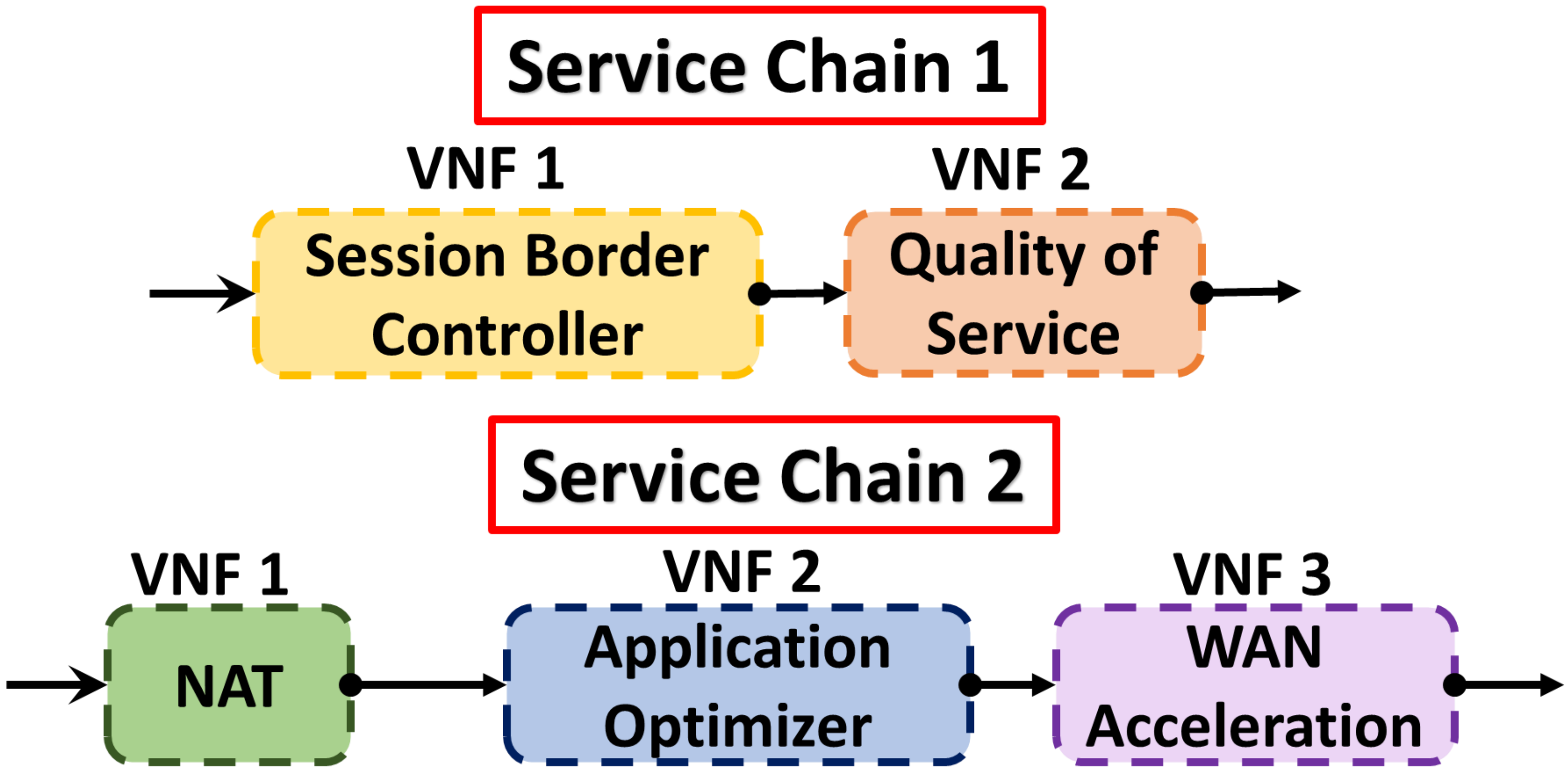
Multiple Virtual Network Function Service Chain Placement and Routing using Column Generation

