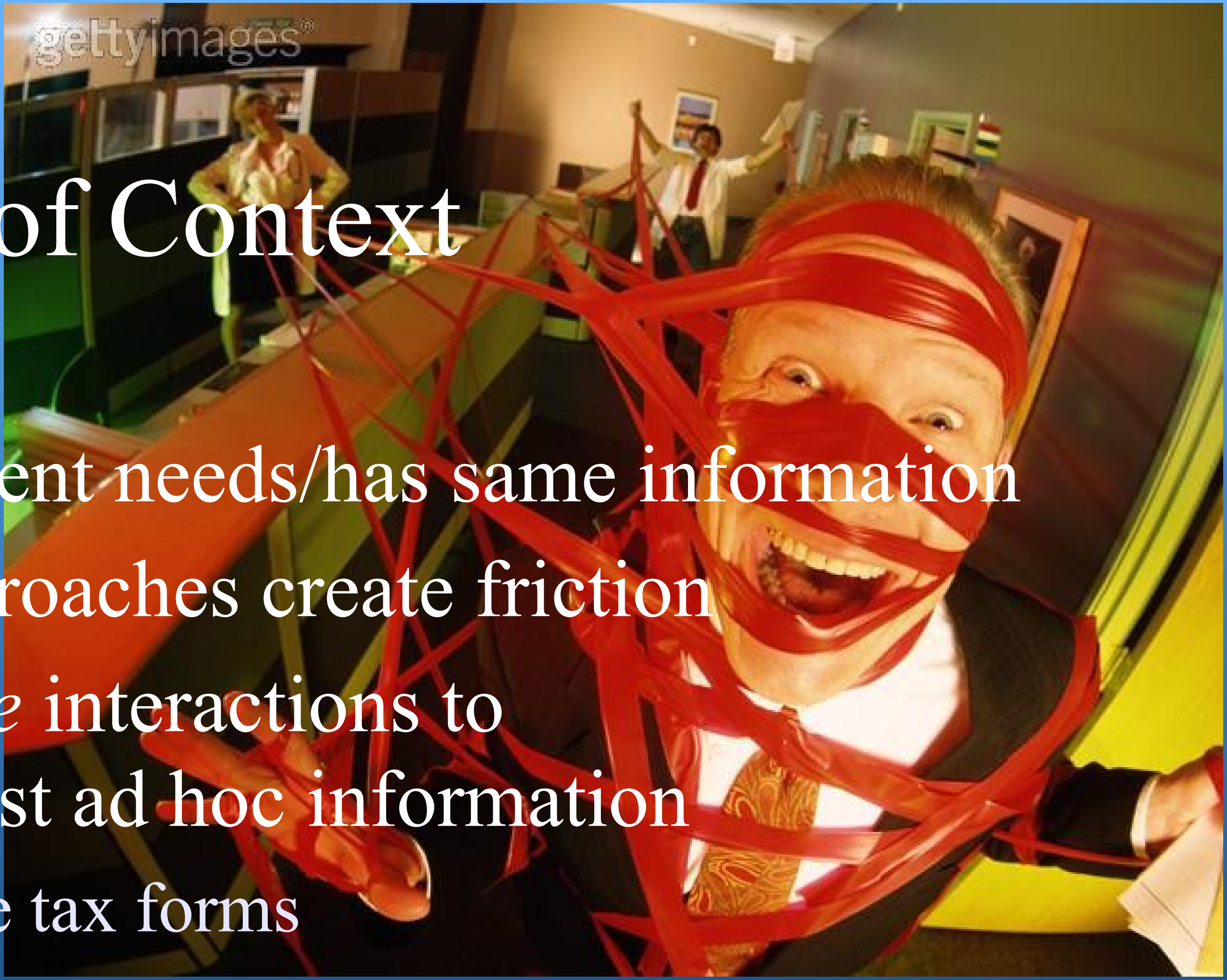
- Useful, but rigid
- Requisite variety
  - (Ashby, 1956)
- Requisite hierarchy
  - (Aulin, 1979;...)
- Hierarchies as networks
  - (Newman, 2003;...)

Adapting networks

- a)Hierarchy
- b)Add interactions
- c)Modules
- d)Shortcuts
- Small-world
- (Bollen & Heylighen)



The Role of Context



- Not every agent needs/has same information
- Uniform approaches create friction
- *Contextualize* interactions to provide/request ad hoc information
  - e.g. Personalize tax forms
- Automatically categorize co-ocurring contexts

A Toy Model: Random Agent Networks

- Inspired partly by random Boolean nets
  - (Kauffman, 1969; Gershenson, 2004;...)
- *N* nodes (agents) solving tasks
- Each with K_i dependencies, chosen randomly
- Task complete once requests from all dependencies are answered
- Dependencies keep tasks in FIFO queue
  - For simplicity, dependencies don't propagate

Random Agent Networks (2)

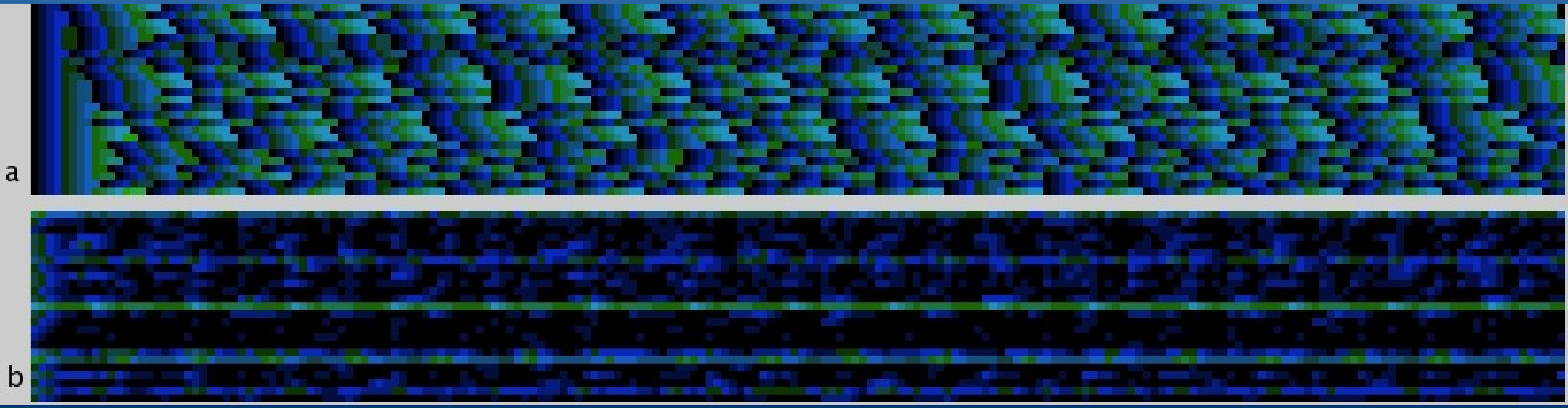
- Time abstracted: 1 timestep for:
  - Send requests to dependencies (transmission delay)
  - Answer 1 request from queue (decision delay)
  - Integrate requests and complete task (decision delay)
- Performance of net as #tasks completed
  - Minimize response delay and idling time (empty queues)
  - Balance tasks request and response
- Sequential updating (deterministic)

