愛知県立大学
Aichi Prefectural University

**2017/9/28 APNOMS2017 Technical Session 7 : 7-3 (15:20~17:00)**

# Queueing Theoretic Approach to Job Assignment Strategy Considering Various Inter-arrival of Job in Fog Computing


Graduate School of Information Science and Technology
Aichi Prefectural University
**Tomoya Mori**, Yoichi Utsunomiya, Xuejun Tian and Takashi Okuda
Email: im171009@cis.aichi-pu.ac.jp


Santol Since 2001
Systems modeling & Analysis, Networking & Telecommunications, and Operations research Laboratory.
Dr. Takashi OKUDA Laboratory

# Agenda

1. Overview
2. Assuming Computing Architecture
3. Queueing Network Model
4. Numerical Examples
5. Discussion
6. Conclusion and Future Work

# Background (1/3)

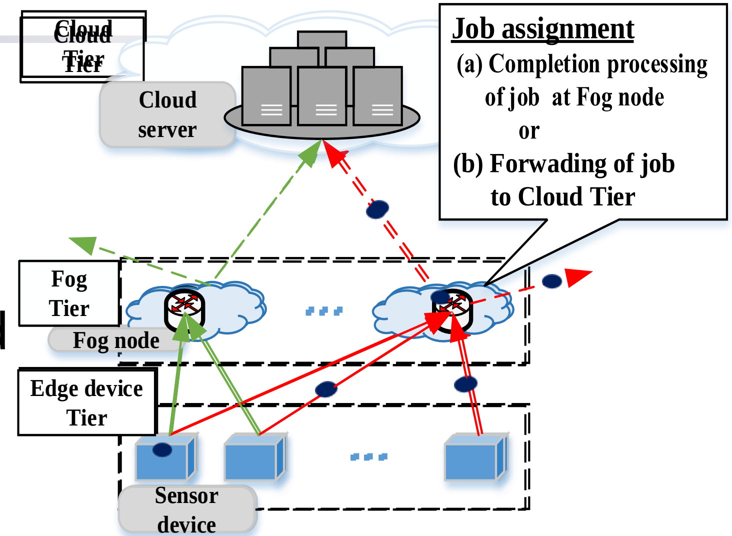
- IoT has recently attracted much attention
- One of the techniques supporting the IoT systems is Cloud computing.

Source: http://reci-asp.net/fudosanit/?p=10506

- **The critical issues** of Cloud computing[1]
  - an increase of network load
  - a delay of feedback control

[1] N. Shiratori, et al, ``Latest Development of IoT Architecture'', The Journal of the IEICE, Vol. 100, No. 3, pp. 214-221, 2017 (in Japanese).

# Background (2/3)




## Fog computing[2]

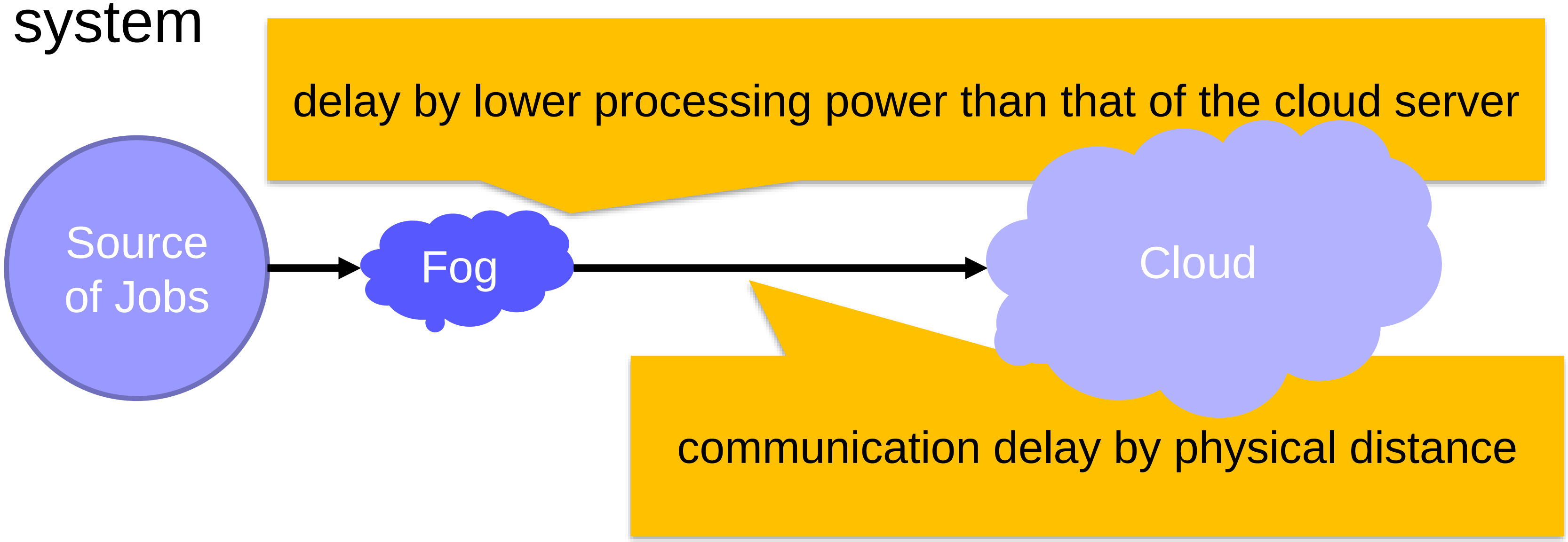
- **roles of conventional routers**
  - fowarding **jobs (data, requests)** to cloud
- **Roles of Fog node**
  - Job assignment

Fog computing is expected to be applied to IoT systems such as Smart Grid and VANETs [3]

[2]Cisco Systems, "*Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are*", https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-overview.pdf.
[3] F. Bonomi, et al. , "Fog Computing and Its Role in the Internet of Things", Proceedings of the first edition of the MCC workshop on Mobile cloud computing, pp. 13-16, 2012.

