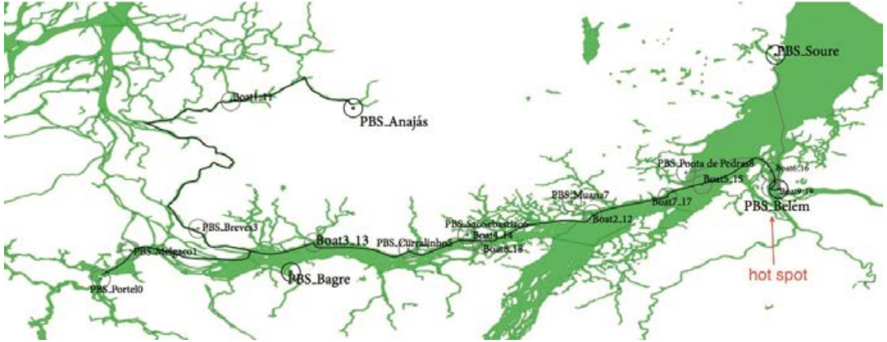
A Constant Approximation for Maximum Throughput Multicommodity Routing

Mengxue Liu (ASU), Andrea Richa (ASU), *Stefan Schmid (Uni Vienna)*, Matthias Rost (TU Berlin)

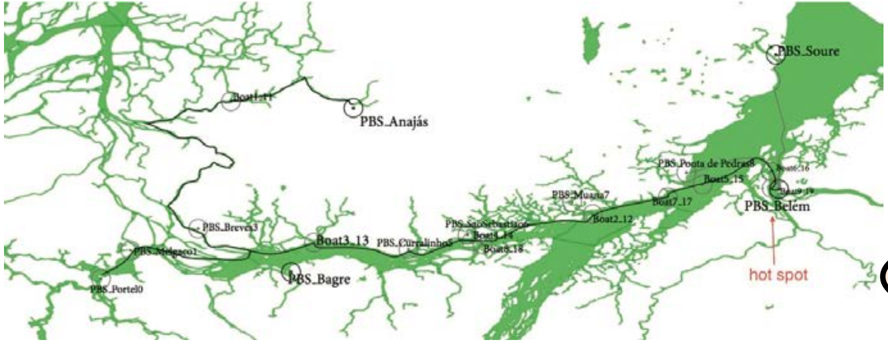
Original Motivation:
Health Care Project in Amazon Riverine

Amazon Riverine



- Complex system of rivers…
- … with poor **communication** infrastructure

Amazon Riverine



How to deliver health care data to hospitals?

- Complex system of rivers…
- … with poor **communication** infrastructure

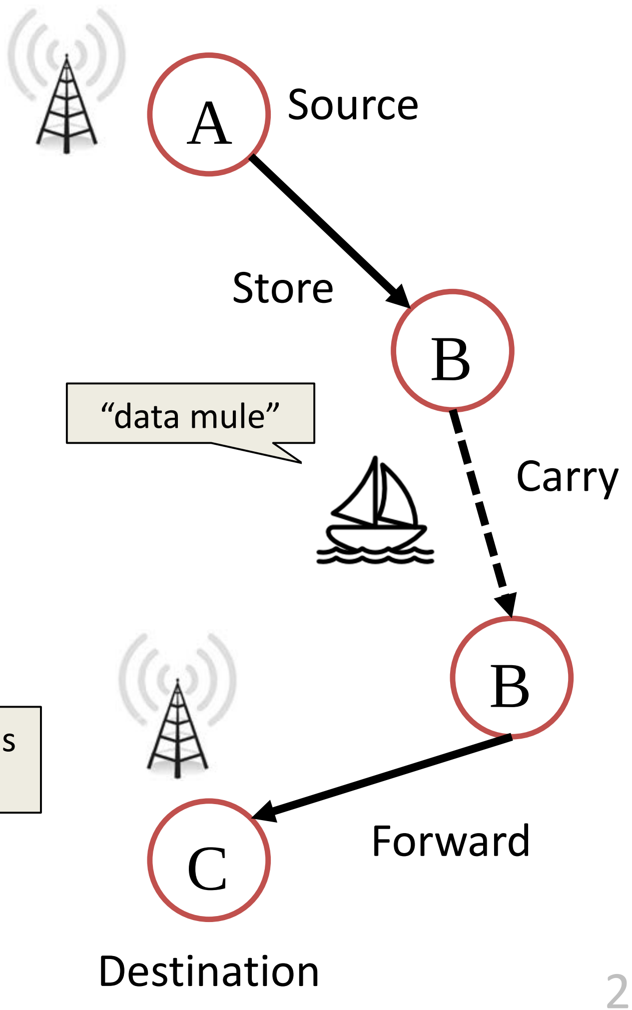
Project and data collection by Andrea Richa (ASU)

Solution: Communication with Boats

- Solution:
  - ***Local nurses*** perform routine clinical examinations, such as **ultrasounds** on pregnant women
  - **Records (files) sent** to the doctors in Belem (city) for evaluation
  - Using regularly **scheduled boats** as ***data mules***

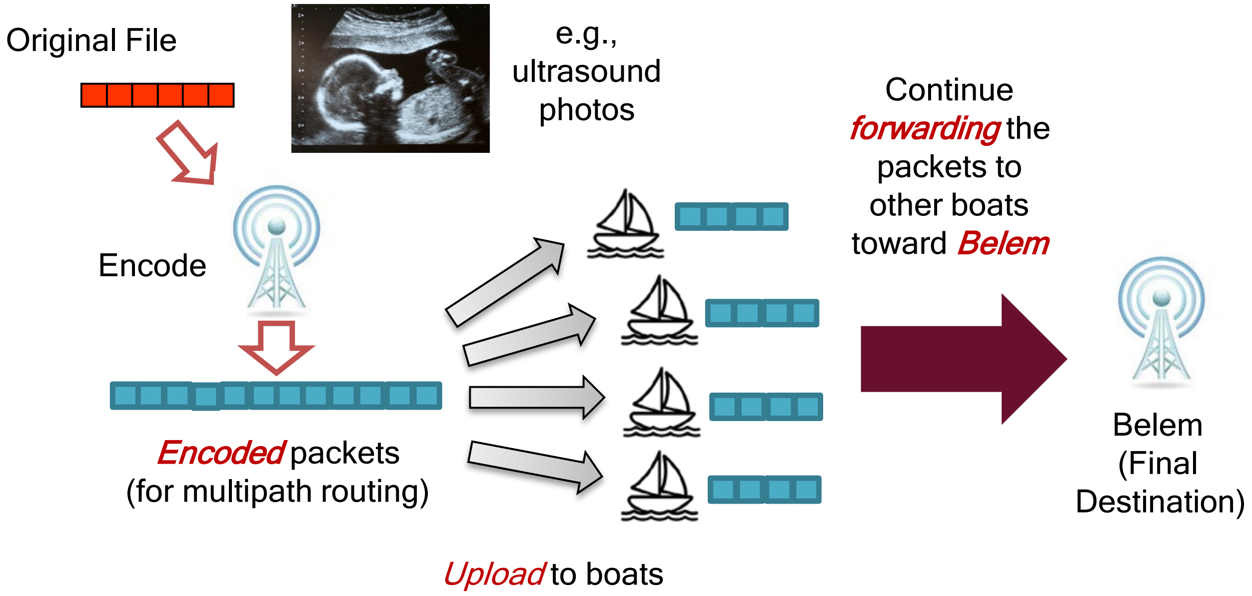
Mobile ubiquitous LAN extension

- Algorithmic problem:
  - Given time schedule of boats (of limited capacity), how to communicate ***maximum amount of data***?