Topologies

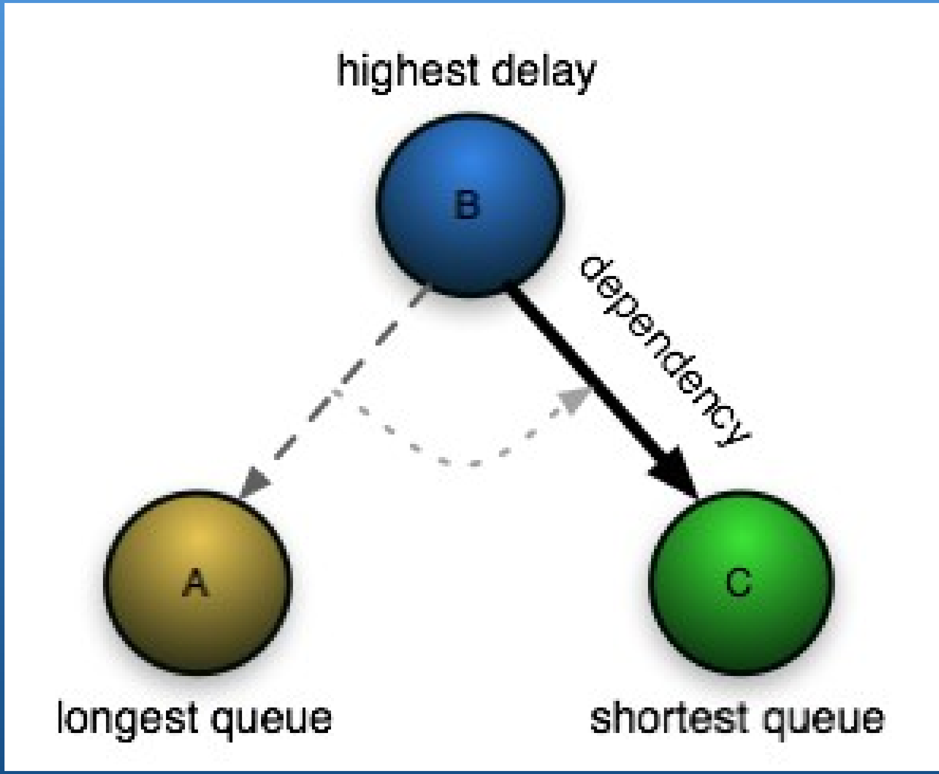
- Homogeneous
  - Every agent has *exactly* K (random) dependencies
- Normal
  - Each agent has K (random) dependencies *on average*
- Scale-free
  - Few with a lot, most with a few: $P(x)=(\gamma-1)x^{-\gamma}$
- Symmetric
  - Non-random, agent depends on K neighbours (CA-like)

RANLab

- http://rans.sourceforge.net
- e.g. N = 25, K = 5, homogeneous topology.
- a) Response delays. b) Queue lengths.
- Lighter colours indicate higher values.



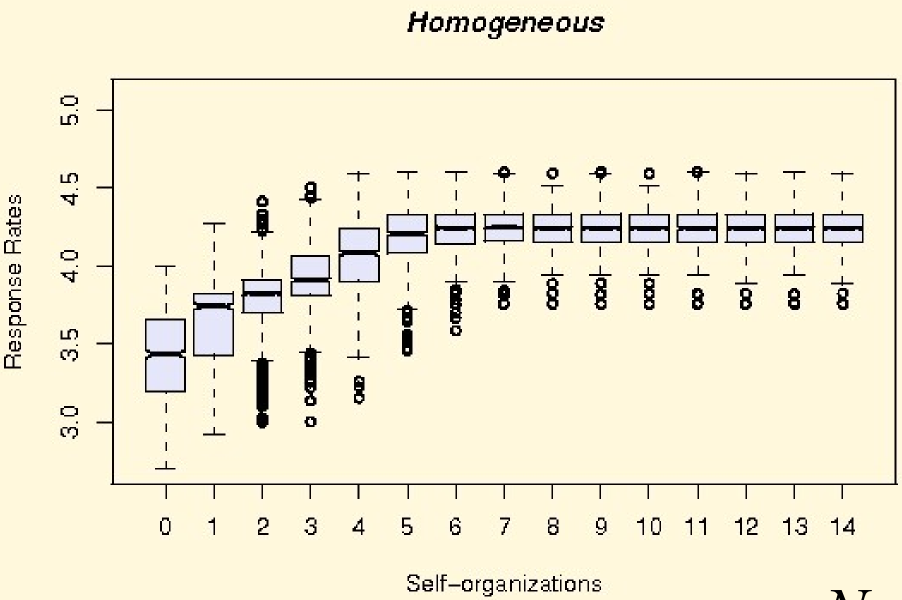
Self-organize!