# Background (3/3)

- Problem of Fog computing
  - Job assignment strategy is **not clear** depending on the system




  - The reasons why it is difficult to consider the strategy
    - Jobs are various
    - Jobs are generated at various intervals

# Objective

- Our work …
  - consider **effective Job assignment strategies** depending on various IoT systems by using **Queueing model**

- In this presentation…
  - focus on the **various inter-arrival of jobs** depending on the IoT system
  - use VCHS (Various Customers Heterogeneous Servers)[4] queueing model to model Fog computing with the Job assignment
  - discuss about an effective Job assignment strategy considering various inter-arrival of jobs

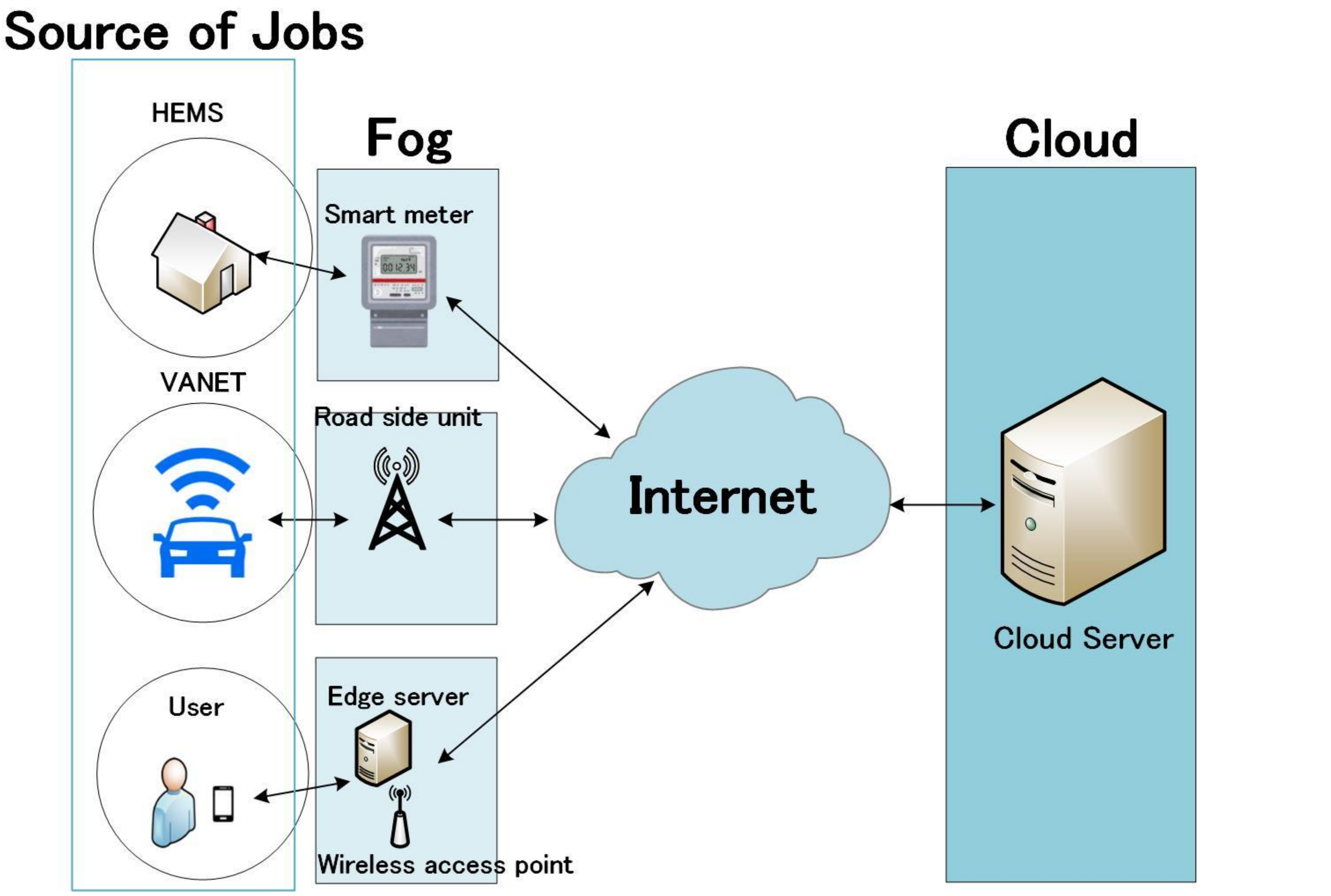
[4] S. Shimizu, et al. “On simulation evaluation for quantitative and qualitative VCHS problems”, IEICE Technical Report, Information Networks, Vol. 110, No. 341, pp. 63-68, 2010 (in Japanese)

# Agenda

1. Overview
2. Assuming Computing Architecture
3. Queueing Network Model
4. Numerical Examples
5. Discussion
6. Conclusion and Future Work

