Profit-Maximizing Virtual Machine Trading in a Federation of Selfish Clouds

Presenter: Shengkai Shi

Jan. 10 , 2013

Problem background

- Cloud federation
- When and how should a selfish cloud trade VMs with others
- Correlated sub-problems

  - VM pricing

  - Job scheduling

  - Server provisioning

Algorithm

- Target:

  - Maximize the individual profit

  - Achieve a satisfactory social welfare

- Feature:

  - Lyapunov optimization

  - Truthful double auction

System model

Service demand

- F individual clouds
- M distinct VM types
- S distinct job types
  - m_s : the type of VM required by type-s job
  - g_s : the number of VM required by type-s job
  - d_s : the SLA of type-s job

System model (cont.)

Service demand

- N_i^m homogenous servers to provision type-m VM in cloud i
  - C_i^m : the maximum number of type-m VM in each server
  - $\sum_{m=1}^{\mathrm{M}} N_i^m$: the number of servers in cloud i
- $r_i^s(t)$: the amount of type-s job arriving at cloud i in t
- $p_i^s(t)$: the given price of type-s job in t

System model

Job scheduling

- $Q_s^i(t)$: the length of the queue for type-s job
- $D_s^i(t)$: the number of type-s jobs dropped
- ξ_s^i : the penalty to drop one type-s job
- $\mu_{ij}^s(t)$: the number of type-s jobs scheduled to cloud j by cloud i
- Queue update:

$$Q_i^s(t+1) = \max\{Q_i^s(t) - \sum_{j=1}^{F} \mu_{ij}^s(t) - D_i^s(t), 0\} + r_i^s(t), \forall s \in [1, S], \forall i \in [1, F]$$

System model (cont.)

Job scheduling

- Z_i^s : a virtual queue reflecting the cumulated response delay of jobs from the respective job queue

$$Z_i^s(t+1) = \max\{Z_i^s(t) + 1_{\{Q_i^s(t)>0\}} \cdot [\varepsilon_s - \sum_{j=1}^{F} \mu_{ij}^s(t)] - D_i^s(t) - 1_{\{Q_i^s(t)=0\}} \cdot \sum_{j=1}^{F} \frac{C_j^{m_s} \cdot N_j^{m_s}}{g_s}, 0\}$$

$$\forall s \in [1, S], \forall i \in [1, F]$$

System model

Server provisioning

- Electricity cost

  - main component of operation cost
- $\beta_i(t)$: the operation cost of cloud i in t
- $n_i^m(t)$: the number of active servers provisioning type-m VMs at cloud i in t
- Constrain:

$$\sum_{j\in[1,F]} \sum_{s:m_{s=m},s\in[1,S]} g_s \mu_{ij}^s(t) \le C_i^m \cdot n_i^m(t)$$

$$n_i^m(t) \le N_i^m, \forall m \in [1,M], \forall s \in [1,F]$$

Double auction

- Buyer and seller
  - $\langle b_i^m(t), \gamma_i^m(t) \rangle$: the unit price and maximum quantity at which cloud i is willing to buy type-m VMs
  - $\langle s_i^m(t), \eta_i^m(t) \rangle$: the unit price and maximum quantity at which cloud i is willing to sell type-m VMs

Double auction (cont.)

- Auctioneer

  - a broker in the cloud federation

  - $\overset{\square}{b}_i^m(t)$ and $\overset{\square}{\gamma}_i^m(t)$: the actual unit charge price and actual quantity for cloud i to buy type-m VMs respectively

  - $\overset{\square}{s}_i^m(t)$ and $\overset{\square}{\eta}_i^m(t)$: the actual unit income and actual quantity for cloud i to sell type-m VMs respectively

Double auction (cont.)

$$\hat{\gamma}_i^m(t) = \sum_{j \in [1,F],\, j \neq i} \alpha_{ij}^m(t), \forall m \in [1, M], i \in [1, F]$$

$$\hat{\eta}_i^m(t) = \sum_{j \in [1,F],\, j \neq i} \alpha_{ji}^m(t), \forall m \in [1, M], \forall i \in [1, F]$$

- Here,
  - $\alpha_{ij}^m(t)$: the number of type-m VMs purchased by cloud i by cloud j from cloud

Double auction (cont.)

