# Autonomy Oriented Computing (AOC) for Web Intelligence (WI): A Distributed Resource Optimization (DRO) Perspective

Xiaolong Jin

Principal-Supervisor: Prof. Jiming Liu

Co-supervisor: Prof. Yuan-Yan Tang

# Outline

- Background on WI and AOC, and problem statements
- DRO perspective on WI
- AOC mechanism and formulation for DRO
- DRO in homogeneous environments
- DRO in heterogeneous environments
- Conclusions and future work

# Background on WI and AOC, and problem statements

# Web Intelligence (WI)

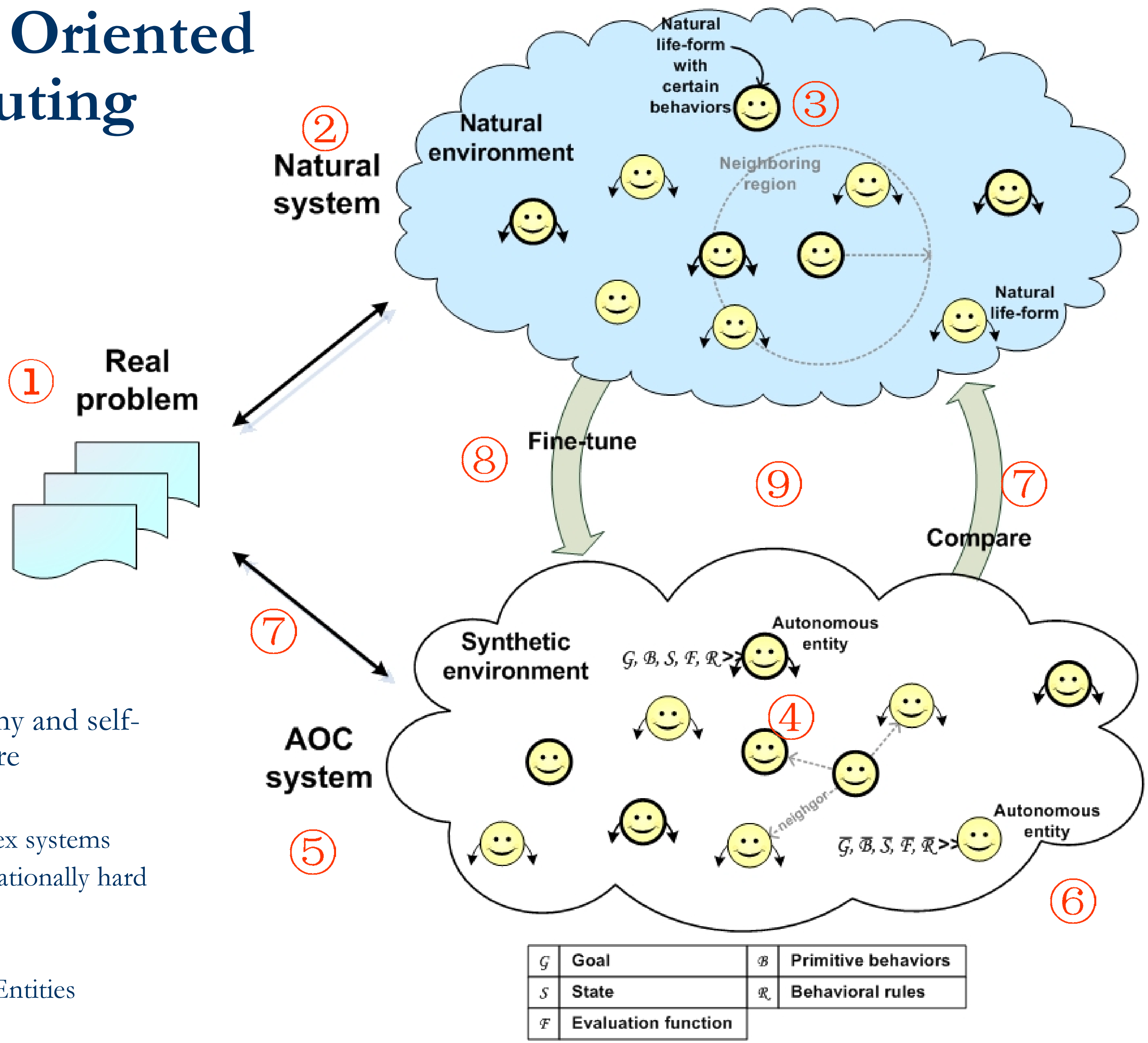
- Proposal: by **Zhong, Liu, Yao, and Ohsuga**; in COMPSAC 2000
- An informal definition:

  "Broadly speaking, Web Intelligence (WI) is a new direction for scientific research and development that explores the *fundamental roles* as well as *practical impacts* of *Artificial Intelligence (AI)* and *advanced Information Technology (IT)* on the next generation of *Web-empowered products, systems, services, and activities*. It is the key and the most urgent research field of IT in the era of Web and agent intelligence."
- The goal of WI: the **Wisdom Web**
  - Proposal: by **Liu, Zhong, Yao, and Ras**; in 2002
  - Purpose: enable human users to gain new practical *wisdom* of working, living, learning, and playing

# Web Intelligence (WI)(Cont.)

- Great challenges for WI [Liu, 2003]
  - Mobilizing distributed resources
    - Distributed Resource Optimization (DRO)
    - Web oriented computing paradigm
      - Large-scale
      - Pervasive and distributed
      - Dynamics and unreliable
      - …
    - …
  - Discovering the best means and ends
  - Enriching social interaction

# Autonomy Oriented Computing




- Proposal:
  by **Liu**, in 2001
- Inspiration: autonomy and self-organization in nature
- Purposes
  - To model complex systems
  - To solve computationally hard problems
- An AOC System:
  - Environment + Entities
  - Interactions
  - System objective function

Problem

# DRO

-Large-scale
-Distributed
-Unreliable
-…

match?

Methodology

# AOC

-Large-scale
-Distributed
-Locally interacting
-…

# Problem Statements

- Problems
  - **Q-1:** What resources to be optimized? What requirements?
  - **Q-2:** How to generalize specific DRO issues?
  - **Q-3:** How to provide an AOC-based computing paradigm for the generalized DRO?
  - **Q-4:** How to refine and validate the above paradigm according to the features of different DRO environments?
    - **Q-4-A:** Homogeneous DRO environments
    - **Q-4-B:** Heterogeneous DRO environments
- Assumptions
  - Features considered: large-scale, distributed, and unreliable
  - Not considered: security, privacy, interoperability, and transportation cost

# DRO perspective on WI

# Q-1: DRO Perspective

- Generalized view of resources
  - Resources → different contents with different functions or utilities
  - Physical or logical
- DRO at four WI levels
  - Internet level
    - E.g. CPU time & processing speed, network width, computers
  - Interface level
    - E.g. Portals
  - Knowledge level
    - E.g. distributed data/knowledge bases
  - Application level
    - E.g. various high-level Web services

# Q-2: A Generalized DRO Scenario

- Assumptions
  - One service request (SR) : one resource
  - No dependency relationship
  - SRs ~ certain distribution
- Generalized scenario
  - Resources
    - DRO environment = resource nodes (RNs) + links
    - Linked resources : *neighbors*
    - Service providers
    - Homogeneous / heterogeneous
    - Unreliable
    - Capacity
  - Service requests
    - Demands to resources
    - Submitted in a distributed fashion
    - No centralized mechanism for distributing service requests
    - Homogeneous / heterogeneous
- Resource optimization
  - How to distribute SRs among different RNs to achieve:
    - Approximate load balancing
    - Approximately optimized resource utilization
    - BTW, reducing the response time for individual SR

# Q-1: WI Requirements on DRO

- *Semantic* : semantic match
- *Correct* : correct match
- *Distributed* : not centralized
- *Optimized* : optimal or sub-optimal utilization
- *Global* : global scale
- *Online* : real-time decision making
- *Robust* : robust to failures and recovery
- *Adaptive* : adaptation to real-time changes (e.g., insertion and remove of resource nodes)
- *Autonomic* : operate by itself

# Q-3:
# AOC mechanism & formulation for DRO

# AOC-Based DRO Mechanism

- Agents carry service requests to search:
  - Idle resource nodes to form new agent teams
  - Existing agent teams to join
- Agents prefer to join agent teams with less load
- When queuing, agents can choose to
  - Remain at current agent teams, or
  - Leave current teams and wander to other resource nodes
- Agents must be served by certain resource nodes
- Agents automatically disappear after being served
- Agents has local information
- Agents indirectly interact via the environment

# AOC Formulation

- Resource environment
  - Environment $E$: $\langle V, L\rangle$
    - $V=\langle rn_1, \cdots, rn_i, \cdots, rn_N\rangle$ : resource nodes
    - $L=\langle l_1, \cdots, l_i, \cdots, l_K\rangle$ : links
  - Resource node $rn$:
    - $si$ : service vector
    - $ps$ : processing speed vector
    - $qts$ : the size of agent team, $qat$, at node $rn$
- Agents
  - State $S$:
    - $wq$ : wandering or queuing
    - $vr$ : radius of the vision range
    - $pos$ : position
    - $rq$ : service request, its size
    - $rs$ : service required
    - $rt$ : response time

