Software Defined Resource Allocation for Service-Oriented Networks

Nan Zhang, Ya-Feng Liu, Hamid Farmanbar, Tsung-Hui Chang, Mingyi Hong, and Zhi-Quan Luo

zhangnan625@pku.edu.cn, yafliu@lsec.cc.ac.cn, hamid.farmanbar@huawei.com, changtsunghui@cuhk.edu.cn, mhong@umn.edu, luozq@cuhk.edu.cn

HUAWEI

香港中文大學（深圳） The Chinese University of Hong Kong

Motivation

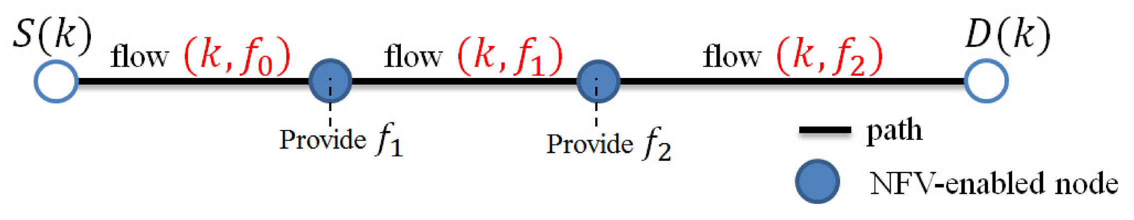
- Today's networks must support diverse service requirements, each service consists of a predefined service function chain (SFC).
- Traditional specialized network hardware provides dedicated network services ⇒ it is costly and inflexible!
- Network function virtualization (NFV) [1]: intelligently integrate a variety of network resources to establish a virtual network (VN) for each request.
- Joint VN embedding and resource allocation [2, 3, 4]:
  - ◇ select function nodes for service function instantiation
  - ◇ route traffic such that each flow gets processed at function nodes in the order defined in the corresponding SFC

Main Contribution

- Perform joint VN embedding and traffic engineering for service-oriented networks.
- Propose a novel problem formulation taking practical network constraints into consideration.
- Show NP-hardness of the formulated problem.
- Develop an efficient penalized successive upper bound minimization (PSUM) algorithm with convergence guarantee.

System Model

- Flow k shall be transmitted from $S(k)$ to $D(k)$ with rate $\lambda(k)$
- SFC of flow k: $\mathcal{F}(k) = (f_1^k \to \cdots \to f_n^k)$
- The set of function nodes that can provide function f: V_f
- Binary variable indicating whether function node i provides function f for flow k: $x_{i,f}(k)$
- Rate of virtual flow (k, f) over link (i,j): $r_{ij}(k,f)$



- Rate of flow k over link (i,j): $r_{ij}(k) = \sum_{f\in\mathcal{F}(k)} r_{ij}(k,f)$ (1)
- In order to reduce communication overhead, each flow k gets served by exactly one node for each $f \in \mathcal{F}(k)$: $\sum_{i\in V_f} x_{i,f}(k) = 1$ (2)
- Each function node provides at most one function for each flow: $\sum_f x_{i,f}(k) \le 1$ (3)

System Model (Cont.)

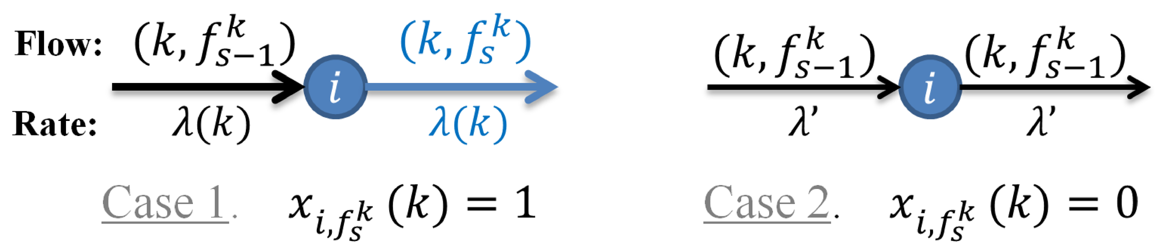
- Link capacity constraint: $\sum_k r_{ij}(k) \le C_{ij}$ (4)
- Node capacity constraint: $\sum_k \sum_f x_{i,f}(k)\lambda(k) \le \mu_i$ (5)
- Network flow conservation constraints:

$$\lambda(k)x_{i,f_s^k}(k) = \sum_{j:(j,i)\in\mathcal{L}} r_{ji}(k, f_{s-1}^k) - \sum_{j:(i,j)\in\mathcal{L}} r_{ij}(k, f_{s-1}^k) \quad (6)$$

$$\lambda(k)x_{i,f_s^k}(k) = \sum_{j:(i,j)\in\mathcal{L}} r_{ij}(k, f_s^k) - \sum_{j:(j,i)\in\mathcal{L}} r_{ji}(k, f_s^k) \quad (7)$$

$$\sum_{j:(S(k),j)\in\mathcal{L}} r_{S(k)j}(k, f_0^k) = \lambda(k) \quad (8)$$

$$\sum_{j:(j,D(k))\in\mathcal{L}} r_{jD(k)}(k, f_n^k) = \lambda(k) \quad (9)$$



Problem Formulation and Analysis

$$\begin{aligned} \min_{\mathbf{r},\mathbf{x}} \quad & g(\mathbf{r}) := \sum_k \sum_{(i,j)} r_{ij}(k) \\ \text{s.t.} \quad & (1)-(9), \\ & r_{ij}(k) \ge 0, r_{ij}(k,f) \ge 0,\ \forall (i,j) \in \mathcal{L}, \\ & x_{i,f}(k) \in \{0,1\},\ \forall k,\ \forall f,\ \forall i. \end{aligned} \quad (P)$$

- Joint VN embedding and resource allocation
- The total link rate objective avoids cycles in choosing routing paths.
- The problem of checking the feasibility of (P) is NP-hard (Proved).
- Suppose $\mu_i \ge \bar{\mu}$ for all i, and $C_{ij} \ge \bar{C}$ for all (i,j), where

$$\bar{\mu} = \sum_{k=1}^{K} \lambda(k),\ \bar{C} = \sum_{k=1}^{K} \lambda(k)(|\mathcal{F}(k)|+1), \quad (10)$$

and $|\mathcal{F}(k)|$ denotes the number of functions in $\mathcal{F}(k)$. Then the LP relaxation of problem (P) always has a binary solution of $\{x_{i,f}(k)\}$.

- The above result suggests that, if the link and node capacity are sufficiently large, then problem (P) and its LP relaxation are equivalent.
- The above lower bounds in (10) are tight.

PSUM Algorithm

- Relax binary variables $\{x_{i,f}(k)\}$ to be real and add a penalty term to the objective function:
  - $\mathbf{x}_f(k) := (x_{i,f}(k))_{i\in V_f}$, then (2) ⇔ $\|\mathbf{x}_f(k)\|_1 = 1$
  - **Fact** [5]: For any $p \in (0,1)$, $\epsilon > 0$, the optimal solution of the following problem must be binary:

$$\begin{aligned} \min \quad & \|\mathbf{x}_f(k) + \epsilon\mathbf{1}\|_p^p := \sum_{i\in V_f} (x_{i,f}(k)+\epsilon)^p \\ \text{s.t.} \quad & \|\mathbf{x}_f(k)\|_1 = 1, x_{i,f}(k) \in [0,1],\ \forall i \in V_f. \end{aligned}$$