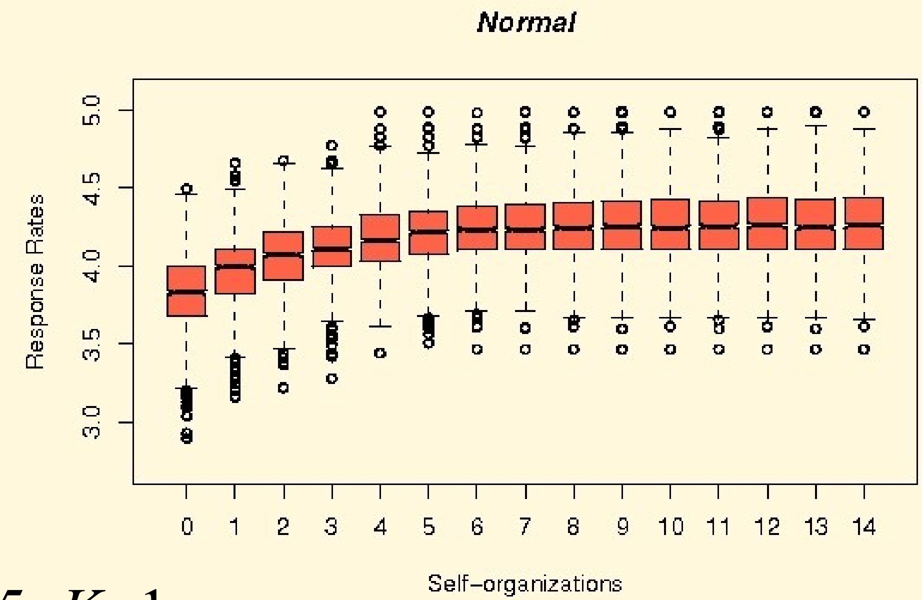


•Long queues=friction → try to reduce them

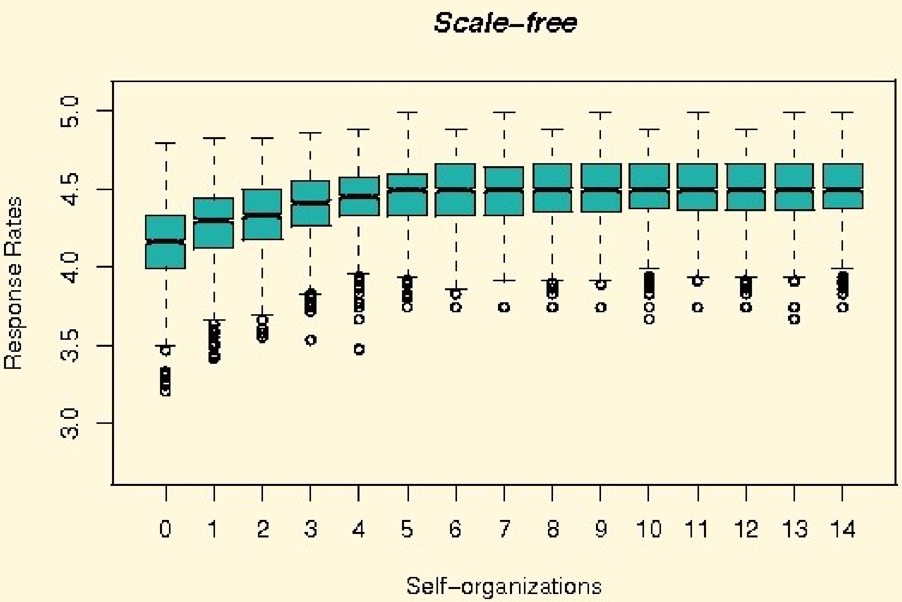
Simulation results

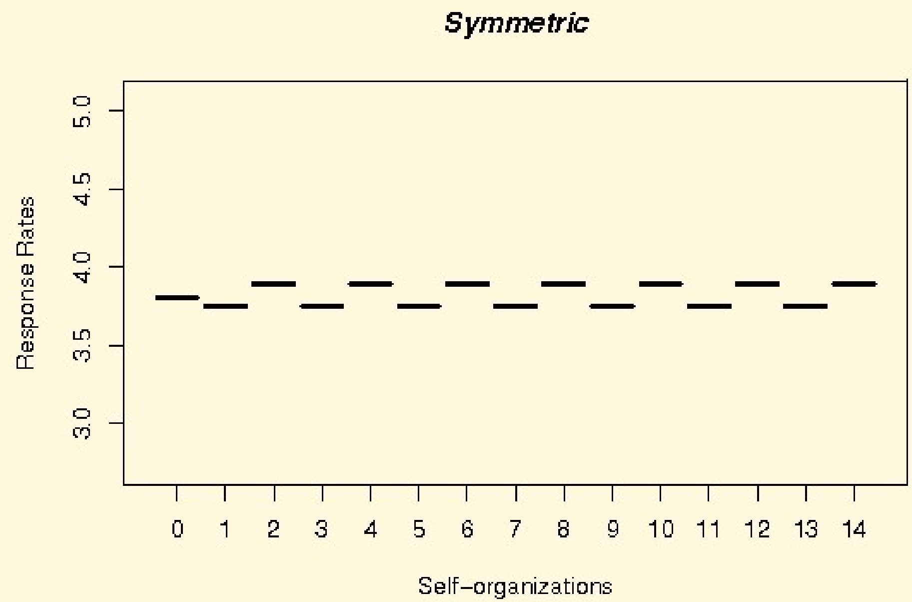
- Normalize topologies to have similar number of dependencies
- 1000 nets generated for each case & topology
- plot **response rate** (avg. tasks completed / timestep) each time self-organization is applied
  - (each 1000 timesteps)

