

## Multi-objective Optimization

### Features:

- Distributed Data Flow
- Fog nodes mobility
- IoT devices mobility
- Migration support
- Partitioning techniques
- Data placement optimization
- Migration Optimization

### Variables:

- QoS
  - latency
- Cost
  - CPU
  - RAM
  - MEM
  - BW
- Energy
  - busyPower
  - idlePower
- Bandwidth

## Notations:

- $N$  – fog nodes  $S = \{s_1, \dots, s_n\}$
- $M$  – modules  $U = \{u_1, \dots, u_m\}$
- $fP^{Mips}$  – matrix  $1 \times N$  representing the MIPS price of each fog node per unit
- $fP^{Ram}$  – matrix  $1 \times N$  representing the memory price of each fog node per unit
- $fP^{Strg}$  – matrix  $1 \times N$  representing the storage price of each fog node per unit
- $fP^{Bw}$  – matrix  $1 \times N$  representing the bandwidth price of each fog node per unit
- $f^{Mips}$  – matrix  $N \times 1$  representing the MIPS capacity of each fog node
- $f^{Ram}$  – matrix  $N \times 1$  representing the memory capacity of each fog node
- $f^{Strg}$  – matrix  $N \times 1$  representing the storage capacity of each fog node
- $f^{bPw}$  – matrix  $N \times 1$  representing the busy power consumption of each fog node per unit
- $f^{iPw}$  – matrix  $N \times 1$  representing the idle power consumption of each fog node per unit
- $f^{wPw}$  – matrix  $N \times 1$  representing the weight of energy consumption (*willing\_to\_waste\_energy*<sup>-1</sup>)
- $m^{Mips}$  – matrix  $M \times 1$  representing the MIPS needed for each application's module
- $m^{Ram}$  – matrix  $M \times 1$  representing the memory needed for each application's module
- $m^{Strg}$  – matrix  $M \times 1$  representing the storage needed for each application's module
- $m^{Bw}$  – matrix  $M \times 1$  representing the bandwidth needed for each application's module
- $K$  – total number of dependencies/edges
- $e^{Cpu}$  – matrix  $1 \times K$  representing the tuple CPU size (MI) needed to be processed
- $e^{Nw}$  – matrix  $1 \times K$  representing the tuple network size (MB) needed to be sent
- $e^{Prob}$  – matrix  $1 \times K$  representing the probability of sending the tuple
- $e^{Pe}$  – matrix  $1 \times K$  representing the periodicity of the producer (i.e., periodic sources)
- $e^S$  – matrix  $1 \times K$  representing the edge source
- $e^D$  – matrix  $1 \times K$  representing the edge destination
- $Z$  – total number of dependencies/edges between different pairs of nodes
- $l^S$  – matrix  $N \times 1$  representing the starting nodes of each pair of nodes
- $l^D$  – matrix  $N \times 1$  representing the ending nodes of each pair of nodes
- $mD$  – matrix  $M \times M$  representing the dependencies between modules
- $mB$  – matrix  $M \times M$  representing the bandwidth needed between modules
- $fL$  – matrix  $N \times N$  representing the latency between each two direct nodes
- $fB$  – matrix  $N \times N$  representing the bandwidth between each two direct nodes
- $D$  – matrix  $N \times M$  representing the nodes where each module can be deployed
- $P$  – matrix  $N \times M$  representing the placement mapping between modules and nodes
- $R$  – matrix  $Z \times N \times N$  representing the routing map between modules
- $e$  – matrix  $1 \times N$ , with all entries to 1
- $\alpha$  – operational weight
- $\beta$  – energetic weight
- $\gamma$  – processing weight
- $\delta$  – latency weight
- $\zeta$  – bandwidth weight

### Preliminary computations:

$$\begin{aligned}
m_i^{Mips} &= \sum_{k \in K} \left( \frac{e_k^{Prob} e_k^{Cpu}}{e_k^{Pe}} \right), e_k^D = i \\
m_i^{Bw} &= \sum_{k \in K} \left( \frac{e_k^{Prob} e_k^{Nw}}{e_k^{Pe}} \right), e_k^S = i \\
m_{B_{i,j}} &= \sum_{k \in K} \left( \frac{e_k^{Prob} e_k^{Nw}}{e_k^{Pe}} \right), e_k^S = i, e_k^D = j \\
m_{D_{i,j}} &= \sum_{k \in K} \left( \frac{e_k^{Prob}}{e_k^{Pe}} \right), e_k^S = i, e_k^D = j
\end{aligned}$$

### Problem formulation:

The cost function is mainly characterized by two components: Operational Cost and Service Quality Cost.

**Operational Cost** ( $C_O$ ) is characterized by the resources allocated in each fog node to support all users' computations, namely: CPU, memory, storage, and bandwidth.

$$C_O = fP^{Mips} \times P \times m^{Mips} + fP^{Ram} \times P \times m^{Ram} + fP^{Strg} \times P \times m^{Strg} + \left( fP^{Bw} \sum_{z \in Z} m_{B_{l_z^S, l_z^D}} \times R_z \right) e'$$

**Power Cost** ( $C_{Pw}$ ) is characterized by the busy/idle power in each fog node to support all users' computations as well as the willing to waste energy to prevent clients to process the whole application.

$$C_{Pw} = \sum_{n \in N} \left( f_n^{iPw} + f_n^{wPw} (f_n^{bPw} - f_n^{iPw}) \times \frac{P_n \times m^{Mips}}{f_n^{Mips}} \right)$$

**Processing Cost** ( $C_P$ ) is characterized by the percentage of unused MIPS using the Jain's fairness index.

$$C_P = \frac{\left( \sum_{n \in N} \frac{P_n \times m^{Mips}}{f_n^{Mips}} \right)^2}{N \times \sum_{n \in N} \left( \frac{P_n \times m^{Mips}}{f_n^{Mips}} \right)^2}$$

**Latency Cost** ( $C_L$ ) is characterized by the total latency on the tuple transmission.

$$C_L = e \left( fL. * \sum_{z \in Z} m_{D_{l_z^S, l_z^D}} \times R_z \right) e'$$

**Bandwidth Cost** ( $C_B$ ) is characterized by the bandwidth usage on the tuple transmission.

$$C_B = e \left( 1. / (fB + \epsilon) . * \sum_{z \in Z} m_{B_{l_z^S, l_z^D}} \times R_z \right) e', \epsilon = 1^{-9}$$

### Final problem:

$$\begin{aligned}
&\underset{P, R}{\text{minimize}} && C = \alpha C_O + \beta C_{Pw} + \gamma C_P + \delta C_L + \zeta C_B \\
&\text{subject to} && P \times m^{Mips} \leq f^{Mips}, \\
& && P \times m^{Ram} \leq f^{Ram}, \\
& && P \times m^{Strg} \leq f^{Strg}, \\
& && P \leq D \\
& && P_{i,j} \in \{0, 1\}, \forall i \in [0, N], \forall j \in [0, M], \\
& && \sum_{i \in N} P_{i,j} = 1, \forall j \in [0, M] \\
& && R_{z,i,j} \in \{0, 1\}, \forall z \in [0, Z], \forall i \in [0, N], \forall j \in [0, N], \\
& && \sum_{j \in N} R_{z,i,j} - \sum_{j \in N} R'_{z,i,j} = P_{i, l_z^S} - P_{i, l_z^D}, \forall z \in [0, Z], \forall i \in [0, N]
\end{aligned}$$