- Penalized problem:

$$\begin{aligned} \min_{\mathbf{z}=(\mathbf{r},\mathbf{x})} \quad & g_\sigma(\mathbf{z}) = g(\mathbf{r}) + \sigma P_\epsilon(\mathbf{x}) \\ \text{s.t.} \quad & (1)-(9), \\ & r_{ij}(k) \ge 0, r_{ij}(k,f) \ge 0,\ \forall (i,j) \in \mathcal{L}, \\ & x_{i,f}(k) \in [0,1],\ \forall k,\ \forall f,\ \forall i, \end{aligned} \quad (P1)$$

where the penalty term:

$$P_\epsilon(\mathbf{x}) = \sum_k \sum_{f\in\mathcal{F}(k)} \|\mathbf{x}_f(k) + \epsilon\mathbf{1}\|_p^p.$$

- Convergence analysis: Suppose the positive sequence $\{\sigma_t\}$ is monotonically increasing and $\sigma_t \to +\infty$, and $\mathbf{z}^t$ is a global minimizer of the penalized problem (P1) with the objective function $g_{\sigma_t}(\mathbf{z})$. Then any limit point of $\{\mathbf{z}^t\}$ is a global minimizer of problem (P).
- Successive Upper bound Minimization (SUM) [6]: solve a sequence of approximate objective functions which are lower bounded by $g_\sigma(\mathbf{z})$:

$$P_\epsilon(\mathbf{x}) \le P_\epsilon(\mathbf{x}^t) + \nabla P_\epsilon(\mathbf{x}^t)^T(\mathbf{x} - \mathbf{x}^t)$$

- PSUM subproblem at the $(t+1)$-th iteration:

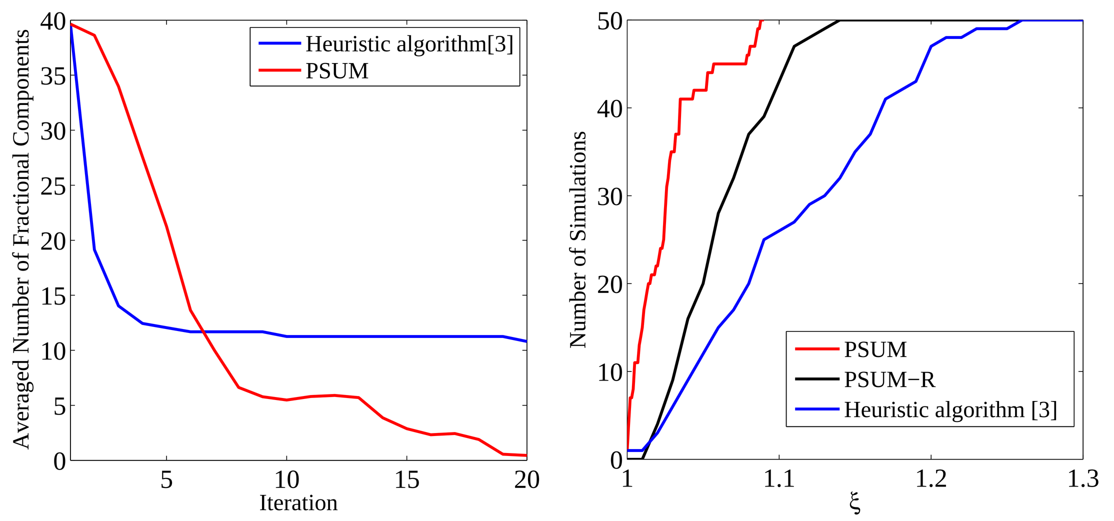
$$\begin{aligned} \min_{\mathbf{r},\mathbf{x}} \quad & g(\mathbf{r}) + \sigma_{t+1}\nabla P_{\epsilon_{t+1}}(\mathbf{x}^t)^T\mathbf{x} \\ \text{s.t.} \quad & (1)-(9), \\ & r_{ij}(k) \ge 0, r_{ij}(k,f) \ge 0,\ \forall (i,j) \in \mathcal{L}, \\ & x_{i,f}(k) \in [0,1],\ \forall k,\ \forall f,\ \forall i, \end{aligned} \quad (\text{PSUM\_sub})$$

where $\sigma_{t+1} = \gamma\sigma_t,\ \epsilon_{t+1} = \eta\epsilon_t$.