- Neighboring region $NR$ : $\langle V', L'\rangle$
  - Resource nodes and links in the vision range
- Neighbors
  - Agents in the neighboring region
- Evaluation functions: $F=\{f_s, f_l\}$
  - $f_s$ : returning the wandering or queuing state
  - $f_l$ : evaluating the load of a resource node $rn$

$$f_l(rn) = f_l(rn.qat) = \sum_{a \in rn.qat} a.rq, \quad \text{or}$$

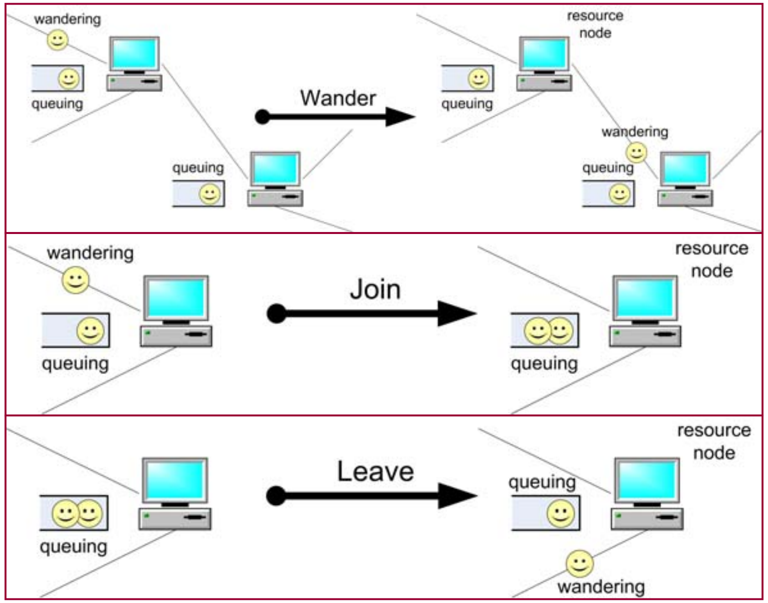
$$f_l(rn) = f_l(rn.qat, rn.ps) = \sum_{a \in rn.qat} \frac{a.rq}{rn.ps_{(a.rs)}}$$

- Goal

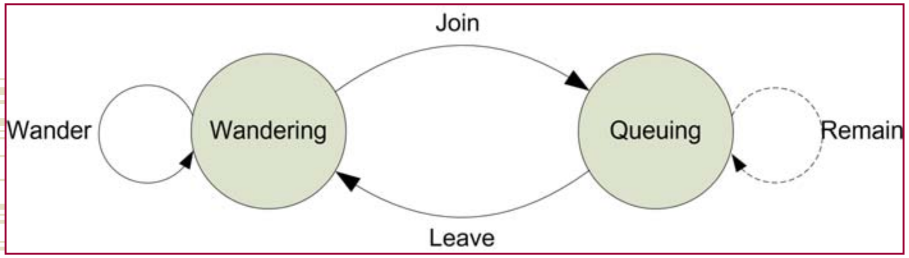
$$g : a.pos = u, \ \text{where } u = \arg\min_{rn \in V} (f_l(rn))$$

# Primitive Behaviors and Behavioral Rules

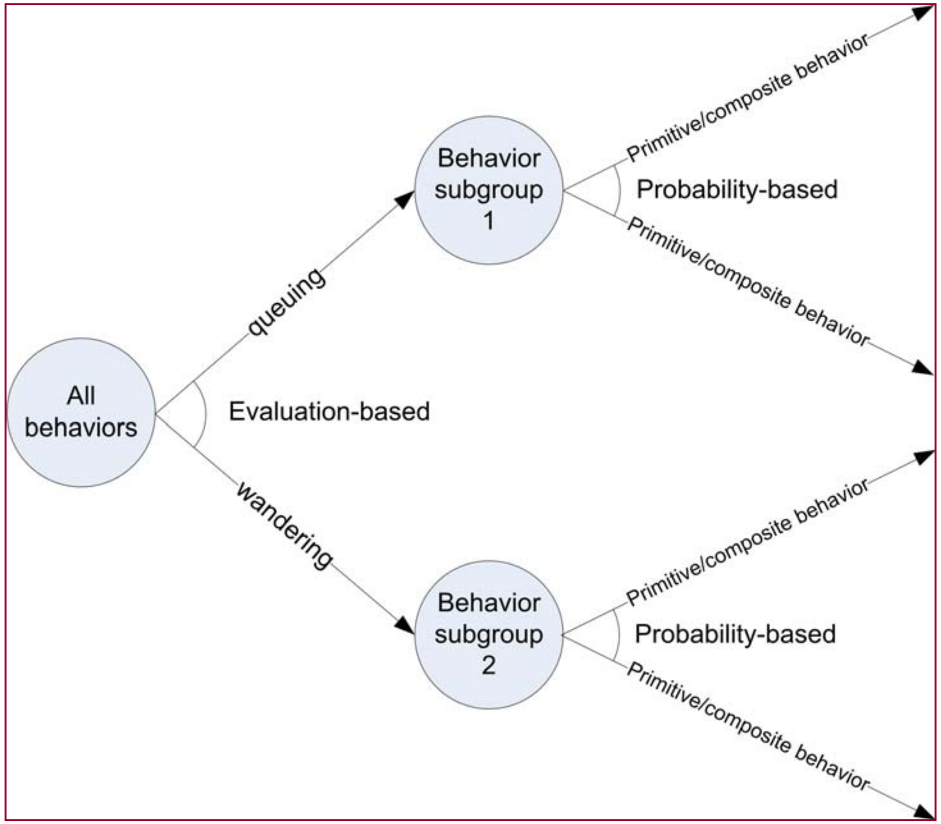
- Primitive behaviors: $B=\{remain, wander, join, leave\}$



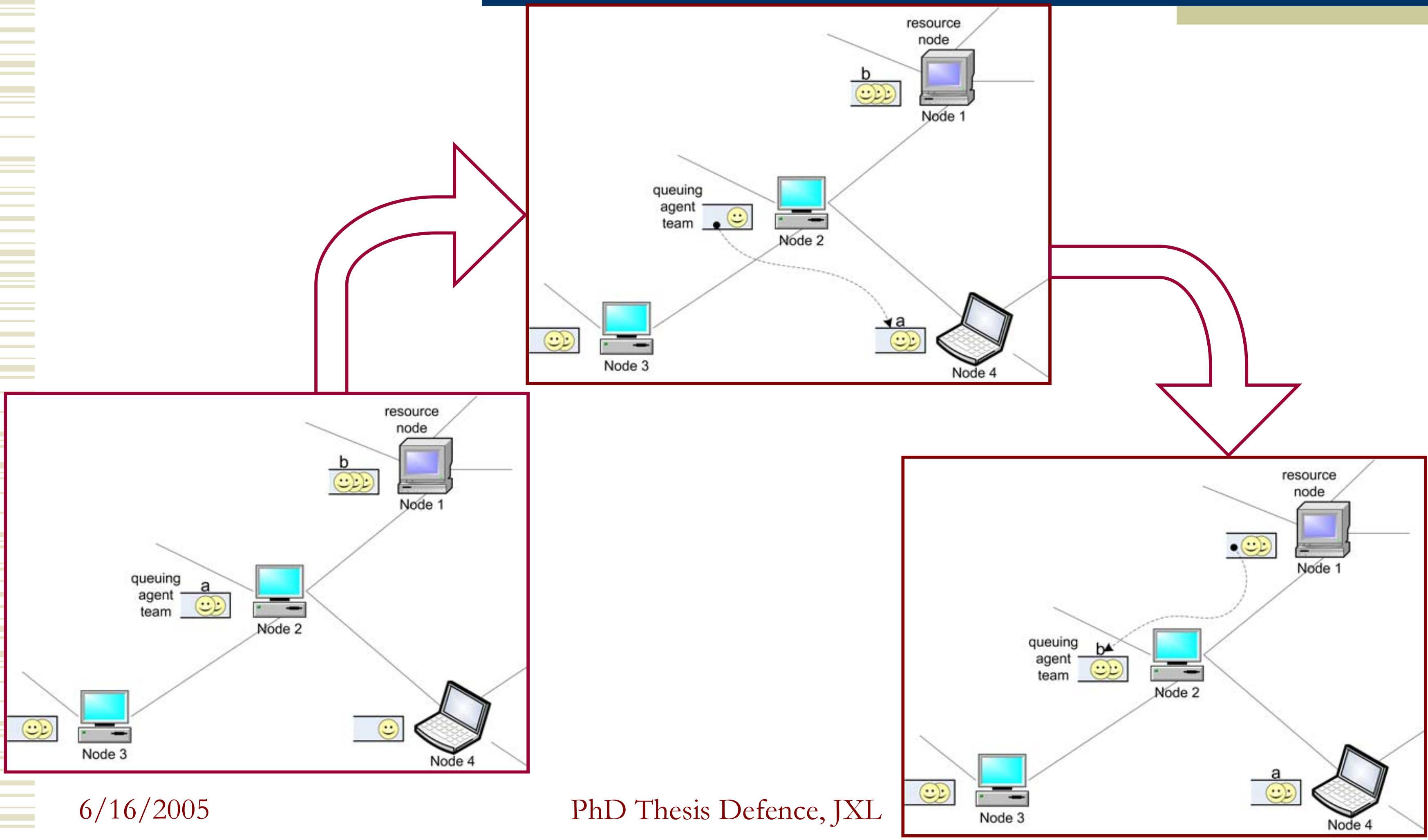

state transition




- Behavioral rules: $R=\{r_h, r_l\}$
  - $r_h$ : the high level, evaluation-based rule
  - $r_l$ : the low level, probability-based rule



