An Approximation Algorithm for Path Computation and Function Placement in SDNs

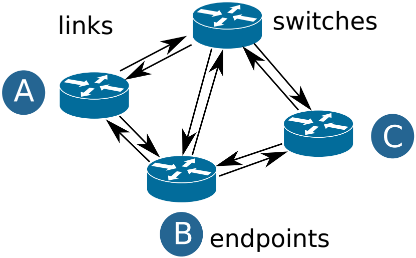
Matthias Rost
Technische Universität Berlin

July 21, SIROCCO 2016

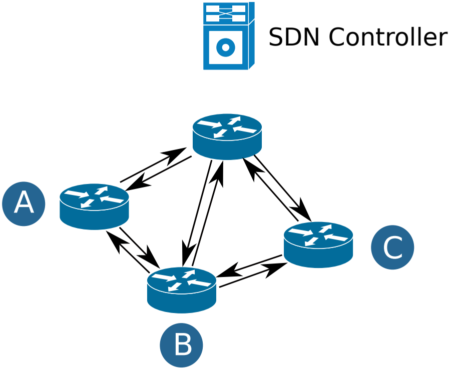
Joint work with *Guy Even* and *Stefan Schmid*

Introduction

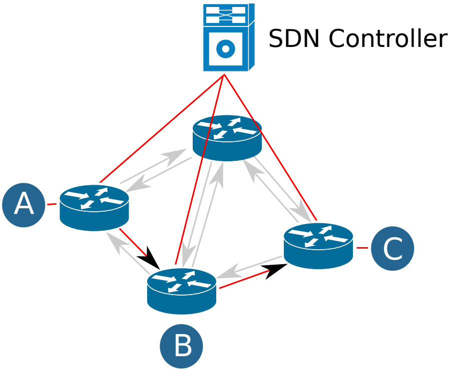
Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]



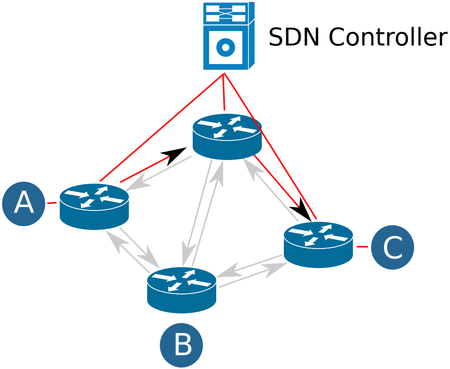
Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]



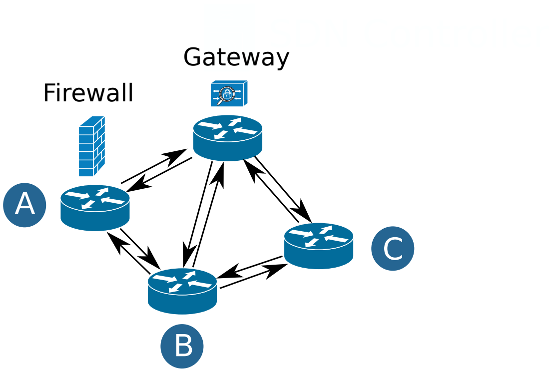
Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]



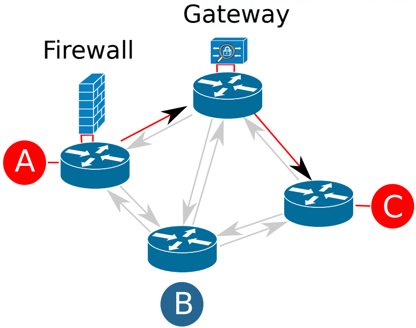
Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]

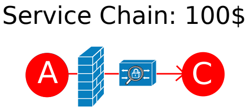


Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]

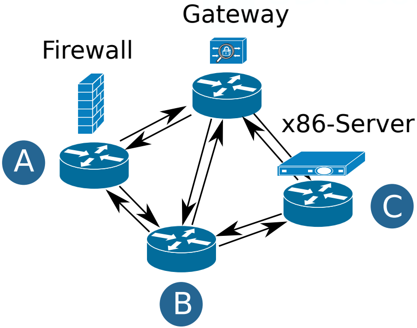


Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]