Throughput of Time-Schedule Networks

- Essentially, a throughput maximization problem over **Delay-Tolerant Network (DTN)** resp. **time-schedule network**



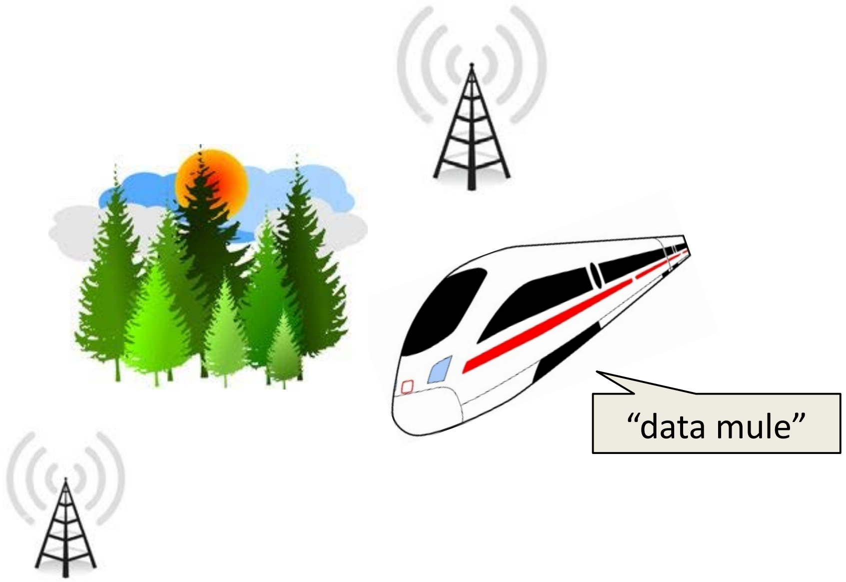
Goal (e.g.):

maximize throughput over DTN

Other Emerging Opportunities for Communication Using Time-Schedule Networks

Smart Devices on the Move: Commute with Their Owners

- E.g., smart devices/***"things" move with their (commuting) owners***: (social) **mobile network**
- Can be exploited for ***communication***: e.g., commuter as "**data mule**" (intermitted connectivity between **hot spots**)



Observation Motivating This Paper

optimal routing **schedules**
in time-schedule network

⟺

maximum throughput in
All-or-Nothing (Splittable) Multicommodity Flow (**ANF**) problem
on the "***connection graph***"

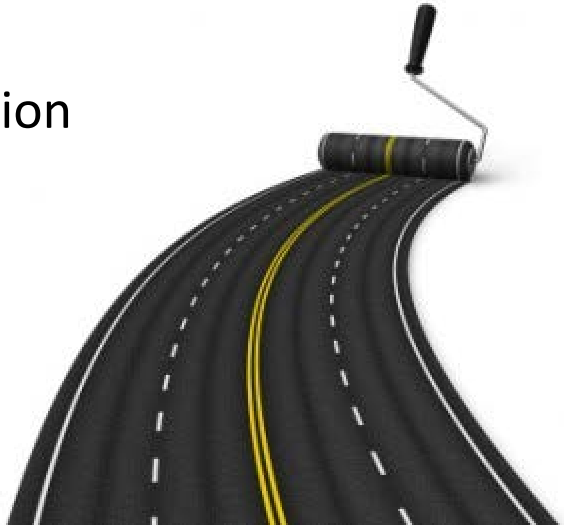
more soon!

Roadmap

1. A Constant Throughput Approximation Algorithm for ANF

2. Application to Optimizing Time-Schedule Networks

3. Extension to Virtual Network Embedding Approximation



Roadmap

1. A Constant Throughput Approximation Algorithm for ANF

2. Application to Optimizing Time-Schedule Networks

3. Extension to Virtual Network Embedding Approximation

The ANF Problem

The ANF Problem

- **ANF** = All-or-Nothing (Splittable) Multicommodity Flow Problem

Input:

- Capacitated directed graph
- (Splittable) ***multi-commodity flows*** with demands

Output:

- ***Maximal*** subset of flows…
- … such that ***capacities*** and ***demands*** are respected (i.e., the all-or-nothing)

The ANF Problem

- **ANF** = All-or-Nothing (Splittable) Multicommodity Flow Problem

Input:

- Capacitated directed graph
- (Splittable) ***multi-commodity flows*** with demands

Output:

- ***Maximal*** subset of flows…
- … such that ***capacities*** and ***demands*** are respected (i.e., the all-or-nothing)

- Well-known: problem is **NP-hard**
- **Challenge:** ***(α, β)-approximation***, α factor *from optimum* and *capacity violation* at most factor β

Our Result

Theorem: Randomized $(\alpha = O(1), \beta = \sqrt{n})$**-approximation** algorithm for throughput maximization (ANF).

Related Work

C. Chekuri, S. Khanna, and F. B. Shepherd (SIAM J. Comput. 2013):

- ***Polylog*** $\boldsymbol{\alpha}$***, but constant*** $\boldsymbol{\beta}$
- Seems non-trivial to modify to *constant* approximation with ***sublinear*** capacity violation

Our Approach: Randomized Rounding

1 Formulate **ILP**

Maximize $\sum_{i=1}^{k} f_i$ subject to

$$\sum_{(s_i,v)\in E} f_{i,(s_i,v)} = f_i \qquad \forall F_i \in \mathcal{F} \qquad (1)$$
$$\sum_{(u,v)\in E} f_{i,(u,v)} = \sum_{(v,u)\in E} f_{i,(v,u)} \qquad \forall F_i \in \mathcal{F}, v \in V \setminus \{s_i,t_i\} \qquad (2)$$
$$\sum_{i=1}^{k} f_{i,(u,v)} \le c_{(u,v)} \qquad \forall (u,v) \in E \qquad (3)$$
$$f_{i,(u,v)} \le f_i \cdot c_{(u,v)} \qquad \forall F_i \in \mathcal{F}, (u,v) \in E \qquad (4)$$
$$f_{i,(u,v)} \ge 0 \qquad \forall F_i \in \mathcal{F}, (u,v) \in E \qquad (5)$$
$$f_i \in \{0,1\} \qquad \forall F_i \in \mathcal{F} \qquad (6)$$

2 **Relax** and solve **LP**: $\tilde{f}$

Maximize $\sum_{i=1}^{k} f_i$ subject to

$$\sum_{(s_i,v)\in E} f_{i,(s_i,v)} = f_i \qquad \forall F_i \in \mathcal{F} \qquad (1)$$
$$\sum_{(u,v)\in E} f_{i,(u,v)} = \sum_{(v,u)\in E} f_{i,(v,u)} \qquad \forall F_i \in \mathcal{F}, v \in V \setminus \{s_i,t_i\} \qquad (2)$$
$$\sum_{i=1}^{k} f_{i,(u,v)} \le c_{(u,v)} \qquad \forall (u,v) \in E \qquad (3)$$
$$f_{i,(u,v)} \le f_i \cdot c_{(u,v)} \qquad \forall F_i \in \mathcal{F}, (u,v) \in E \qquad (4)$$
$$f_{i,(u,v)} \ge 0 \qquad \forall F_i \in \mathcal{F}, (u,v) \in E \qquad (5)$$
$$f_i \in \{0,1\} \qquad \forall F_i \in \mathcal{F} \qquad (6)$$

~~$\{0,1\}$~~ **(0,1)**

3 **Round** acc. to $\tilde{f}$ values (**randomized**)

