Developing Communication-aware Service Placement Frameworks in the Cloud Economy

Chao Chen, Ligang He, Bo Gao, Stephen A. Jarvis
Multitenant cloud system

- Cloud system with multiple tenants and services
  - Cloud Tenants: Users renting services and virtual machines.
  - Cloud system (Datacenter)
    - Cloud providers: deliver a level of QoS (Quality-of-Service), such as: Amazon Web Services, Microsoft Azure, Google Cloud Platform, etc.
    - Cloud Services: hosted by a collection of virtual machines running different jobs: data analysis, web servers, etc.
Example: NASDAQ OMX

- **T1:** UploadData
- **T2:** StartVMs
- **T3:** CreateNetwork
- **T4:** DistinctData
- **T5:** MapReduce
- **T6:** StoreResult

**S3:**
Amazon simple Storage Service

**EC2:**
Amazon Elastic Compute

**VPC:**
Amazon Virtual Private Cloud

**Direct Connect:**
Amazon Direct Connect

**EMR:**
Amazon Elastic MapReduce
Challenges

1. Task/Service invocations may vary according to dynamic system information, and it may be difficult to know the full picture of the tasks/workflows in the Cloud. This work is *service-oriented*, which does not focus on allocating resources for a set of specific tasks or workflows, but aim to allocate resources based on the interaction patterns between services.

2. When the services interact with each other, data might be(cached) communicated between them. If the Virtual machine(VMs) that host the services with frequent communications among themselves can be placed to the same Physical Machine(PM), the communication cost could be significantly reduced. *(The cost is huge!)*
Challenges

1. Task/Service invocations may vary according to dynamic system information, and it may be difficult to know the full picture of the tasks/workflows in the Cloud. This work is *service-oriented*, which does not focus on allocating resources for a *set of specific tasks or workflows*, but aim to allocate resources based on *the interaction patterns between services*.

*Leontief Input-Output Model in economy*

2. When the services interact with each other, data might be(cached) communicated between them. If the Virtual machine(VMs) that host the services with frequent communications among themselves can be placed to the same Physical Machine(PM), the communication cost could be significantly reduced. *(The cost is huge!)*

*Genetic algorithm to find out the “optimal” solution*
Leontief Input-Output Model
Leontief Input-Output Model

<table>
<thead>
<tr>
<th></th>
<th>Agriculture</th>
<th>Manufacturing</th>
<th>Services</th>
<th>Open Sector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agriculture</td>
<td>34.69</td>
<td>4.92</td>
<td>5.62</td>
<td>39.24</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>5.28</td>
<td>61.82</td>
<td>22.99</td>
<td>60.02</td>
</tr>
<tr>
<td>Services</td>
<td>10.45</td>
<td>25.95</td>
<td>42.03</td>
<td>130.65</td>
</tr>
<tr>
<td>Total Gross Output</td>
<td>84.56</td>
<td>163.43</td>
<td>219.03</td>
<td></td>
</tr>
</tbody>
</table>
Leontief Input-Output Model

### Exchange of Goods and Services in the U.S. for 1947 (in billions of 1947 dollars)

<table>
<thead>
<tr>
<th></th>
<th>Agriculture</th>
<th>Manufacturing</th>
<th>Services</th>
<th>Open Sector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agriculture</td>
<td>34.69</td>
<td>4.92</td>
<td>5.62</td>
<td>39.24</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>5.28</td>
<td>61.82</td>
<td>22.99</td>
<td>60.02</td>
</tr>
<tr>
<td>Services</td>
<td>10.45</td>
<td>25.95</td>
<td>42.03</td>
<td>130.65</td>
</tr>
<tr>
<td><strong>Total Gross Output</strong></td>
<td><strong>84.56</strong></td>
<td><strong>163.43</strong></td>
<td><strong>219.03</strong></td>
<td><strong>490.79</strong></td>
</tr>
</tbody>
</table>
Leontief Input-Output Model

Exchange of Goods and Services in the U.S. for 1947 (in billions of 1947 dollars)