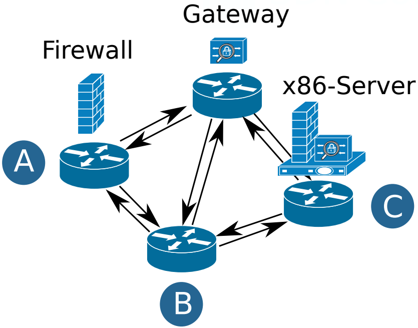




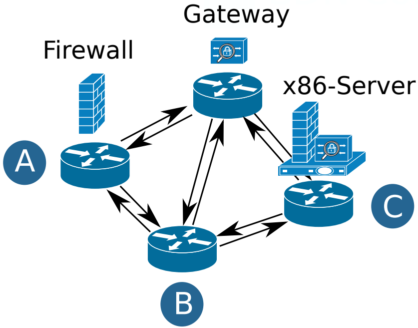
Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]

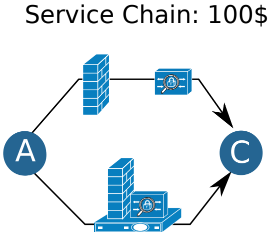


Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]

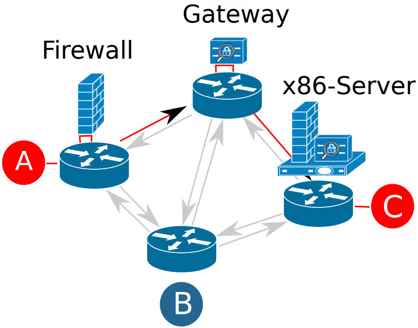


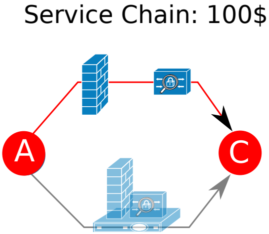
Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]



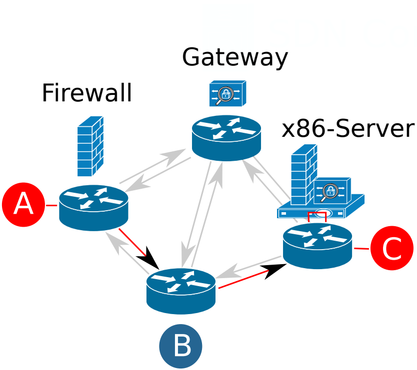


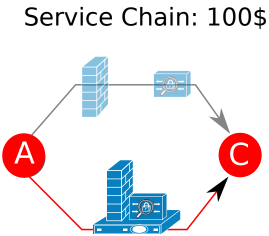
Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]



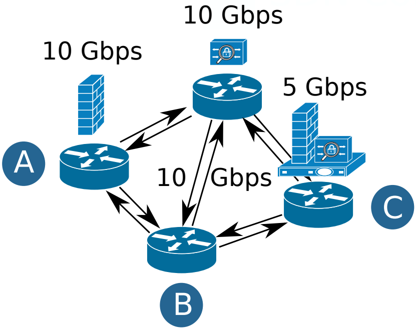


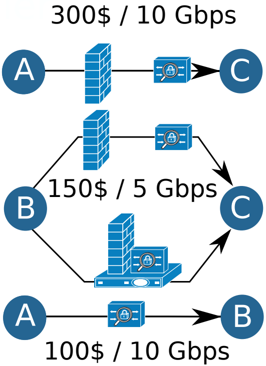
Opportunities: Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [Kreutz et al., 2015]



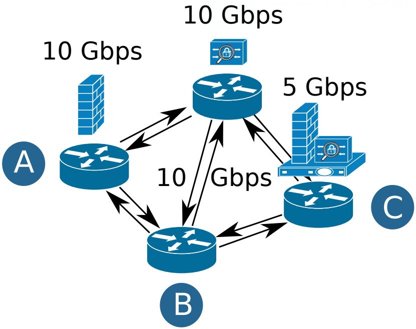


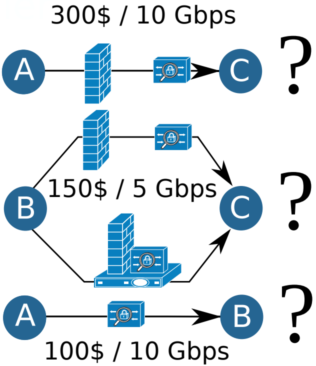
Path Computation and Function Placement Problem (PCFP)