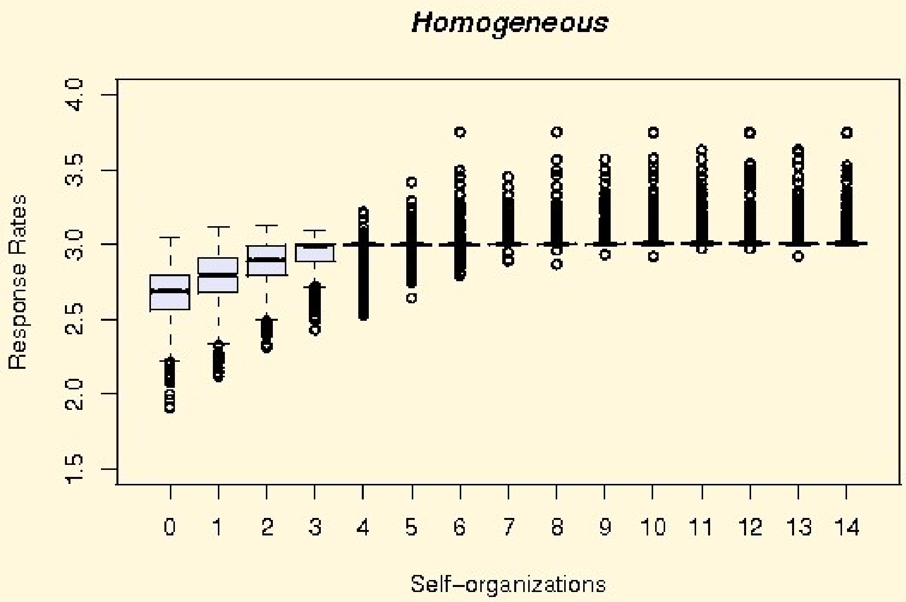

Homogeneous
Response Rates
Self–organizations

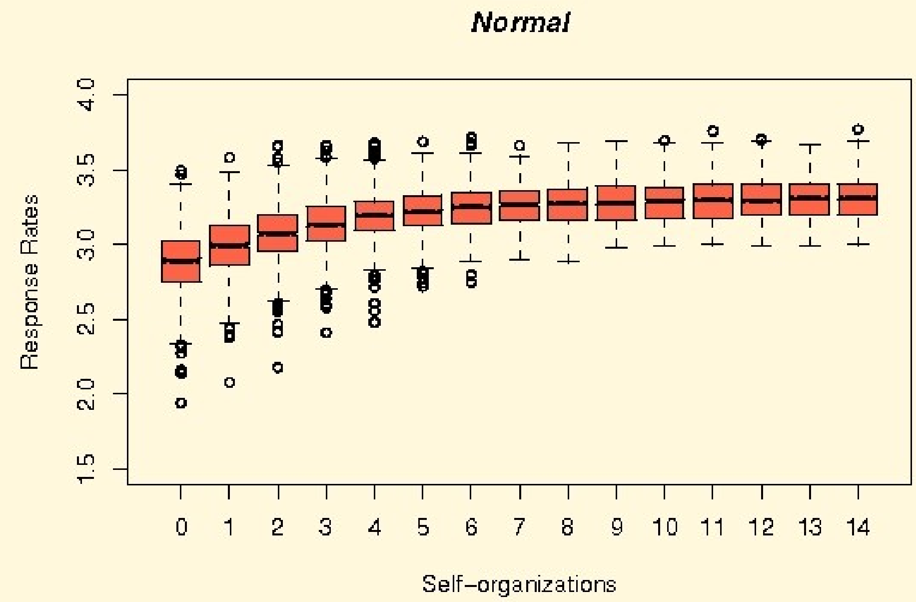

Normal
Response Rates
Self–organizations

N=15, *K*=1


Scale–free
Response Rates
Self–organizations

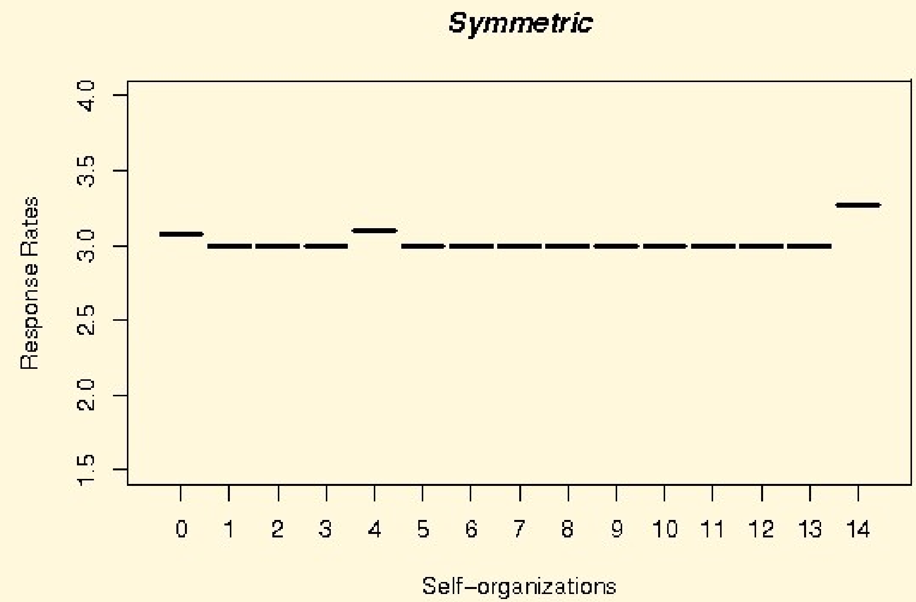

Symmetric
Response Rates
Self–organizations

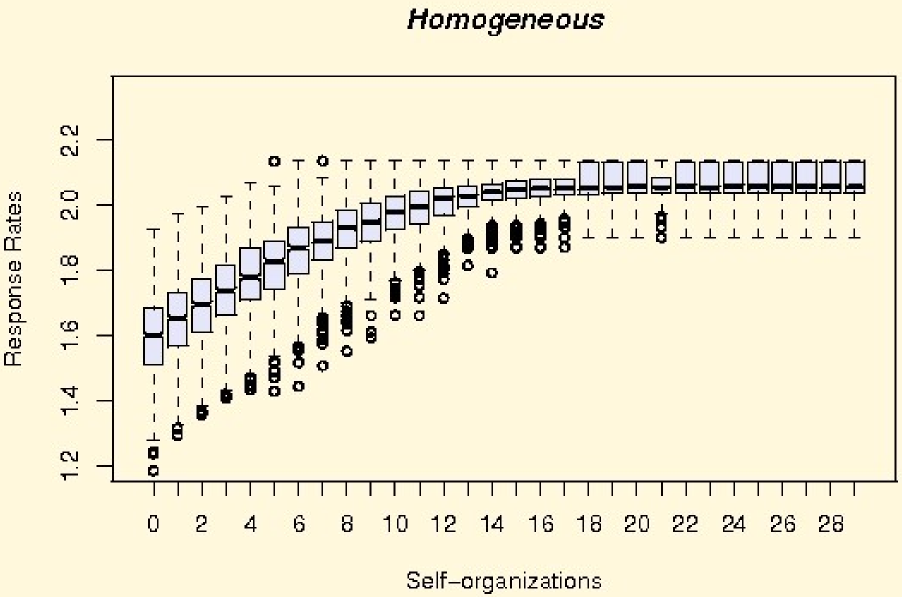

Homogeneous
Response Rates
1.5 2.0 2.5 3.0 3.5 4.0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
Self-organizations

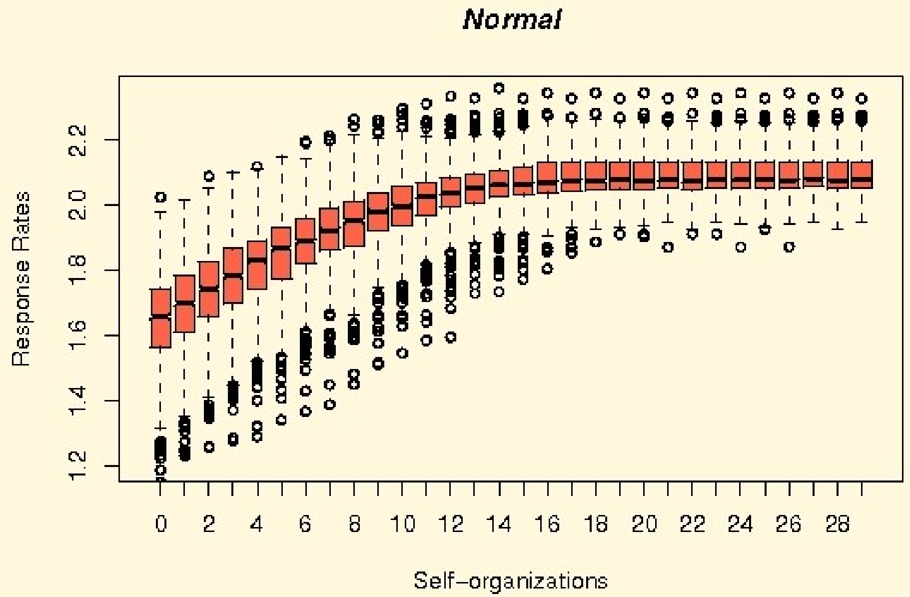

Normal
Response Rates
1.5 2.0 2.5 3.0 3.5 4.0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
Self-organizations

N=15, *K*=2

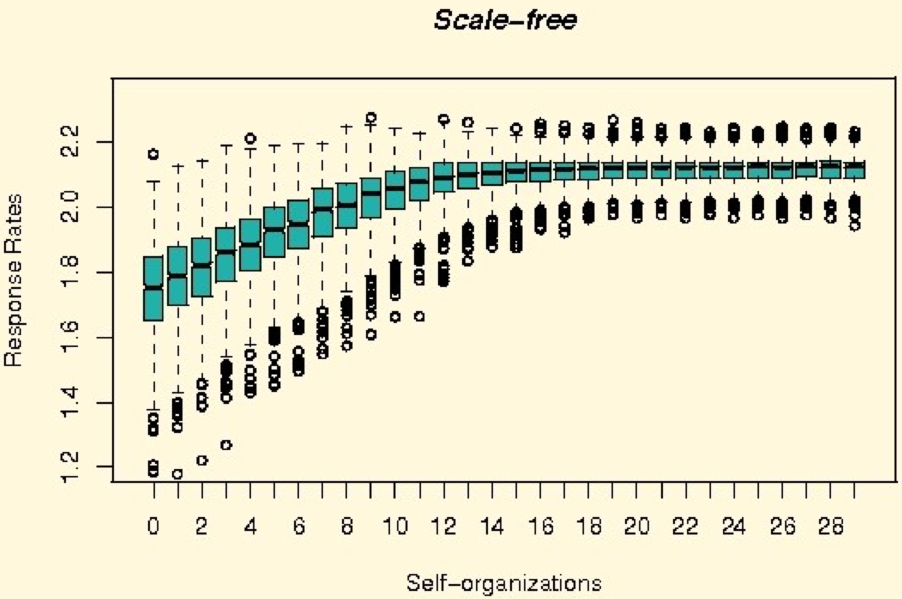

Scale-free
Response Rates
1.5 2.0 2.5 3.0 3.5 4.0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
Self-organizations

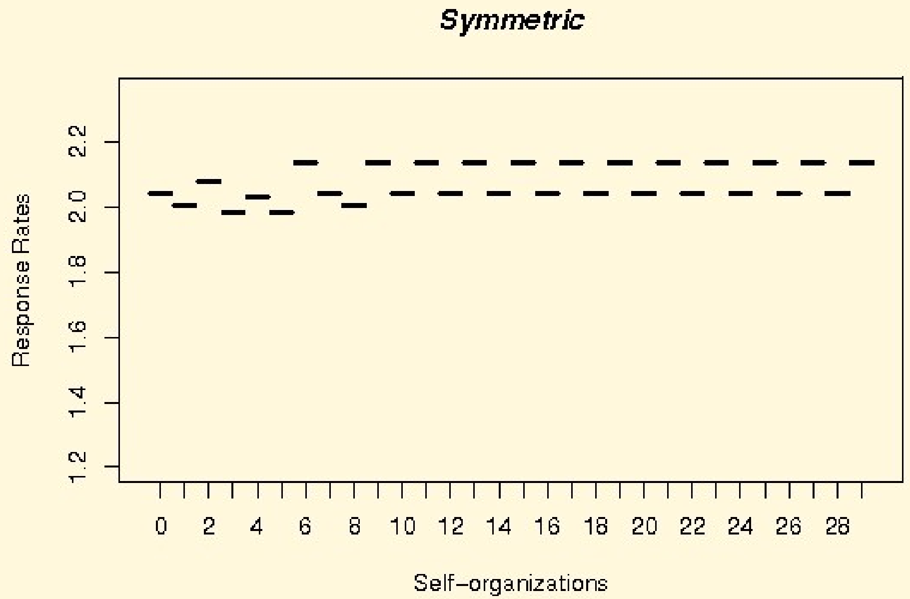

Symmetric
Response Rates
1.5 2.0 2.5 3.0 3.5 4.0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
Self-organizations