<table>
<thead>
<tr>
<th></th>
<th>Agriculture</th>
<th>Manufacturing</th>
<th>Services</th>
<th>Open Sector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agriculture</td>
<td>34.69</td>
<td>4.92</td>
<td>5.62</td>
<td>39.24</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>5.28</td>
<td>61.82</td>
<td>22.99</td>
<td>60.02</td>
</tr>
<tr>
<td>Services</td>
<td>10.45</td>
<td>25.95</td>
<td>42.03</td>
<td>130.65</td>
</tr>
<tr>
<td>Total Gross Output</td>
<td>84.56</td>
<td>163.43</td>
<td>219.03</td>
<td></td>
</tr>
</tbody>
</table>

consumption matrix

\[
C = \begin{bmatrix}
.4102 & .0301 & .0257 \\
.0624 & .3783 & .1050 \\
.1236 & .1588 & .1919 \\
\end{bmatrix}
\]

demand vector

\[
d = \begin{bmatrix}
39.24 \\
60.02 \\
130.65 \\
\end{bmatrix}
\]
Leontief Input-Output Model

Exchange of Goods and Services in the U.S. for 1947 (in billions of 1947 dollars)

<table>
<thead>
<tr>
<th></th>
<th>Agriculture</th>
<th>Manufacturing</th>
<th>Services</th>
<th>Open Sector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agriculture</td>
<td>34.69</td>
<td>4.92</td>
<td>5.62</td>
<td>39.24</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>5.28</td>
<td>61.82</td>
<td>22.99</td>
<td>60.02</td>
</tr>
<tr>
<td>Services</td>
<td>10.45</td>
<td>25.95</td>
<td>42.03</td>
<td>130.65</td>
</tr>
<tr>
<td>Total Gross Output</td>
<td>84.56</td>
<td>163.43</td>
<td>219.03</td>
<td></td>
</tr>
</tbody>
</table>

consumption matrix

\[
C = \begin{bmatrix}
.4102 & .0301 & .0257 \\
.0624 & .3783 & .1050 \\
.1236 & .1588 & .1919
\end{bmatrix}
\]

demand vector

\[
d = \begin{bmatrix}
39.24 \\
60.02 \\
130.65
\end{bmatrix}
\]

x - Equilibrium Production Level

\[
x = Cx + d
\]

\[
x = (I - C)^{-1}d = \begin{bmatrix}
82.40 \\
138.85 \\
201.57
\end{bmatrix}
\]
Modern industry ecosystem

Households buy output of business as the final demand

Households

Goods

Business

Labor

Business purchases from other business to produce their own goods

Business

Business
Cloud system returns the result as production

Cloud tenant requests services as the final demand

Service communicates each other service to meet external demand

Results

Service

Service

Service

Service
## Consumption Matrix

<table>
<thead>
<tr>
<th></th>
<th>Service 0</th>
<th>Service 1</th>
<th>Service 2</th>
<th>Service 3</th>
<th>Service 4</th>
<th>External demand</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service 0</td>
<td>c{0,0}</td>
<td>c{0,1}</td>
<td>c{0,2}</td>
<td>c{0,3}</td>
<td>c{0,4}</td>
<td>λ0</td>
</tr>
<tr>
<td>Service 1</td>
<td>c{1,0}</td>
<td>c{1,1}</td>
<td>c{1,2}</td>
<td>c{1,3}</td>
<td>c{1,4}</td>
<td>λ1</td>
</tr>
<tr>
<td>Service 2</td>
<td>c{2,0}</td>
<td>c{2,1}</td>
<td>c{2,2}</td>
<td>c{2,3}</td>
<td>c{2,4}</td>
<td>λ2</td>
</tr>
<tr>
<td>Service 3</td>
<td>c{3,0}</td>
<td>c{3,1}</td>
<td>c{3,2}</td>
<td>c{3,3}</td>
<td>c{3,4}</td>
<td>λ3</td>
</tr>
<tr>
<td>Service 4</td>
<td>c{4,0}</td>
<td>c{4,1}</td>
<td>c{4,2}</td>
<td>c{4,3}</td>
<td>c{4,4}</td>
<td>λ4</td>
</tr>
</tbody>
</table>