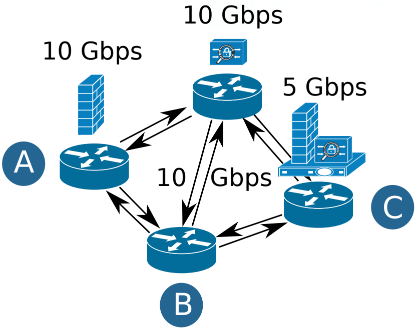


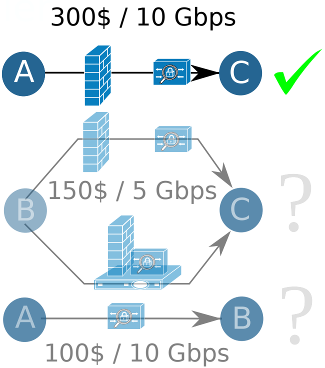
Path Computation and Function Placement Problem (PCFP)



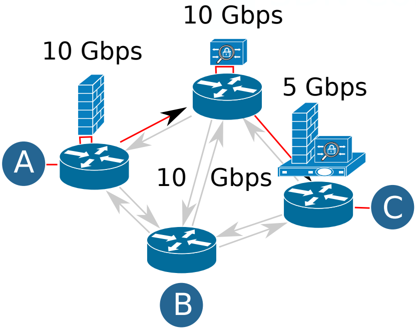


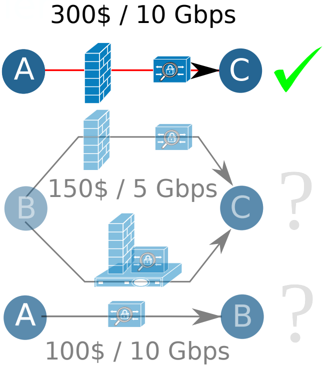
Path Computation and Function Placement Problem (PCFP)



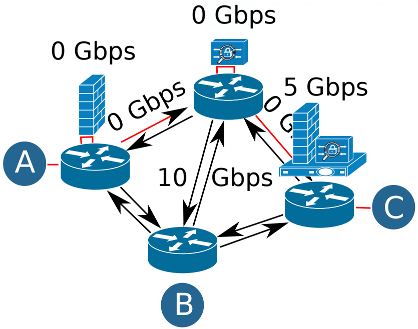


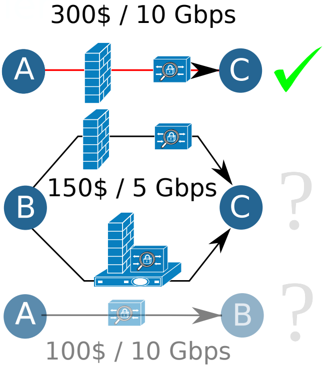
Path Computation and Function Placement Problem (PCFP)



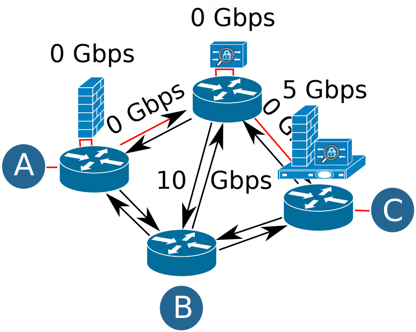


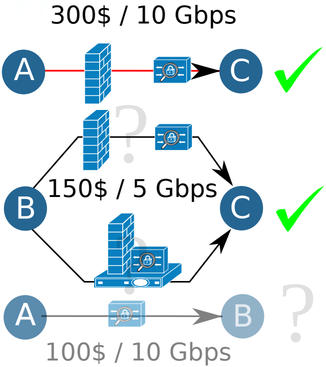
Path Computation and Function Placement Problem (PCFP)



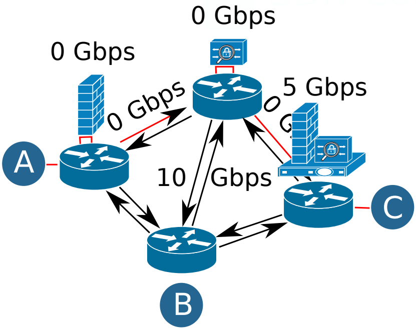


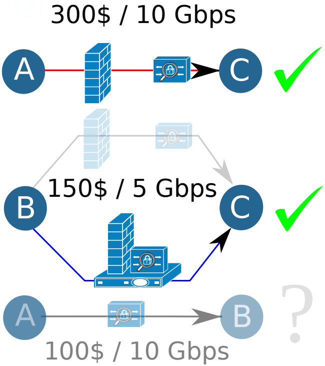
Path Computation and Function Placement Problem (PCFP)