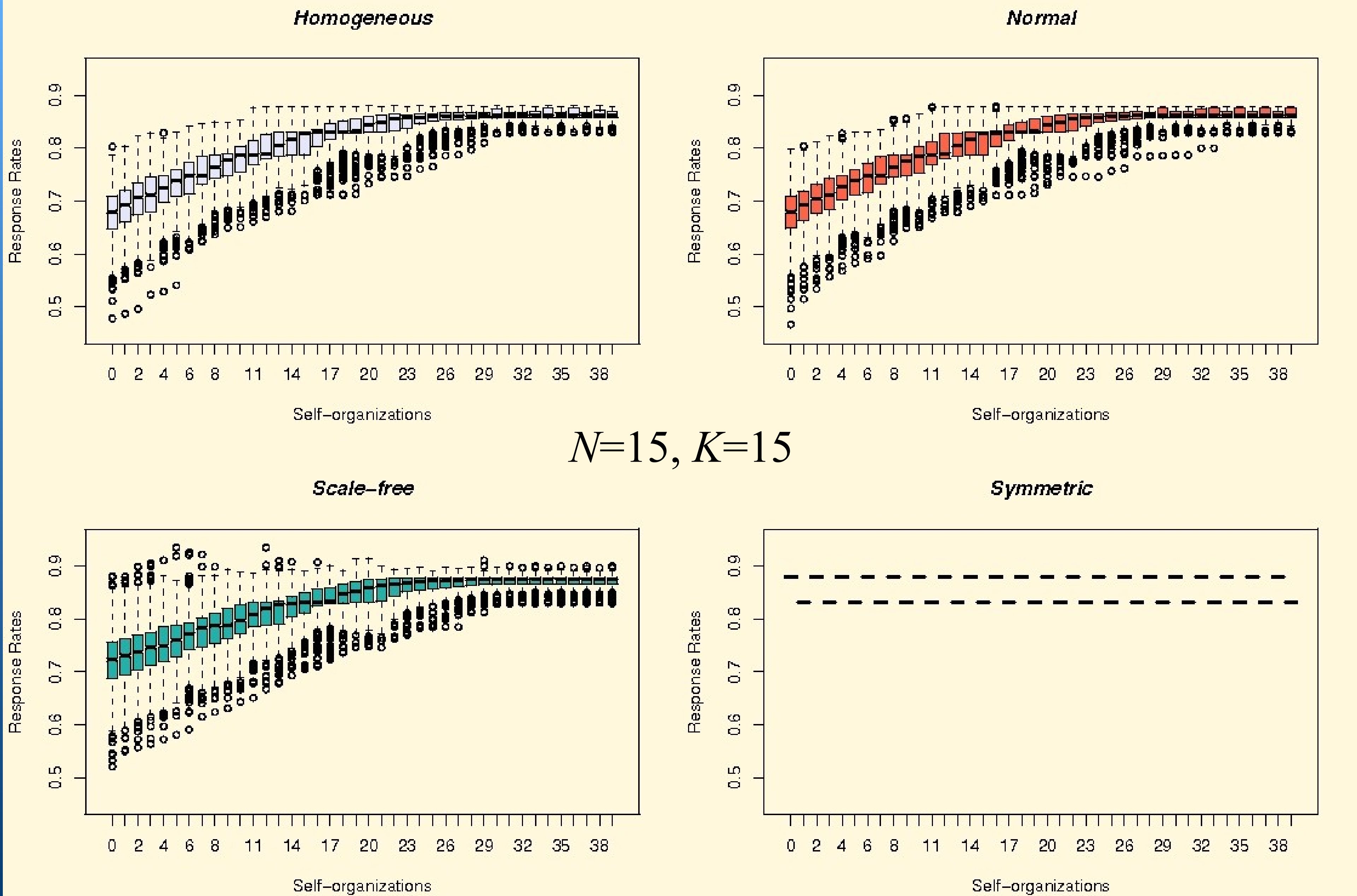

Homogeneous
Response Rates
Self-organizations

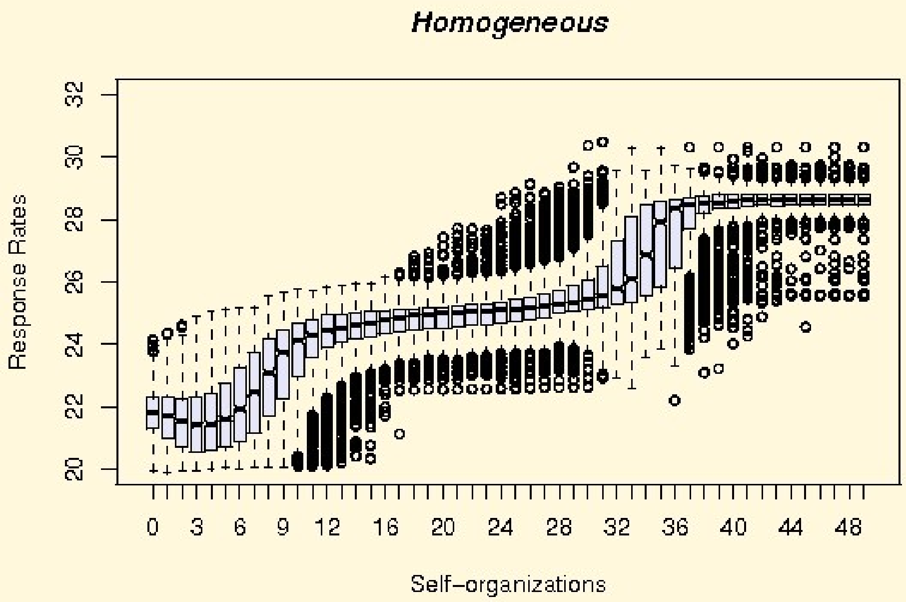

Normal
Response Rates
Self-organizations

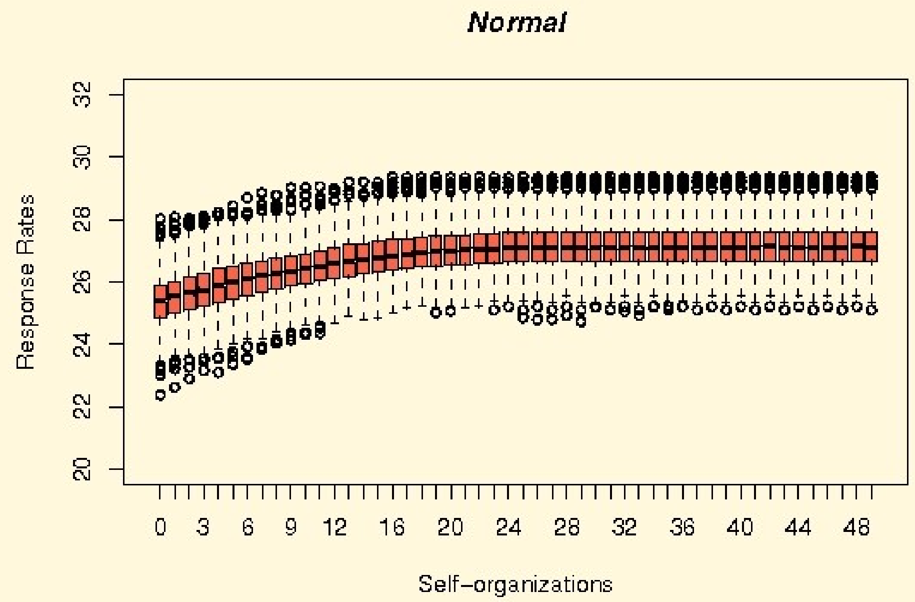
N=15, K=5


Scale-free
Response Rates
Self-organizations

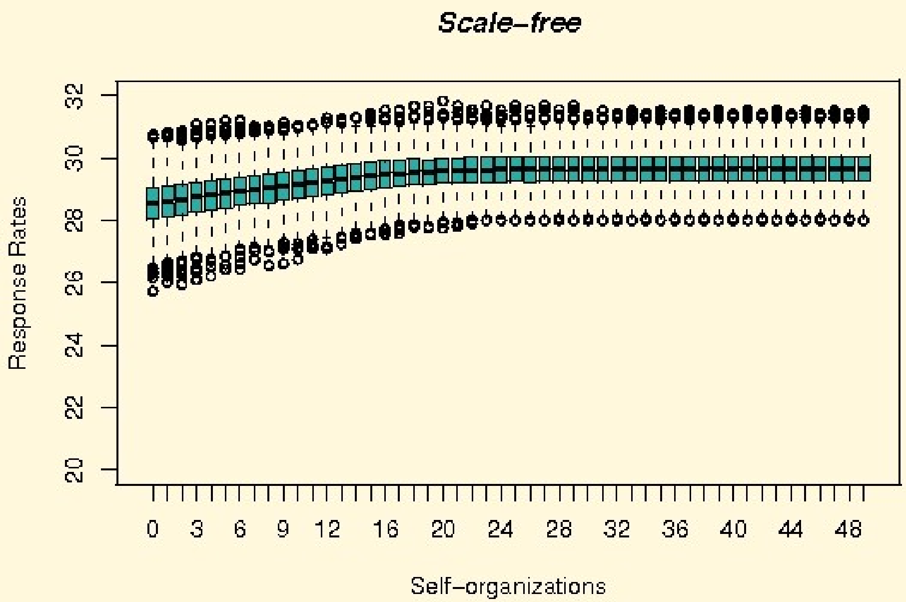

Symmetric
Response Rates
Self-organizations

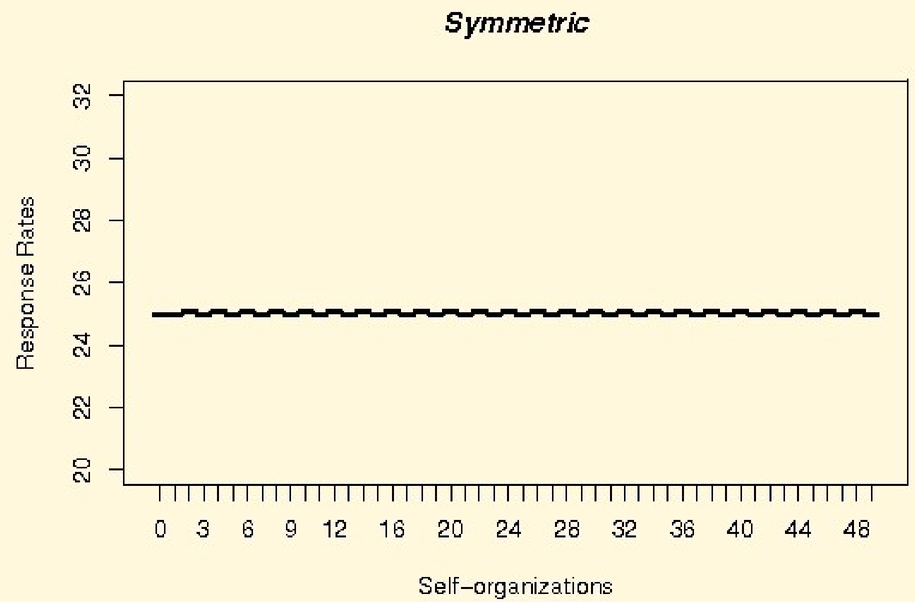

Homogeneous
Normal
Scale-free
Symmetric
N=15, K=15
Response Rates
Self-organizations
0.5
0.6
0.7
0.8
0.9
0 2 4 6 8 11 14 17 20 23 26 29 32 35 38