BY
ABHISHEK GUPTA
FRIDAY GROUP MEETING
NOVEMBER 11, 2016

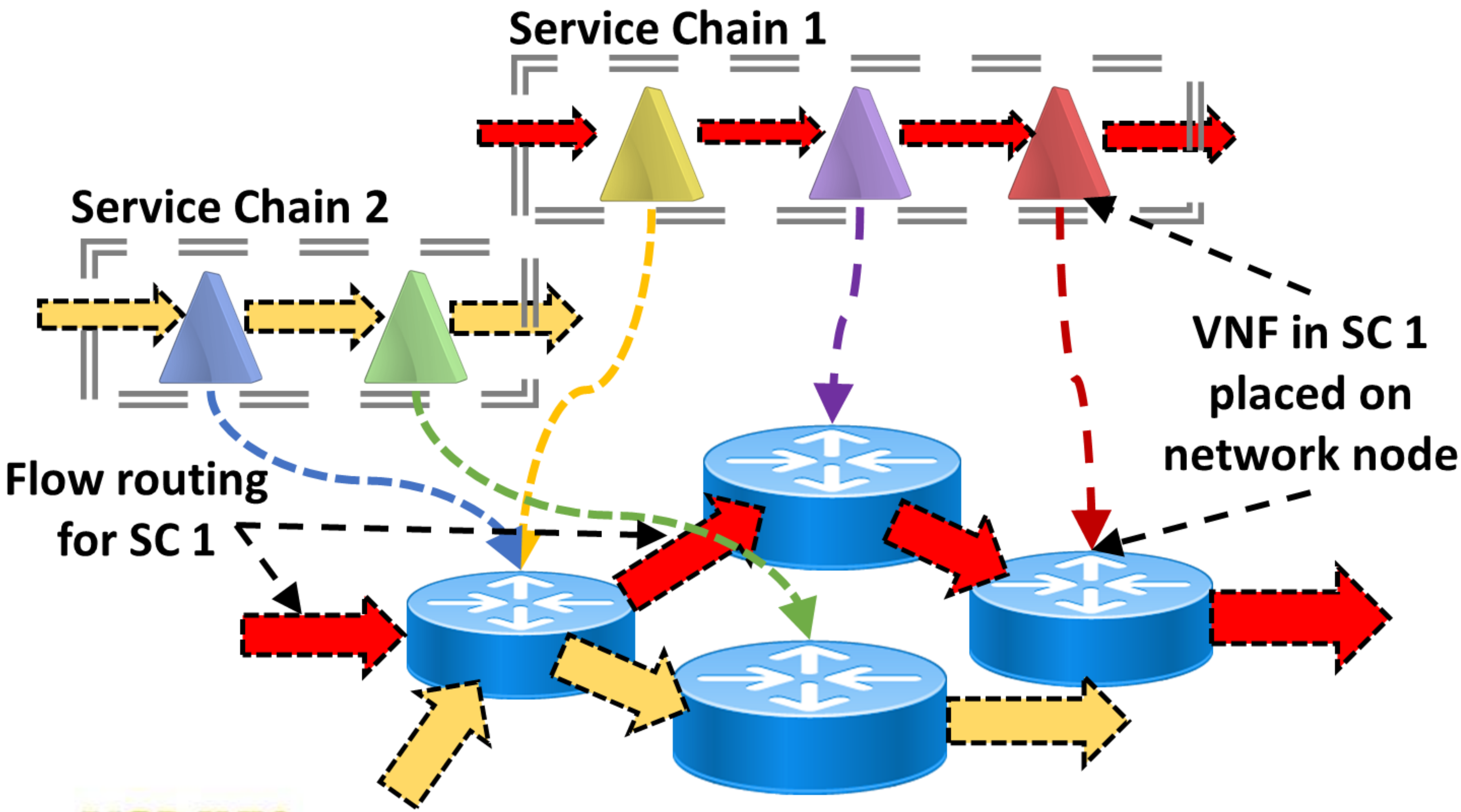
UCDAVIS

Virtual Network Function (VNF) Service Chain (SC)

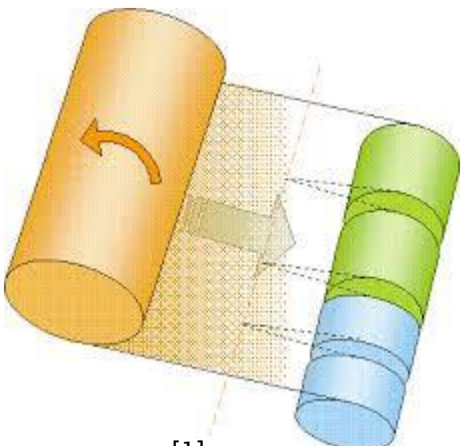


UC DAVIS
UNIVERSITY OF CALIFORNIA

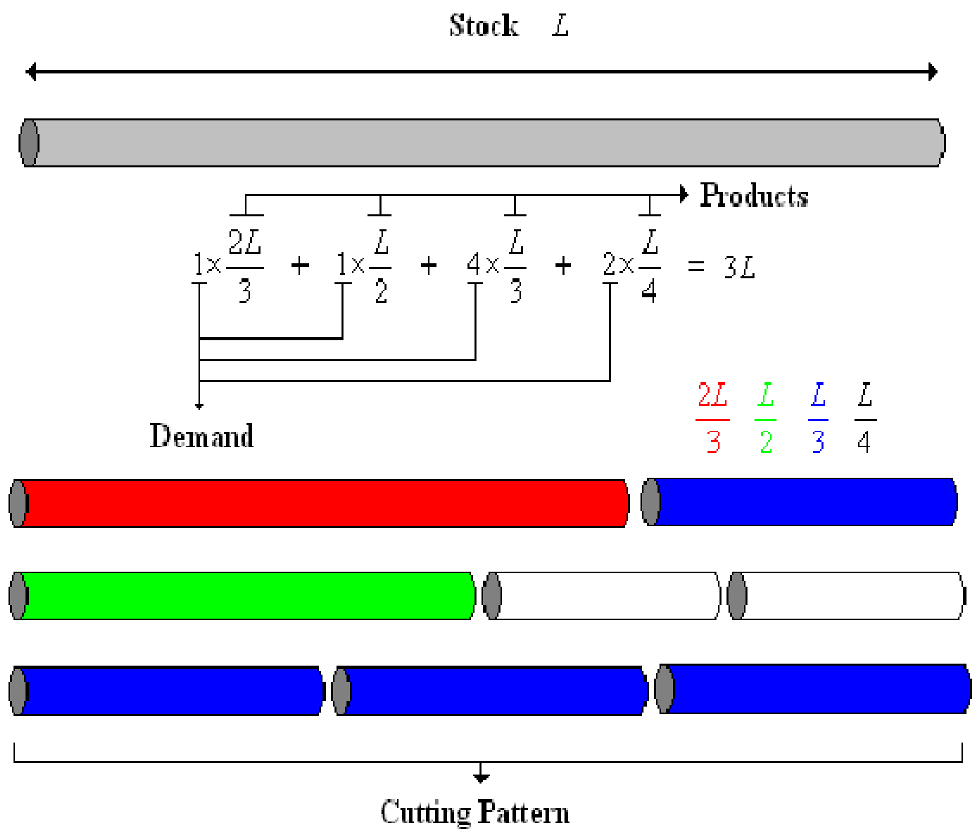
Multiple VNF SC Placement and Routing



Cutting Stock Problem



[1]



[1] http://www.pepto.se/link.asp?href=pages/dot_eng.html&language=2.
[2] https://www.researchgate.net/publication/228428085_A_New_Heuristic_Algorithm_for_the_One-Dimensional_Cutting_Stock_Problem/figures?lo=1

Continued…

- Number of patterns can increase exponentially with increase in set of orders of smaller widths, i.e., we have both more demands per roll width and more roll widths (finals)

width of final	45	36	31	14
number of orders	97	610	395	211

$$\text{cutting pattern } \mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{pmatrix}$$

$$\begin{aligned} 45a_1 + 36a_2 + 31a_3 + 14a_4 &\leq 100 \\ a_1, a_2, a_3, a_4 &\geq 0 \text{ integer} \end{aligned} \qquad (*)$$

[3] http://www.unc.edu/depts/stat-or/courses/provan/STOR614_web/lect12_cutstock.pdf

Continued…

The object is to determine a set $\mathbf{a}^1, \mathbf{a}^2, \ldots, \mathbf{a}^r$ of cutting patterns and the number x_j of copies of pattern $\mathbf{a}^j = \begin{pmatrix} a_{1j} \\ a_{2j} \\ a_{3j} \\ a_{4j} \end{pmatrix}$ produced in order to minimize the waste, or equivalently, to **minimize the number of rolls used**.

width of final	45	36	31	14
number of orders	97	610	395	211

- How to minimize the number of raw rolls used?