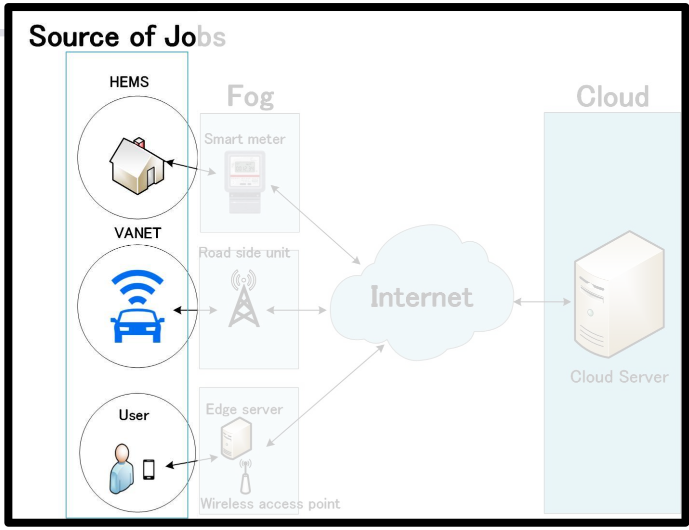
# Schema

Assuming Computing Architecture



# Job

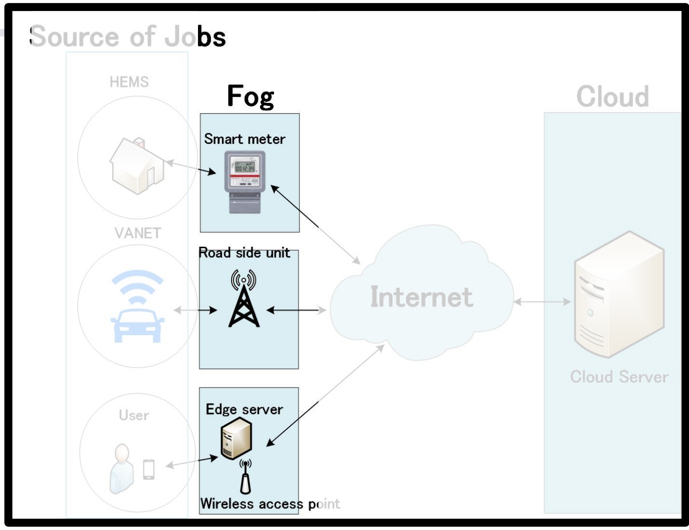





- generated at various intervals
  - HEMS：regular interval
  - VANETs：random
  - User：random，burst
- has two attributes;
  - RV : request volume (amount for a processing of a job)
    - represented by size : S (Small)，M (Medium)，L (Large)
  - DS : delay-sensitiveness (degree of allowance of delay)
    - represented by degree : H (High)，M (Middle)，L (Low)
- Assume that size of all jobs is equal

# Fog tier

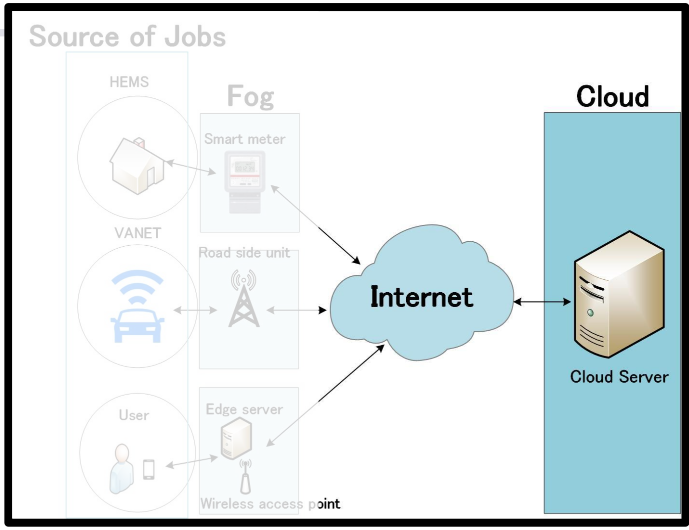
Assuming Computing Architecture




- consists of various devices (***Fog node***) depending on IoT system
- each fog node receives jobs at various intervals
  - Smart meter (HEMS) regular interval
  - RSU (VANETs) random
  - Access point (User) random, burst

# Cloud tier



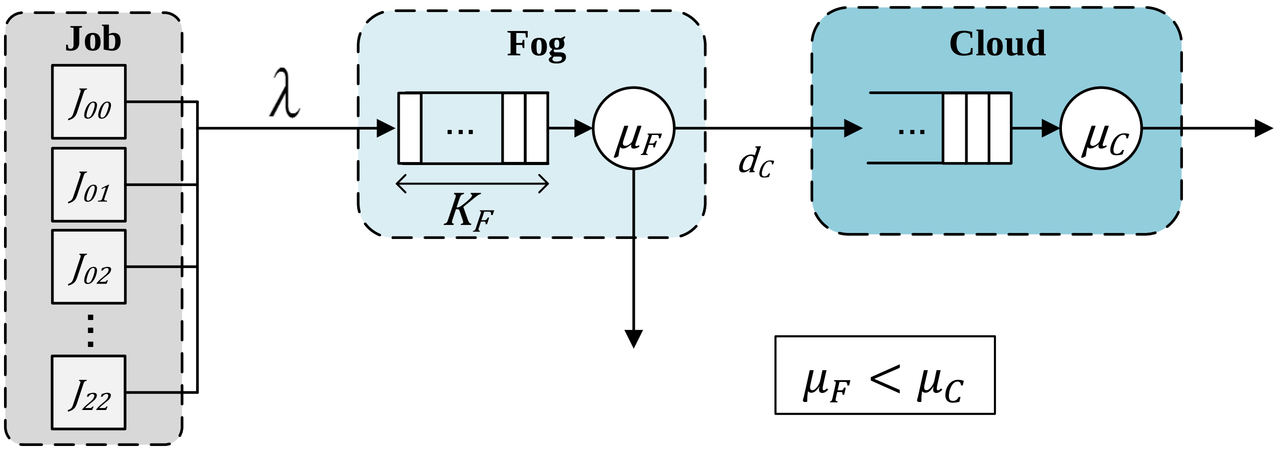



- One cloud server
- has a **delay time** of which jobs arrive at Cloud because of the distance between Fog and Cloud

# Agenda

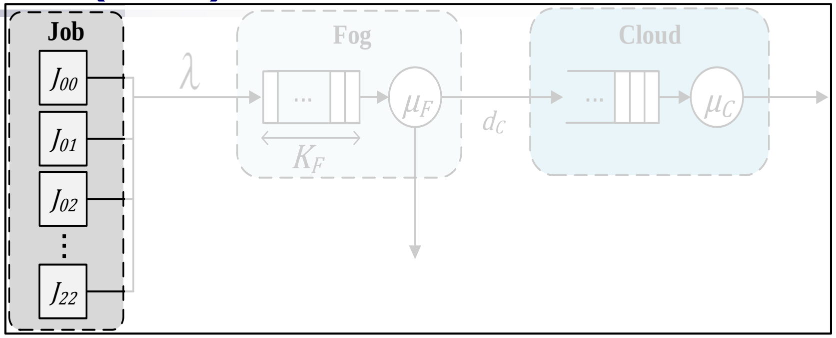
1. Overview
2. Assuming Computing Architecture
3. Queueing Network Model
4. Numerical Examples
5. Discussion
6. Conclusion and Future Work