# Indirect Interactions



# System Objective Function

$$\Phi(\mathbf{V},\mathbf{A}) = std_{rl} = \sqrt{\frac{\sum_{j=1}^{N}(rn_j.rl - \frac{\sum_{i=1}^{N} rn_i.rl}{N})^2}{N}}$$

- $\boldsymbol{V}=\{rn_1,\cdots,rn_i,\cdots,rn_N\}$ : the set of resource nodes
- $\boldsymbol{A}=\{rn_1.qat,\ \cdots,rn_i.qat,\cdots,rn_N.qat\}$: the set of queuing agent teams at resource nodes in $\boldsymbol{V}$
- $rn_i.rl = \sum_{a \in rn_i.qat} \frac{a.rq}{rn_i.ps_{(a.rs)}}$

# Q-4-A:
# DRO in homogeneous environments

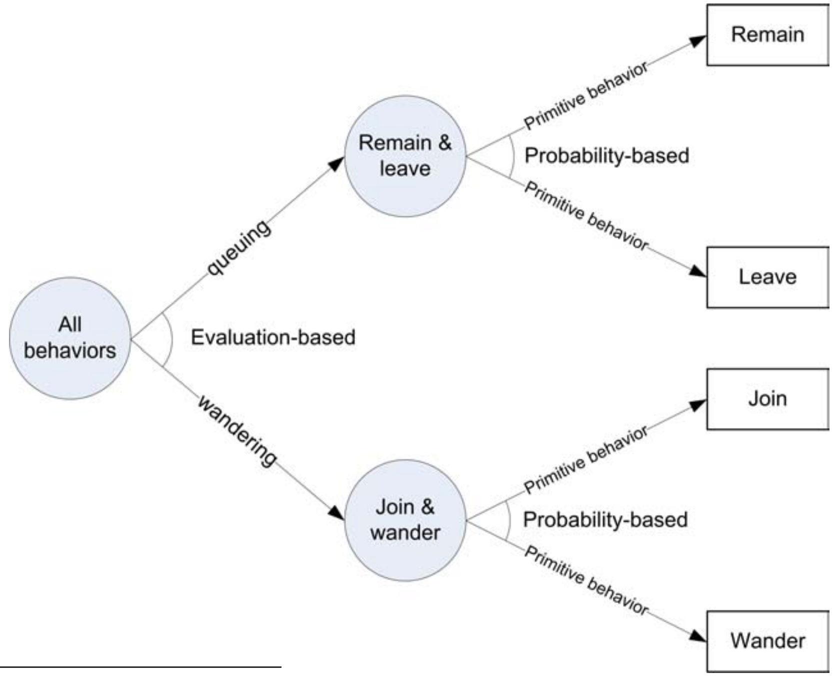
# Refined DRO Mechanism

- Homogeneous environments
  - Homogeneous RNs : same service + same processing speed
  - Homogeneous SRs : same required service + same size
- Refined DRO mechanism
  - The size of an agent team = Service request load
  - Agents' decision-making based on
    - Probabilities
    - The size of agent teams encountered
  - Agents' preference: relatively small agent teams
  - Resource nodes' capability : A maximum size for agent teams

- At each step, how many wandering agents join agent teams of a certain size depends on:
  - The total number of currently wandering agents
  - The numbers of currently existing agent teams of various sizes
- At each step, how many queuing agents leave teams of a certain size depends on:
  - The total number of currently existing teams of this size
- In the above sense, the proposed mechanism is adaptive

# Refined AOC Formulation

- Homogeneous environments
  - Relative load of RN: $rn.rl = rn.qts$
  - The maximum team size $m \Leftrightarrow$ the capacity of RNs:
    - $rn.cp = m$
    - $rn.rl \leq m$
- Agents
  - State:
    - Two vectors: $\langle \zeta_0(t), \cdots, \zeta_s(t), \cdots, \zeta_{m-1}(t) \rangle$ and $\langle \iota_2(t), \cdots, \iota_s(t), \cdots, \iota_m(t) \rangle$
      - $\zeta_s(t)$: probability for joining teams of size $s$
      - $\iota_s(t)$: probability for leaving teams of size $s$
      - $\zeta_s(t)$ and $\iota_s(t)$ are fixed
    - Vision range $a.vr=1$
  - Evaluation function: $f_l(rn)=rn.qts$
  - Behavioral rule:




- System objective function: $\Phi(\mathbf{V}, \mathbf{A}) = \sqrt{\frac{\sum_{j=1}^{N}(rn_j.qts - \frac{\sum_{i=1}^{N} rn_i.qts}{N})^2}{N}}$

# Performance Studies

- Instantaneous DRO Scenario
  - A small time interval → no new service request + no handled service request
  - **I-1.** Can the mechanism achieve a steady state where load is balanced?
  - **I-2.** Is the mechanism robust to tolerate the dynamic changes in the environment and adapt their outcome?
- Ongoing DRO Scenario
  - A long time interval→ new service requests + handled service requests
  - **I-3.** How does the arrival speed of service requests affect the performance? How to determine an appropriate arrival speed?
  - **I-4.** Is the mechanism robust to tolerate the dynamic changes in the environment and adapt their outcome?

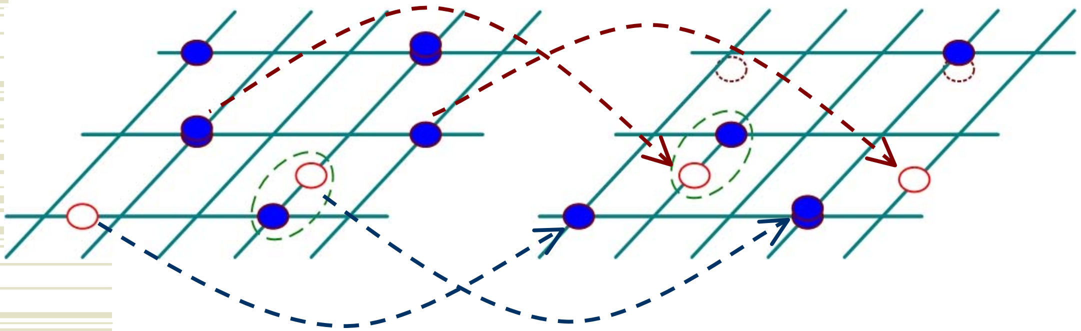
# AOC-based Instantaneous DRO Model

The **quantitative changes** of:

**Agents teams of size one:**

$$\frac{dq_1(t)}{dt} = j_0 w(t) - j_1 w(t) + l_2 q_2(t)$$

- Wandering agents join idle nodes or existing agent teams
- Queuing agents leave existing agent teams



**Agents teams of size two:**

$$\frac{dq_2(t)}{dt} = j_1 w(t) - l_2 q_2(t)$$

**Wandering agents:**

$$\frac{dw(t)}{dt} = l_2 q_2(t) - \sum_{s=0}^{1} j_s w(t)$$

**Idle resource nodes:**

$$\frac{dq_0(t)}{dt} = -j_0 w(t)$$

$$\frac{dq_1(t)}{dt} = j_0 w(t) - j_1 w(t) + l_2 q_2(t)$$

$$\frac{dq_s(t)}{dt} = j_{s-1} w(t) - j_s w(t) + l_{s+1} q_{s+1}(t) - l_s q_s(t)$$

$$\frac{dq_m(t)}{dt} = j_{m-1} w(t) - l_m q_m(t)$$

$$\frac{dq_0(t)}{dt} = -j_0 w(t)$$

$$\frac{dw(t)}{dt} = \sum_{s=2}^{m} l_s q_s(t) - \sum_{s=0}^{m-1} j_s w(t)$$

$$j_s(t) = \begin{cases} \frac{q_s(t)}{w(t)}, & \text{if } \bar{j}_s(t) - \frac{q_s(t)}{w(t)} \geq 0, \\ \frac{q_s(t)}{w(t)}, & \text{if } \frac{\Phi(t)}{\Psi(t)} \geq 1, \\ \bar{j}_s(t) + \frac{\Phi(t)}{\Psi(t)}\left(\frac{q_s(t)}{w(t)} - \bar{j}_s(t)\right), & \text{otherwise,} \end{cases}$$

where

$$\bar{j}_s(t) = \frac{j_s^p \cdot j_s^d(t)}{\sum_{i=0}^{m-1}(j_i^p \cdot j_i^d(t))} o$$

$$\Phi(t) = \sum_{i=0}^{m-1} sgn\left(\bar{j}_s(t) - \frac{q_s(t)}{w(t)}\right)\left(\bar{j}_s(t) - \frac{q_s(t)}{w(t)}\right),$$

$$\Psi(t) = \sum_{i=0}^{m-1} sgn\left(\frac{q_s(t)}{w(t)} - \bar{j}_s(t)\right)\left(\frac{q_s(t)}{w(t)} - \bar{j}_s(t)\right),$$

$$sgn(x) = \begin{cases} 1, & \text{if } x > 0, \\ 0, & \text{otherwise.} \end{cases}$$

$$l_s(t) = \begin{cases} \bar{l}_s(t), & \text{if } N = 0, \\ \frac{\bar{l}_s(t)}{N+1}, & \text{otherwise,} \end{cases}$$