[3] http://www.unc.edu/depts/stat-or/courses/provan/STOR614_web/lect12_cutstock.pdf

Exponential number of variables

width of final	45	36	31	14
number of orders	97	610	395	211

$$\mathbf{a}^j = \begin{pmatrix} a_{1j} \\ a_{2j} \\ a_{3j} \\ a_{4j} \end{pmatrix}$$

$$\begin{array}{rcccccccccc}
\min z & = & x_1 & + & x_2 & + & \ldots & + & x_r & & \\
& & a_{11}x_1 & + & a_{12}x_2 & + & \ldots & + & a_{1r}x_r & = & 97 \\
& & a_{21}x_1 & + & a_{22}x_2 & + & \ldots & + & a_{2r}x_r & = & 610 \\
& & a_{31}x_1 & + & a_{32}x_2 & + & \ldots & + & a_{3r}x_r & = & 395 \\
& & a_{41}x_1 & + & a_{42}x_2 & + & \ldots & + & a_{4r}x_r & = & 211 \\
& & \multicolumn{9}{l}{x_1, x_2, \ldots, x_r \geq 0 \text{ integer}}
\end{array}$$

This has a **large** number of variables, in addition to being an integer program. We will make an initial simplification by solving the **relaxed LP**. To handle the large number of columns we will use the **revised simplex method** with **delayed column generation**.

[3] http://www.unc.edu/depts/stat-or/courses/provan/STOR614_web/lect12_cutstock.pdf

Column Generation (CG)

- CG generates patterns for the cutting stock on demand.
- By generating patterns on demand we keep problem size small and make optimization scalable.
- This is done by splitting the problem into 2 sub-problems:
  - Reduced Master Problem (RMP)
  - Pricing Problem (PP)
- To begin CG process an initial pattern is required.
- This pattern is provided by a heuristic.

Reduced Master Problem (RMP)

- RMP contains all patterns generated by CG till the current iteration.
- Each modified RMP solved as LP until CG terminates.
- Final RMP is solved as an ILP.
- As a result, we have e-optimal procedure (e: difference between ILP and final LP objective values).
- Since CG procedure is provably e-optimal, it is not a heuristic/meta-heuristic

Pricing Problem (PP)

- PP generates patterns which are added to RMP. Hence, contains constraints only pertinent to pattern generation.
- Each solution of the PP is a pattern, which is added to RMP as a column.
- PP is solved as an ILP.
- Column (pattern) is added to RMP.
- After solving the RMP, the dual values associated with RMP constraints are used to build the PP objective.
- PP Objective is called **"Reduced Cost"**

UCDAVIS
UNIVERSITY OF CALIFORNIA

Model

width of final	45	36	31	14
number of orders	97	610	395	211

$$\begin{aligned} 45a_1 + 36a_2 + 31a_3 + 14a_4 &\le 100 \\ a_1, a_2, a_3, a_4 &\ge 0 \text{ integer} \end{aligned} \qquad (*)$$

RMP

$$\begin{array}{rccccccccc} \min z = & x_1 & + & x_2 & + & \ldots & + & x_r & & \\ & a_{11}x_1 & + & a_{12}x_2 & + & \ldots & + & a_{1r}x_r & = & 97 \\ & a_{21}x_1 & + & a_{22}x_2 & + & \ldots & + & a_{2r}x_r & = & 610 \\ & a_{31}x_1 & + & a_{32}x_2 & + & \ldots & + & a_{3r}x_r & = & 395 \\ & a_{41}x_1 & + & a_{42}x_2 & + & \ldots & + & a_{4r}x_r & = & 211 \\ & \multicolumn{9}{l}{x_1, x_2, \ldots, x_r \ge 0 \text{ integer}} \end{array}$$