# Queueing Network Model(1/6)




- Queueing network model has two VCHS queueing model
  - the fog node
    - service rate $\mu_F$, a queue of length $K_F$
  - the cloud server
    - service rate $\mu_C$, an unlimited queue

# Queueing Network Model(2/6)

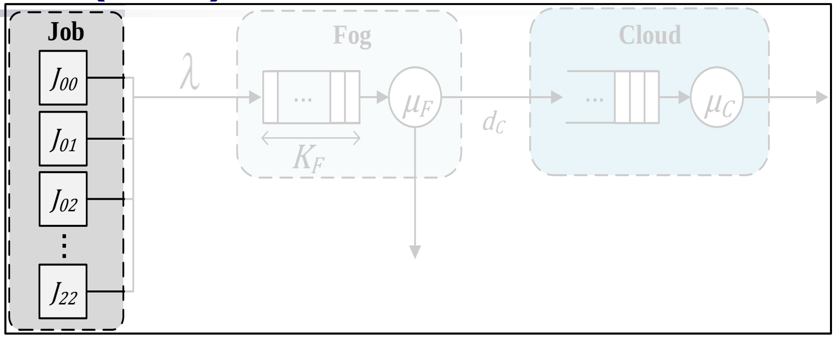



- Job $J_{ij}(i, j = 0, \ldots, 2)$
  - has two attribute;
    - RV : request volume
    - DS : delay sensitiveness

| Attribute | $J_{00}$ | $J_{01}$ | $J_{02}$ | $J_{10}$ | $J_{11}$ | $J_{12}$ | $J_{20}$ | $J_{21}$ | $J_{22}$ |
|---|---|---|---|---|---|---|---|---|---|
| RV | S | | | M | | | L | | |
| DS | H | M | L | H | M | L | H | M | L |

# Queueing Network Model(2/6)

- Job $J_{ij}(i,j=0,\dots,2)$




- has two attribute;
  - RV : request volume
  - DS : delay sensitiveness

$H$ is the service time for a server with service rate 1.0 to process $J_{ij}$

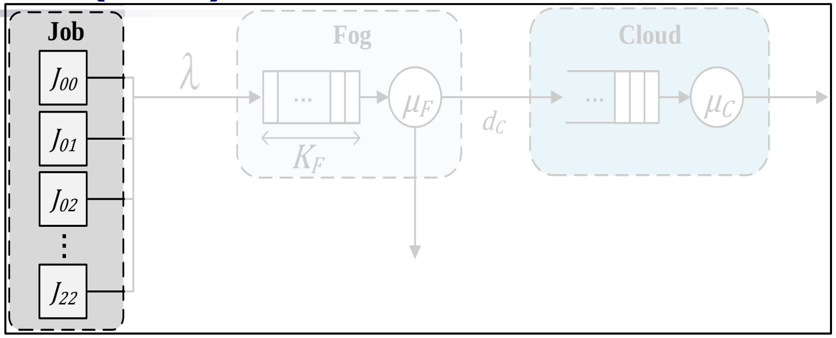
- has an exponential distribution with mean $h$

$$h=\begin{cases} h_s, & i=0 \\ h_m, & i=1 \\ h_l, & i=2 \end{cases}$$

$$where\ h_s<h_m<h_l$$

# Queueing Network Model(2/6)

- Job $J_{ij}(i, j = 0, \ldots, 2)$




  - □ has two attribute;
    - RV : request volume
    - DS : delay sensitiveness

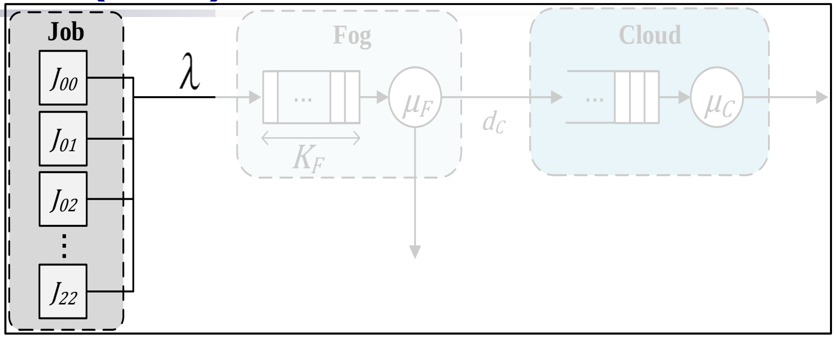
$V_{DS}$ represents the value of the degree of DS for $J_{ij}$

- has an exponential distribution with mean $v$

$$v = \begin{cases} w_s, & i = 0 \\ w_m, & i = 1 \\ w_l, & i = 2 \end{cases}$$

$$where\ w_s < w_m < w_l$$

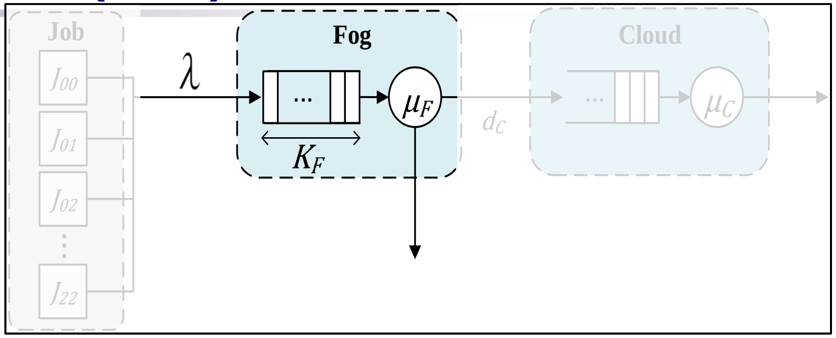
# Queueing Network Model(3/6)