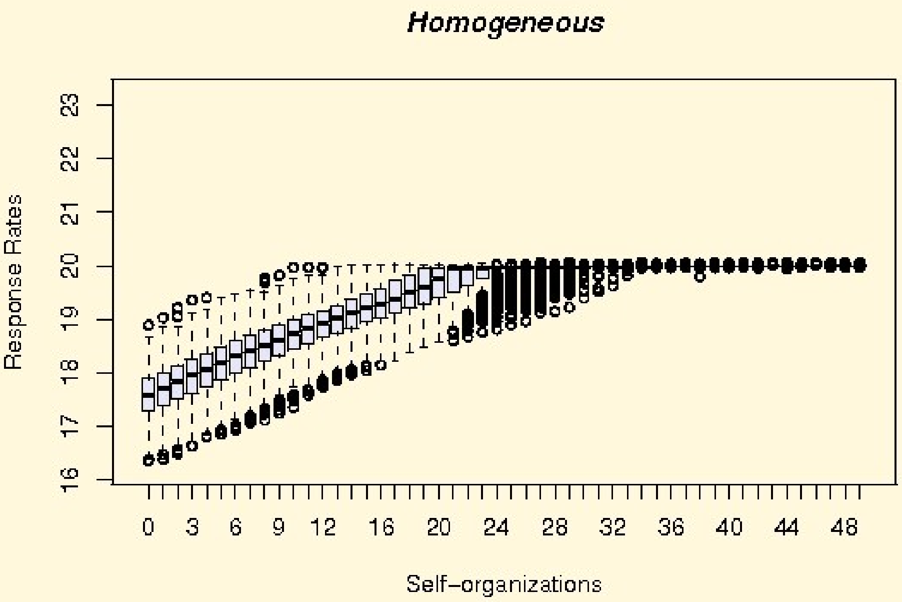

Homogeneous
Response Rates
20 22 24 26 28 30 32
0 3 6 9 12 16 20 24 28 32 36 40 44 48
Self–organizations


Normal
Response Rates
20 22 24 26 28 30 32
0 3 6 9 12 16 20 24 28 32 36 40 44 48
Self–organizations

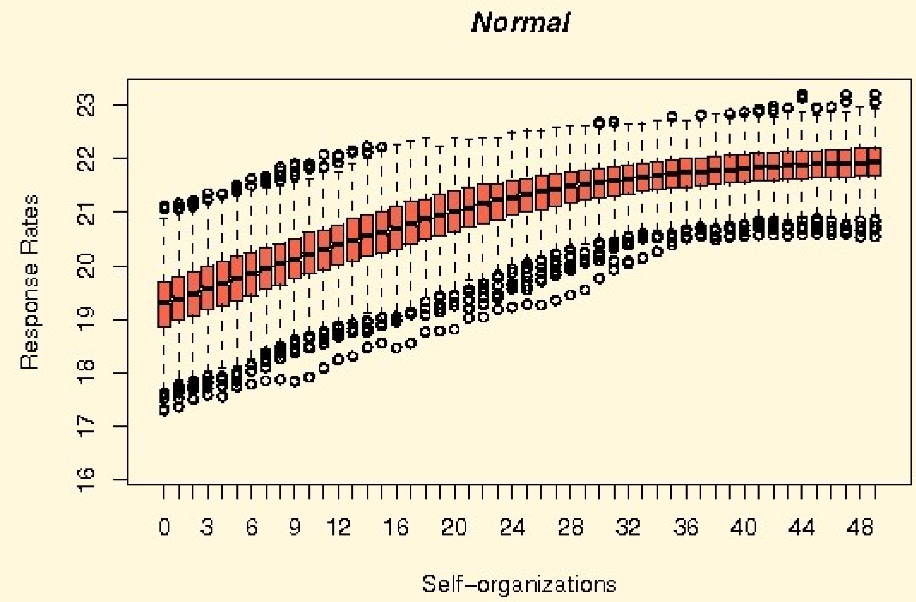
N=100, K=1


Scale–free
Response Rates
20 22 24 26 28 30 32
0 3 6 9 12 16 20 24 28 32 36 40 44 48
Self–organizations


Symmetric
Response Rates
20 22 24 26 28 30 32
0 3 6 9 12 16 20 24 28 32 36 40 44 48
Self–organizations

