Exploiting Classes of Virtual Machines for Scalable IaaS Cloud Management

C. Canali
R. Lancellotti

Dipartimento di Ingegneria “Enzo Ferrari”
Università di Modena e Reggio Emilia
Challenges of Cloud computing

- **Vision from a IaaS perspective:**
  - continuous growth

- **More VMs, more data, ...**
  - More data centers
  - Larger data centers

- **Growth by $10^2$ in 15 years**

- **Scalability problems due to the infrastructure size:**
  - Monitoring of so many VMs
  - Management of infrastructure
  - placement of VMs over physical nodes
VM placement challenges

- Large number of VMs
- → Many physical nodes
- Multiple metrics
- Sampling at multiple times
  - Complementary workload patterns
VM placement challenges

- Large number of VMs
- → Many physical nodes
- Multiple metrics
- Sampling at multiple times
  - Complementary workload patterns
- → A huge, multidimensional Tetris game...
Reference scenario

VMs of class 1

... P₁ → Prediction → F₁ → Consolidation model

Past resource utilization

Future resource utilization

VM placement decision

Infrastructure description

VMs of class c
Assumptions

- **VM placement:** periodic task
  - We consider consolidation window of 24 hour
- **Cloud provider has knowledge of VMs classes**
  - Information from PaaS/SaaS provider to IaaS provider
  - e.g., Elastic map-reduce, Elastic load balancer
  - IaaS can monitor and classify VMs (proposals available in literature)
Consolidation models

• **Consolidation model:**
  - Solution of optimization problem
  - Input: future resource requirements (per-VM or per-class), Infrastructure description

• **Available solutions:**
  - Multi-dimensional bin packing (MBP)
  - First Fit Decreasing Heuristic (FFD) – special case of bin packing: we consider only one dimension
  - Class-based placement (CBP)
Multi-dimensional bin packing

- **Single bin-packing problem** for whole data center
- Classes of VMs not considered
- **Multi-dimensional problem:**
  - Multiple time intervals
  - Multiple resources
Problem formulation

• Objective function:

\[ \min \sum_{n \in N} O_n \]

\[ \text{Minimize number of nodes used} \]

• Subject to:

\[ \sum_{n \in N} I_{n,m} = 1 \quad \forall m \in M \]

Resource requirement of VM \( m \) at time \( t \)

\[ \sum_{m \in M} R_{m,t} I_{n,m} \leq V_n O_n \quad \forall n \in N, \forall t \in T \]

Node capacity constraint

\[ I_{n,m} = \{0,1\} \]

Decision variable: VM \( m \) on node \( n \)

\[ O_n = \{0,1\} \]

Available resources on node \( n \)

\[ \forall n \in N \]

Decision variable: Node \( n \) is on/off
Computational challenges

● Number of node capacity constraints grows with:
  – Number of nodes
  – Number of time intervals considered

● Addressing scalability problems:
  – Wall time limit on optimizer
  – Reduce number of time intervals (e.g., instead of 5min intervals can consider 1h, 4h, 12h, 1d...)
  – Use of heuristics instead of optimal solution
  – Special case: if only one time interval is considered multi-dimensional bin packing → bin packing (FFD)
Class-based VM placement

- Build a small consolidation solution (B-block)
- Replicate solution as a building block
- Solve residual problem (E-Block)
Class-based VM placement

- Additional input:
  → number of B-blocks \( \bar{b} \)
- Choice: \( \bar{b} = \text{n. of VMs in class with minimum cardinality} \)
- Impact of \( \bar{b} \)
  → open issue
- Two bin packing problems (B- E-blocks)
- Major dimensionality reduction
B-block problem formulation

- **Objective function:**
  \[ \min \sum_{n \in N_b} O_n \] (Minimize number of nodes used)

- **Subject to:**
  \[ \sum_{n \in N_b} I_{n,m} = 1 \] (for class \( c \) at time \( t \))

  \[ \sum_{c \in C} \sum_{m \in B_c} R_{c,t} I_{n,m} \leq V_n O_n \] (Resource requirement constraint)

  \[ I_{n,m} = \{0,1\} \]

  \[ O_n = \{0,1\} \]

