ETRI Electronics and Telecommunications Research Institute

Optimal Service Placement using Pseudo Service Chaining Mechanism

[Playnet-MANO]

2016. 11. 15.

Taeheum Na : {taeheum@etri.re.kr}

Network SW Platform Research Section

ETRI

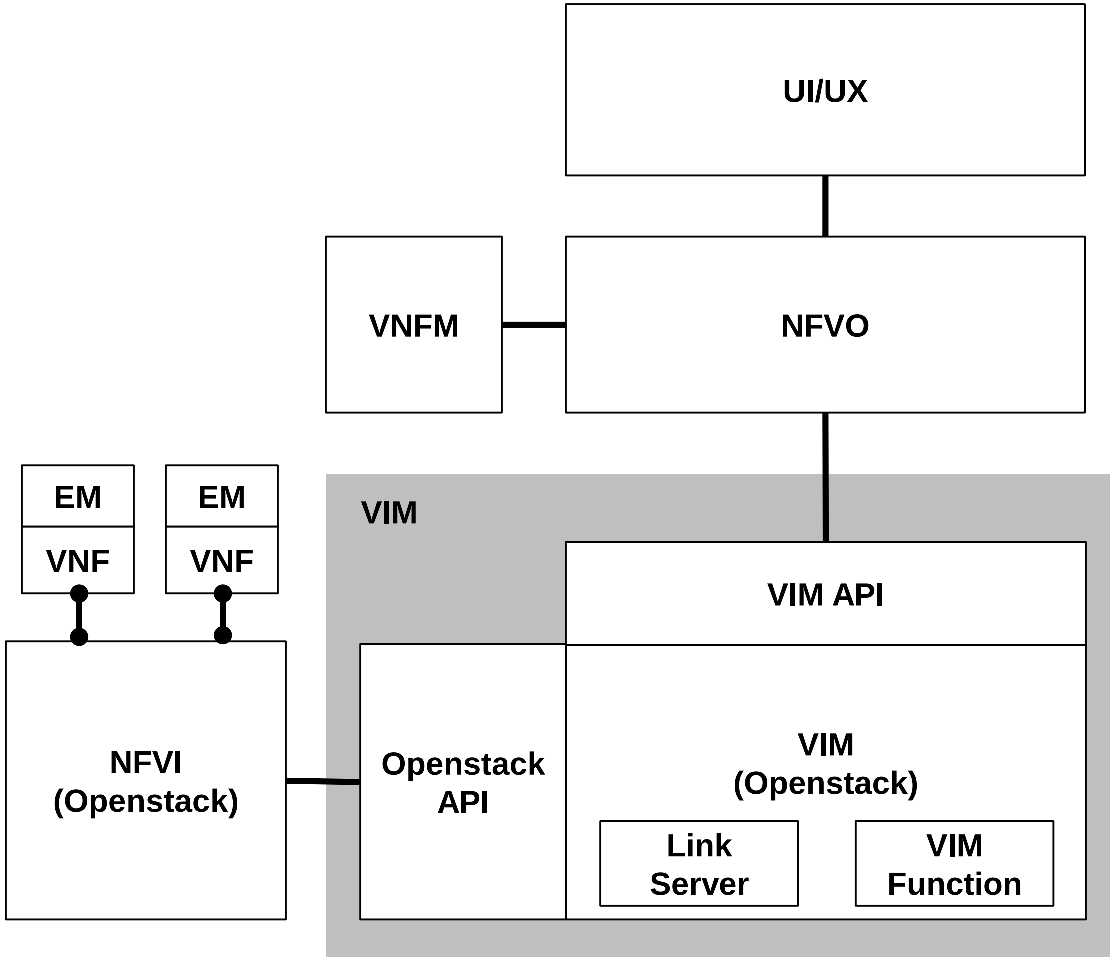
IT R&D Global Leader

Contents

Playnet-MANO

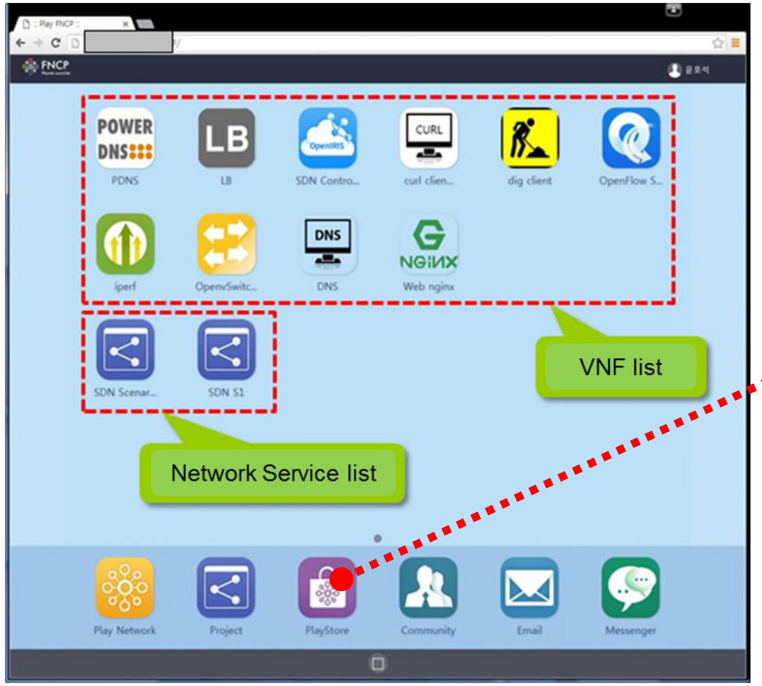
Playnet-MANO

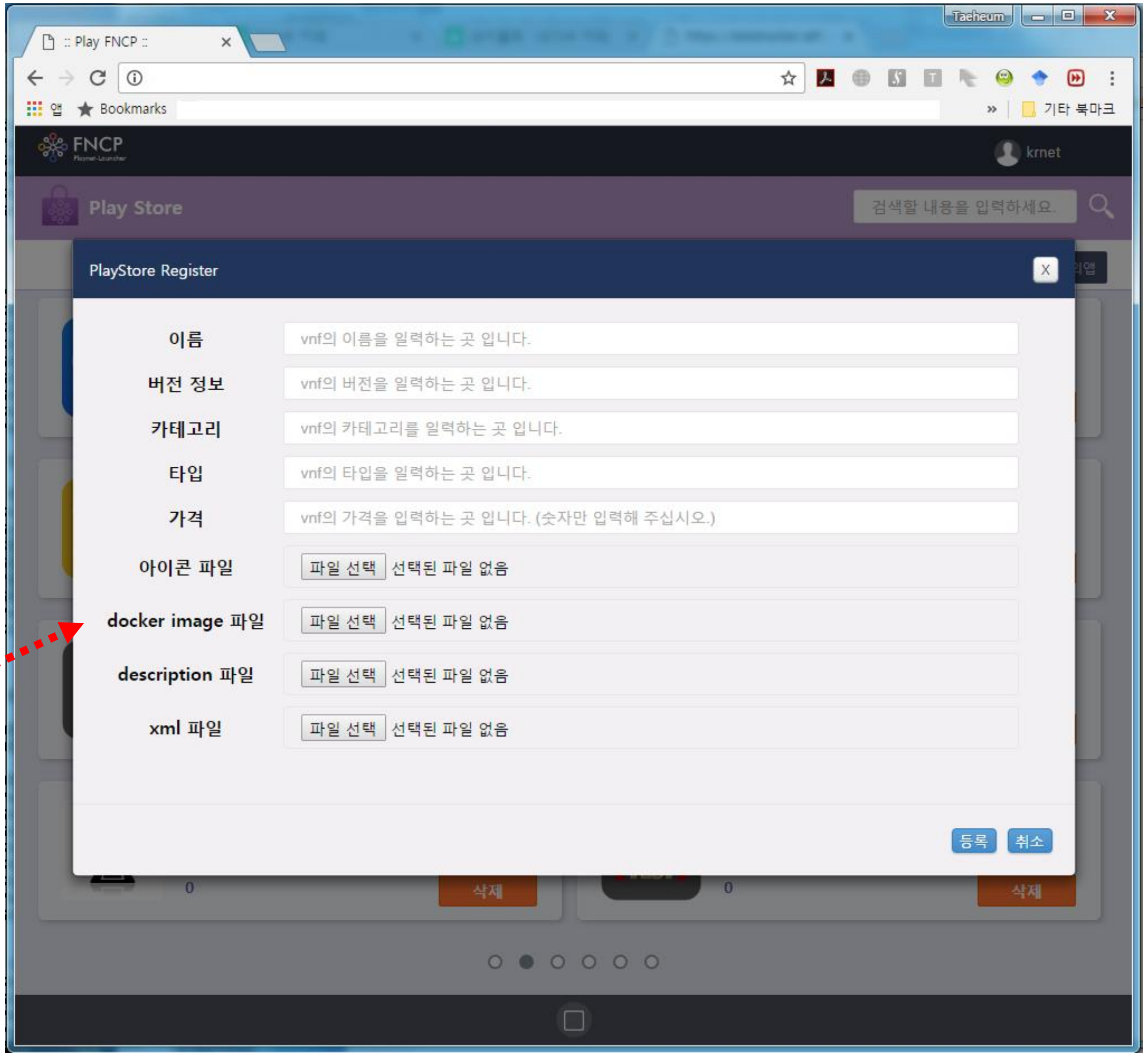
- Playnet-MANO = Playground for virtualized network application
  - Open Source MANO (OSM) based NFV Environment
  - Extended VIM functionality: OpenStack (liberty)
    - Container and KVM based virtualization
    - Using Nova-docker plugin
    - Consideration for point-to-point link (E-line type)
  - Saving & loading Network Service (NS)
    - Save and load NS using VNFFG format