With probability $\tilde{f}_i$, set $f_i = 1$, otherwise set it to 0
Scale up the fractional flow $\tilde{f}_{i,e}$ from the LP solution on edge e for commodity i by $\frac{1}{\tilde{f}_i}$, i.e., $f_{i,e} = \tilde{f}_{i,e} \times \frac{1}{\tilde{f}_i}$, for i s.t. $f_i = 1$
If the solution is greater than an α fraction of the optimal solution, return this solution; otherwise, repeat at most $\theta(\log|V|)$ times

fractional OPT

ILP

Maximize $\sum_{i=1}^{k} f_i$ subject to

$$\sum_{(s_i,v)\in E} f_{i,(s_i,v)} = f_i \qquad \forall F_i \in \mathcal{F} \qquad (1)$$

$$\sum_{(u,v)\in E} f_{i,(u,v)} = \sum_{(v,u)\in E} f_{i,(v,u)} \forall F_i \in \mathcal{F}, v \in V \setminus \{s_i, t_i\} \qquad (2)$$

$$\sum_{i=1}^{k} f_{i,(u,v)} \leq c_{(u,v)} \qquad \forall (u,v) \in E \qquad (3)$$

$$f_{i,(u,v)} \leq f_i \cdot c_{(u,v)} \qquad \forall F_i \in \mathcal{F}, (u,v) \in E \qquad (4)$$

$$f_{i,(u,v)} \geq 0 \qquad \forall F_i \in \mathcal{F}, (u,v) \in E \qquad (5)$$

$$f_i \in \{0,1\} \qquad \forall F_i \in \mathcal{F} \qquad (6)$$

maximize sum of integral flows

s.t. flow *conservation*

capacity

all-or-nothing

ILP

$$\text{Maximize } \sum_{i=1}^{k} f_i \text{ subject to}$$

$$\begin{array}{lll}
\sum_{(s_i,v)\in E} f_{i,(s_i,v)} = f_i & \forall F_i \in \mathcal{F} & (1) \\
\sum_{(u,v)\in E} f_{i,(u,v)} = \sum_{(v,u)\in E} f_{i,(v,u)} & \forall F_i \in \mathcal{F}, v \in V \setminus \{s_i, t_i\} & (2) \\
\sum_{i=1}^{k} f_{i,(u,v)} \leq c_{(u,v)} & \forall (u,v) \in E & (3) \\
f_{i,(u,v)} \leq f_i \cdot c_{(u,v)} & \forall F_i \in \mathcal{F}, (u,v) \in E & (4) \\
f_{i,(u,v)} \geq 0 & \forall F_i \in \mathcal{F}, (u,v) \in E & (5) \\
f_i \in \{0,1\} & \forall F_i \in \mathcal{F} & (6)
\end{array}$$

A **trick**: implied by (3), but there to *strengthen relaxation* (bounds violation)!

Randomized Rounding

Impact on edge capacities?

- After rounding up, we need to ***rescale*** flows by $1/f_i$, s.t. f_i=1 for each i

Randomized Rounding

- After rounding up, we need to ***rescale*** flows by $1/f_i$, s.t. $f_i=1$ for each i
- Constant approximation of **objective** follows from *Chernoff*:

$$Pr[OPT_{ALG} < \frac{1}{3} \cdot OPT_{IP}] \leq e^{-2/9}$$

- **Capacity violation** from ***modified LP*** formulation and *Hoeffding* bounds:

$$1 + \epsilon' \cdot \sqrt{2 \log |V| \cdot k}$$

size

flows

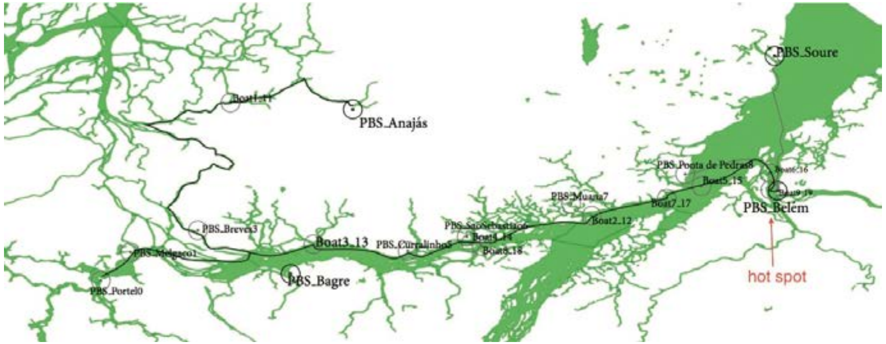
Roadmap

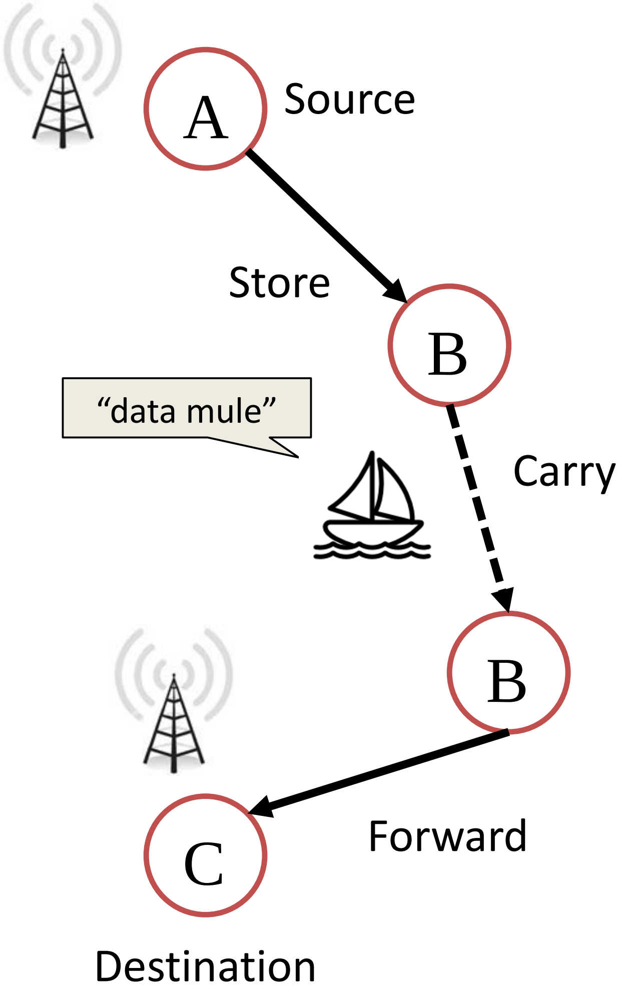
1. A Constant Throughput Approximation Algorithm for ANF ✓

2. Application to Optimizing Time-Schedule Networks

3. Extension to Virtual Network Embedding Approximation

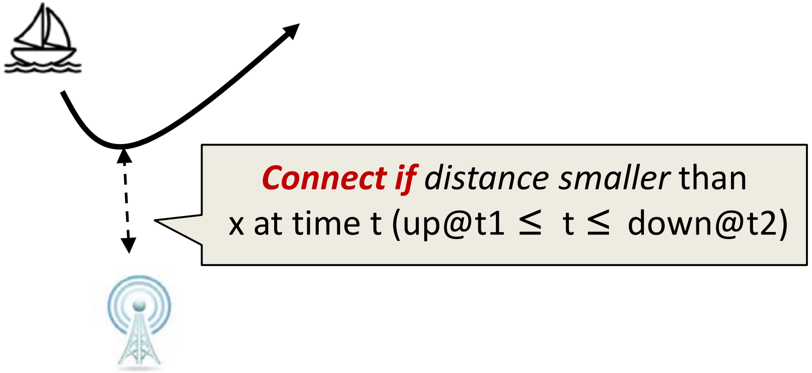
Recall: Health Care in Amazon Riverine



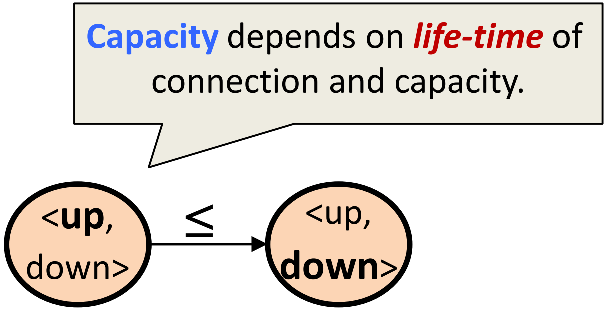


Model

- *m* **moving nodes** : has schedule P_i
- *n* **stationary nodes** (up-/down-load data)
- *k* **commodities** between (possibly different sources and destinations)