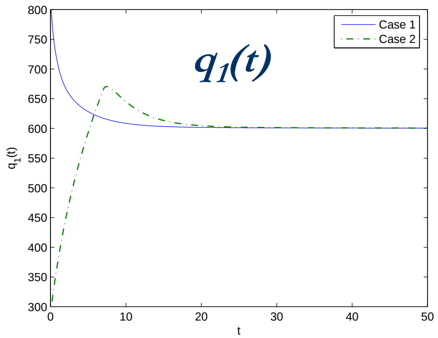
where

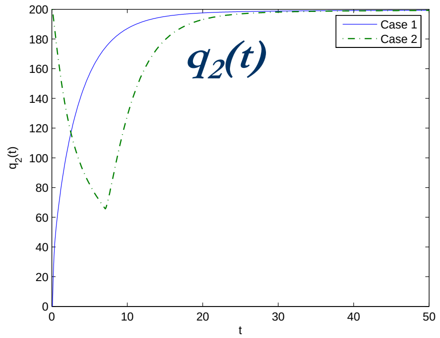
$$\bar{l}_s(t) = \frac{l_s^p \cdot l_s^d(t)}{\sum_{i=2}^{m}(l_i^p \cdot l_i^d(t))},$$

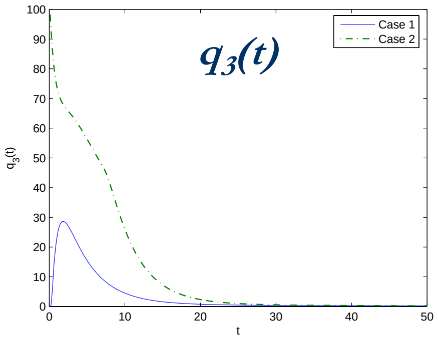
and $N$ is the consecutive times when $\forall i \in \{1, 2, \cdots, N\}$, $j_{s-1}(t-i) > \bar{j}_{s-1}(t-i)$ and $j_{s-1}(t-N-1) \leq \bar{j}_{s-1}(t-N-1)$.

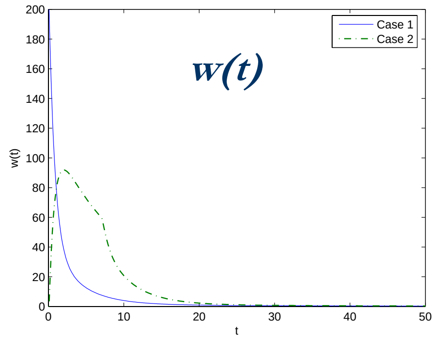
# I-1. Study on: Global stability

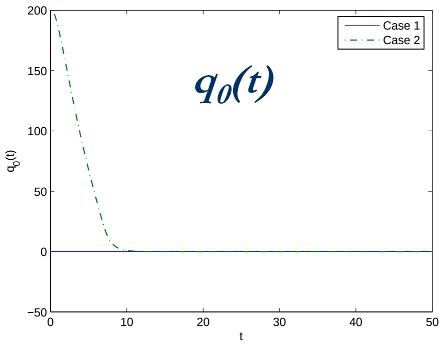
- **Setting**
  - Case I: $q_0(0)=0$, $q_1(0)=800$, $q_2(0)=0$, $q_3(0)=0$, $w(0)=200$
  - Case II: $q_0(0)=200$, $q_1(0)=300$, $q_2(0)=200$, $q_3(0)=100$, $w(0)=0$
- **Observations**
  - Any initial agent distribution →
    - a steady state, a load-balanced state, an optimal resource utilization
    - all characterizing parameters, nonnegative
  - Different agent distributions → the same steady state → globally stable

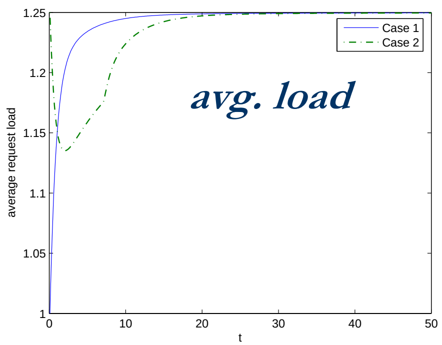


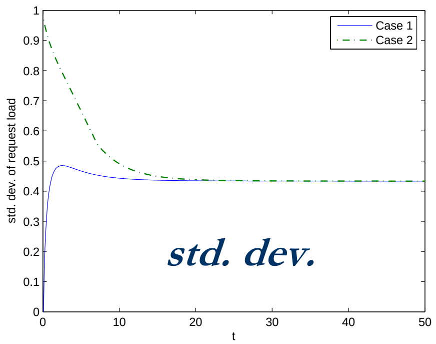



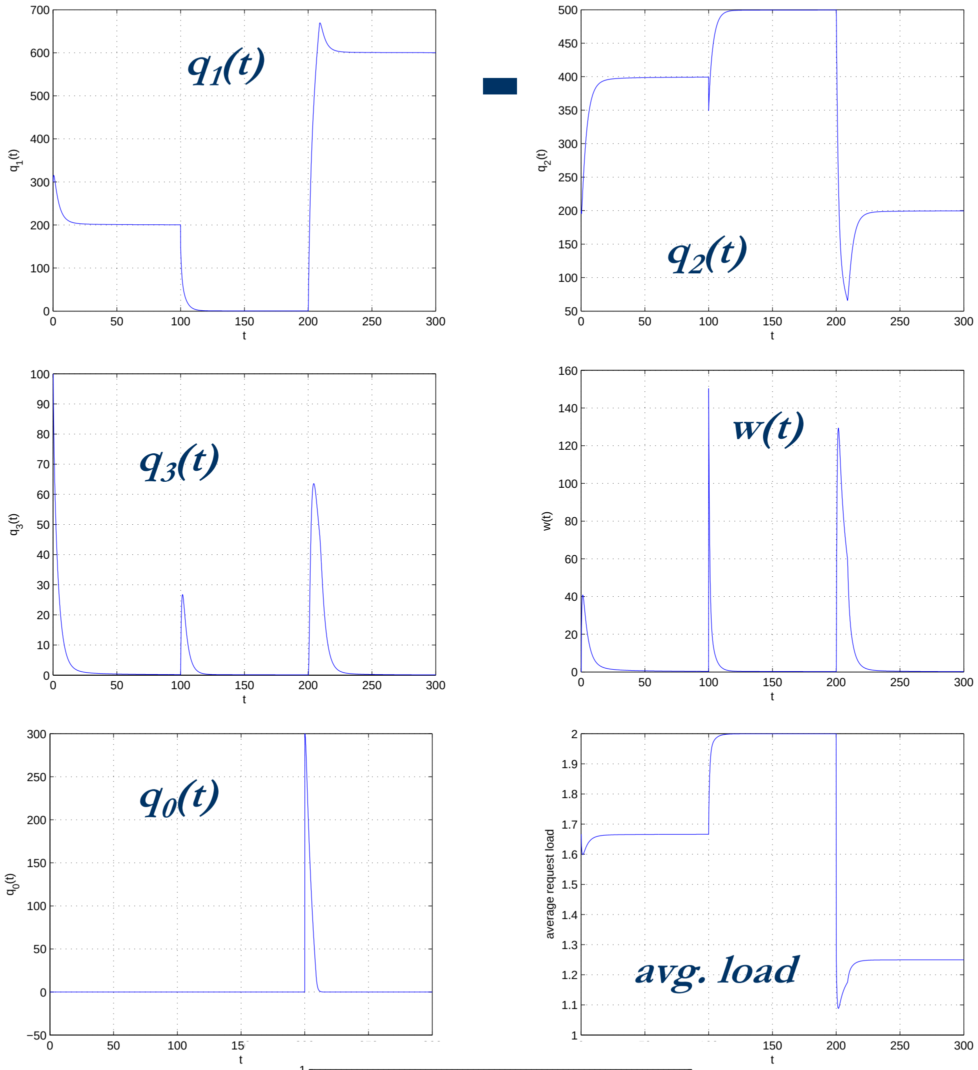



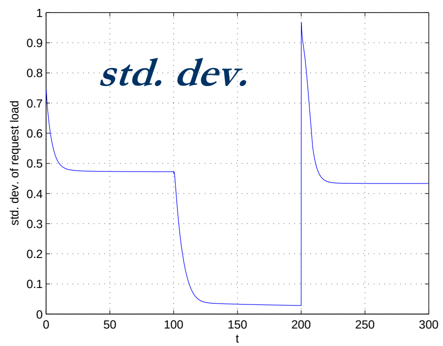






## I-2. **Study on:** Robustness and adaptation

- **Setting:**
  - $m=3$, $S(0)=1000$, $q_1(0)=300$, $q_2(0)=200$, $q_3(0)=100$, $w(0)=0$, $Q(0)=600$, $Q(100)=500$, *and* $Q(200)=800$
  - At time $t=100$ : *100* nodes fail
  - At time $t=200$ : *200* new nodes added

