Cost-aware Virtual Machines Placement Problem under constraints over a Distributed Cloud Infrastructure

Eya DHIB, Khaled BOUSSETTAy, Nawel ZANGAR and Nabil TABBANE

MEDIATRON Laboratory, SUP’COM, Tunisia
L2TI Laboratory, Galilée Institut, Paris 13 university, France
Summary

1. Massively Multiplayers Online Gaming (MMOG)

2. Distributed MMOG architecture

3. Cost-aware Virtual Machines Placement Problem

4. Experiments results

5. Conclusion & perspectives
**MMOG**

- Massively Multiplayers Online Games
- Popular large scale game service (20 millions worldwide players in 2010*)

![Avatars]

- Delay-sensitive game service:

  ![Diagram](image.png)

  ➔ How maintain tradeoff between cost and delay for better game experience?

Distributed MMOG architecture
Cost-aware Virtual Machines Placement Problem (1/2)

• **Goal:**
  - Find optimal VMs placement that minimize resources allocation cost under delay constraint.

• **Notifications:**
  - multiple multidimensional knapsack problem
    - M Datacenters = M knapsacks
    - V VMs = V items to be placed in the knapsacks
    - \( w_{m,v} = (w_{m,v}^1, w_{m,v}^2, ..., w_{m,v}^K) \): **required resources** vector
      ( \( w_{m,v}^1 \): CPU, \( w_{m,v}^2 \): memory, \( w_{m,v}^3 \): bandwidth, \( w_{m,v}^4 \): space disk)
    - \( b_m = (b_m^1, b_m^2, ..., b_m^K) \): **capacity resources** vector
      ( \( b_m^1 \): CPU, \( b_m^2 \): memory, \( b_m^3 \): bandwidth, \( b_m^4 \): space disk)
Cost-aware Virtual Machines Placement Problem (2/2)

- **Notifications:**
  - \(c_{m,v}\): allocation cost of \(v^{th}\) item if placed in \(m^{th}\) knapsack
  - \(x_{m,v}\): decision variable
    - \{1\}: if \(v^{th}\) item is placed in \(m^{th}\) knapsack
    - \{0\}: otherwise
  - \(D_{m,v}\): response delay of \(v^{th}\) item if placed in \(m^{th}\) knapsack
  - \(Dt_{max}\): threshold delay
Cost-aware Virtual Machines Placement Problem

\[
\text{minimize} \left( \sum_{m=1}^{M} \sum_{v=1}^{V} c_{m,v} x_{m,v} \right) \tag{1}
\]

\[
D_{m,v} x_{m,v} \leq D_{t_{\text{max}}}, \forall m, \forall v \tag{2}
\]

\[
\sum_{v=1}^{V} w_{m,v}^{k} x_{m,v} \leq b_{m}^{k}, m = 1..M, k = 1..4 \tag{3}
\]

\[
\sum_{m=1}^{M} x_{m,v} = 1 \tag{4}
\]

\[
x_{m,v} \in \{0,1\} \tag{5}
\]
N, Z, Vmi, πZ = NULL

1. Sort all Vmi depending on performances and allocation cost in ascending order
2. Decide necessary amount of resources to allocate V ^Mi based on controller

\[ VM_i \geq VM_i^\hat{} \]

1. Calculate the allocation cost C(z; pz; v) relative to the VMi
2. Calculate the processing delay D(z; pz; v) relative to the Vmi
3. \[ z = \text{linprog}(VM_i, C(z; pz; v), D(z; pz; v)); \]

4. πZ = NULL?
   - Yes: Upgrade to the next configuration scenario
   - No: VM not placed

End
Experiments scenario

Architecture

AMAZON EC2 : M = 33 datacenters distributed all over the world

- 10 Physical servers
  - 2048 Mb memory
  - $10^5$ Mb storage space
  - $10^4$ Mb bandwidth
  - 4 processors; 100 MIPS

- Several VMs
- VMs characteristics offered by AMAZON EC2 platform
  - V : "World of Warcraft" (WoW) Cloud game
  - Delay threshold = 500 ms
Experiments results: Cost

- arbitrary VMs placement over data centers
- powerful VMs placement in first closest datacenter: No attention to cost
- costless VMs placement in first available datacenter: No attention to delay
- VMs placement with attention to cost and delay

![Bar chart showing monthly allocation cost ($) vs. number of users per zone. The chart includes bars for Random, GAC, MAC, and Contribution. The graph indicates a decrease in cost with a 25% reduction compared to the baseline.](image)
Experiments results: Delay

arbitrary VMs placement over data centers

powerfull VMs placement in first closest datacenter: **No attention to cost**

costless VMs placement in first available datacenter: **No attention to delay**

VMs placement **with attention to cost and delay**

![Graph showing delay versus number of users per zone](image)

80%
### Experiments results: Comparision

<table>
<thead>
<tr>
<th></th>
<th>GAC</th>
<th>Random</th>
<th>MAC</th>
<th>Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cost</strong></td>
<td>Highest</td>
<td>High</td>
<td>Cheapest</td>
<td>Cheap</td>
</tr>
<tr>
<td><strong>Delay</strong></td>
<td>Shortest</td>
<td>Long with high number of players</td>
<td>Longest</td>
<td>Short</td>
</tr>
</tbody>
</table>
Conclusion & perspectives

• Contribution:
  - Improvement of resources management for a Cloud gaming service.
  - Optimizing the overall resources allocation cost and placement under delay constraint.

• Results:
  - successfully the balance between allocation cost, resources placement and delays.

• Perspectives:
  - Impact of VMs placement problem on the Quality of Experience (QoE) of Cloud gaming users.
  - Dynamic VMs placement problem for MMOG over time
Thank you

Questions?