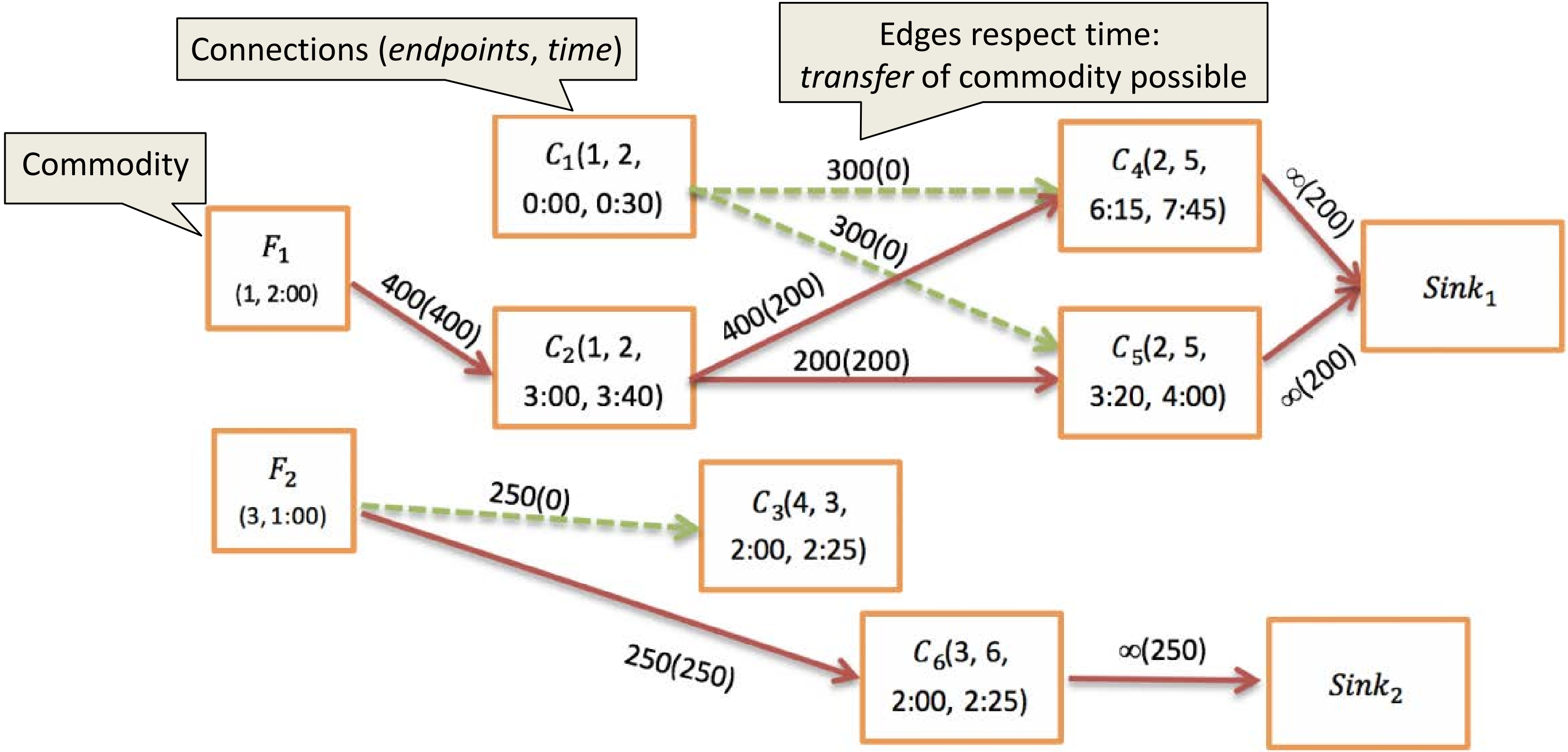


… can be transformed into a connection graph and MCF problem:

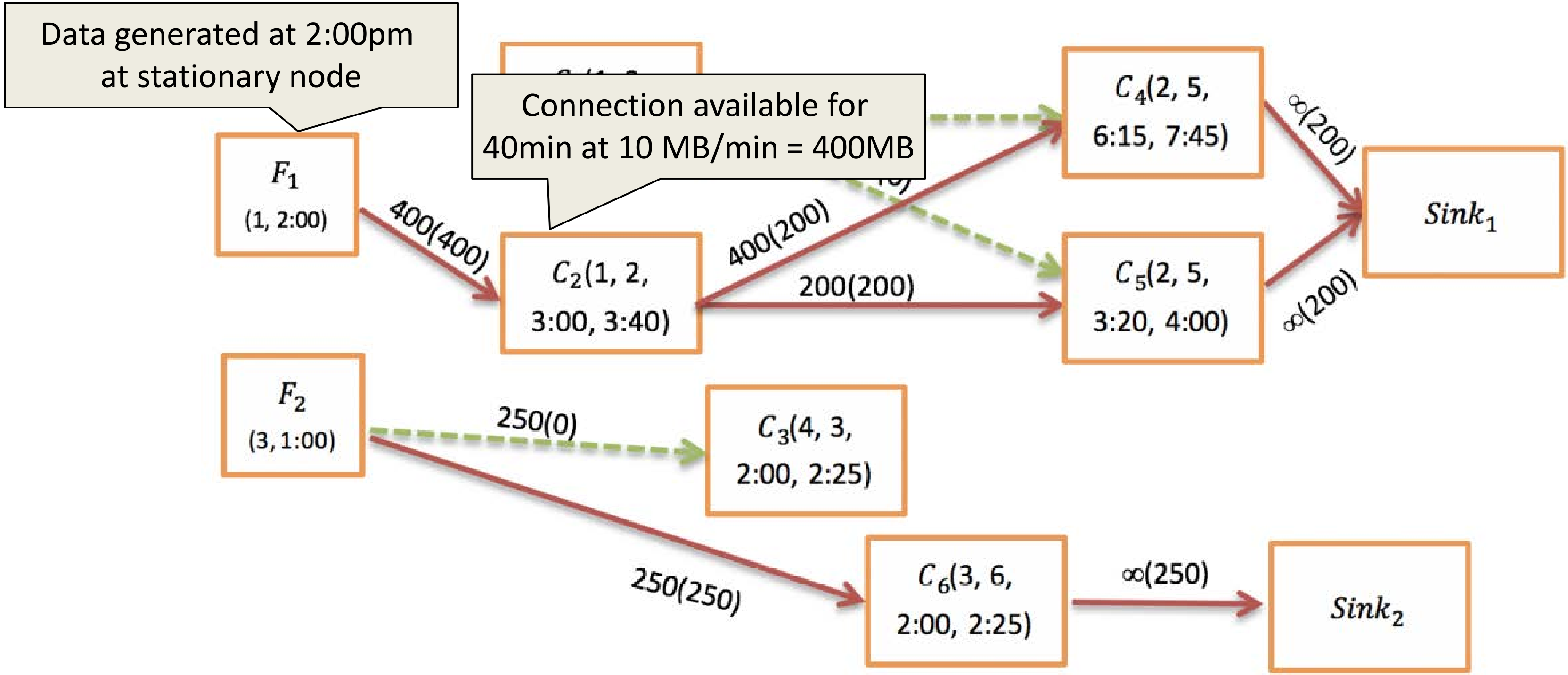


- Vertices:
  - moving and stationary nodes **plus** commodity sources **plus** connection nodes
- Directed edges:
  - From connection node C_x to connection node C_y if they ***share a common object and*** $Up_x \leq Down_y$