- **Observations**
  - Successfully enduing relatively large-scale resource failures
    - Converging to steady states
  - Quickly responding to a drastic increase in the availability of resource nodes so as to re-balance the load
  - In general,
    - Robust to tolerate dynamic changes
    - Promptly adapting the results of dynamic changes

# AOC-based Ongoing DRO Model

The **quantitative changes** of:

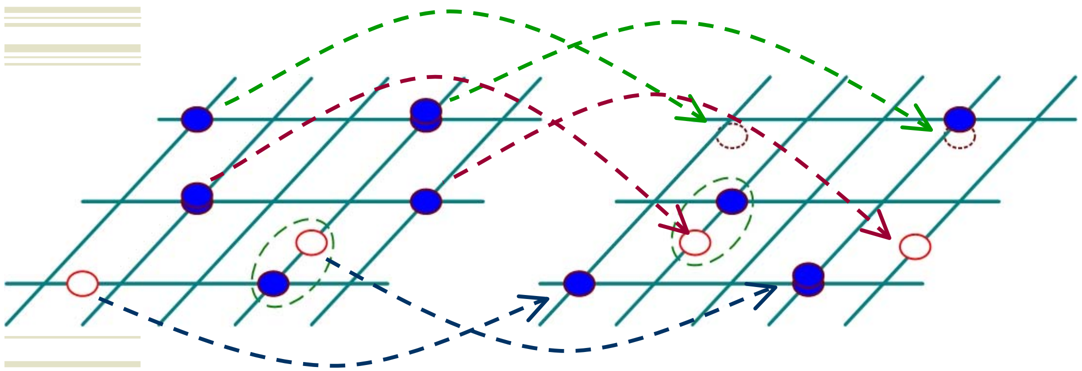
**Agents teams of size one:**

$$\frac{dq_1(t)}{dt} = j_0 w(t) - j_1 w(t) + l_2 q_2(t) + f_2 q_2(t) - f_1 q_1(t)$$

- Wandering agents join idle nodes or existing agent teams
- Queuing agents leave existing agent teams
- Old service requests are finished after being served a unit of service time

**Agents teams of size two:**

$$\frac{dq_2(t)}{dt} = j_1 w(t) - l_2 q_2(t) - f_2 q_2(t)$$



**Wandering agents:**

$$\frac{dw(t)}{dt} = l_2 q_2(t) - \sum_{s=0}^{1} j_s w(t) + g(t)$$

- *g(t)* : newly generated agents for new tasks

**Idle resource nodes:**

$$\frac{dq_0(t)}{dt} = -j_0 w(t) + f_1 q_1(t)$$

$$\frac{dq_1(t)}{dt} = j_0 w(t) - j_1 w(t) + l_2 q_2(t) + f_2 q_2(t) - f_1 q_1(t)$$

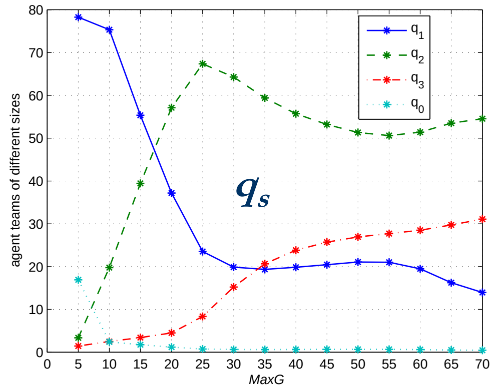
$$\frac{dq_s(t)}{dt} = j_{s-1} w(t) - j_s w(t) + l_{s+1} q_{s+1}(t) - l_s q_s(t) + f_{s+1} q_{s+1}(t) - f_s q_s(t)$$

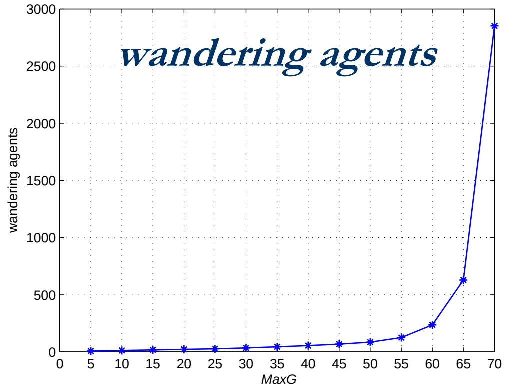
$$\frac{dq_m(t)}{dt} = j_{m-1} w(t) - l_m q_m(t) - f_m q_m(t)$$

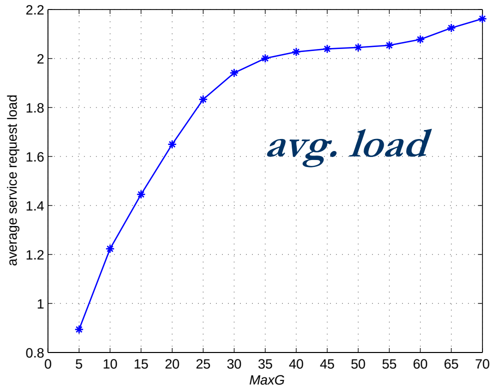
$$\frac{dq_0(t)}{dt} = -j_0 w(t) + f_1 q_1(t)$$

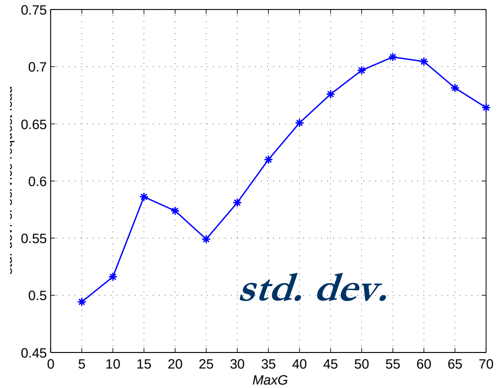
$$\frac{dw(t)}{dt} = \sum_{s=2}^{m} l_s q_s(t) - \sum_{s=0}^{m-1} j_s w(t) + g(t)$$

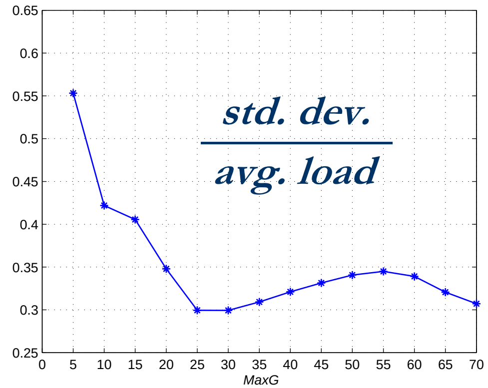
# I-3. Study on: The effects of arrival speeds of service requests