PP

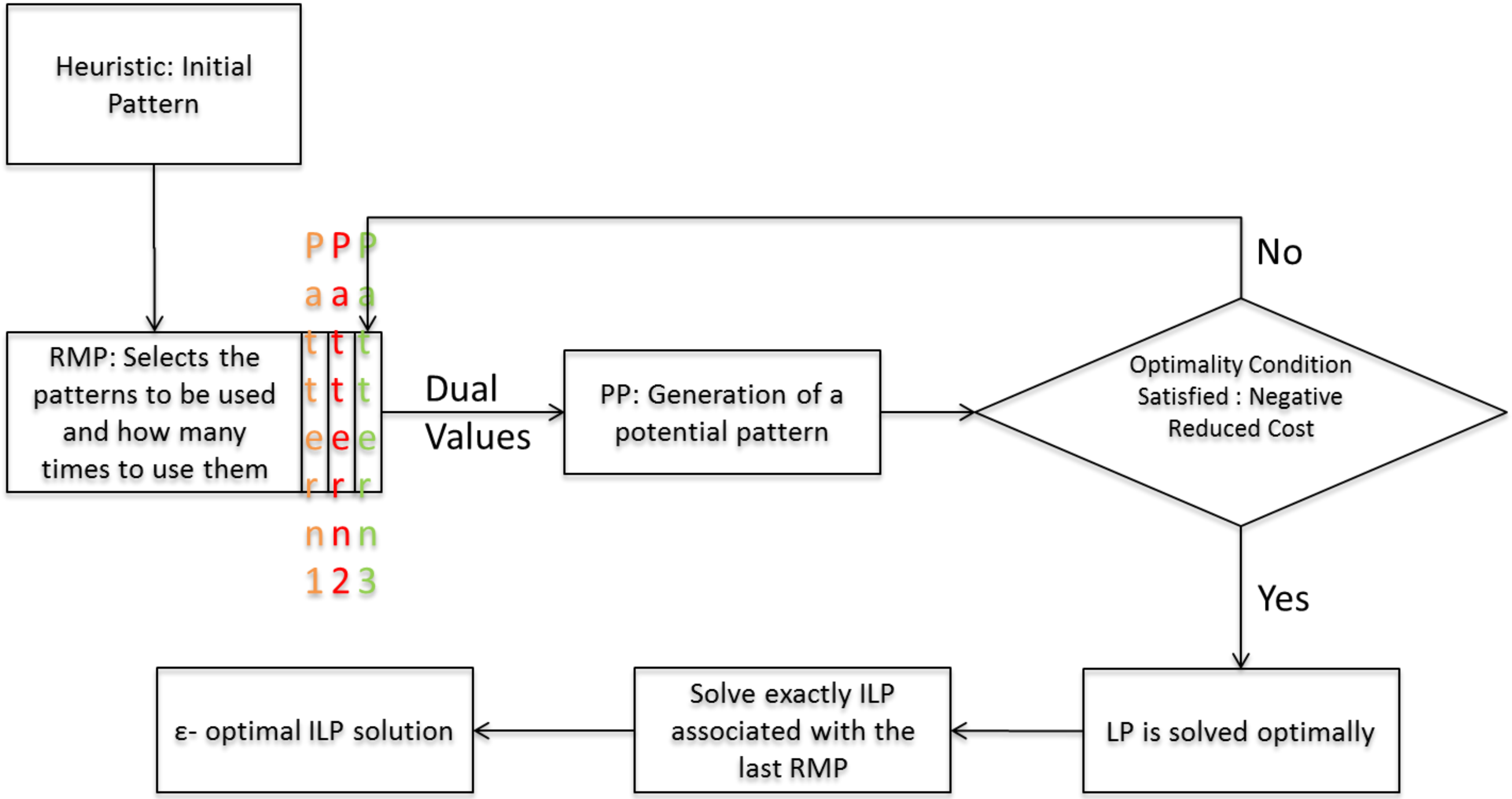
Minimize:

$$1 - \sum_i \pi_i A_i$$

subject to:

$$\sum_i W_i A_i \le W$$

CG scheme for cutting stock problem

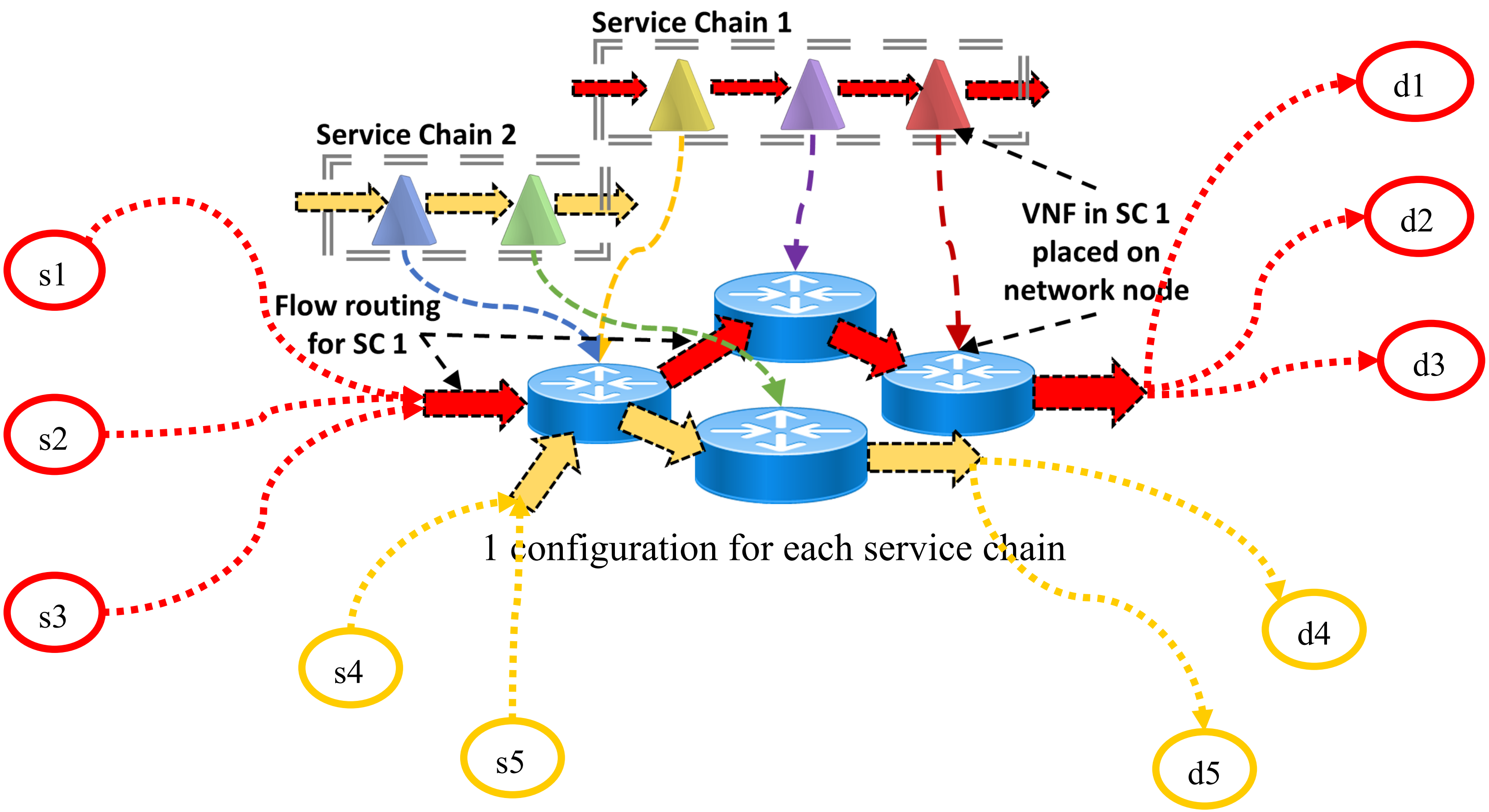


UCDAVIS
UNIVERSITY OF CALIFORNIA

Our Problem – Concept of Configurations

- Each placement of a service chain is a configuration.
- We allow only one configuration per service chain at present.
- So multiple traffic flow in demand of the same service chain will use the same configuration.
- With the usual flow capacity constraints and core capacity constraints