- PSUM-R: combine PSUM with a Rounding technique
  1. Perform $t_{\max}$ PSUM iterations to obtain $\{(\bar{x}_{i,f}(k))_{i\in V_f}\}$;
  2. For nonbinary $\bar{\mathbf{x}}_f(k)$: if $\bar{x}_{j,f}(k) = \max_{i\in V_f} \bar{x}_{i,f}(k) \ge \theta$, then set $x_{j,f}(k) = 1$; otherwise find the node $v \in V_f$ with the maximum remaining computational capacity and set $x_{v,f}(k) = 1$;
  3. Determine $\mathbf{r}$: solve (P) with $\mathbf{x}$ being fixed and the objective function being $g + \tau\Delta$, and modify (4) by $\sum_k r_{ij}(k) \le C_{ij} + \Delta$.

Simulation Results

- Simulation scenario: a mesh network
  - 100 nodes and 684 direct links
  - 5 service functions, $|V_f| = 10$ candidate nodes for each function
  - $\mathcal{F}(k) = (f_1^k \to f_2^k)$ and $(S(k), D(k))$ are uniformly randomly chosen for each flow $(f_1^k \ne f_2^k, S(k), D(k) \notin V_{f_s^k}, s = 1,2)$
  - Parameter setting: $C_{ij} \sim [0.5, 5.5]$, $\mu_i \sim [0.5, 8]$, $K = 30$, $\lambda(k) = 1, \forall k$
- Compare with the modified heuristic algorithm in [3].
- Parameters setting: $T_{\max} = 20$, $\sigma_1 = 2$, $\epsilon_1 = 0.001$, $\gamma = 1.1$, $\eta = 0.5$, $t_{\max} = 7$, $\theta = 0.9$, $\tau = 5$.
- Randomly generate 50 instances of problem (P).



Left: the averaged number of fractional components varies with iterations;
Right: the number of simulations with $g^*_{\text{PSUM}}/g^*_{\text{LP}} \le \xi$ varies with ξ.

- The solutions returned by PSUM gradually converge to some feasible binary solutions.
- In 50 simulations, PSUM and PSUM-R successfully find the feasible solution 48 times while the heuristic algorithm only succeeds 9 times.
- PSUM can approximately solve problem (P) by returning a feasible solution with good quality and is easily implemented.
- PSUM-R achieves a good balance of solution quality and algorithm efficiency.

References

[1] M. Chiosi, D. Clarke, P. Willis, A. Reid, J. Feger et al. Network Functions Virtualization: Introductory White Paper. *SDN and OpenFlow World Congress*, Darmstadt, Germany, Oct. 2012.

[2] H. Zhang, S. Vrzic, G. Senarath, N. D. Dào, H. Farmanbar, J. Rao, C. Peng, and H. Zhuang. 5G Wireless Network: MyNET and SONAC. *IEEE Network*, vol. 29, no. 4, pp. 14–23, Jul. 2015.

[3] X. Li, J. Rao, and H. Zhang. Engineering Machine-to-Machine Traffic in 5G. *IEEE Internet of Things Journal*, vol. 3, no. 4, pp. 609–618, Aug. 2016.

[4] M. Charikar, Y. Naamad, J. Rexford, and X. K. Zou. Multi-commodity Flow with In-network Processing. *Manuscript*, www.cs.princeton.edu/~jrex/papers/mopt14.pdf.

[5] B. Jiang, Y.-F. Liu, and Z. Wen, Lp-norm regularization algorithms for optimization over permutation matrices. *SIAM Journal on Optimization*, vol. 26, no. 4, pp. 2284–2313, 2016.

[6] M. Razaviyayn, M. Hong, and Z.-Q. Luo. A Unified Convergence Analysis of Block Successive Minimization Methods for Nonsmooth Optimization. *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1126–1153, 2013.