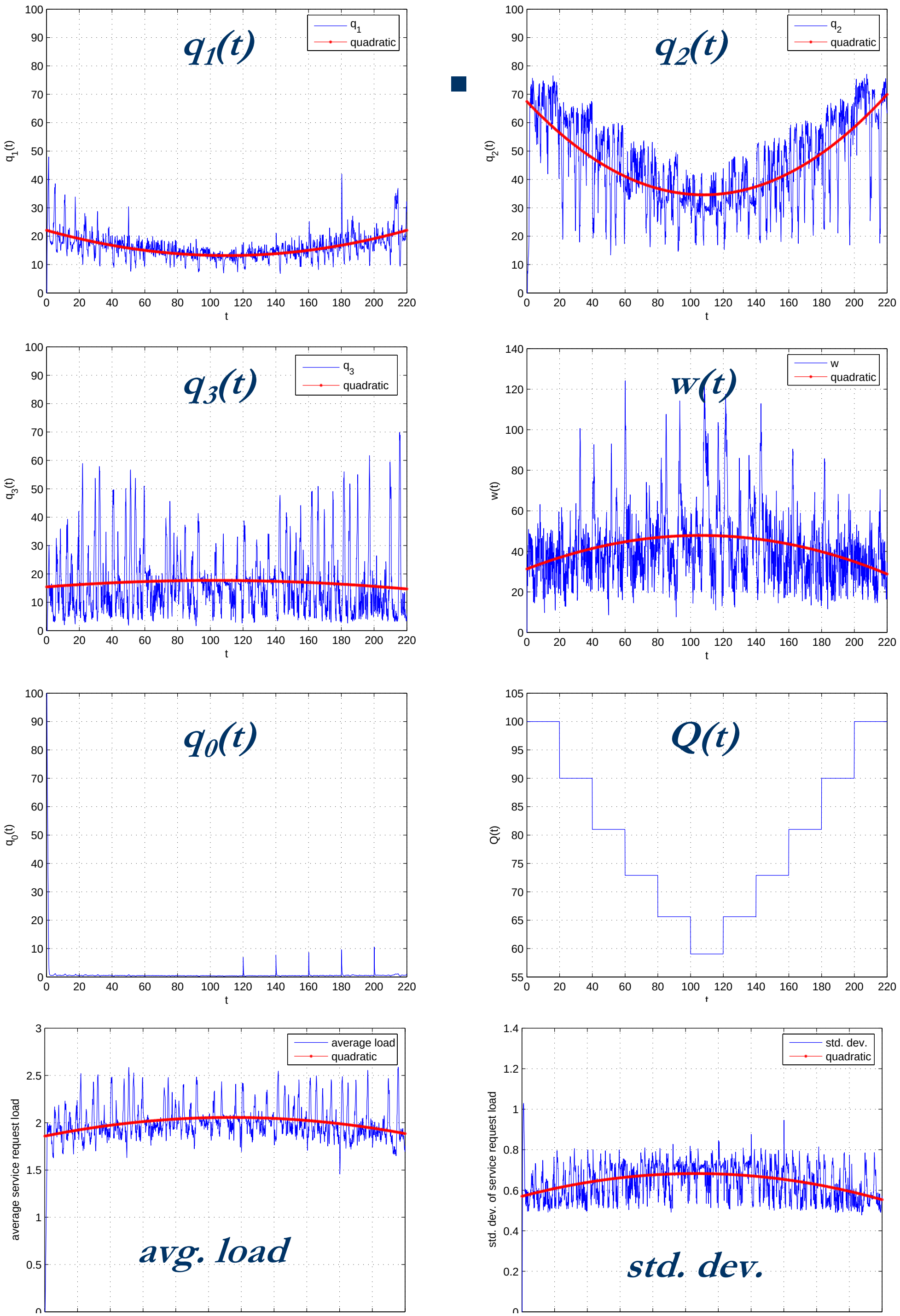

- **Setting:**
  - $g(t)=random([1,MaxG])$
  - $m=3$, $\lambda=10$, $Q(0)=100$, $S(0)=0$, $q_1(0)=0$, $q_2(0)=0$, $q_3(0)=0$, $w(0)=0$

- **Observations**
  - The arrival speed greatly affects the performance of the proposed mechanism. Fixing the service time of service requests,
    - A small arrival speed → less-loaded
    - A large arrival speed → over-loaded
  - Fixing service time $\lambda$, an appropriate arrival speed should be set according:

    $$MaxG \approx (m \cdot Q)/\lambda$$

  - The service requests arrived during a unit of service time (i.e., $MaxG \cdot \lambda$) should approximate the capacity of the whole resource environment (i.e., $m \cdot Q$):

    $$MaxG \cdot \lambda \approx m \cdot Q$$




# I-4. Study on: Robustness and adaptation

- **Setting:**
  - *MaxG=30*
  - $m=3$, $\lambda=10$, $Q(0)=100$, $S(0)=0$, $q_1(0)=0$, $q_2(0)=0$, $q_3(0)=0$, $w(0)=0$
  - At the first *100* steps : *0.1* percent of resource nodes failure per *20* steps
    - *0.1* percent of $q_0(t)$, $q_1(t)$, $q_2(t)$, $q_3(t)$ failure, respectively
  - At the later *100* steps : *0.1* percent of resource nodes recovered and are added to $q_0(t)$, per *20* steps
  - If a resource node with an agent team fails, queuing agents at this node become wandering agents
- **Observation**
  - The avg. service requst load and its std. dev. : no great changes
    - Successfully enduring resource failures
    - Quickly responding to increases in the availability of resource nodes

  - In general, the proposed mechanism is robust
    - It can tolerate failures and recovery of resource nodes without being greatly affected its performance

# Summary

- Instantaneous DRO Scenario
  - Given any initial agent distribution →a steady, load-balanced state
  - Different initial agent distributions →the same steady state → the proposed mechanism is globally stable
  - The proposed mechanism can tolerate large-scale failures and recovery of resource nodes → it is robust to endure dynamic changes occurred, adapt them, and finally reach a new steady state
- Ongoing DRO scenario
  - The arrival speed of service requests greatly affects the performance of the proposed mechanism
  - An appropriate arrival speed of service requests should be set according to:

    $$MaxG \approx \{m \cdot Q\} / \lambda$$
  - The proposed AOC-based DRO mechanism is robust and adaptive to tolerate failures and recovery of resource nodes without being greatly affected its performance

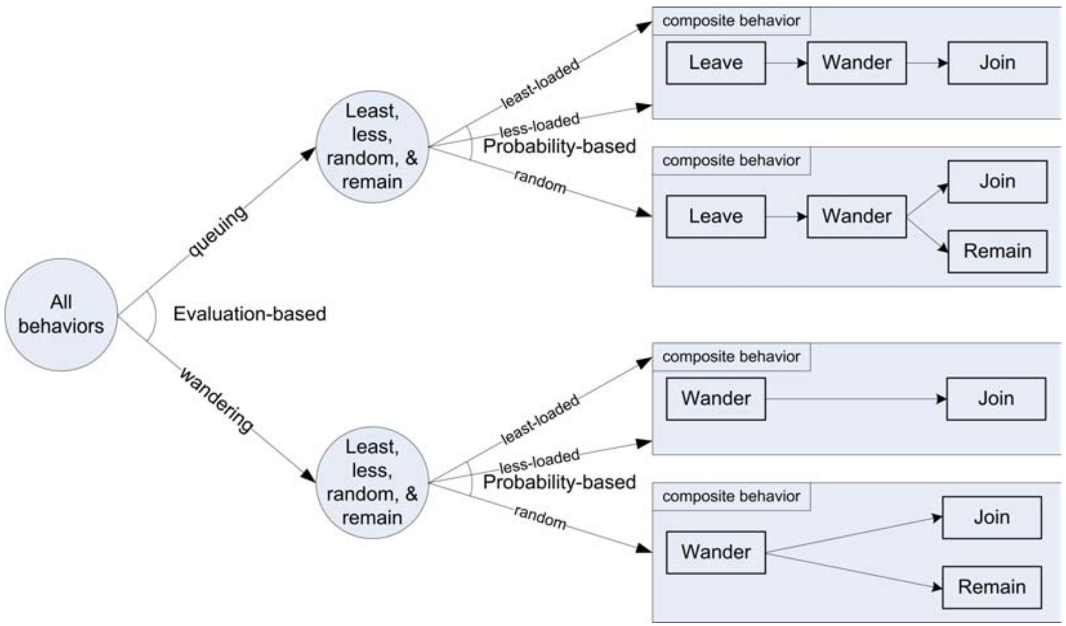
# Q-4-B:
# DRO in heterogeneous environments

# Characterization of Heterogeneous Environments

- Heterogeneous resources & heterogeneous service requests
- Topology of resource networks
  - scale-free with a power of *3*
- Service request characterization
  - *interarrival times*, *sizes*, and *service times* ~ exponential distribution
  - $\lambda_{iat}$ : exponential distribution of interarrival times
  - $\lambda_{ts}$ : exponential distribution of sizes
- Failures and recovery of resource nodes
  - Exponential distributions
  - $\lambda_{fti}$ : exponential distribution of failures
  - $\lambda_{rti}$ : exponential distribution of recovery

# Refined AOC Mechanism and Formulation

- Three composite behaviors: combinations of the primitive behaviors
  - Least-loaded move
  - Less-loaded move
  - Random move
- State description
  - Probabilities vector: $pcb=\langle p_{least}, p_{less}, p_{random}\rangle$
  - $p_{least}, p_{less}, p_{random}$ are fixed
  - $p_{least}$ & $p_{less}$ : relatively large
  - $p_{random}$ : relatively small
- Behavioral rules



# Performance Studies

- **II-1.** How does the probability combination affect the performance of the proposed mechanism? Are all composite behaviors necessary?
  - **II-1-A.** Unsaturated situations
  - **II-1-B.** (Approximately) saturated situations

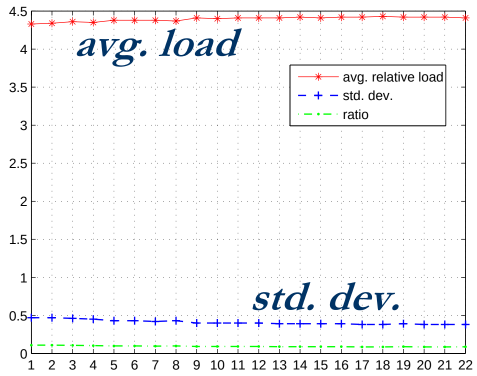
- **II-2.** Whether the proposed mechanism is robust to endure the failures and recovery of resource nodes, and adapt the outcome?