- Economic properties for double auction

  - truthfulness

  - individual rationality

  - ex-post budget balance

Truthfulness

- Definition:

A truthful double auction is employed at the auctioneer, where sellers and buyers bid their true values of the prices and quantities, in order to maximize their individual utilities

- $\tilde{b}_i^m(t)$ and $\tilde{s}_i^m(t)$: the true value of buying and selling a type-m VM respectively
- $\tilde{\gamma}_i^m(t)$ and $\tilde{\eta}_i^m(t)$: the true value of the quantity to buy and sell type- m VMs respectively

Individual selfishness

- Revenue

  - job service charge paid by customers

$$\Phi_1^i = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{s \in [1,S]} E\{ p_i^s(t) \cdot r_i^s(t) \}, \forall i \in [1, F]$$

  - income from selling VMs

$$\Phi_2^i = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{m \in [1,M]} E\{ \hat{s}_i^m(t) \cdot \hat{\eta}_i^m(t) \}, \forall i \in [1, F]$$

Individual selfishness (cont.)

- Cost

  - cost for running active servers

$$\Psi_1^i = \lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\sum_{s\in[1,S]} E\{\beta_i(t)\cdot\sum_{m=1}^{M} n_i^m(t)\}, \forall i \in [1,F]$$

  - penalties for dropped jobs

$$\Psi_2^i = \lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\sum_{s\in[1,S]} E\{\xi_i^s \cdot D_i^s(t)\}, \forall i \in [1,F]$$

  - cost for buying VMs from other clouds

$$\Psi_3^i = \lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1} E\{\sum_{m\in[1,M]} \hat{b}_i^m(t)\cdot\hat{\gamma}_i^m(t)\}, \forall i \in [1,F]$$

Individual selfishness (cont.)

- Profit maximization

$$\max \Phi_1^i + \Phi_2^i - \Psi_1^i - \Psi_2^i - \Psi_3^i$$

- Social welfare

  - overall profit of the cloud federation

$$\sum_{i \in [1,F]} \Phi_1^i + \Phi_2^i - \Psi_1^i - \Psi_2^i - \Psi_3^i$$

  - maximization

$$\sum_{i \in [1,F]} \Phi_1^i - \Psi_1^i - \Psi_2^i$$

Dynamic algorithms

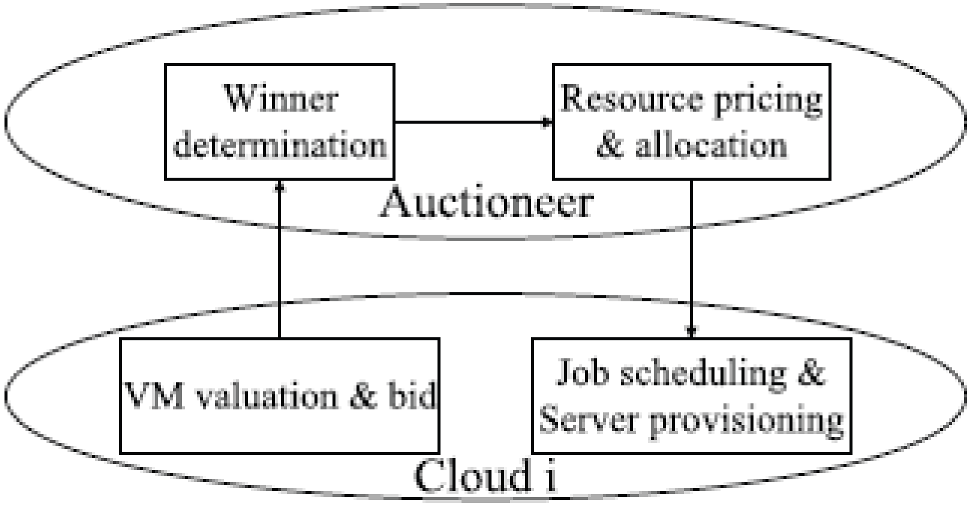

Winner determination
Resource pricing & allocation
Auctioneer
VM valuation & bid
Job scheduling & Server provisioning
Cloud i