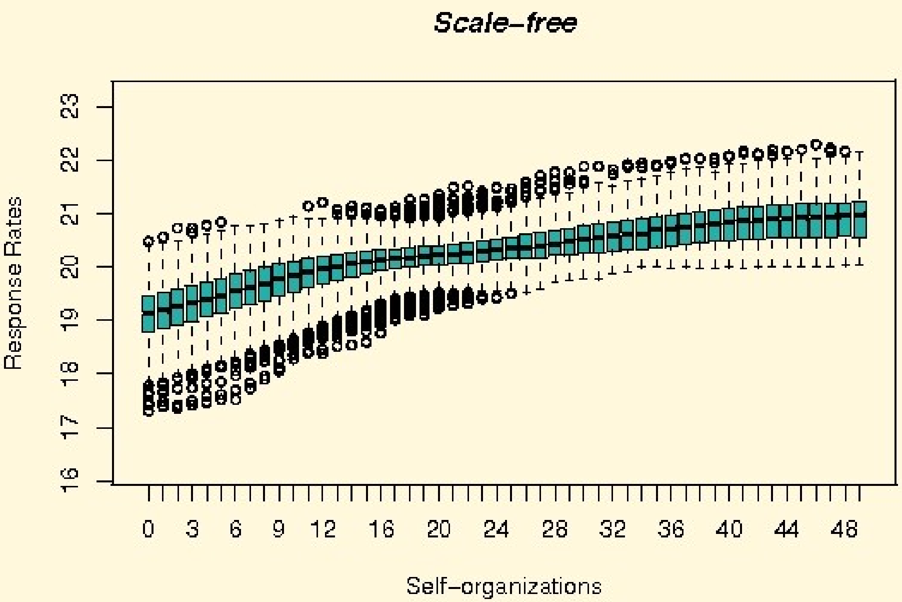


Homogeneous
Response Rates
Self-organizations

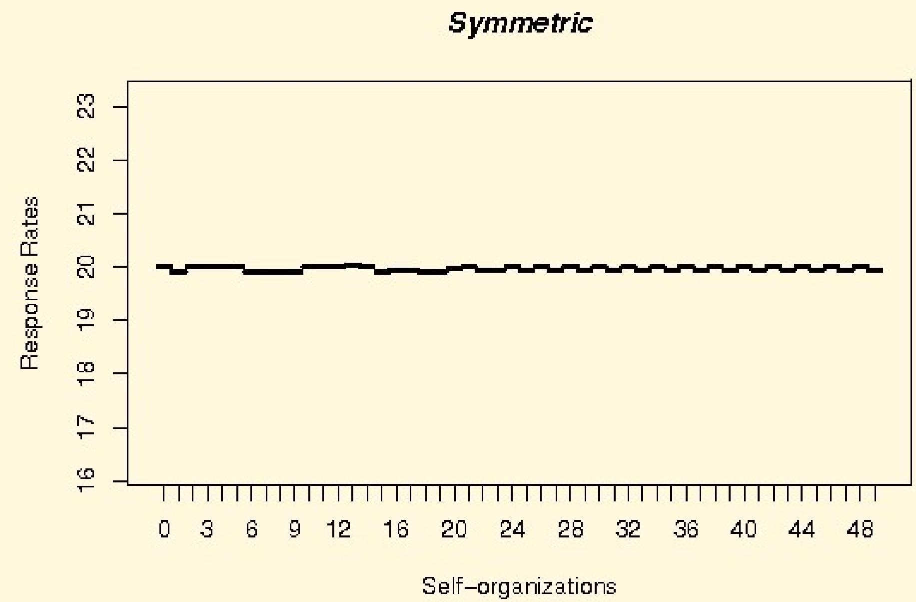


Normal
Response Rates
Self-organizations

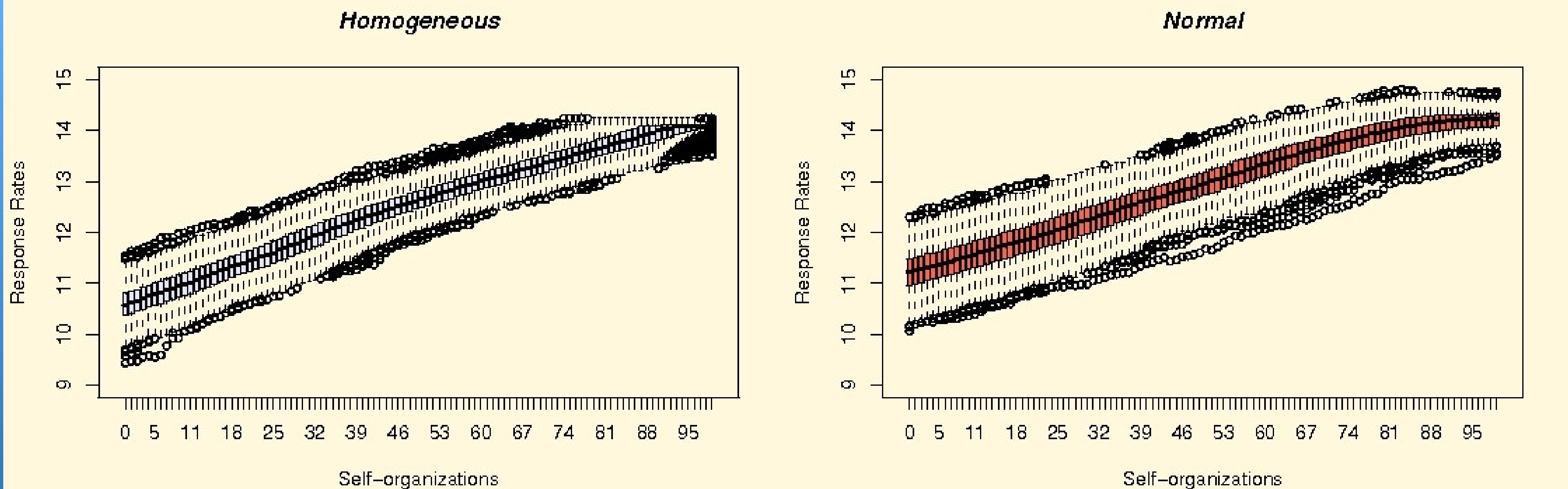
N=100, K=2



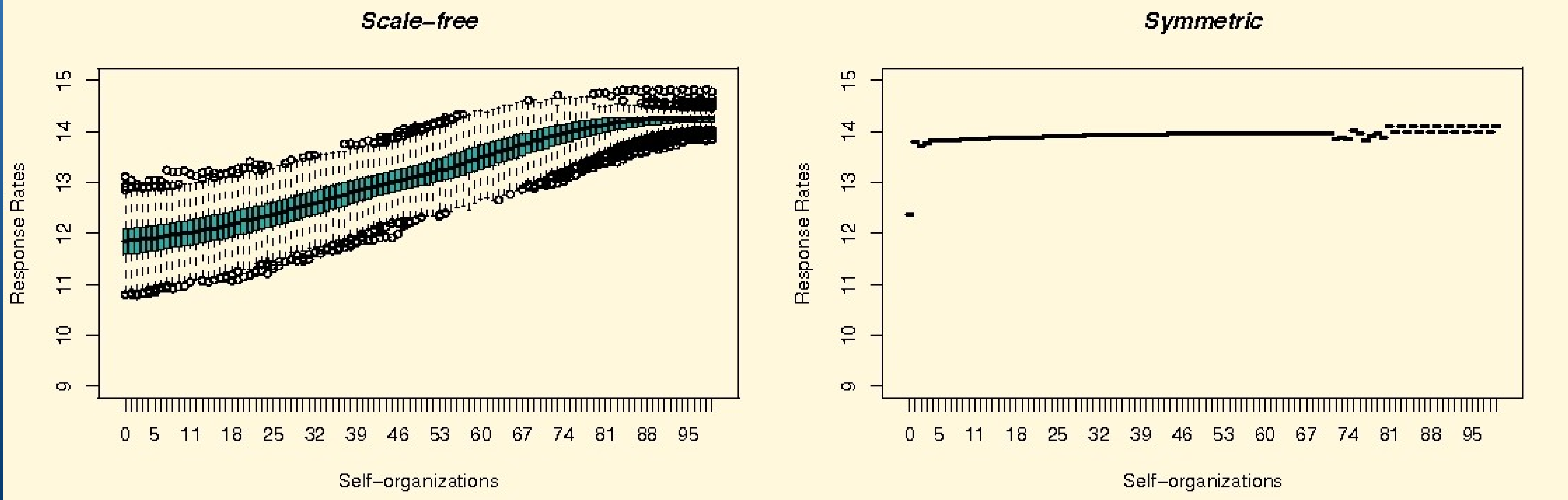
Scale-free
Response Rates
Self-organizations