- Job $J_{ij}(i, j = 0, \dots, 2)$
  - One of the nine jobs is randomly generated
  - the inter-arrival time $T_a$
    - has a probability distribution decided by *the squared coefficient of variation*
      - $Ca^2$ (= variance $\sigma^2$ $/(mean\ \lambda^{-1})^2$)

| | |
|---|---|
| $Ca^2 = 0$ | deterministic distribution |
| $Ca^2 = 1$ | exponential distribution |
| $Ca^2 > 1$ | hyper-exponential distribution |

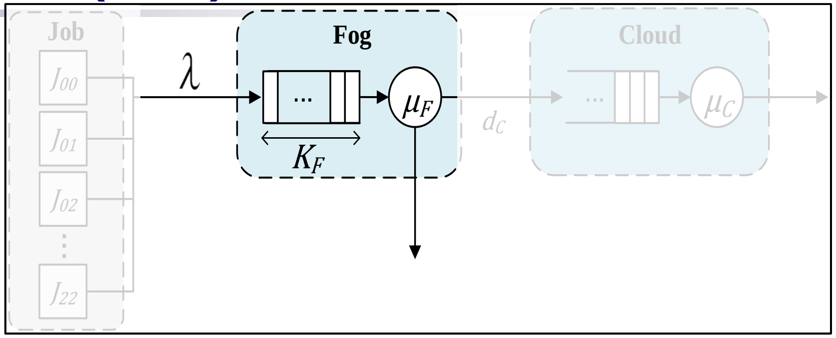
# Queueing Network Model(4/6)




- Fog tier
  - Service discipline
    - FCFS (First Come First Served)
      In this talk, I don’t consider a priority of jobs

# Queueing Network Model(5/6)

- Fog tier
  - Flow of a processing of $J_{ij}$

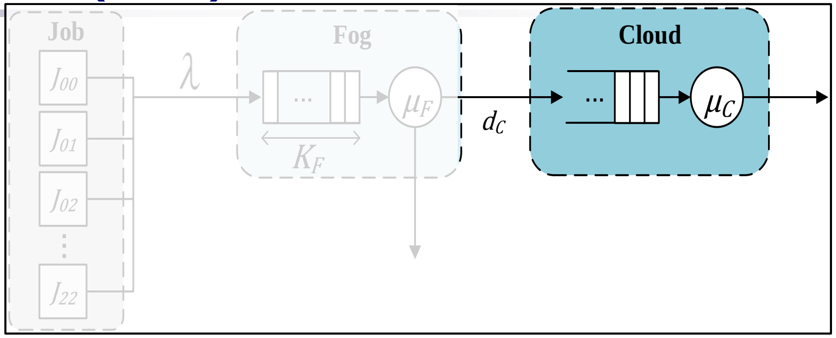



Please note we assume

- the fog node can accurately detect the value of attributes of jobs
- the time which the fog node requires at Job assignment can be ignored

- assign $J_{ij}$ to Cloud tier
- assign $J_{ij}$ to Cloud tier

# Queueing Network Model(6/6)



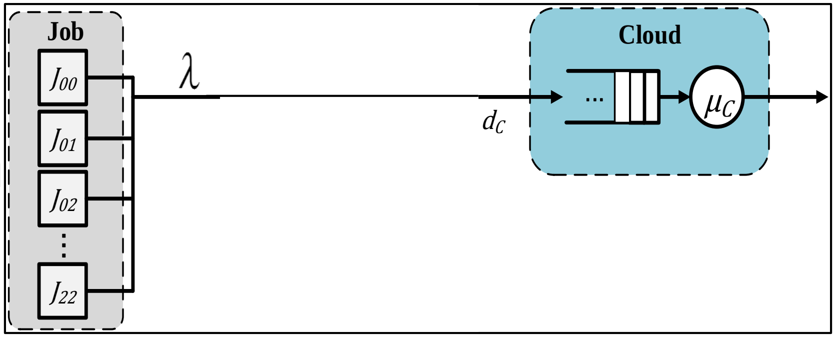

- Cloud tier
  - delay time $d_C$
    - a deterministic amount because we assume that size of all jobs is equal.
  - Service discipline
    - FCFS (First Come First Served)
  - It takes $H/\mu_C$ to process $J_{ij}$ assigned by the Fog tier

# Agenda

1. Overview
2. Assuming Computing Architecture
3. Queueing Network Model
4. Numerical Examples
5. Discussion
6. Conclusion and Future Work

# Simulation settings (1/3)

- simulate 4 types
  - only Cloud computing
  - Fog computing + Cloud com
    - RV-S
    - DS-S
    - Rand-S




- Condition of one simulation
  - a million jobs arrives at the Fog
  - If all jobs in the system have been processed, one simulation is completed

We use a discrete event-based simulation with C++[5]

[5]Mesquite Software, http://www.mesquite.com.