Playnet Architecture

- Features of Playnet
  - VNF Management
    - GUI based Registering / deleting VNF at Playnet Store
    - Instantiation/termination/configuration



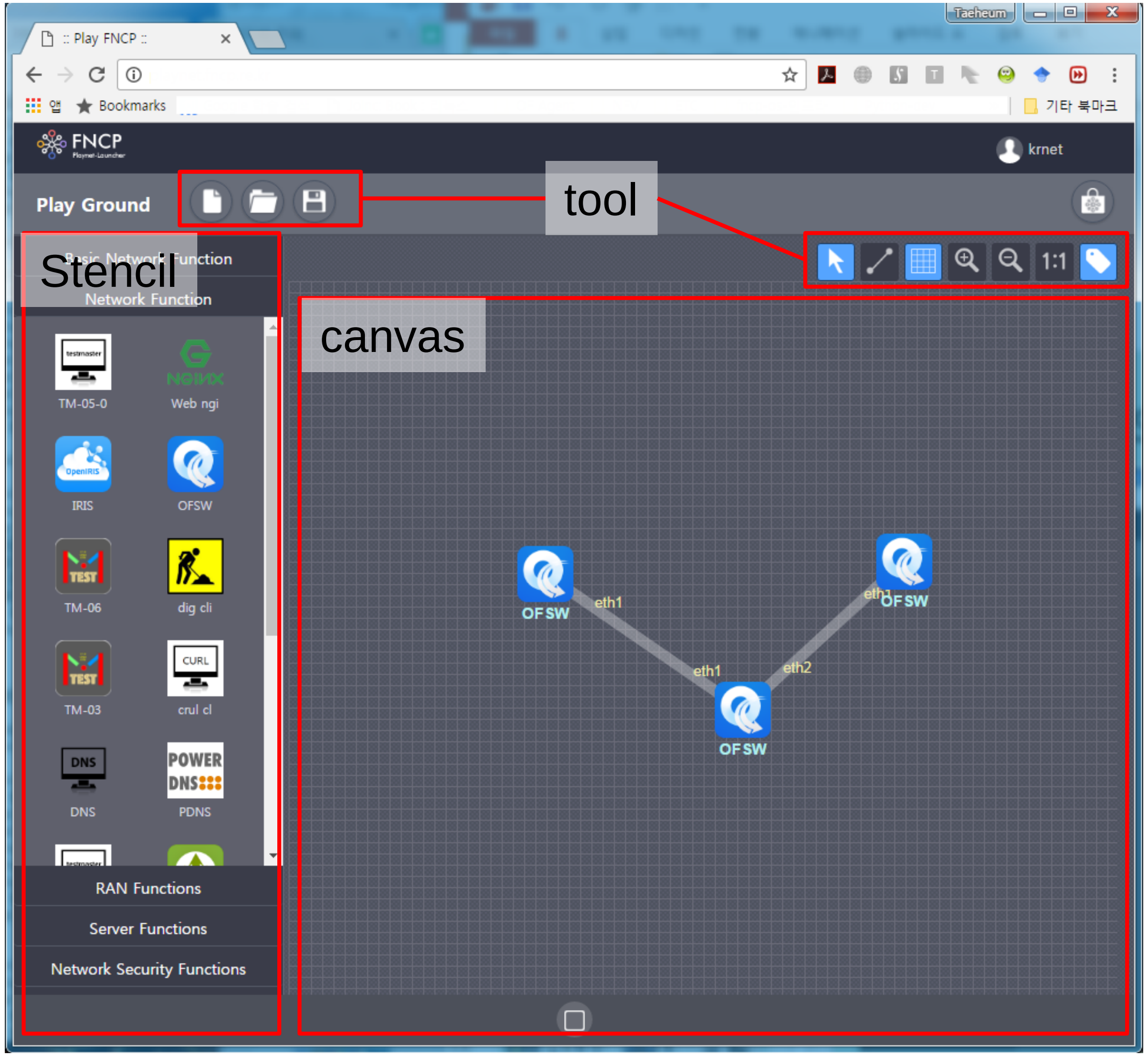


Playnet Architecture

❖ **Features of Playnet**

- Network Service Management
  - ➢ Network Services are managed as projects
  - ➢ Project save → configuration for VNFs are also saved
  - ➢ Loading the project → all VNF instances are activated
    - ✓ Configurations for VNFs are also synchronized
  - ➢ Project termination → all VNF instances are terminated
- Creating/deleting link among VNFs
  - ➢ Point-to-Point, Multipoint-to-Multipoint
- Flavor management

- → When NS is loaded, need to consider optimal placement for the link performance



Related work

❖ IETF Standard

- draft-irtf-nfvrg-resource-management-service-chain-03
- draft-lee-sfc-dynamic-instantiation-01
- Both draft document mention traffic localization

➔ Our work can be one of the use case

❖ ETSI Standard

- VNFFG Descriptor (VNFFGD)
- Virtual Link Descriptor (VLD)
  - ➢ Throughput/Bandwidth requirement, QoS
- Virtual Link Record (VLR)
  - ➢ Allocated_capacity

➔ need more specific parameter for the link

number of transaction (VLD), weight of transaction (VLD), amount of transmitted data (VLR)