Continued…



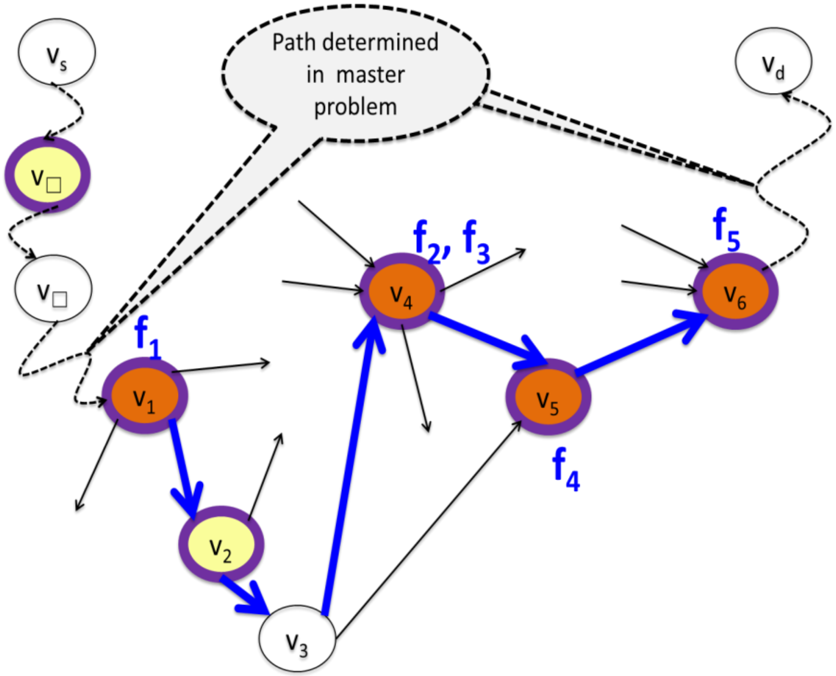
Service Chain 1
Service Chain 2
VNF in SC 1
placed on
network node
Flow routing
for SC 1
s1
s2
s3
s4
s5
d1
d2
d3
d4
d5
1 configuration for each service chain

Function of RMP and PP

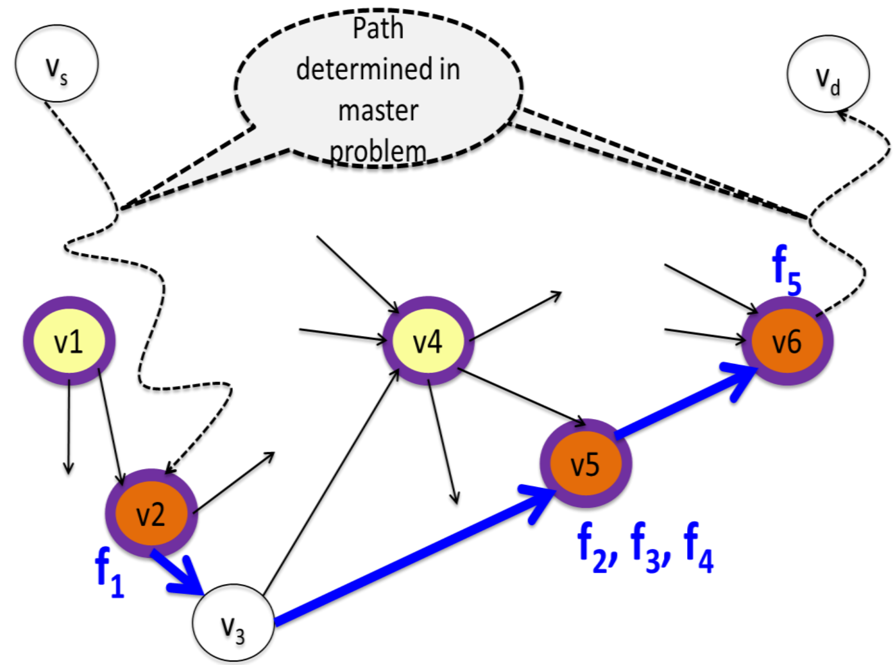
- RMP decides the routes from source to location of first VNF and location of last VNF to destination.
- Each service chain has a PP instance associated with it.
- PP decides location of all VNFs in a service chain and route to traverse these locations.

