Destination Selection Algorithm in a Server Migration Service

Asato Yamanaka¹, Yukinobu Fukushima¹, Tutomu Murase², Tokumi Yokohira¹ and Tatsuya Suda³

The Graduate School of Natural Science and Technology, Okayama University, Japan¹
Cloud System Research Laboratories, NEC Corporation, Japan²
University Netgroup Inc, USA³
IaaS cloud service is attracting attention

NW-Apps (e.g., online games application) can operate their servers at a data center without any initial cost

Location of server is fixed at a data center

It is difficult for an IaaS provider to provide NW-Apps with good communication QoS
Server migration service (SMS) can improve communication QoS in IaaS cloud service.

In SMS, we need to:

- Satisfy SLA of as many clients as possible
- Decrease the migrating server’s negative impact (network impact) on its background traffic

by appropriately determining where to locate servers.

![Diagram showing network with NW-App's client, NW-App's server, and Working Place (WP). The diagram highlights negative impact.](image-url)
Research Objective

- Previous research of server migration
  - Live migration of VMs [1]
    - Decreasing downtime of a migrating server
    - No consideration of network impact
    - Server’s destination is determined in advance

- Research objective
  - Proposing destination selection algorithms that try to decrease network impact while satisfying SLA of as many clients as possible

Model of Network Application (NW-App)
Model of Network Environment
Procedures of Server Migrasion Service

NW-App developer and the server migration service provider make an SLA about communication QoS.

The provider always checks whether the SLAs (delays, throughputs, packet loss, etc) are satisfied or not.

Not satisfied

The provider executes destination selection algorithm, determines destination WP and moves server in the new WP (Only one server can migrate).
Quantification of Network Impact

Definition

Number of bits of background traffic that suffer negative effect for the period in which server migration traffic is being transmitted

\[ NI = \sum_{i=1}^{n-1} A_i T_i \]

- \( A_i \): Bit rate of background traffic on \( L_i \) [bps]
- \( T_i \): Migration time on link \( L_i \) [s]
# Destination Selection Algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Destination WP</th>
<th>Expected effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Impact Algorithm (MIA)</td>
<td>Migrate to WP with minimum impact</td>
<td>Decrease total NW impact</td>
</tr>
<tr>
<td>Maximum Remaining space Algorithm (MRA)</td>
<td>Migrate to WP with the maximum remaining space</td>
<td>Increase the migration success rate in the future</td>
</tr>
<tr>
<td>Maximum Covering Algorithm (MCA)</td>
<td>Migrate to WP with the maximum coverage</td>
<td>Decrease the number of migrations in the future</td>
</tr>
</tbody>
</table>

**Coverage**: The number of routers which the communication QoS is satisfying the SLA between router and WP

**SLA**: less than 3ms

## Selection flow

1. List candidates for destination WP (non-full-capacity, all SLAs satisfied)
2. Among the candidates, select a destination WP based on each algorithm
Example of Server Migration

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Destination WP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Impact Algorithm (MIA)</td>
<td>Migrate to WP with minimum impact</td>
</tr>
<tr>
<td>Maximum Remaining space Algorithm (MRA)</td>
<td>Migrate to WP with the maximum remaining space</td>
</tr>
<tr>
<td>Maximum Covering Algorithm (MCA)</td>
<td>Migrate to WP with the maximum coverage</td>
</tr>
</tbody>
</table>

SLA: less than 3ms
### Simulation Models

**Parameter** | **Value**
--- | ---
Number of routers | 14
Number of WPs | 14
Capacity of WP | 4
Number of servers | 7
Number of clients | 1000
Number of servers in NW App 1 | 4
Number of servers in NW App 2 | 3
SLA | 15~23ms
Link bandwidth | 10Gbps
Background traffic | 1Gbps
Server size | 500Mbyte

**Evaluation Index**
1. number of *accommodatable clients*
2. Total network impact divided by number of accommodatable clients

*Accommodatable clients: The number of clients whose SLA are satisfied*
Evaluation Result (Accommodatable Clients)

Full-cover WP can accommodate all servers
All clients are accommodated

MCA shows 35~40% smaller accommodatable clients than MIA and MRA

Upper bound on end-to-end delay (x) [ms]
Why MCA shows smaller accommodatable clients than MIA and MRA

MCA

Non-full-cover WP

Full-cover WP

MIA, MRA

Full-cover WP is filled up later than MCA

Evaluation Result (Explanation)
Evaluation Result (Network Impact)

MRA always shows larger impact than MIA and MCA

MCA shows smaller impact than MIA
Conclusions and Future Works

Conclusions

- Propose and evaluate destination selection algorithms
- When the capacity of the full-cover WP is smaller than the total number of servers
  - MIA shows the best performance
- When the capacity of the full-cover WP is equal to or larger than the total number of servers
  - All the algorithm accommodates almost the same number of clients
  - MCA shows the best performance as to network impact

Future Works

- Realization of push-out function
- Performance evaluate in terms of downtime of NW-Apps
- Design of an algorithm for server replication