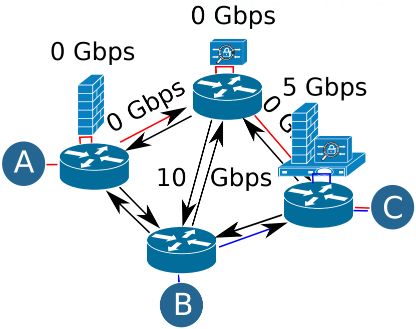


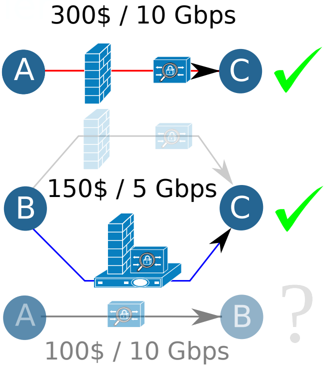
Path Computation and Function Placement Problem (PCFP)



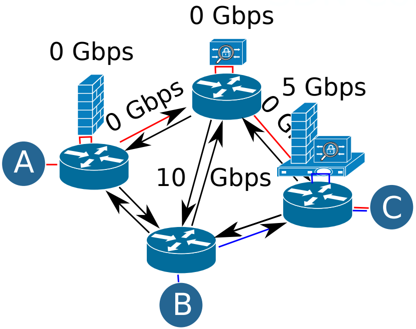


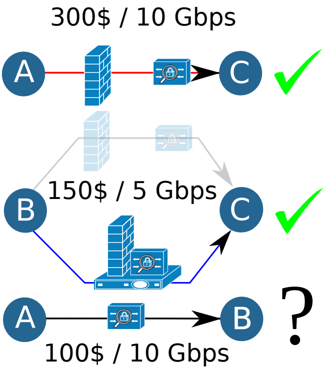
Path Computation and Function Placement Problem (PCFP)



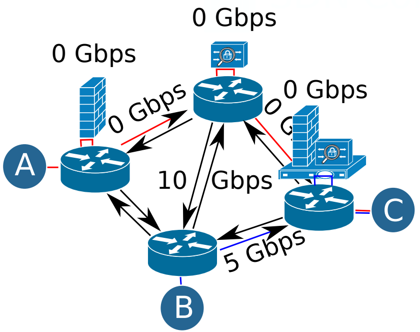


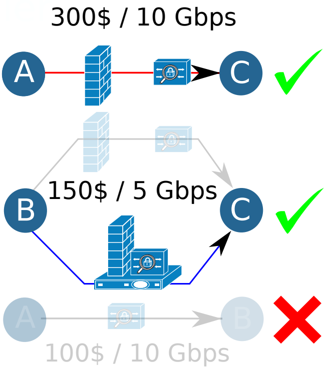
Path Computation and Function Placement Problem (PCFP)



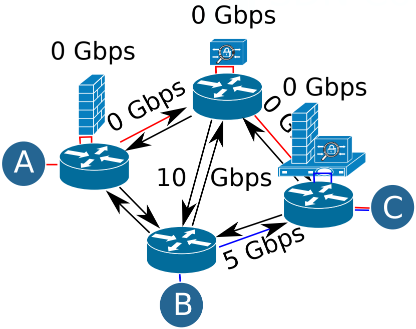


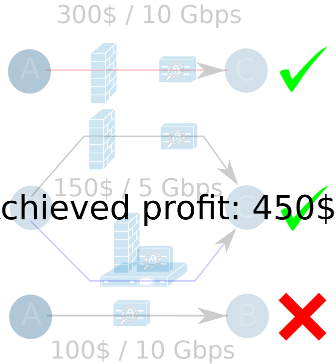
Path Computation and Function Placement Problem (PCFP)





Path Computation and Function Placement Problem (PCFP)





Achieved profit: 450$

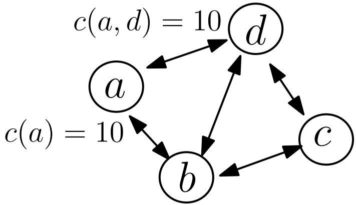
Formal Definition of PCFP

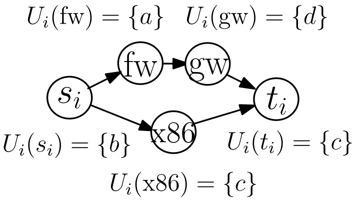
Substrate Network

- Directed network $N = (V, E)$
- capacities $c : V \cup E \to \mathbb{R}_{\geq 0}$

Requests

- Acyclic graph $G_i = (X_i, Y_i)$
- mapping restrictions $U_i : X_i \cup Y_i \to 2^V \cup 2^E$
- benefit, demand: $b_i, d_i \in \mathbb{R}_{\geq 0}$
- start, target: $s_i, t_i \in X_i$