Pseudo Service Chaining Mechanism

IT R&D Global Leader ETRI

Pseudo Service Chaining Mechanism

❖ Goal

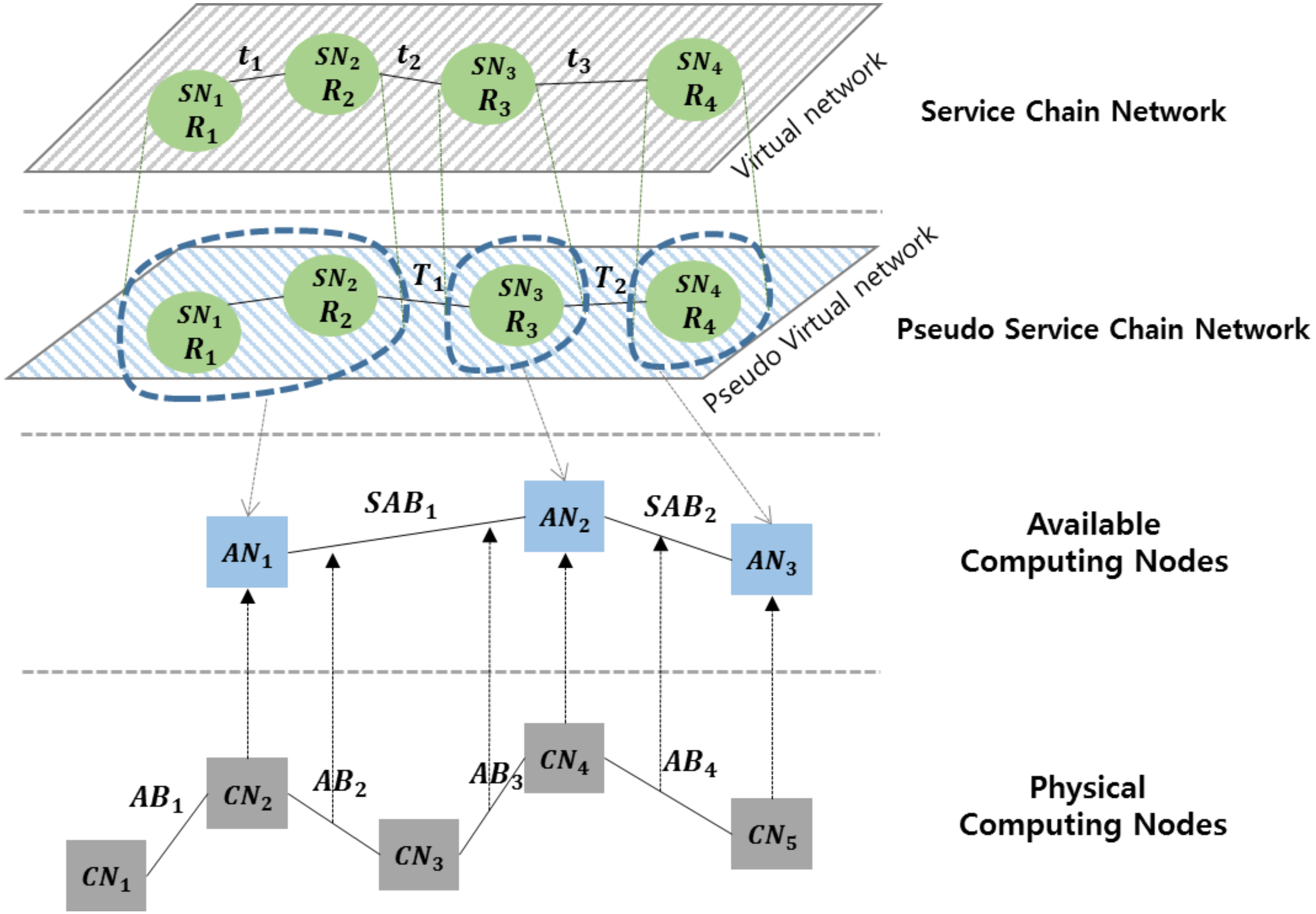
- By localizing SFs (=Minimize the number of entity in SFPs) based on link description metric
- Saving core network bandwidth
- By avoiding capsulation, save the computation resource
- Getting more better performance of virtual link

❖ Assumption

- Doesn't consider scaling, failover and policy
- Metric of Link parameter is decided by Operator (SFC user) at first
- Based on monitoring, it can be updated

Pseudo Service Chaining Mechanism

- Overview of mechanism
  - Phase 1: Calculation of Virtual Link Costs
    - Based on VLD parameters calculate link cost
    - Selecting pseudo virtual node (PVN)
  - Phase 3: Placement PVN
  - Phase 2: selection of available computing nodes
  - It is recursively conducted



Pseudo Service Chaining Mechanism

Phase 1: Calculation of Virtual Link Costs

- Transaction among service nodes
- Transaction weight at virtual link
- Volume of traffic at virtual link

$$c_i = w_i \times NROM(t_i) \times b_i$$

where, 0 < I < number of virtual link

Table 1. Parameter definitions for calculation of virtual link costs.

Notation	Definition
t_i	Amount of transactions at a virtual link *i*
w_i	Transaction weight for a virtual link *i*
b_i	Volume of traffic at a virtual link *i*
c_i	Cost of a virtual link *i*
L_c	List of virtual links in the order of cost

Pseudo Service Chaining Mechanism

- Phase 2: Selection of available computing nodes
  - Based on resource requirement of instance
  - Available compute node
    - 1st available compute node
      - ✓ Available resource > resource requirement of PVN

$$L_r = List\ of\ Available\ computing\ Nodes\ \{\boldsymbol{R_i}\ \geq SUM[V_{r_i}, V_{r_{i\pm1}}]\}$$

Where, 0 < *i* < number of service node

    - 2nd available compute node
      - ✓ Available resource > minimum resource requirement of SN

$$L_r = List\ of\ Available\ computing\ Nodes\ \{\boldsymbol{R_i}\ \geq MIN[V_{r_1}, V_{r_i}]\}$$