# Simulation settings (2/3)

## Performance evaluation metrics

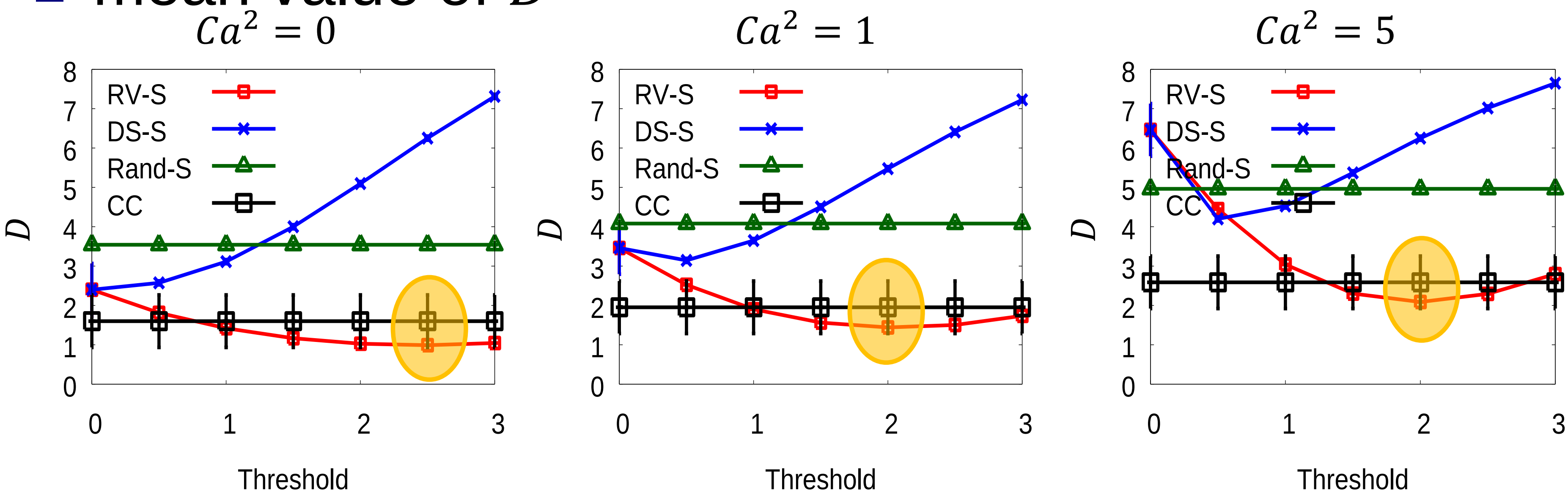
- *delay ratio*
  - $D = D_s + D_w$
  - $D_s = Time\ in\ the\ sytem\ /h$
    - *Time in the system*
      - the time that a job spent in the queueing network
  - $D_w = Waiting\ time\ /v$
    - *Waiting time*
      - the time that a job waited for the process by any servers
- *probability of block*
  - $B = \frac{Total\ number\ of\ jobs\ blocked\ by\ the\ fog\ tier}{Total\ number\ of\ jobs\ which\ arrive\ at\ the\ fog\ tier}$

# Simulation settings (3/3)

| Item | Symbol | Parameter |
|---|---|---|
| mean arrival rate | $\lambda$ | 1.0 |
| squared coefficient of variation | $Ca^2$ | 0,1,5 |
| mean service time of each jobs | $\{h_s, h_m, h_l\}$ | {1.0,2.0,3.0} |
| mean value of delay-sensitiveness of each jobs | $\{w_s, w_m, w_l\}$ | {1.0,2.0,3.0} |
| service rate of the Fog tier | $\mu_F$ | 1.5 |
| length of the queue in the Fog tier | $K_F$ | 10 |
| service rate of the Cloud tier | $\mu_C$ | 3.0 |
| delay time between the Fog tier and the Cloud tier | $d_C$ | 1.0 |
| threshold | ― | 0.0～3.0 |

# Simulation results (1/2)

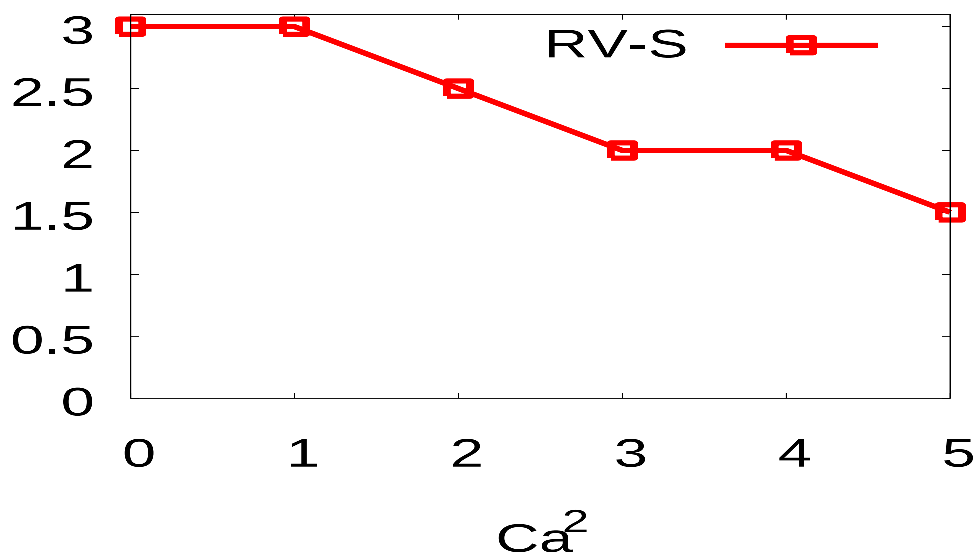
- mean value of $D$




- RV-S is the best
- In any $Ca^2$, the difference between the minimum of $D$ by RV-S and **the $D$ by only Cloud computing** is the same
- The number of jobs which the Fog tier can process is different depending on $Ca^2$

# Simulation results (2/2)

- Max of $threshold$ which can satisfy $B$ less than .001




- According to increasing $Ca^2$, max of $threshold$ which can satisfy $B$ less than .001 is **declining**
  - To ensure any $B$, we need to set appropriate $threshold$ depending on $Ca^2$

# Agenda

1. Overview
2. Assuming Computing Architecture
3. Queueing Network Model
4. Numerical Examples
5. Discussion
6. Conclusion and Future Work

# Discussion

- need to take two points into account
  1. delay-sensitiveness
     - **The more large** $Ca^2$, the time of processing jobs at the Fog is **the more bottleneck**
       - consider the Job assignment strategy by a model which has more fog nodes
       - add a priority control in the queue of the fog node by regarding delay-sensitiveness as a priority
  2. a job with a high importance
     - Even if the queue of the fog node is full,
       we should assign such jobs to the Cloud

# Agenda

1. Overview
2. Assuming Computing Architecture
3. Queueing Network Model
4. Numerical Examples
5. Discussion
6. Conclusion and Future Work