Symmetric
Response Rates
Self-organizations

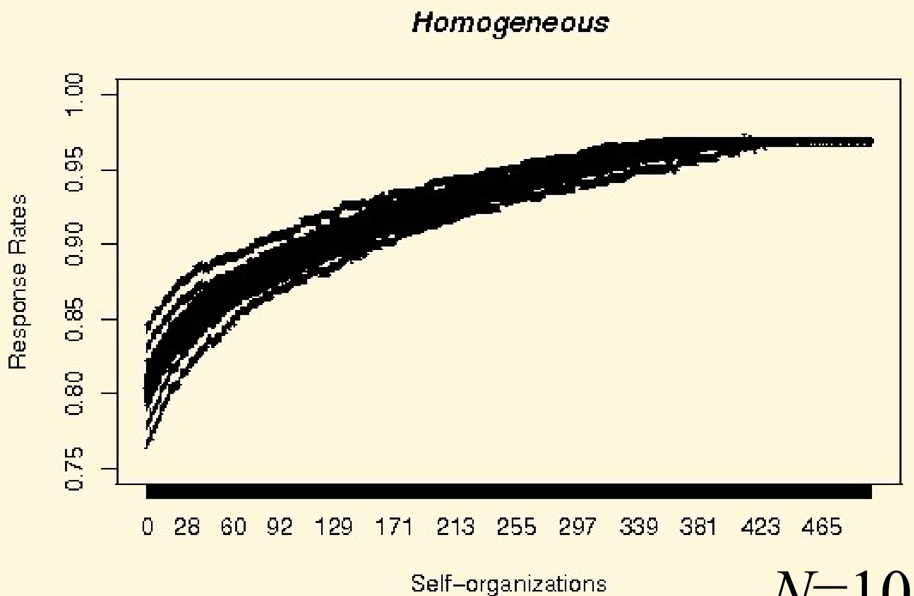

Homogeneous
Normal
Response Rates
Self−organizations
9
10
11
12
13
14
15
0 5 11 18 25 32 39 46 53 60 67 74 81 88 95

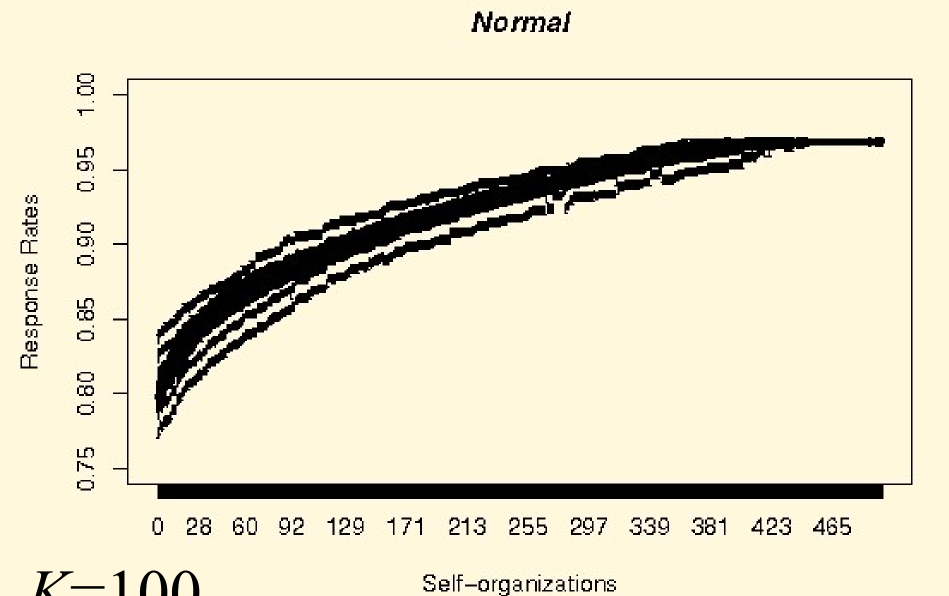
N=100, *K*=5

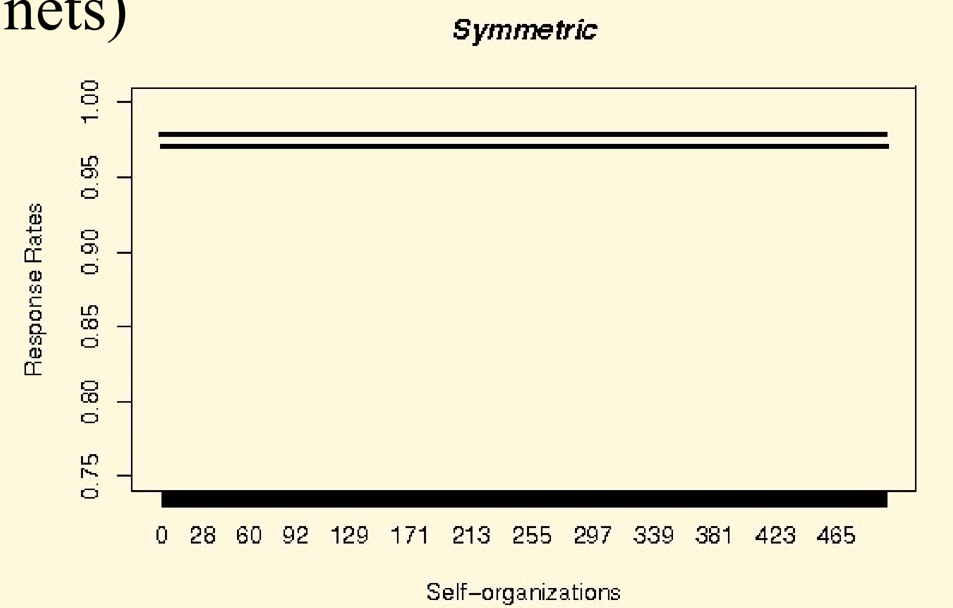

Scale−free
Symmetric
Response Rates
Self−organizations
9
10
11
12
13
14
15
0 5 11 18 25 32 39 46 53 60 67 74 81 88 95

N=100, K=100

(25 nets)







RAN Discussion

- Many open questions, but illustrates the benefit of self-organization in bureaucracies
- Showed that only few modifications on *random* networks lead to near optimality
  - Model doesn't take cost into account...
- RANs self-adaptive to changes in demands
- Weights can model diversity of delays

Future Work

- Study RAN robustness (damage of nodes)
- Phase transitions? (order/chaos)
- Refine model to make more realistic
  - e.g. Include costs
  - Domain expertise
- Possible implementation?

Conclusions

- Presented different ways in which self-organization can improve bureaucracy efficiency
  - Decrease delays → reduce friction
- Adaptability and robustness
  - Speed of reaction and decision will allow adaptation while preserving functionality
- Similar ideas could be used in different organizations