Continued…

NFV node   NFV node not hosting VNFs   NFV node hosting VNFs



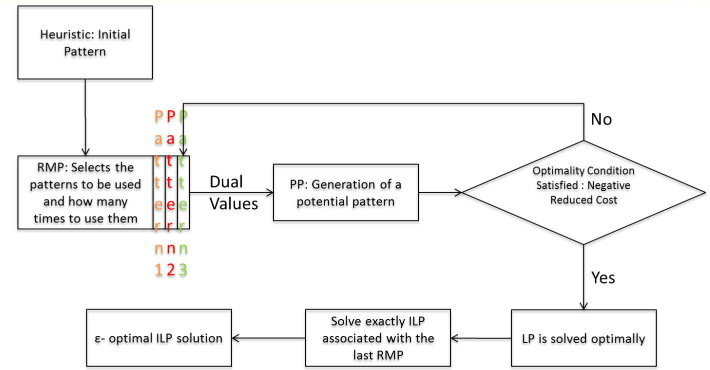
First potential configuration

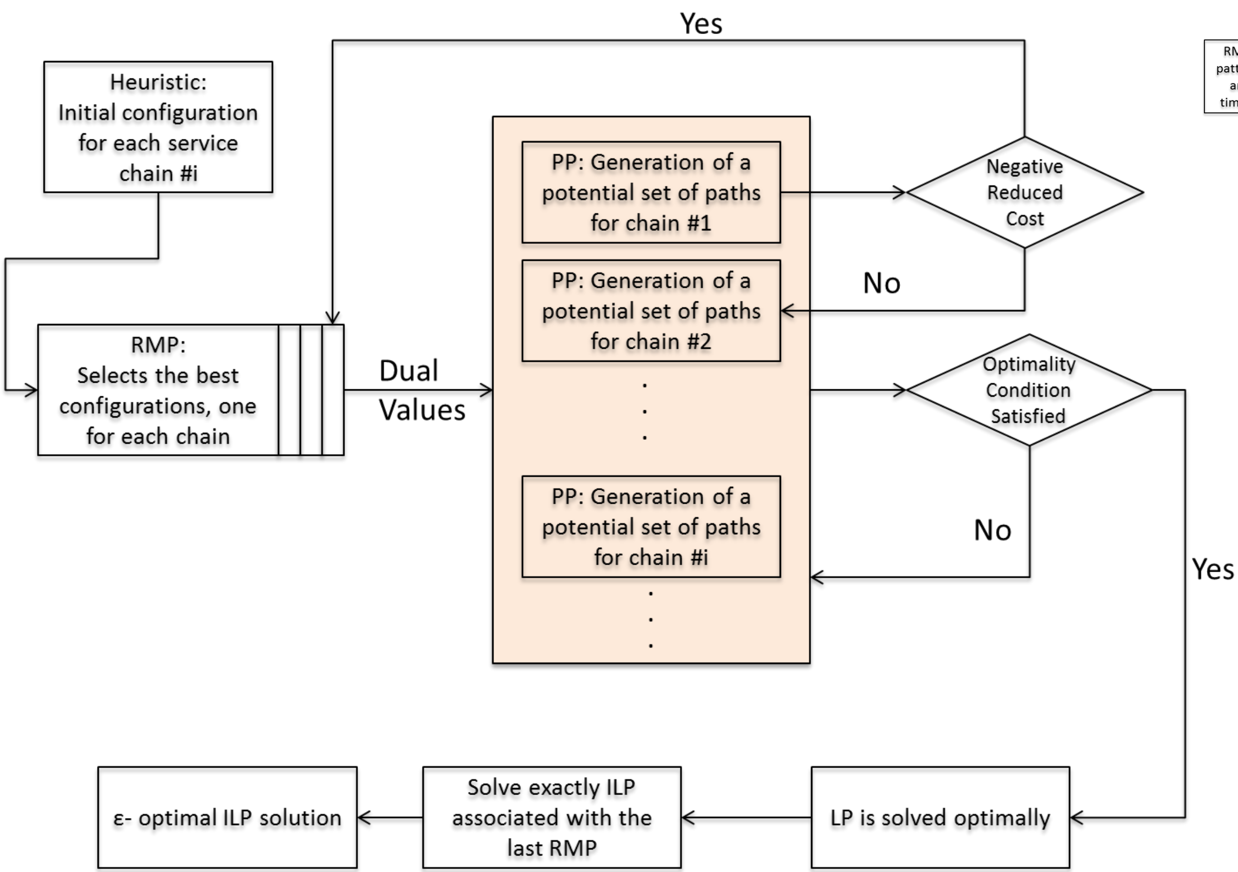


Second potential configuration

UCDAVIS UNIVERSITY OF CALIFORNIA

Methodology Pipeline (Round-robin scheme)





Model

RMP

Objective: Minimize bandwidth consumed:

$$\min \quad \underbrace{\sum_{c\in C}\sum_{\gamma\in\Gamma_c}\sum_{\ell\in L}\overbrace{\left(\sum_{(s,d)\in\mathcal{SD}} D^c_{sd}\right)}^{\text{Overall traffic using } c}\sum_{i\in I} b^{\gamma}_{i\ell}}_{\text{COST}_\gamma}\, z_\gamma +$$

$$\sum_{c\in C}\sum_{\ell\in L}\sum_{(s,d)\in\mathcal{SD}} D^c_{sd}\left(y^{\text{first}(c),sd}_\ell + y^{\text{last}(c),sd}_\ell\right). \qquad (2)$$

Constraints:

$$\sum_{\gamma\in\Gamma_c} z_\gamma = 1 \qquad c\in C \qquad (3)$$

$$\sum_{\gamma\in\Gamma}\sum_{v\in V^{\text{NFV}}} a^{\gamma}_{vf}\, z_\gamma \ge 1 \qquad f\in F \qquad (4)$$

$$\sum_{c\in C}\sum_{\gamma\in\Gamma_c}\sum_{f\in F} n^{\text{CORE}}_f \sum_{(v_s,v_d)\in\mathcal{SD}} D^c_{sd}\, a^{\gamma}_{vf}\, z_\gamma \le n^{\text{CORE}} \qquad v\in V^{\text{NFV}} \qquad (5)$$

$$\sum_{c\in C}\sum_{(v_s,v_d)\in\mathcal{SD}} D^c_{sd}\left(y^{\text{first}(c),sd}_\ell + y^{\text{last}(c),sd}_\ell + \sum_{\gamma\in\Gamma}\sum_{i=1}^{n_c-1} b^{\gamma}_{i\ell}\, z_\gamma\right) \le \text{CAP}_\ell \qquad \ell\in L \qquad (6)$$

$$\sum_{\gamma\in\Gamma} a^{\gamma}_{vf} z_\gamma \ge x^f_v \qquad f\in F, v\in V^{\text{NFV}} \qquad (7)$$

$$\sum_{\gamma\in\Gamma} a^{\gamma}_{vf} z_\gamma \le M\, x^f_v \qquad f\in F, v\in V^{\text{NFV}}. \qquad (8)$$