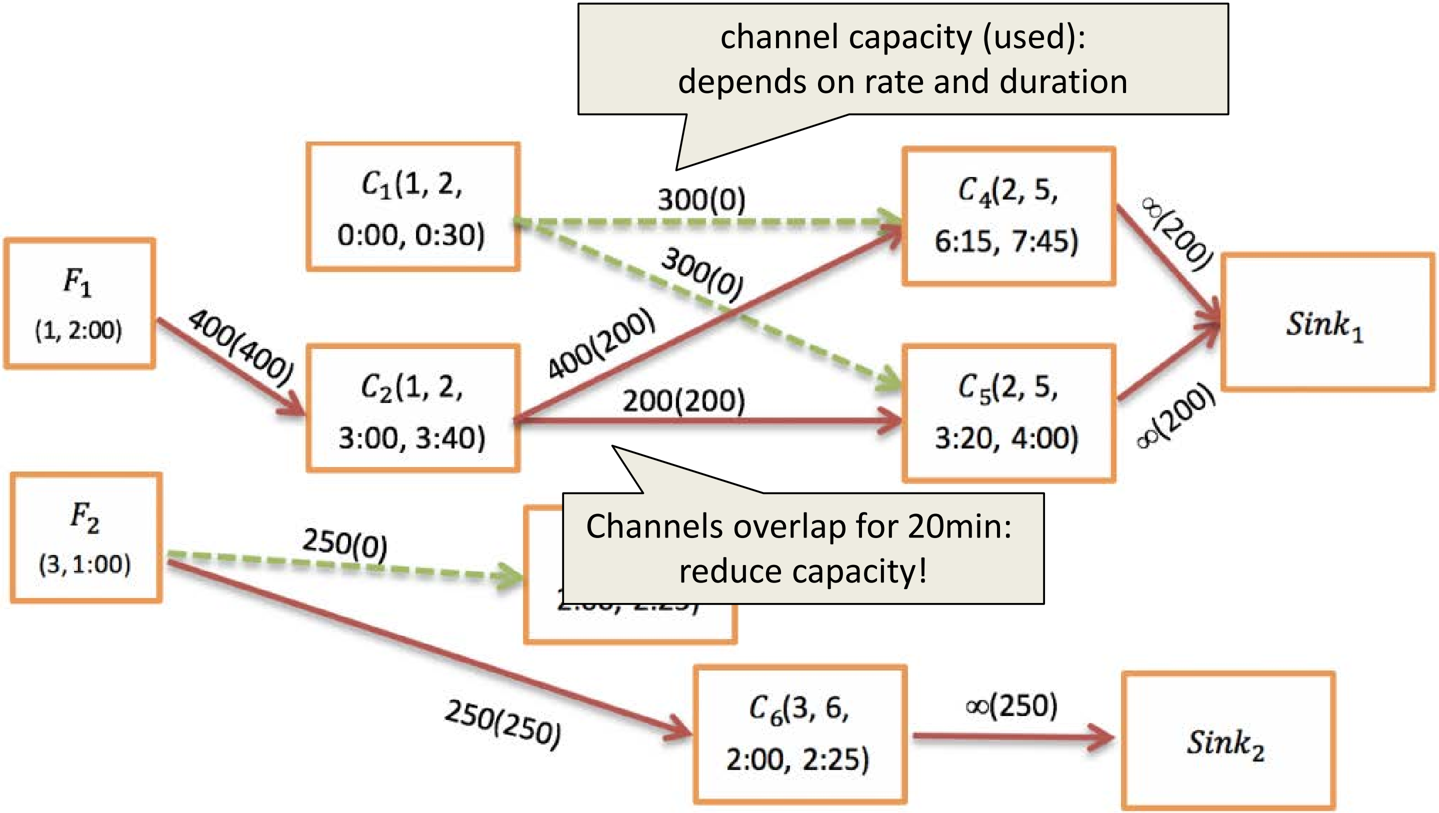
Connection Graph Example



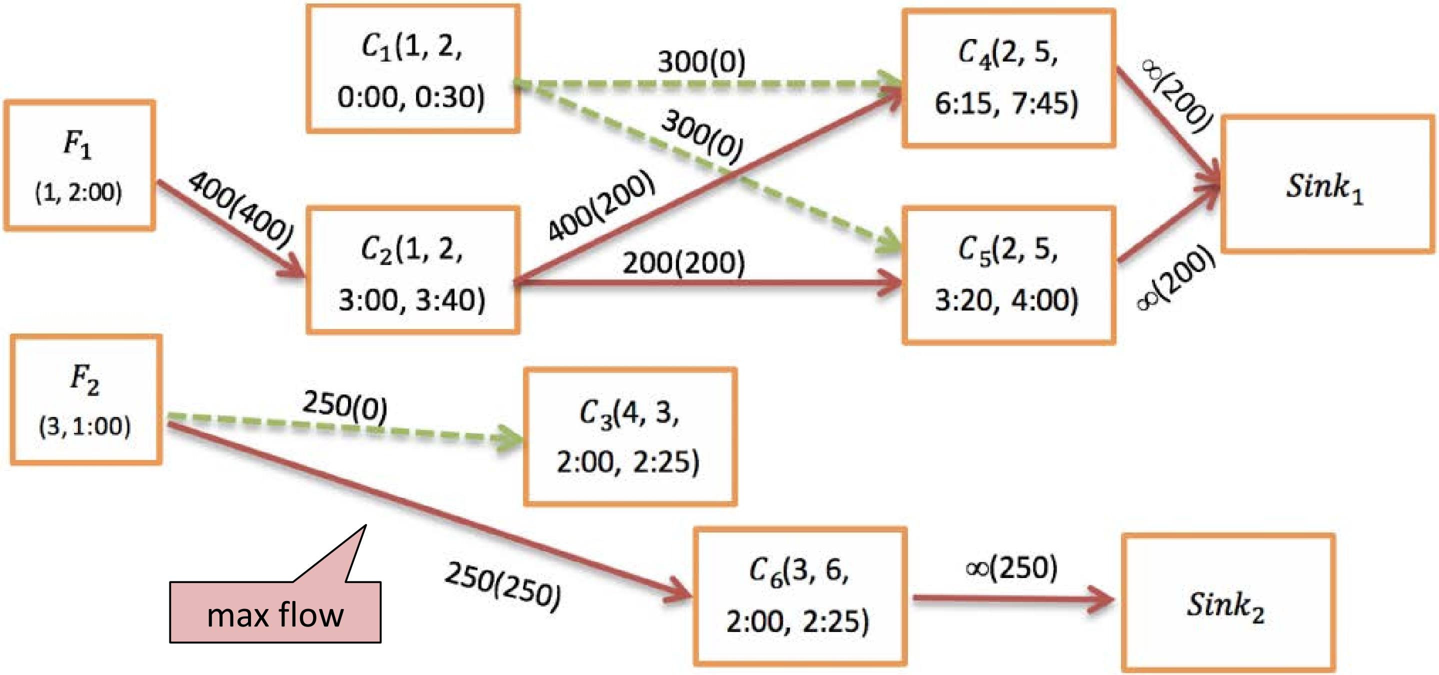
Connection Graph Example



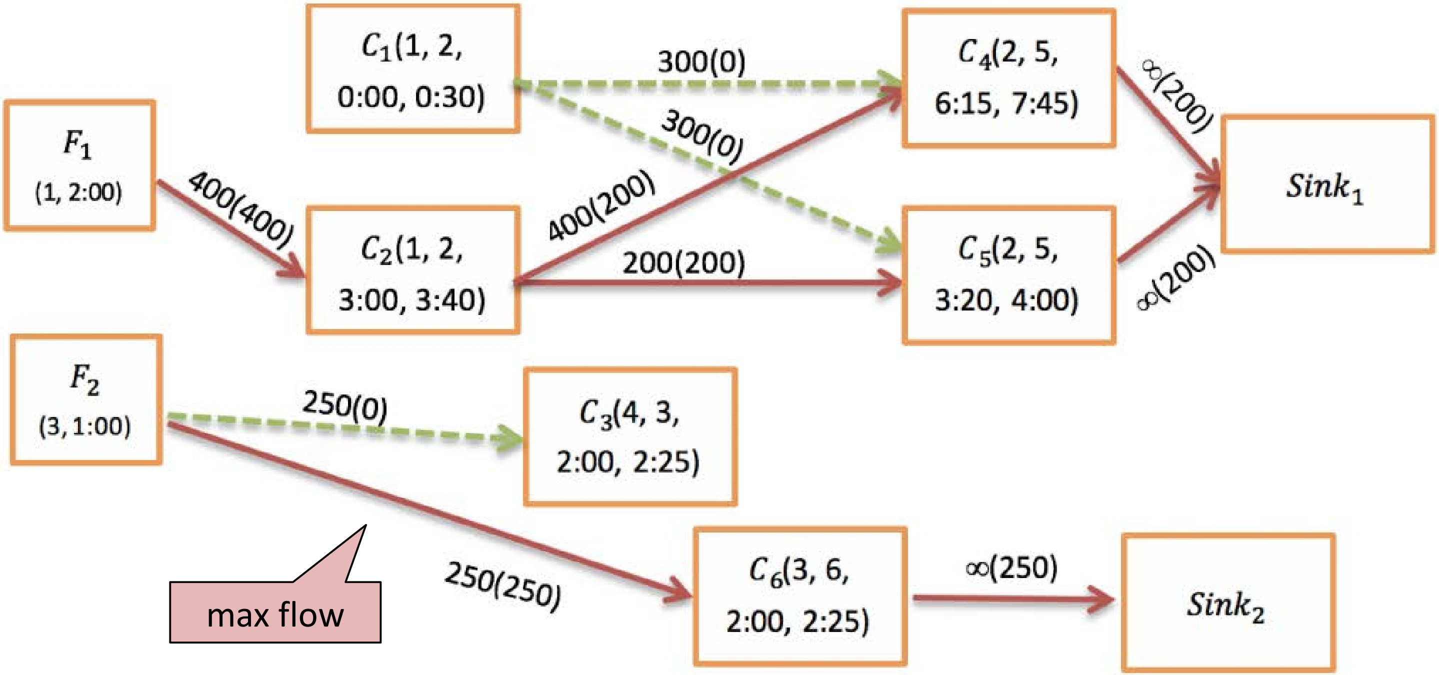
Connection Graph Example



Connection Graph Example



Connection Graph Example



Polynomial-time computable!

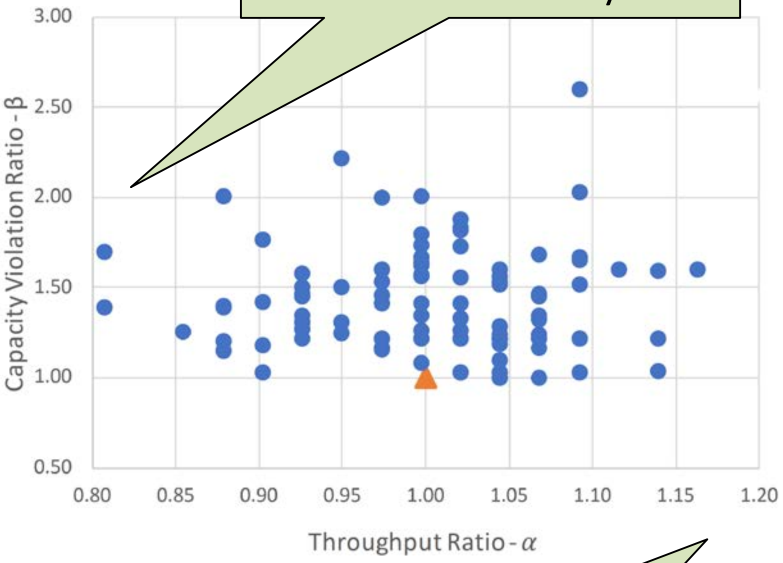
How good is the algorithm in practice?

Case Study: State of Para, Brazil