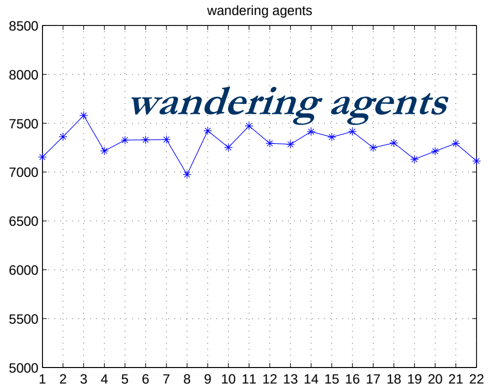
# II-1-A. **Study on:** the probability combination in an unsaturated situation

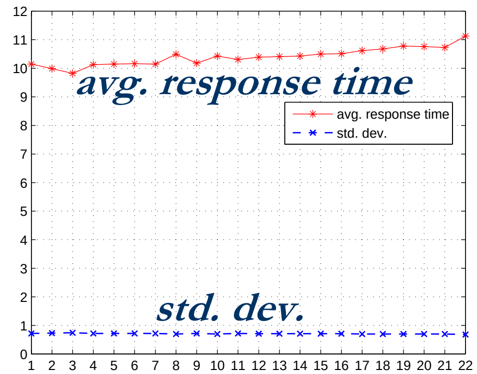

- **Setting:**
  - $\lambda_{iat}$=*0.75*, $\lambda_{ts}$=*100*
  - Processing speeds of services 1 & 2: *200* & *100*
- **Observations**
  - An experiment $p_{random}$ =1.0 :
    - Avg load : 2.5 + Std. Dev. : 3.1
    - The mechanism: effective
  - Large $p_{least}$, → small $std_{rl}$ → more optimized utilization
    - a large number of wandering agents
    - a low service request load
  - Large $p_{less}$ → large $std_{rl}$ → low degree of resource optimization
    Large $p_{less}$ → relatively short response time → the less-loaded move is necessary

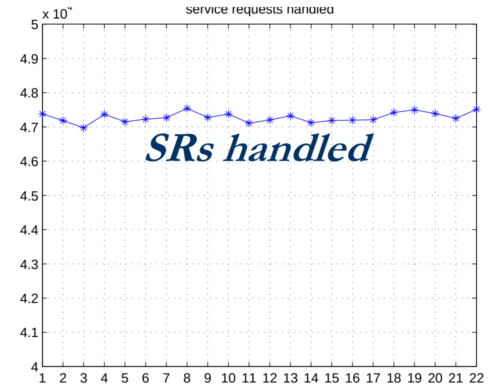
- Random move is also necessary
  - $p_{random}$= 0 → wandering agents are relatively hard to find suitable resource nodes → a lot of wandering agents
  - Random move helps agents move to new areas such that they can possibly find suitable resource nodes
- In general, in a relatively optimal combination
  - $p_{least}$ and $p_{less}$ : close *0.5*
  - $p_{random}$ : a relatively small, nonzero value, say, *0.01*～*0.1*

## II-1-B. **Study on:** the probability combination in a saturated situation



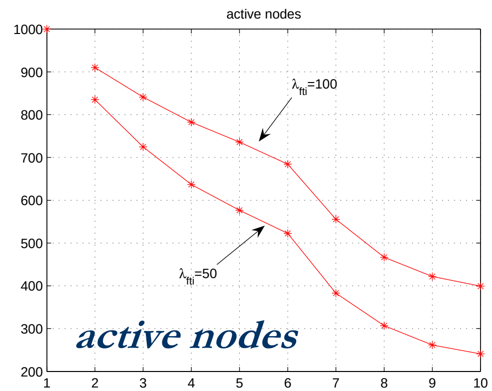



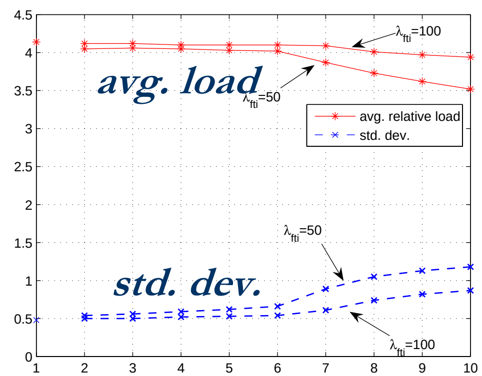



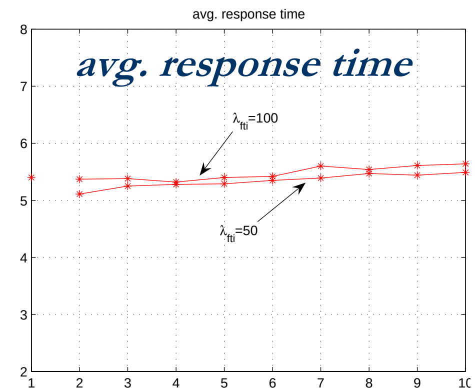

- **Setting:**
  - $\lambda_{iat}=0.45$, $\lambda_{ts}=100$
  - Processing speeds of services 1 & 2: *200* & *100*
- **Observations**
  - An experiment $p_{random}$ =1.0 :
    - Avg load : 4.5 + Std. Dev. : 2.8
    - The mechanism: effective
  - Random move, not necessary
    - All regions may have the same load situation
    - Least-loaded move or less-loaded move is enough for agents

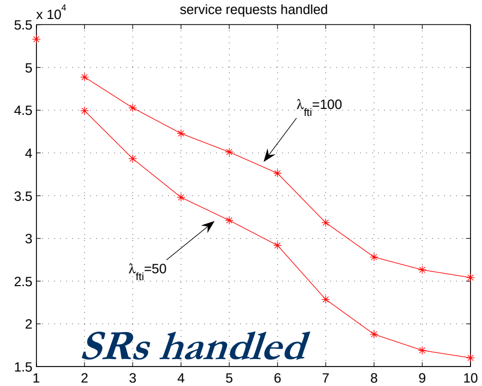
- Different $p_{least}$ and $p_{less}$ $\nrightarrow$ different performance
  - Since a least-loaded move is computationally harder than a less-loaded move, only performing less-loaded move is more reasonable for agents
- In general,
  - If saturated: less-loaded move only
  - If unsaturated: relatively large $p_{least}$ and $p_{less}$, and relatively small $p_{random}$

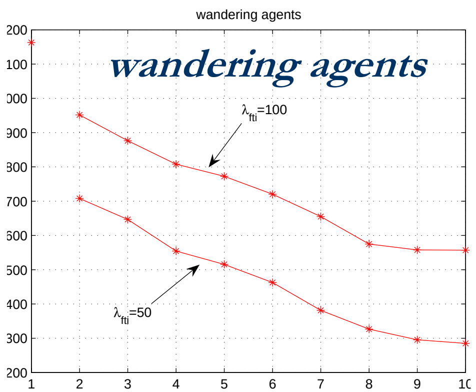
# II-2. **Study on:** Robustness and adaptation




- **Setting:**
  - $\lambda_{iat}$=0.45, $\lambda_{ts}$=100
  -

| # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| $\lambda_{fli}$ | – | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| $\lambda_{rti}$ | – | 10 | 20 | 30 | 40 | 50 | 100 | 150 | 200 | 250 |
| $\lambda_{fli}$ | – | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 |
| $\lambda_{rti}$ | – | 10 | 20 | 30 | 40 | 50 | 100 | 150 | 200 | 250 |

Note: '–' denotes that in this case, no resource nodes fail and recover.

- **Observations**
  - The proposed mechanism is robust
    - it can endure failures and recovery of resource nodes
  - The effects are mainly determined by the distributions of the f&R time intervals, i.e., $\lambda_{fti}$ and $\lambda_{rti}$
    - $\lambda_{fti}>\lambda_{rti}$: no much effect on the average load
    - $\lambda_{fti}<\lambda_{rt}$: the effects becomes remarkable: low average load + high standard deviation
    - The smaller the value of $\lambda_{fti}$ (or, the larger the value of $\lambda_{rti}$), the greater the effects
  - The effects on the average response time are not obvious

# Summary

- In an unsaturated resource environment
  - Less-loaded move & random move : necessary
  - The probability combination determines the performance
  - In an optimized probability combination: $p_{least} \sim p_{less} \sim 0.5$ and $p_{random} \sim$ small, but nonzero

- In a saturated resource environment
  - The probability combination has no great effect
  - Least-loaded move and random move: not necessary
  - Less-loaded move is enough

- Robustness and adaptation
  - The mechanism is robust to endure failures and recovery and adapt to the outcome
  - If resource nodes can quickly recover from failure, no great effects will be caused

# Conclusions and Future Work

- Conclusions & contribution
  - Surveyed related work on Web Intelligence (WI) and Autonomy Oriented Computing (AOC) (Chapter 2)
  - Presented a brief DRO perspective on WI. Specifically, gave a generalized view of distributed resources on the Web, and described a generalized and abstracted scenario for DRO (Chapter 4)
  - Provided an AOC-based DRO mechanism and the corresponding AOC formulation (Chapter 5)
  - Presented an AOC-based DRO mechanism for homogeneous resource environments and validated it through macroscopical characterization, numerical simulation, and experimentation (Chapter 6)

# Conclusion and Future Work (Cont.)

- Presented an AOC-based DRO mechanism for heterogeneous resource environments and validated it through experimentation (Chapter 7)

- Validated AOC as an effective methodology for distributed resource optimization on the Web in that it satisfies the WI requirements, e.g., adaptive, robust, optimized, etc.. (Chapters 6 & 7)

- Future work
  - Service request interdependency
  - Agent behavioral variation
  - Implementation in a realistic Web environment

# Acknowledgement


- Heartfelt gratitude to Prof. Jiming Liu
- Thanks to Prof. Yuan-Yan Tang, Dr. Kwok-Wai Cheung, Dr. Yiu-Ming Cheung, Dr. Chun-Hung Li, and other professors in our department
- Special thanks to Mr. Hoi Fung Lam
- …