Figure: Key algorithm modules

Individual profit maximization

- Drift-plus-penalty framework in Lyapunov optimization theory
- One-shot optimization problem:

$$\max \varphi_1^i(t) + \varphi_2^i(t) + \varphi_3^i(t)$$

where

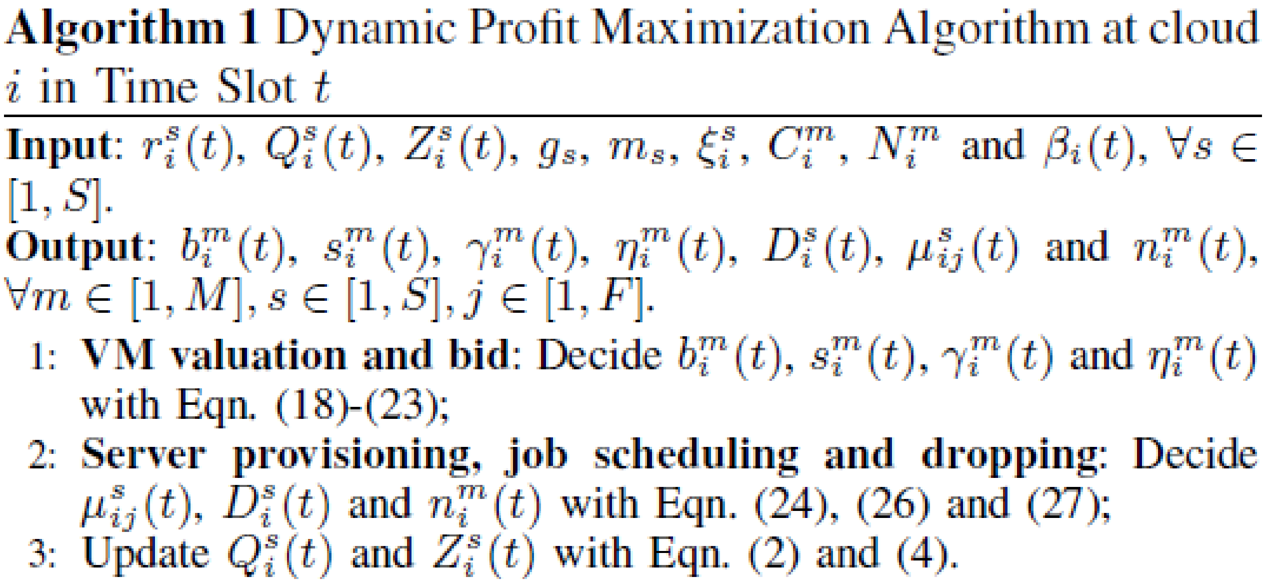
$$\varphi_1^i = V \sum_{m\in[1,M]} [\hat{s}_i^m(t)\hat{\eta}_i^m(t) - \hat{b}_i^m(t)\hat{\gamma}_i^m(t) - \beta_i(t) n_i^m(t)]$$

$$\varphi_2^i = \sum_{s\in[1,S]} \sum_{j\in[1,F]} \mu_{ij}^s(t)[Q_i^s(t) + Z_i^s(t)]$$

$$\varphi_3^i = \sum_{s\in[1,S]} D_i^s(t)[Q_i^s(t) + Z_i^s(t) - V \cdot \xi_i^s]$$

Individual profit maximization (cont.)