Data from State of Para, Brazil

Results

Capacity violation much lower than in analysis...



... and good approximation!

Roadmap

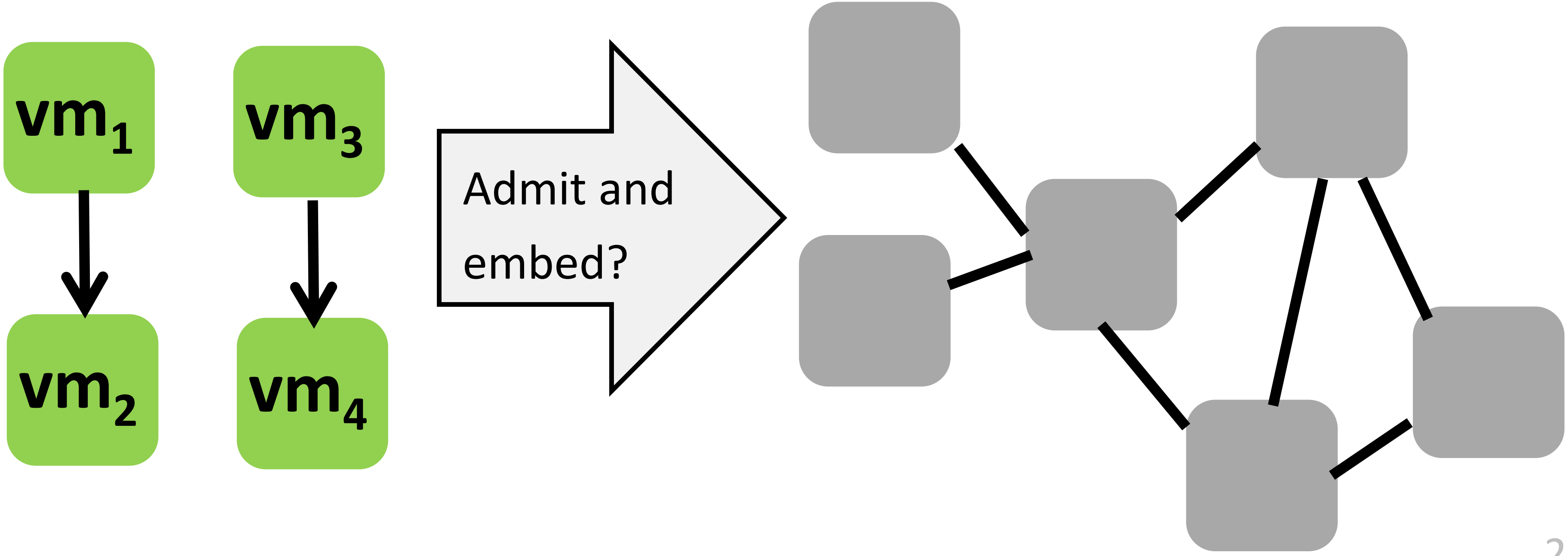
1. A Constant Throughput Approximation Algorithm for ANF ✓
2. Application to Optimizing Time-Schedule Networks ✓
3. Extension to Virtual Network Embedding Approximation

Can we use similar techniques for admitting and routing VNets?