Route from v_s to first function location:

$$\sum_{\ell\in\omega^+(v_s)} y^{\text{first}(c),sd}_\ell = 1 - x^{\text{first}(c)}_{v_s} \qquad c\in C, \quad (v_s,v_d)\in\mathcal{SD}: D^c_{sd}>0 \qquad (9)$$

$$\sum_{\ell\in\omega^-(v)} y^{\text{first}(c),sd}_\ell \ge x^{\text{first}(c)}_v \qquad c\in C, \quad (v_s,v_d)\in\mathcal{SD}: D^c_{sd}>0, v\in V^{\text{NFV}}\setminus\{v_s\} \qquad (10)$$

$$\sum_{\ell\in\omega^+(v)} y^{\text{first}(c),sd}_\ell - \sum_{\ell\in\omega^-(v)} y^{\text{first}(c),sd}_\ell = -x^{\text{first}(c)}_v \qquad c\in C, (v_s,v_d)\in\mathcal{SD}: D^c_{sd}>0, v\in V^{\text{NFV}}\setminus\{v_s\} \qquad (11)$$

$$\sum_{\ell\in\omega^+(v)} y^{\text{first}(c),sd}_\ell - \sum_{\ell\in\omega^-(v)} y^{\text{first}(c),sd}_\ell = 0 \qquad c\in C, (v_s,v_d)\in\mathcal{SD}: D^c_{sd}>0, \quad v\in V\setminus(V^{\text{NFV}}\cup\{v_s\}). \qquad (12)$$

Route from last function location to v_d:

$$\sum_{\ell\in\omega^-(v_d)} y^{\text{last}(c),sd}_\ell = 1 - x^{\text{last}(c)}_{v_d} \qquad c\in C, \quad (v_s,v_d)\in\mathcal{SD}: D^c_{sd}>0 \qquad (13)$$

$$\sum_{\ell\in\omega^+(v)} y^{\text{last}(c),sd}_\ell \ge x^{\text{last}(c)}_v \qquad c\in C, \quad (v_s,v_d)\in\mathcal{SD}: D^c_{sd}>0, v\in V^{\text{NFV}}\setminus\{v_d\} \qquad (14)$$

$$\sum_{\ell\in\omega^+(v)} y^{\text{last}(c),sd}_\ell - \sum_{\ell\in\omega^-(v)} y^{\text{last}(c),sd}_\ell = x^{\text{last}(c)}_v \qquad c\in C, (v_s,v_d)\in\mathcal{SD}: D^c_{sd}>0, v\in V^{\text{NFV}}\setminus\{v_d\} \qquad (15)$$

$$\sum_{\ell\in\omega^+(v)} y^{\text{last}(c),sd}_\ell - \sum_{\ell\in\omega^-(v)} y^{\text{last}(c),sd}_\ell = 0 \qquad c\in C, (v_s,v_d)\in\mathcal{SD}: D^c_{sd}>0, \quad v\in V\setminus(V^{\text{NFV}}\cup\{v_d\}). \qquad (16)$$

PP

[PP_SC(c)]

$$\text{RED\_COST}_\gamma = \text{COST}_\gamma - \sum_{f\in F} u^{(4)}_f \sum_{v\in V} a_{vf} + \sum_{v\in V^{\text{NFV}}} u^{(5)}_v \sum_{f\in F_c} n^{\text{CORE}}_f \sum_{(v_s,v_d)\in\mathcal{SD}} D^c_{sd}\, a_{vf}$$
$$+ \sum_{\ell\in L}\sum_{(v_s,v_d)\in\mathcal{SD}} u^{(6)}_\ell D^c_{sd} \sum_{i=1}^{n_c-1} b^{\gamma}_{i\ell}$$
$$- \sum_{f\in F}\sum_{v\in V^{\text{NFV}}} u^{(7)}_{vf} a_{vf} + \sum_{f\in F}\sum_{v\in V^{\text{NFV}}} u^{(8)}_{vf} a_{vf} - u^{(3)}_c. \qquad (17)$$

Constraints:

$$\sum_{f\in F_c} n^{\text{CORE}}_f \sum_{(v_s,v_d)\in\mathcal{SD}} D^c_{sd}\, a_{vf} \le n^{\text{CORE}} \qquad v\in V^{\text{NFV}} \qquad (18)$$

$$\sum_{(v_s,v_d)\in\mathcal{SD}} D^c_{sd} \sum_{i=1}^{n_c-1} b^{\sigma_i(c),\sigma_{i+1}(c)}_\ell \le \text{CAP}_\ell \qquad \ell\in L \qquad (19)$$

$$\sum_{v\in V^{\text{NFV}}} a_{v\sigma_i(c)} = 1 \qquad i=1,2,\ldots,n_c \qquad (20)$$

$$\sum_{\ell\in\omega^+(v)} b^{\gamma}_{1\ell} \ge a_{v,\sigma_1(c)} - a_{v,\sigma_2(c)} \qquad v\in V^{\text{NFV}} \qquad (21)$$

$$\sum_{\ell\in\omega^+(v)} b^{\gamma}_{1\ell} \le 1 - a_{v,\sigma_2(c)} \qquad v\in V^{\text{NFV}} \qquad (22)$$

$$\sum_{\ell\in\omega^-(v)} b^{\gamma}_{1\ell} \le 1 - a_{v,\sigma_1(c)} \qquad v\in V^{\text{NFV}} \qquad (23)$$

$$\sum_{\ell\in\omega^+(v)} b^{\gamma}_{i\ell} - \sum_{\ell\in\omega^-(v)} b^{\gamma}_{i\ell} = a_{v,\sigma_i(c)} - a_{v,\sigma_{i+1}(c)} \qquad v\in V^{\text{NFV}}, i=1,2,\ldots,n_c-1 \qquad (24)$$

$$\sum b^{\gamma}_{i\ell} - \sum b^{\gamma}_{i\ell} = 0 \qquad v\in V\setminus V^{\text{NFV}}, i=1,2,\ldots,n_c-1 \qquad (25)$$