- Dynamic profit maximization algorithm



Algorithm 1 Dynamic Profit Maximization Algorithm at cloud i in Time Slot t

```
Input: $r_i^s(t)$, $Q_i^s(t)$, $Z_i^s(t)$, $g_s$, $m_s$, $\xi_i^s$, $C_i^m$, $N_i^m$ and $\beta_i(t)$, $\forall s \in [1, S]$.
Output: $b_i^m(t)$, $s_i^m(t)$, $\gamma_i^m(t)$, $\eta_i^m(t)$, $D_i^s(t)$, $\mu_{ij}^s(t)$ and $n_i^m(t)$, $\forall m \in [1, M], s \in [1, S], j \in [1, F]$.
  VM valuation and bid: Decide $b_i^m(t)$, $s_i^m(t)$, $\gamma_i^m(t)$ and $\eta_i^m(t)$ with Eqn. (18)-(23);
  Server provisioning, job scheduling and dropping: Decide $\mu_{ij}^s(t)$, $D_i^s(t)$ and $n_i^m(t)$ with Eqn. (24), (26) and (27);
  Update $Q_i^s(t)$ and $Z_i^s(t)$ with Eqn. (2) and (4).
```

VM valuation

- True value of the price to buy (sell)

  - equal to this value, profit remains the same

  - lower than this value, a profit gain (loss) occurs

  - higher than this value, a profit loss (gain) occurs

VM valuation (cont.)

- True values of the buy and sell prices

$$\tilde{b}_i^m(t) = \frac{Q_i^{s_m^*}(t) + Z_i^{s_m^*}(t)}{V \cdot g_{s_m^*}}$$

$$\tilde{s}_i^m(t) = \begin{cases} \dfrac{Q_i^{s_m^*}(t) + Z_i^{s_m^*}(t)}{V \cdot g_{s_m^*}} & if\ \dfrac{Q_i^{s_m^*}(t) + Z_i^{s_m^*}(t)}{V \cdot g_{s_m^*}} > \beta_i(t) \Big/ C_i^m \\ \beta_i(t) \Big/ C_i^m & Otherwise \end{cases}$$

where

$$s_m^* = \arg \max_{s' \in [1,S], m_{s'} = m} \left\{ W_i^{s'}(t) \right\}$$

$$W_i^{s'}(t) = \frac{Q_i^{s'}(t) + Z_i^{s'}(t)}{g_{s'}}$$

VM valuation (cont.)

- True volumes of type-m VMs to buy and sell

$$\tilde{\gamma}_i^m(t) = \sum_{i\in[1,F]} C_i^m N_i^m$$

$$\tilde{\eta}_i^m(t) = C_i^m N_i^m$$

  - $\tilde{\gamma}_i^m(t)$: the number of all potential type-m VMs

- Bidding price

$$b_i^m(t) = \tilde{b}_i^m(t), s_i^m(t) = \tilde{s}_i^m(t)$$

$$\gamma_i^m(t) = \tilde{\gamma}_i^m(t), \eta_i^m(t) = \tilde{\eta}_i^m(t)$$

Double auction

- Winner determination

  - sort bids in non-decreasing (for sellers) and non-increasing (for buyers) orders:

$$\theta_1^m(t) \geq \theta_2^m(t) \geq \ldots \geq \theta_N^m(t)$$

$$v_1^m(t) \leq v_2^m(t) \leq \ldots \leq v_N^m(t)$$

  - find $j' = \arg\max v_j^m(t) \leq \theta_j^m(t)$, the largest index in the sorted sequence of sell-bids such that,

$$v_{j'}^m(t) \leq \theta_2^m(t), v_{j'+1}^m(t) > \theta_2^m(t)$$

Price and allocation

- Uniform clearing price

  - the actual charge price to each buyer cloud

$$\hat{b}_i^m(t) = \begin{cases} \theta_2^m(t)\, ifb_i^m(t) wins \\ 0 \quad Otherwise \end{cases}$$

  - the actual price paid to each seller cloud

$$\hat{s}_i^m(t) = \begin{cases} v_{j'}^m(t)\, ifs_i^m(t) wins \\ 0 \quad Otherwise \end{cases}$$

Price and allocation (cont.)