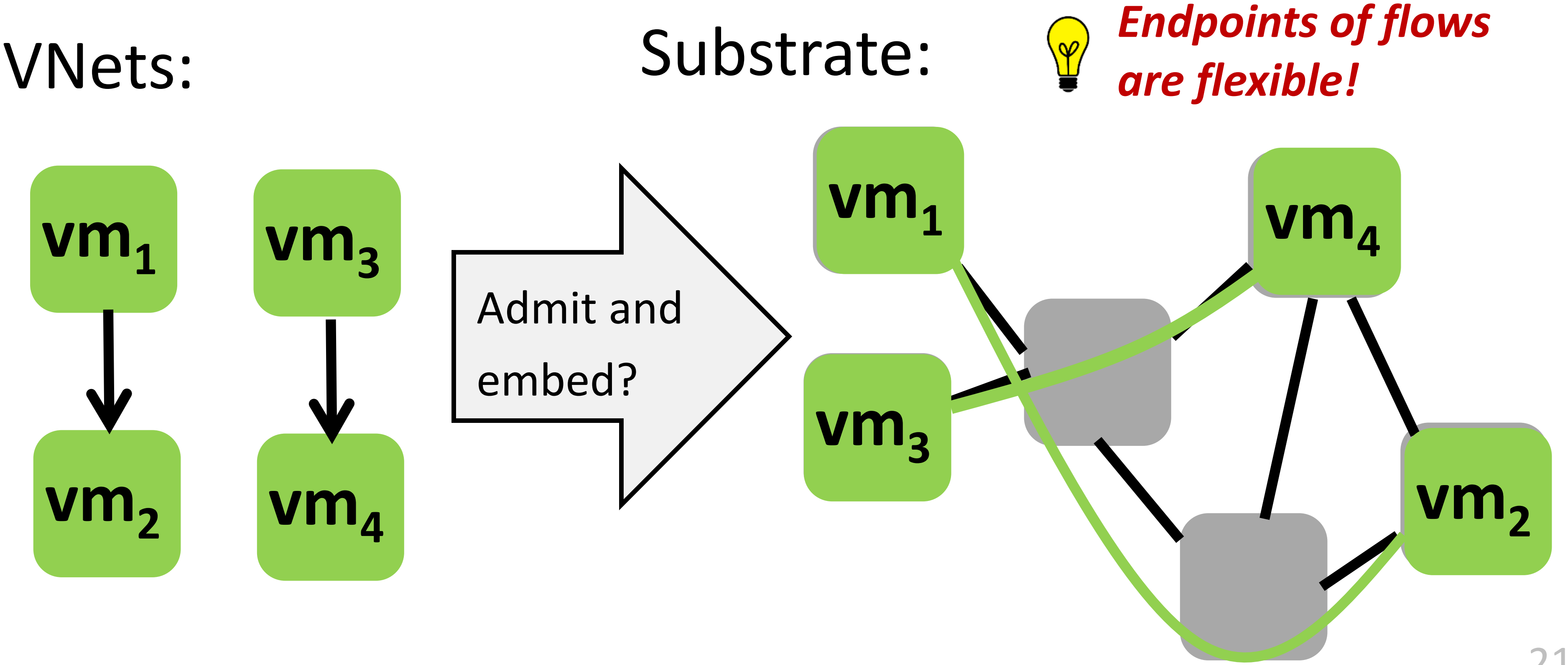
Recall: VNet Embedding Problem

VNets:

Substrate:



Recall: VNet Embedding Problem



Challenge: Decomposable ILP Formulations
Randomized Rounding based on MCF Can Fail!

$$\max \sum_{r\in\mathcal{R}} b_r x_r \tag{1}$$

$$\sum_{u\in V_S^{r,i}} y^u_{r,i} = x_r \qquad \forall r\in\mathcal{R}, i\in V_r \tag{2}$$

Each virtual node ***should be placed***

$$\sum_{u\in V_S\setminus V_S^{r,i}} y^u_{r,i} = 0 \qquad \forall r\in\mathcal{R}, i\in V_r \tag{3}$$

Forbidden node mappings

$$\begin{bmatrix} \sum\limits_{(u,v)\in\delta^+(u)} z^{u,v}_{r,i,j} \\ - \sum\limits_{(v,u)\in\delta^-(u)} z^{v,u}_{r,i,j} \end{bmatrix} = \begin{bmatrix} y^u_{r,i} \\ -y^u_{r,j} \end{bmatrix} \forall \begin{bmatrix} r\in\mathcal{R}, (i,j)\in E_r, \\ u\in V_S \end{bmatrix} \tag{4}$$

Induce a **flow** for each virtual edge

$$z^{u,v}_{r,i,j} = 0 \qquad \forall \begin{bmatrix} r\in\mathcal{R}, (i,j)\in E_r, \\ (u,v)\in E_S\setminus E_S^{r,i,j} \end{bmatrix} \tag{5}$$

Forbidden edge mappings

$$\sum_{i\in V_r, \tau_r(i)=\tau} d_r(i)\cdot y^u_{r,i} = a_r^{\tau,u} \qquad \forall r\in\mathcal{R}, (\tau,u)\in R_S^V \tag{6}$$

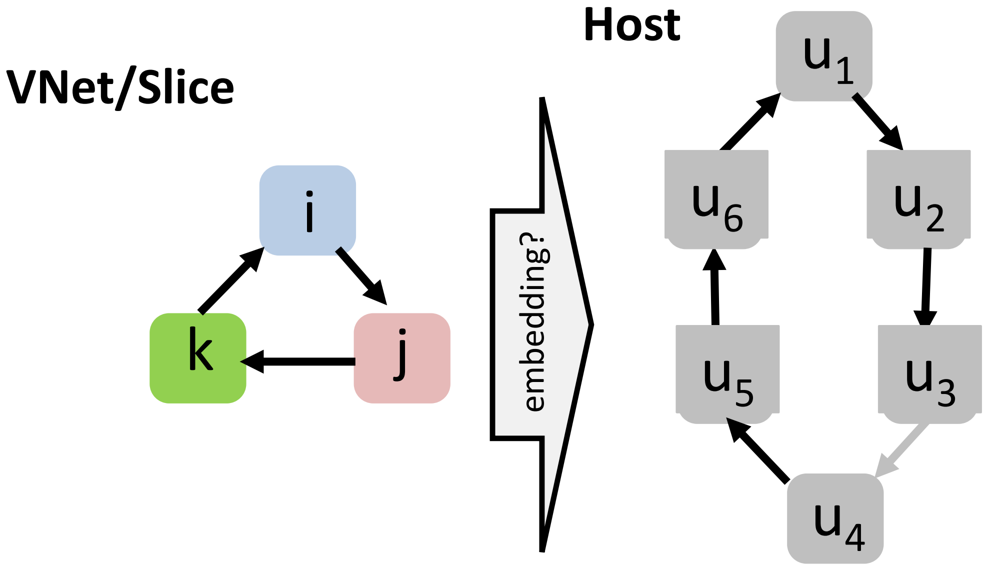
$$\sum_{(i,j)\in E_r} d_r(i,j)\cdot z^{u,v}_{r,i,j} = a_r^{u,v} \qquad \forall r\in\mathcal{R}, (u,v)\in E_S \tag{7}$$

Compute cumulative ***allocations***

$$\sum_{r\in\mathcal{R}} a_r^{x,y} \leq d_S(x,y)\ \forall (x,y)\in R_S \tag{8}$$

Respect resource ***capacities***

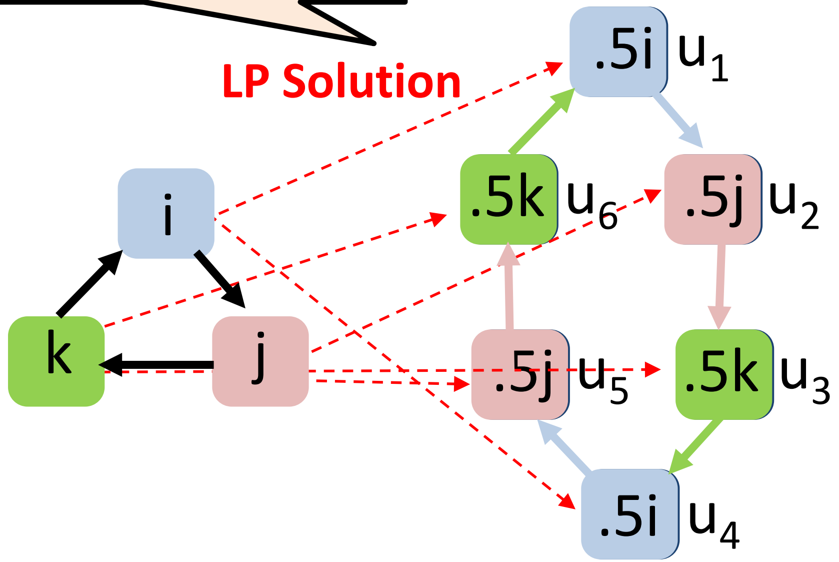
Challenge: Decomposable ILP Formulations
Randomized Rounding based on MCF Can Fail!