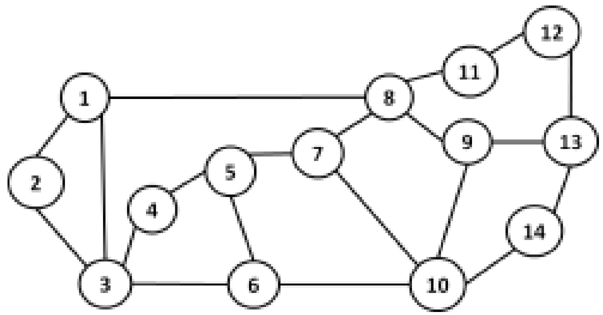
$$\sum_{\ell\in\omega^-(v)} b^{\gamma}_{(n_c-1)\ell} \ge a_{v,\sigma_{n_c}(c)} - a_{v,\sigma_{n_c-1}(c)} \qquad v\in V^{\text{NFV}} \qquad (26)$$

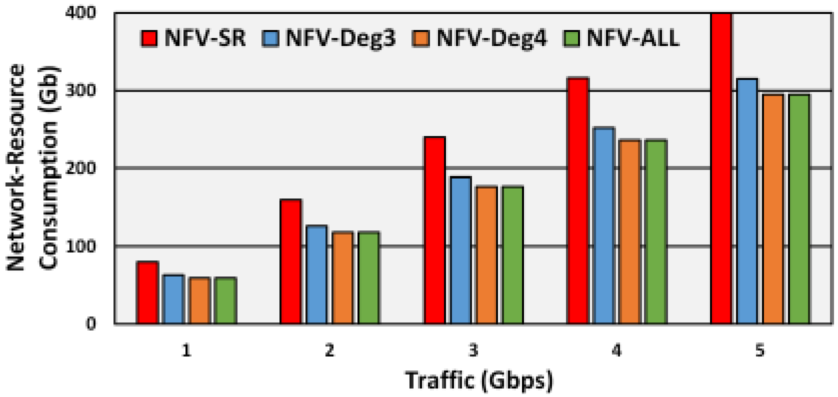
$$\sum_{\ell\in\omega^+(v)} b^{\gamma}_{(n_c-1)\ell} \le 1 - a_{v,\sigma_{n_c}(c)} \qquad v\in V^{\text{NFV}}. \qquad (27)$$

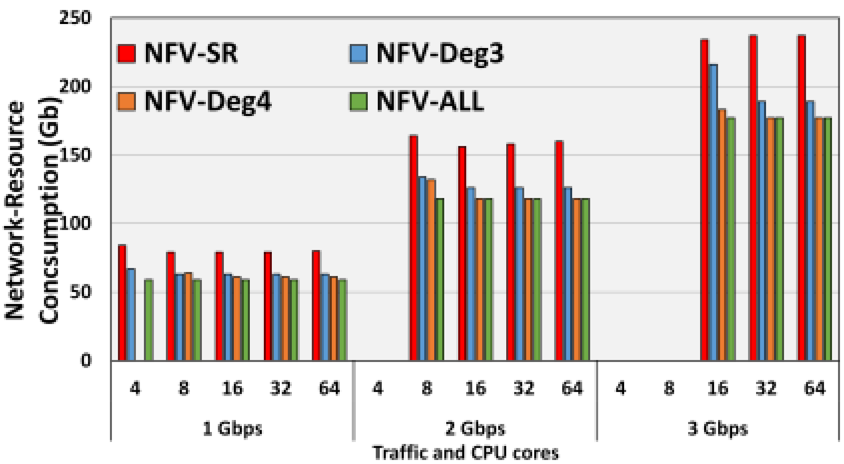
UC DAVIS
UNIVERSITY OF CALIFORNIA

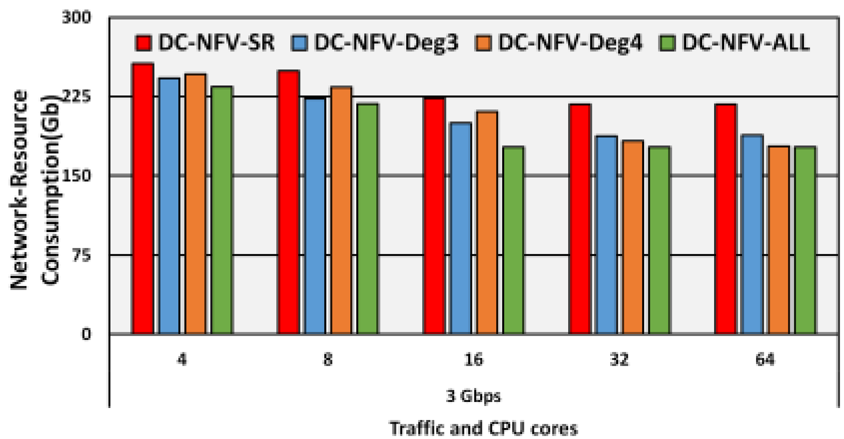
Some initial results

- 20 uniformly distributed traffic flows, 13 service chains, and 33 VNFs.









UC DAVIS
UNIVERSITY OF CALIFORNIA

Issues at present

- Single configuration for each service chain.
  - Allowing more than one configuration per service chain introduces quadratic terms in the model, but can be resolved easily.

Work Directions

- Network-enable Cloud (NeC) being ideal for Network Function Virtualization Infrastructure (NFVI) in core networks.
  - Number of configurations per service chain.
  - Varying number of DC and NFV nodes.

- Pareto function which tries to find a balance between the reduction in bandwidth consumption with number of nodes to be virtualized.
  - Number of nodes to virtualize and by how much in terms of compute resources.
  - Number of configurations per service chain.

Continued…

- Adding latency constraints on each type of service.
  - Topology made of access, metro area and core network so we can see the placement across access, metro and core.
  - For services with strict latency constraints, probably metro and access will be the possible locations.
  - CORD can also be a focus.
  - Datacenter topologies to show if the method.
- Nature of the service chains.
  - Share large number of VNFs (not as one instance of the VNF).
  - Distinct VNFs across service chains.
  - Compute resource requirement of service chains.
  - Latency requirement of service chains.
- Developing more sophisticated heuristics, increasing scalability and performance (speed, optimality) of CG.
- Altering the ways in which the PPs are processed.