- Uniform clearing price

  - the number of type-m VMs bought by cloud

$$\hat{\gamma}_i^m(t) = \begin{cases} \sum_{j=1}^{j'-1} L_j^m(t) \, ifb_i^m(t) wins \\ 0 \qquad Otherwise \end{cases}$$

  - the number of type-m VMs sold by cloud

$$\hat{\eta}_i^m(t) = \begin{cases} \eta_i^m(t) \, ifs_i^m(t) wins \\ 0 \quad Otherwise \end{cases}$$

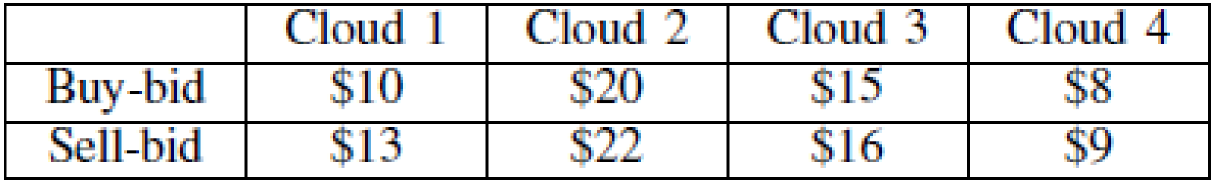
Price and allocation (cont.)

- Uniform clearing price

  - the number of type-m VMs sold from cloud j to cloud i

$$\alpha_{ij}^{m}(t) = \begin{cases} \eta_j^m(t)\, if s_i^m(t) and b_i^m(t) win \\ 0 \qquad Otherwise \end{cases}$$

Example



	Cloud 1	Cloud 2	Cloud 3	Cloud 4
Buy-bid	$10	$20	$15	$8
Sell-bid	$13	$22	$16	$9

- Buy-bidder winner: Cloud 2
- Sell-bidder winner: Cloud 4
- Actual clearing buy price: $15
- Actual clearing sell price: $13

Job scheduling

- The number of type-m jobs scheduled to local servers

$$\mu_{ii}^{s}(t)=\begin{cases}\dfrac{C_i^{m_s}\cdot N_i^{m_s}-\sum_{j\neq i}\alpha_{ij}^{m_s}(t)}{g_s} & if\ \dfrac{Q_i^s(t)+Z_i^s(t)}{V\cdot g_s}>\beta_i(t)\Big/C_i^{m_s}\ and s=s_{m_s}^{*}\\ 0 & Otherwise\end{cases}$$

- The number of type-m jobs scheduled to other cloud $(j\neq i)$

$$\mu_{ij}^{s}(t)=\begin{cases}\dfrac{\alpha_{ij}^{m_s}(t)}{g_s} & \begin{matrix} if \\ s=s_{m_s}^{*}\end{matrix}\\ 0 & Otherwise\end{cases}$$

Job dropping

- The number of type-s jobs dropped by cloud i

$$D_i^s(t) = \begin{cases} D_s^{(\max)}(t)\, if Q_i^s(t) + Z_i^s(t) > V \cdot \xi_i^s \\ 0 \qquad\qquad Otherwise \end{cases}$$