# Conclusion and Future Work

- Conclusion
  - RV-S is the best strategy of three simple strategies in any $Ca^2$
  - There are two points to consider an effective Job assignment strategy depending on the system

- Future Work
  should consider ・・・
  - to propose an effective Job assignment strategy
  - to use more practical data to model job, fog, and so on
  - the time of Job assignment

愛知県立大学
Aichi Prefectural University

# Thank you for your attention!!

Santol Since 2001
Systems modeling & Analysis,
Networking & Telecommunications,
and Operations research Laboratory.
Dr. Takashi OKUDA Laboratory

愛知県立大学
Aichi Prefectural University

**2017/9/28 APNOMS2017 Technical Session 7 : 7-3 (15:20~17:00)**

# Queueing Theoretic Approach to Job Assignment Strategy Considering Various Inter-arrival of Job in Fog Computing


Graduate School of Information Science and Technology
Aichi Prefectural University
**<u>Tomoya Mori</u>**, Yoichi Utsunomiya, Xuejun Tian and Takashi Okuda
Email: im171009@cis.aichi-pu.ac.jp


Santol Since 2001
Systems modeling & Analysis, Networking & Telecommunications, and Operations research Laboratory.
Dr. Takashi OKUDA Laboratory

# Related works


- R. Deng, R. Lu, C. Lai, T. H. Luan and H. Liang, "*Optimal Workload Allocation in Fog-Cloud Computing Towards Balanced Delay and Power Consumption*", IEEE Internet of Things Journal, Vol. 3, Issue. 6, pp. 1171-1181, 2016.

- X. Q. Pham, A. A. Alsaffar, P. V. Nam, N. Q. Thai, N. D. Tri, N. T. Thu and E. N. Huh, "*An Efficient Task Scheduling Based on Cloud and Fog Collaboration*", 2017 Annual Conference of the Korean Institute of Communication Sciences, pp. 1242-1243, 2017.

# Performance evaluation metrics ($D_s, D_w$)

- $D_s$ implies the degree of the delay caused by the system to the mean service time for jobs.
- $D_w$ implies the degree of the delay caused by the system to the delay-sensitiveness for jobs.

# Why is the Fog a single server?

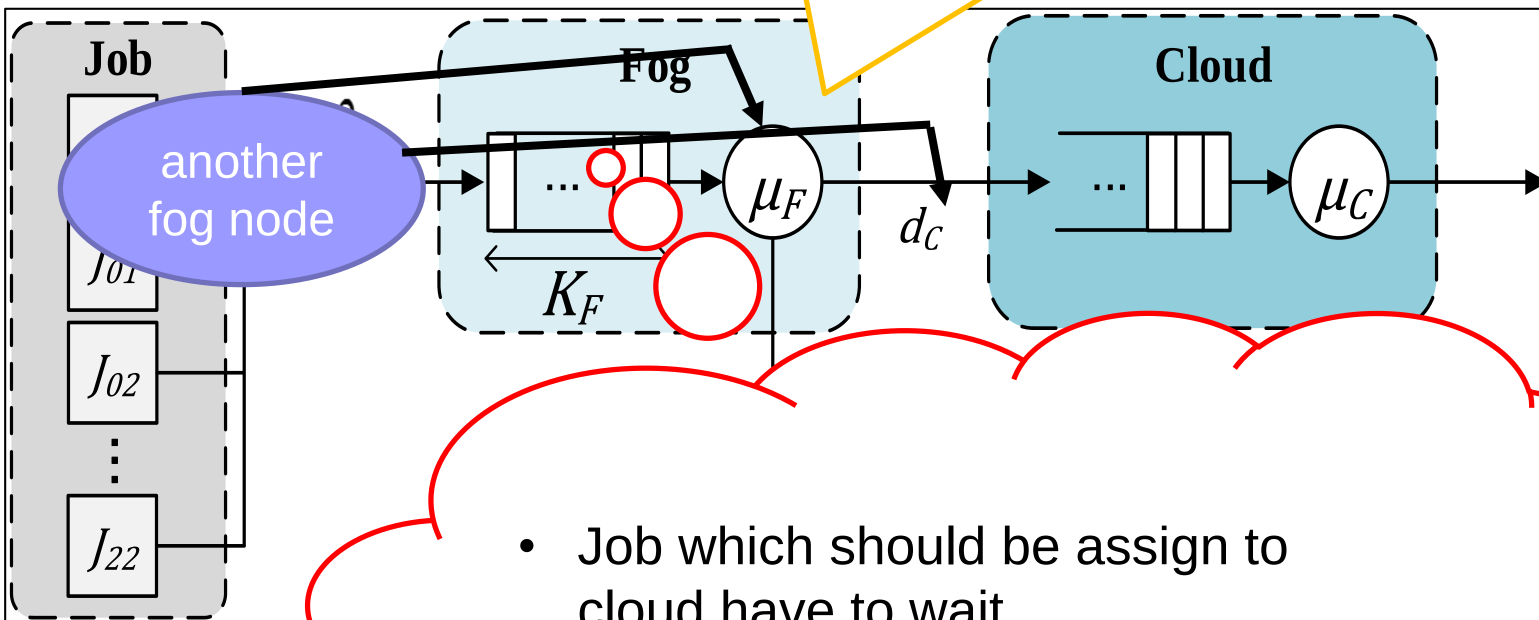
- Nearly, the number of devices which has computer of equivalent high processing power is increasing.

- In this talk, assume the Fog consists of these devices.

- Total processing power of five servers with service rate 1.0 equals to the processing power of one server with service rate 5.0.

- Why is the time of processing jobs at the Fog  the more bottleneck?