Where, 0 < *i* < number of service node

  - Sort in descending order

IT R&D Global Leader ETRI

Pseudo Service Chaining Mechanism

Phase 3: Greedy placement

- **Multiple-Knapsack Problem**

$$x_{jk} \begin{cases} 1, if \;\; vm_j \; is \; assigned \; in \; pvm_k \\ 0, otherwise \end{cases} \quad (5)$$

$$\sum_{j=1}^{n} Vr_j \, x_{jk} \; < R_i \quad (6)$$

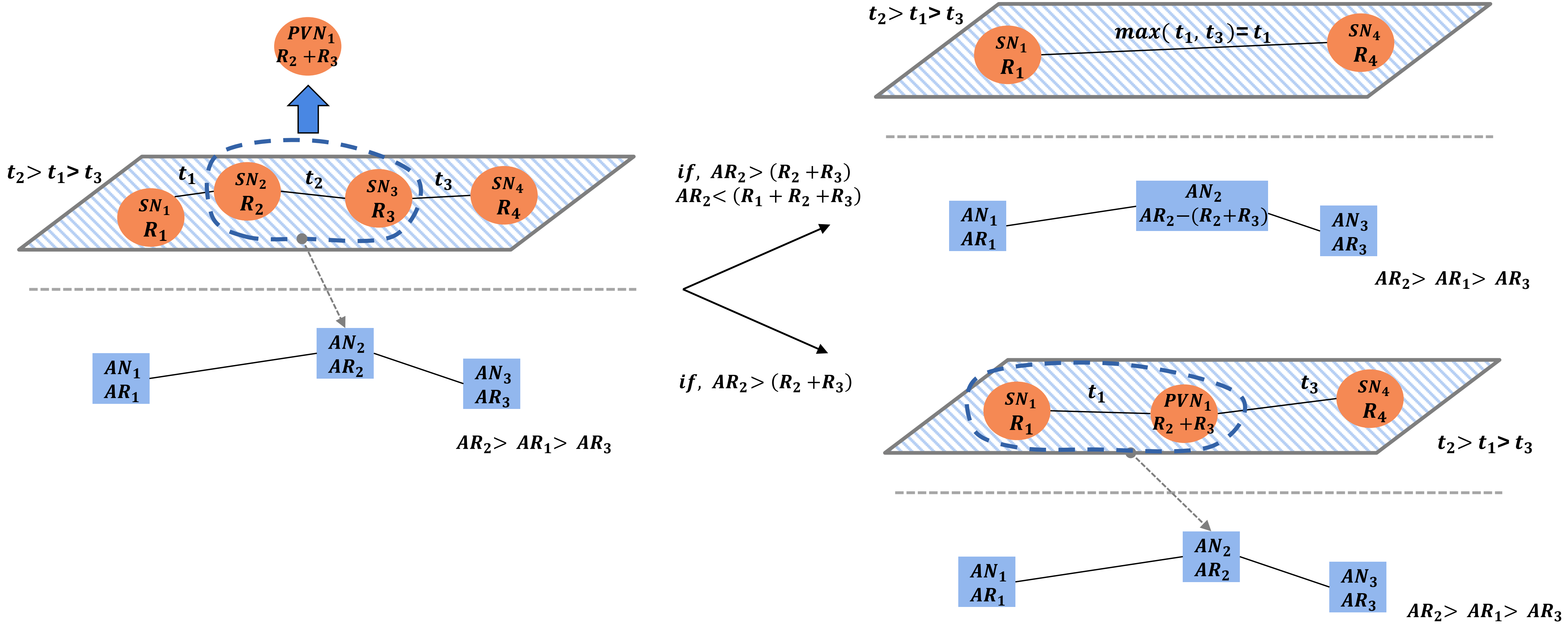
$$pw_k \; = \sum_{j=1}^{n} w_j \, x_{jk} \quad (7)$$

$$y_{ik} \begin{cases} 1, if \;\; pvm_k \; is \; assigned \; in \; AN_i \\ 0, otherwise \end{cases} \quad (8)$$

$$maximizes \; z = \sum_{i=1}^{m} \sum_{j=1}^{n} pw_k \, y_{ik} \quad (9)$$

Pseudo Service Chaining Mechanism

- **Phase 3: Greedy placement**
  - Maximize the sum of cost in the allocated PVM



Conclusion

- Result
  - improvement of 14% in RTT
  - improvement of 37% in UDP receive rate

- Analysis
  - Better performance for Loss-rate of UDP
  - Decrease round trip time
  - Less CPU usage of host node(Interrupt)

Question?

taeheum@etri.re.kr