Here,

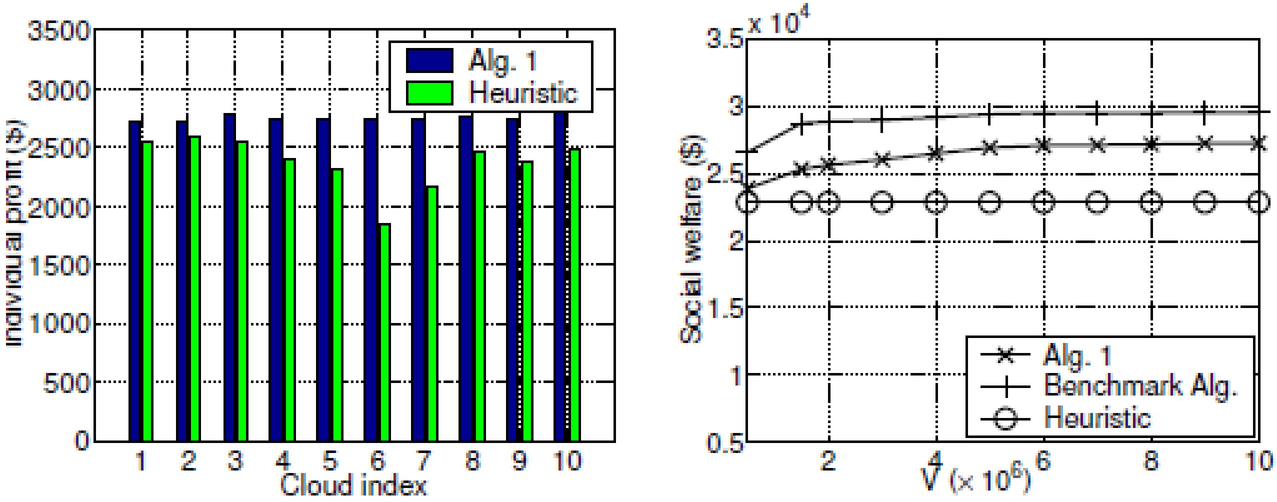
- $D_s^{(\max)}$: the maximum number of type-s jobs dropped by cloud i

Server provisioning

- The number of activated servers at cloud i to provision type-m VM

$$n_i^m(t) = \frac{\sum_{s \in [1,S], m_s = m} \mu_{ii}^s(t) \cdot g_s + \sum_{j \neq i} \alpha_{ji}^m(t)}{C_i^m}$$

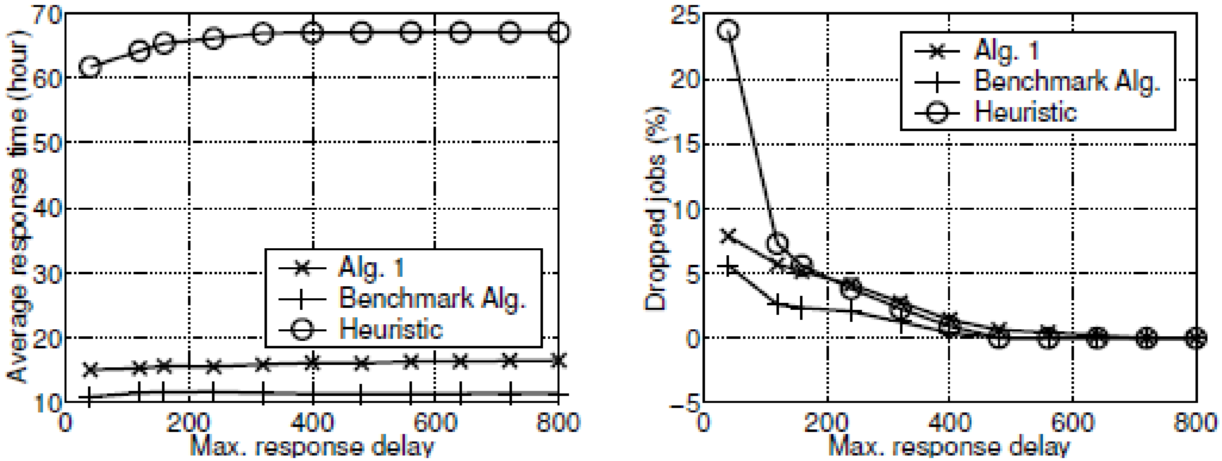
Simulation results



(a) Individual Profit with $V = 10^7$ (b) Social Welfare

Figure: Comparisions of individual profit and social welfare

Simulation results (cont.)



(a) Average response delay (b) Averaged job-drop percentage

Figure: Comparisons of job scheduling delay and drop percentage

Conclusion

- Selfishness of individual cloud is addressed
- A truthful, individual-rational, ex-post budget-balanced double auction
- Online VM pricing, job scheduling, and server provisioning
- Theoretical analysis and simulation study

Thank you!