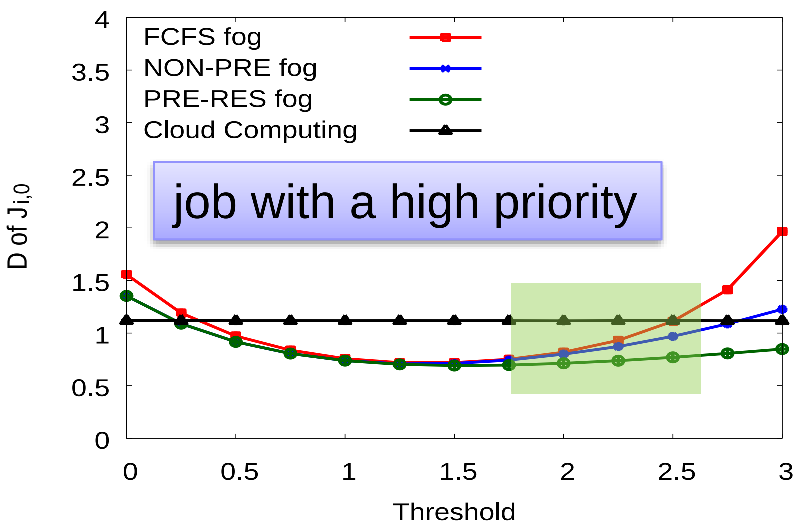
running a processing of job

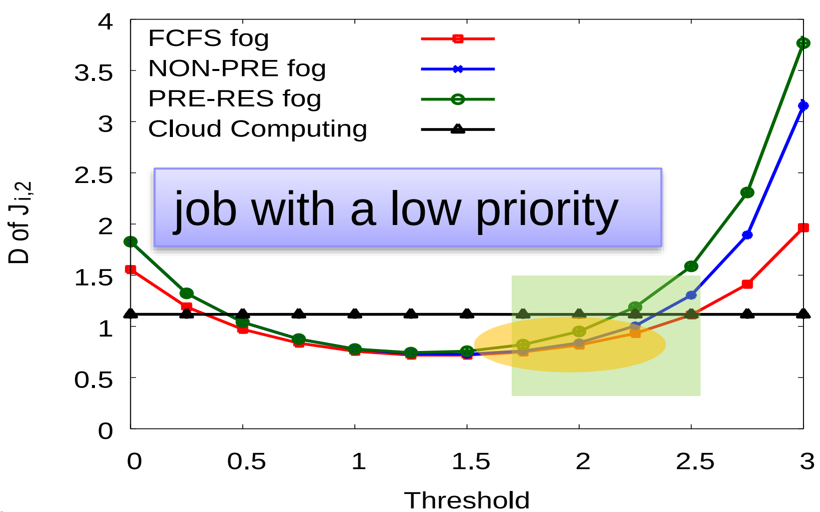



- Job which should be assign to cloud have to wait
- Job with a high DS have to wait

# Simulation results (1/2)

- *threshold* vs. *D*



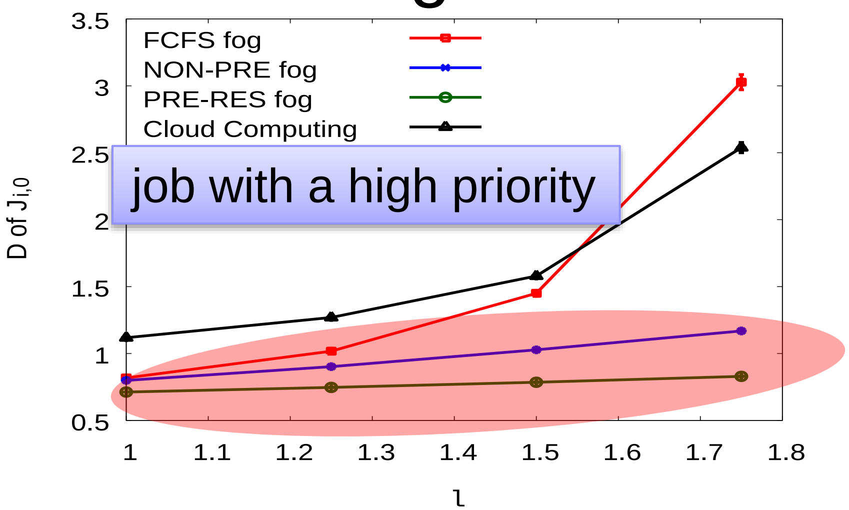




Combining a priority control with the Job assignment (graph of Blue and Green) is more effective than just doing the Job assignment (Red graph).

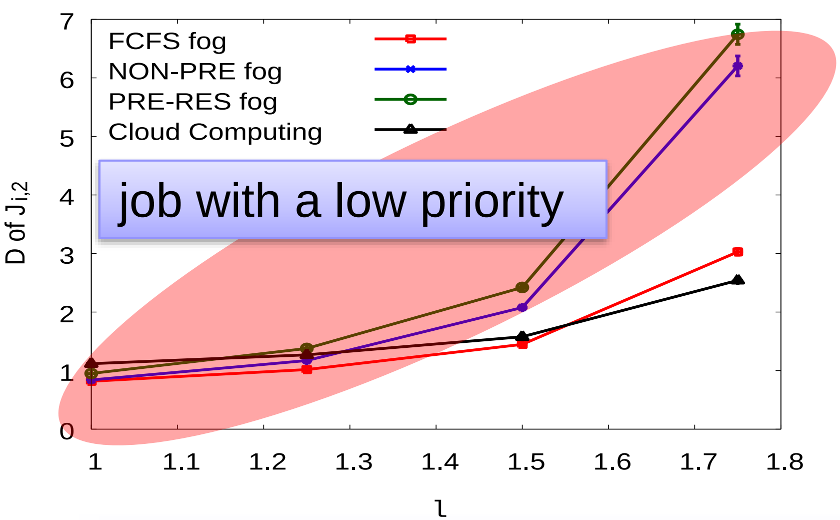
Even if Jobs with a low priority are postponed (Blue and Green graph), it is possible to lower $D$ than that of Cloud Computing (graph of Black).

# Simulation results (2/2)

- increasing of the arrival rate vs. $D$ ($threshold = 2$)




For the jobs with a delay-sensitiveness, combining a priority control with the Job assignment is also effective if the arrival rate increases.




For the jobs with a not delay-sensitiveness, we need to think about how far they allow latency.