# Publications

- **Books (3)**
  - Jiming Liu, **Xiaolong Jin**, and Kwok Ching Tsui, *Autonomy Oriented Computing: From Solving Computational Problems to Characterizing Complex Behavior*, Springer, December, 2004, http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-147-72-36093939-0,00.html;
  - Jiming Liu, **Xiaolong Jin**, Shiwu Zhang, and Jianbing Wu, *Multi-Agent Systems: Models and Experimentation* (in Chinese), Tsinghua University Press, November, 2003;
  - Jiming Liu (Author), **Xiaolong Jin** and Shiwu Zhang (Translators), *Multi-Agent Systems: Principles and Techniques* (in Chinese), Tsinghua University Press, November, 2003.
- **Thesis (1)**
  - **Xiaolong Jin**, *Autonomy Oriented Computing (AOC) for Web Intelligence (WI): A Distributed Resource Optimization Perspective*, PhD thesis, Department of Computer Science, Hong Kong Baptist University, March, 2005.
- **Edited Proceedings (1)**
  - **Xiaolong Jin** and Jianliang Xu, eds., *Proceedings of the First HKBU-CSD Postgraduate Research Symposium (PG Day)*, Department of Computer Science, Hong Kong Baptist University, Technical Report COMP-05-002, January, 2005.
- **Invited Book Chapters (5)**
  - **Xiaolong Jin** and Jiming Liu, *Autonomy Oriented Computing (AOC) for Web Intelligence (WI): A Distributed Resource Optimization Perspective*, N. Zhong, J. Liu, eds., Annual Review of Intelligent Informatics, 2005;
  - Jiming Liu, **Xiaolong Jin**, and Kwok Ching Tsui, *Autonomy Oriented Computing (AOC)*, Submitted to the Encyclopedia of Computer Science and Computer Engineering, John Wiley & Sons, 2005;
  - **Xiaolong Jin** and Jiming Liu, *From Individual Based Modeling to Autonomy Oriented Computation*, Matthias Nickles, Michael Rovatsos, and Gerhard Weiss, eds., Agents and Computational Autonomy, LNAI 2969, pp. 151-169, Springer, 2004;

# Publications (Cont.)

- **Xiaolong Jin**, Jiming Liu, and Yuanshi Wang, *Agent-Supported WI Infrastructure: Case Studies in Peer-to-Peer Networks*, Y.-Q. Zhang, A. Kandel, T. Y. Lin, and Y. Y. Yao, eds., Computational Web Intelligence: Intelligent Technology for Web Applications, Chapter 24, pp. 515-538, World Scientific Publishing, 2004;
- **Xiaolong Jin** and Jiming Liu, *Agent Networks: Topological and Clustering Characterization*, N. Zhong, J. Liu, eds., Intelligent Technologies for Information Analysis, Chapter 13, pp. 291-310, Springer, 2004.

## Journal Papers (6)

- Bingcheng Hu, Jiming Liu, and **Xiaolong Jin**, *Multi-Agent RoboNBA Simulation: From Local Behaviors to Global Characteristics*, Accepted by the Special Issue on Agent-Directed Simulation at Simulation;
- Jiming Liu, **Xiaolong Jin**, and Yuanshi Wang, *Agent-Based Load Balancing on Homogeneous Minigrids: Macroscopic Modeling and Characterization*, IEEE Transactions on Parallel and Distributed Systems, vol. 16, no. 7, pp. 586-598, 2005;
- **Xiaolong Jin** and Jiming Liu, *Characterizing Autonomic Task Distribution and Handling in Grids*, Engineering Applications of Artificial Intelligence, vol. 17, no. 7, pp. 809-823, 2004;
- Jiming Liu, **Xiaolong Jin**, and Kwok Ching Tsui, *Autonomy Oriented Computing (AOC): Formulating Computational Systems with Autonomous Components*, IEEE Transaction On Systems, Man, and Cybernetics – Part A: Systems and Humans (in press);
- Jiming Liu, **Xiaolong Jin**, and Yi Tang, *Multi-Agent Collaborative Service and Distributed Problem Solving*, Cognitive Systems Research, vol. 5, no. 3, pp. 191-206, 2004;
- Jiming Liu, **Xiaolong Jin**, and Jing Han, *Distributed Problem Solving without Communication – An Examination of Computationally Hard Satisfiability Problems*, International Journal of Pattern Recognition and Artificial Intelligence, vol. 16, no. 8, pp. 1041-1064, 2002.

## Conference Papers (17)

- **Xiaolong Jin** and Jiming Liu, *Resource Optimization in Heterogeneous Grid Environments*, Submitted to the 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05), in Compiegne University of Technology, France, September 2005;

# Publications (Cont.)

- **Xiaolong Jin** and Jiming Liu, *AOC-Based Load Balancing on Homogeneous Minigrids*, Submitted to the 2005 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'05), in Compiegne University of Technology, France, September 2005;
- Tingting Wang, Jiming Liu, and **Xiaolong Jin**, *Minority Game Strategies for Dynamic Multi-Agent Role Assignment*, in Proceedings of the 2004 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'04), pp. 316-322, Beijing, China, September 2004;
- Bingcheng Hu, Jiming Liu, and **Xiaolong Jin**, *From Local Behaviors to Global Performance in a Multi-Agent System*, in Proceedings of the 2004 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'04), pp. 309-315, Beijing, China, September 2004;
- **Xiaolong Jin**, Jiming Liu, and Yuanshi Wang, *Modeling Agent-Based Task Handling in a Peer-to-Peer Grid*, in Proceedings of the 2004 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'04), pp. 288-294, Beijing, China, September 2004;
- Yi Tang, Jiming Liu, and **Xiaolong Jin**, *Aggregating Local Behaviors Based upon Lagrange Method*, in Proceedings of the 2004 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'04), pp. 413-416, Beijing, China, September 2004;
- **Xiaolong Jin** and Jiming Liu, *The Dynamics of Peer-to-Peer Tasks: An Agent-Based Perspective*, in Proceedings of the Third International Workshop on Agents and Peer-to-Peer Computing (AP2PC 2004), New York, USA, July 2004;
- Long Gan, Jiming Liu, and **Xiaolong Jin**, *Agent-Based, Energy Efficient Routing in Sensor Networks*, in Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS'04), New York, USA, July 2004;
- **Xiaolong Jin** and Jiming Liu, *Properties of Clustering Coefficient in Random Agent Networks* (Excellent Paper Award), in Proceedings of the Second International Conference on Active Media Technology (ICAMT'03), pp. 73-82, Chongqing, China, May 2003;
- Bingcheng Hu, Jiming Liu, and **Xiaolong Jin**, *Phase Transitions in RoboNBA*, in Proceedings of the 5th ACM Postgraduate Research Day (Hong Kong), pp. 73-79, Hong Kong, January 2004;
- **Xiaolong Jin**, Jiming Liu, and Yuanshi Wang, *Modeling Agent-Based Task Handling in a Peer-to-Peer Grid*, in Proceedings of the 5th ACM Postgraduate Research Day (Hong Kong), pp. 87-96, Hong Kong, January 2004;
- Yuanshi Wang, Jiming Liu, and **Xiaolong Jin**, *Modeling Agent-Based Load Balancing with Time Delays*, in Proceedings of the 2003 IEEE/WIC International Conference on Intelligent Agent Technology (IAT'03), pp. 189-195, Halifax, Canada, October 2003;

# Publications (Cont.)

- **Xiaolong Jin** and Jiming Liu, *Efficiency of Emergent Constraint Satisfaction in Small-World and Random Networks*, in Proceedings of the 2003 IEEE/WIC International Conference on Intelligent Agent Technology (IAT'03), pp. 304-310, Halifax, Canada, October 2003;
- **Xiaolong Jin** and Hongge Liu, *Research on the Completeness of Pangu Knowledge Base*, in Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (ICMLC'03), Xi'an, China, August 2003;
- **Xiaolong Jin** and Jiming Liu, *Agent Network Topology and Complexity*, in Proceedings of the Second International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS'03), pp. 1020-1021, Melbourne, Australia, July 2003;
- Yi Tang, Jiming Liu and **Xiaolong Jin**, *Adaptive Compromises in Distributed Problem Solving*, in Proceedings of the Fourth International Conference on Intelligent Data Engineering and Automated Learning (IDEAL'03), LNCS 2690, pp. 31-40, Springer, Hong Kong, China, March 2003;
- **Xiaolong Jin** and Jiming Liu, *Multiagent SAT (MASSAT): Autonomous Pattern Search in Constrained Domains*, in Proceedings of the Third International Conference on Intelligent Data Engineering and Automated Learning (IDEAL'02), LNCS 2462, Hujun Yin et. al. Eds., pp. 318-328, Springer, Manchester, UK, August 2002;

## Technical Reports (4)

- Jiming Liu, **Xiaolong Jin**, and Yuanshi Wang, *Agent-Based Load Balancing on Homogeneous Minigrids: Macroscopic Modeling and Characterization*, Department of Computer Science, Hong Kong Baptist University, Technical Report Comp-04-005, September 2004;
- **Xiaolong Jin**, Jiming Liu, and Yuanshi Wang, *Characterizing the Dynamics of Agent-Based Peer-to-Peer Computing*, Department of Computer Science, Hong Kong Baptist University, Technical Report Comp-04-004, May 2004;
- **Xiaolong Jin** and Jiming Liu, *An Autonomy-Oriented, Distributed Approach to Satisfiability Problems*, Department of Computer Science, Hong Kong Baptist University, Technical Report Comp-04-003, May 2004;
- Jiming Liu, **Xiaolong Jin**, and Kwok Ching Tsui, *Autonomy Oriented Computing (AOC): Formulating Computational Systems with Autonomous Components*, Department of Computer Science, Hong Kong Baptist University, Technical Report Comp-04-001, March 2004.

**Thank you!**

**Q. & A.**