- **E-block problem formulation is similar**
Experimental setup

- Number of VMs from 150 to 1200
- 44 classes, each class [8-50] VMs
- Focus on CPU (only trace available) – Utilization: [0-100%]
- Each physical node has capacity of 800%
- Time intervals considered:
  - 5m (288 int.)
  - 1h (24 int.)
  - 12h (2 int.)
  - 1d (1 int.)
- IBM ILOG CPLEX Optimizer v12
- Maximum time for consolidation: 1800s (30m)
Experimental results

**MBP:**
- Optimal solution only for small problems ($\leq 200$)
- Reducing dimensionality improves scalability
- No acceptable solutions for large problems ($\geq 1200$)

**CBP:**
- Always reaches solution even with 5m time interval
- Solves to optimality for medium problems ($\leq 700$)

<table>
<thead>
<tr>
<th>VMs</th>
<th>CBP 5m</th>
<th>MBP 1d</th>
<th>MBP 12h</th>
<th>MBP 1h</th>
<th>MBP 5m</th>
</tr>
</thead>
<tbody>
<tr>
<td>150</td>
<td>S/S</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>200</td>
<td>S/S</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>250</td>
<td>S/S</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>300</td>
<td>S/S</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>400</td>
<td>S/S</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>N</td>
</tr>
<tr>
<td>500</td>
<td>S/S</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>N</td>
</tr>
<tr>
<td>600</td>
<td>S/S</td>
<td>L</td>
<td>L</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>700</td>
<td>S/S</td>
<td>L</td>
<td>L</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>800</td>
<td>L/S</td>
<td>L</td>
<td>L</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>900</td>
<td>L/S</td>
<td>L</td>
<td>L</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>1000</td>
<td>L/S</td>
<td>L</td>
<td>L</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>1100</td>
<td>L/S</td>
<td>L</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>1200</td>
<td>L/S</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>
Solution quality: small number of VMs

- Solution quality: relative to LP relaxation of the problem
  - Lower is better
- FFD: low quality results
- MBP:
  - 5m: best solution
  - Time interval reduction → lower quality
  - VM set size growth → lower quality
- CBP quality remains stable with problem size
Computation time

- FFD: very fast but inaccurate
- When problem size grows, MBP becomes slower may result in sub-optimal solutions (quality reduction)
- CBP: very fast → scalable solution for larger problems

<table>
<thead>
<tr>
<th>Consolidation model</th>
<th>150 VMs (B/E)</th>
<th>200 VMs</th>
<th>250 VMs</th>
<th>300 VMs</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBP 5m (B/E)</td>
<td>0.43/0.46</td>
<td>0.49/0.28</td>
<td>0.54/0.49</td>
<td>0.98/0.40</td>
</tr>
<tr>
<td>FFD 1d</td>
<td>0.05</td>
<td>0.05</td>
<td>0.06</td>
<td>0.07</td>
</tr>
<tr>
<td>MBP 1d</td>
<td>0.21</td>
<td>11.36</td>
<td>45.28</td>
<td>147.73</td>
</tr>
<tr>
<td>MBP 12h</td>
<td>4.13</td>
<td>79.39</td>
<td>1800(L)</td>
<td>1800(L)</td>
</tr>
<tr>
<td>MBP 1h</td>
<td>32.87</td>
<td>91.20</td>
<td>1800(L)</td>
<td>1800(L)</td>
</tr>
<tr>
<td>MBP 5 min</td>
<td>233.09</td>
<td>270.59</td>
<td>1800(L)</td>
<td>1800(L)</td>
</tr>
</tbody>
</table>
Solution quality: large number of VMs

- **MBP:**
  - VM set size growth
  - → lower quality
  - → need time interval reduction

- **FFD:** always worst performing

- **CBP:**
  - VM set size growth
  - → always reaches solution
  - → solution quality improves
Conclusions and future work

- The challenge of VM placement in cloud computing
- **Proposal of Class-based placement technique**
- Better scalability compared to alternatives:
  - Can manage larger problems
  - Higher quality solution within the same time frame
- **Future work:**
  - New experiments: larger data centers, more resources
  - Analysis of B-block size ($b$ parameter):
    - impact on performance, automatic estimation
  - Different optimization strategies (e.g., dynamic programming)
Acknowledgment

- The authors acknowledge the support of the UniMORE-FAR2014 project **SAMMClouds**
- SAMMClouds: **Secure and Adaptive Management of MultiClouds**
Exploiting Classes of Virtual Machines for Scalable IaaS Cloud Management

C. Canali

R. Lancellotti

Dipartimento di Ingegneria “Enzo Ferrari”
Università di Modena e Reggio Emilia