Challenge: Decomposable ILP Formulations

nding based on MCF Can Fail!

Valid LP solution: virtual node mappings sum to 1 and each virtual node connects to its neighboring node **with half a unit of flow**…

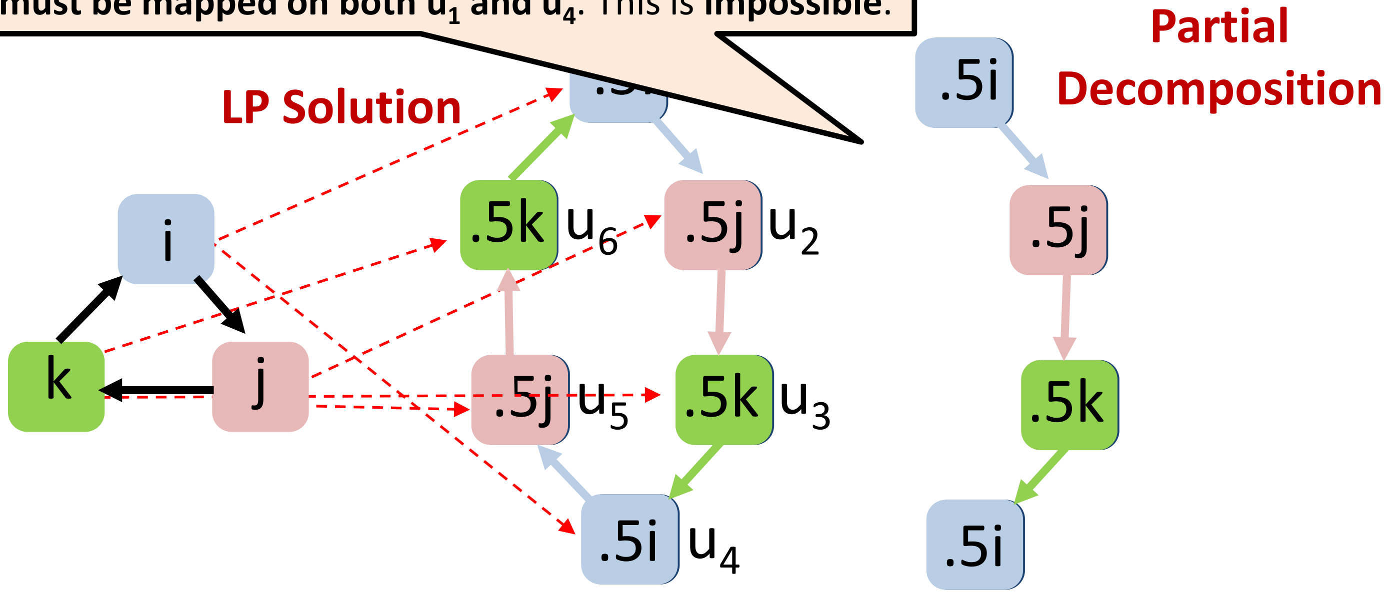


Relaxations of classic MCF formulation **cannot be decomposed** into convex combinations of valid mappings (so we **need different formulations**!)

Formulations

MCF Can Fail!

Impossible to decompose and extract **any single valid mapping**.
Intuition: Node *i* is mapped to u_1 and **the only neighboring node** that hosts *j* is u_2, so *i* must be fully mapped on u_1 and *j* on u_2. Similarly, *k* must be mapped on u_3. But flow of virtual edge (*k*,*i*) leaving u_3 only leads to u_4, so ***i* must be mapped on both u_1 and u_4**. This is **impossible**.



Relaxations of classic MCF formulation **cannot be decomposed** into convex combinations of valid mappings (so we **need different formulations**!)

Approximation of VNet Embedding

1. To avoid dependencies, decompose VNet into **forests and cycles**, orient

2. Formulate alternative, decomposable **ILP**

$$\max \sum_{r \in \mathcal{R}} b_r x_r$$

Cons. (2) - (7) for $G_r^{\mathcal{F}}$ on variables $(x_r, \vec{y}_r, \vec{z}_r, \vec{a}_r)[\mathcal{F}_r]$	$\forall r \in \mathcal{R}$
Cons. (2) - (7) for $G_r^{C_k}$ on variables $(x_r, \vec{y}_r, \vec{z}_r, \vec{a}_r)[C_k, w]$	$\forall r \in \mathcal{R}, C_k \in \mathcal{C}_r, w \in V_{S,t}^{C_k}$
$x_r = \sum_{u \in V_S^{r,i}} y_{r,i}^u$	$\forall r \in \mathcal{R}, i \in V_r$
$y_{r,i}^u = y_{r,i}^u[\mathcal{F}]$	$\forall r \in \mathcal{R}, i \in V_r^{\mathcal{F}}, u \in V_S^{r,i}$
$y_{r,i}^u = \sum_{w \in t_r^{C_k}} y_{r,i}^u[C_k, w]$	$\forall \begin{bmatrix} r \in \mathcal{R}, i \in V_r, u \in V_S^{r,i}; \\ C_k \in \mathcal{C}_r : i \in V_r^{C_k} \end{bmatrix}$
$0 = y_{r,t_r^{C_k}}^u[C_k, w]$	$\forall \begin{bmatrix} r \in \mathcal{R}, C_k \in \mathcal{C}_r, w \in V_{S,t}^{C_k}, \\ u \in V_{S,t}^{C_k} \setminus \{w\} \end{bmatrix}$
$a_r^{\tau,u} = \sum_{i \in V_r, \tau_r(i) = \tau} d_r(i) \cdot y_{r,i}^u$	$\forall r \in \mathcal{R}, (\tau, u) \in R_S^V$
$a_r^{u,v} = a_r^{u,v}[\mathcal{F}] + \sum_{C_k \in \mathcal{C}_r, w \in V_{S,t}^{C_k}} a_r^{u,v}[C_k, w]$	$\forall r \in \mathcal{R}, (u,v) \in E_S$
$\sum_{r \in \mathcal{R}} a_r^{x,y} \leq d_S(x,y)$	$\forall (x,y) \in R_S$

3. Apply **randomized rounding**

foreach $r \in \mathcal{R}$ **do** // preprocess requests
  compute LP Formulation 2 for request r maximizing x_r
  if $x_r < 1$ **then remove** request r from the set $\mathcal{R}$
compute LP Formulation 2 for $\mathcal{R}$ maximizing $\sum_{r \in \mathcal{R}} b_r \cdot x_r$
foreach $r \in \mathcal{R}$ **do** // perform decomposition
  compute $\mathcal{D}_r = \{(f_r^k, m_r^k)\}_k$ from LP solution
do // perform randomized rounding
  foreach $r \in \mathcal{R}$ **select** m_r^k **with probability** f_r^k
while (solution is *not* (α, β, γ)-approximate and maximal rounding tries are not exceeded)

Conclusion

- Constant approximation with for **throughput maximization**
- Can be used to solve transmission scheduling on **time-schedule networks**
- Non-trivial **augmentation** required (*future work*: improve)
- But does not appear in ***simulations***
- Applications to **virtual network embedding** problem: works too!
- *Future work*: derandomization


Thank you!
Questions?

Further Reading

- Robust data mule networks with remote healthcare applications in the Amazon region: A fountain code approach Mengxue Liu, Thienne Johnson, Rachit Agarwal, Alon Efrat, Andrea Richa, Mauro Margalho Coutinho. **HealthCom** 2015.

- Charting the Complexity Landscape of Virtual Network Embeddings
Matthias Rost and Stefan Schmid. **IFIP Networking**, Zurich, Switzerland, May 2018.