\( c\{2,1\} = 2 \)
Consumption relation

Service 1  Service 2
VM ➔ VM ➔ VM
VM

Service 1  Service 2
VM ➔ VM

Service 1  Service 2
VM ➔ VM

PM 3

PM 1

PM 2

c[2,1] = 2
c[2,1] = 2

c[2,1] = 2
c[2,1] = 2

c[2,1] = 2
c[2,1] = 2

WARWICK
Consumption relation

Service 1 -> Service 2

VM

PM 3

c\{2,1\} = 2

c\{2,1\} = 3

Service 1 -> Service 2

VM

PM 1

c\{2,1\} = 2

c\{2,1\} = 1

Service 1 -> Service 2

VM

PM 2

c\{2,1\} = 2

c\{2,1\} = 3

WARWICK
Consumption relation

Service 1  Service 2
VM → VM → VM
PM 3

Service 1  Service 2
VM
PM 1

Service 1  Service 2
VM → VM
PM 2

\[ c_{2,1} = 2 \]

\[ c_{2,1} = 3 \]

\[ c_{2,1} = 0 \]

\[ c_{2,1} = 3 \]
Communication Cost

\[ C(M) = \sum_{k=1}^{N} \sum_{j=1}^{M} \sum_{i=1}^{M} \beta_{ijk} \]  

(6)

\[ \beta_{ijk} = \begin{cases} 
  e_{ji} \times (f(j, R_j, v_{jk}) \times p_{ji} - f(i, R_i, v_{ik})) & \text{if } \alpha_{ijk} \leq c_{ij} \\
  0 & \text{otherwise} 
\end{cases} \]  

(7)

The objective is to find a VM-to-PM mapping such that \( C(M) \) is minimized, subject to certain constraints. This can be formalized as Eq. 8, where \( x_i \) is the number of VMs's

\[
\text{minimize} \quad C(M),
\]

\[
\text{subject to:} \quad \forall i: 1 \leq i \leq M, \sum_{k=1}^{N} v_{ik} = x_i
\]

\[ v_{ik} \geq 0 \]

The ratio of the number of VM\{i\} to the number of VM\{j\} in the PM n\{k\}

The total amount of data that have to be communicated in the Cloud caused by the inadequate resource capacity of S\{j\} in PM n\{k\} comparing with that of S\{j\} in the same PM.
Designing Genetic Algorithm for VMs Allocation Problem

1. Start

2. Generate a population of individuals of size $N$: $x_1, x_2, x_3, \ldots, x_N$

3. Calculate the fitness of each individual:
   $f(x_1), f(x_2), f(x_3), \ldots, f(x_N)$

4. Is the termination criterion satisfied?
   - Yes: Stop Evolution
   - No: Select a pair of individuals for crossover

5. With the crossover probability $P(c)$, exchange parts of the two selected individuals and create two offsprings

6. With the mutation probability $P(m)$, randomly change an individual

7. Replace the previous population with the new population

Calculate it via Eq. 5

$$C(\mathcal{M}) = \sum_{k=1}^{N} \sum_{j=1}^{M} \sum_{i=1}^{M} \beta_{ijk}$$
Genetic Algorithm: two points Crossover

Parent A

4  2  11

Parent B

8  1  9

Crossover point

Offspring A

4  1  9

Offspring B

8  2  11

Crossover point
Impact of the increase in external demands

(a) computation-intensive workflow

(b) general workflow

(c) communication-intensive workflow

(d) the three types of workflow combined
Comparing CAGA with mini-nodes and the round-robin heuristic in terms of communication cost

(a) computation-intensive workflow

(b) general workflow

(c) communication-intensive workflow
Comparing CAGA with mini-nodes and the round-robin heuristic in terms of number of used PMs.
Convergence speed of CAGA

a: general workflow

b: computation-intensive workflow

c: communication-intensive workflow
THANK YOU!
Question?