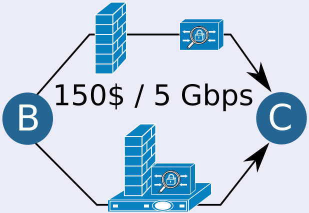


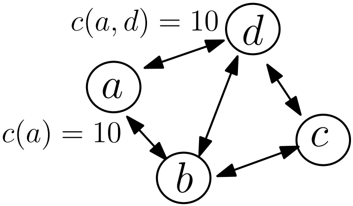
Formal Definition of PCFP

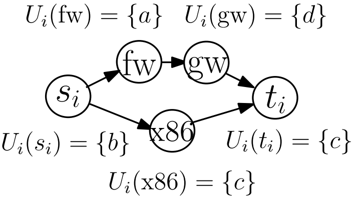
Substrate Network

- Directed network $N = (V, E)$
- capacities $c : V \cup E \rightarrow \mathbb{R}_{\geq 0}$

Requests







Formal Definition of PCFP

Substrate Network

- Directed network $N = (V, E)$
- capacities $c : V \cup E \rightarrow \mathbb{R}_{\geq 0}$

Requests

- Acyclic graph $G_i = (X_i, Y_i)$
- mapping restrictions $U_i : X_i \cup Y_i \rightarrow 2^V \cup 2^E$
- benefit, demand: $b_i, d_i \in \mathbb{R}_{\geq 0}$
- start, target: $s_i, t_i \in X_i$

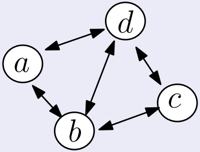
Task

Find set $I' \subseteq I$ of requests to embed and valid realizations $\bar{p}_i$ for $i \in I'$, s.t.

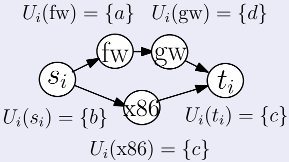
1. $\bar{p}_i$ represents a path from $s_i \rightsquigarrow t_i$
2. capacities of substrate nodes and edges is not violated
3. the profit $\sum_{i \in I'} b_i$ is maximized.

Formal Definition of PCFP

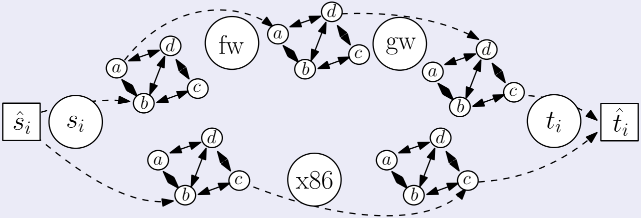
Substrate



Request r_i



Valid Realizations via Product Networks: $pn(N, r_i)$

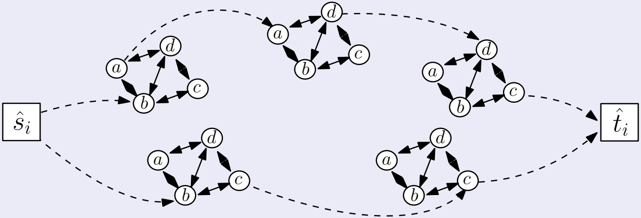


Formal Definition of PCFP

Valid Realizations

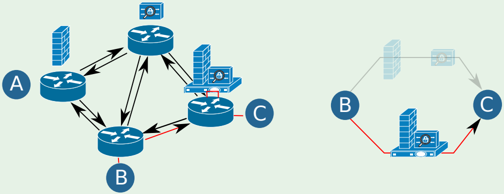
Any $\hat{s}_i$-$\hat{t}_i$ path in $pn(N, r_i)$ represents a valid realization of request r_i.

Valid Realizations via Product Networks: $pn(N, r_i)$

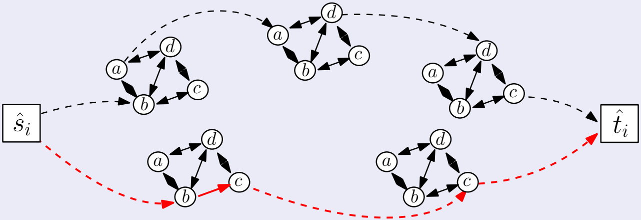


Formal Definition of PCFP

Valid Realizations: Example 1

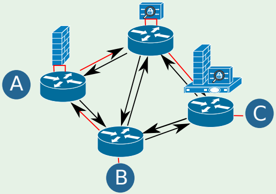


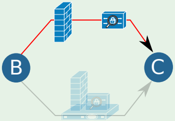
Valid Realizations in Product Networks $pn(N, r_i)$



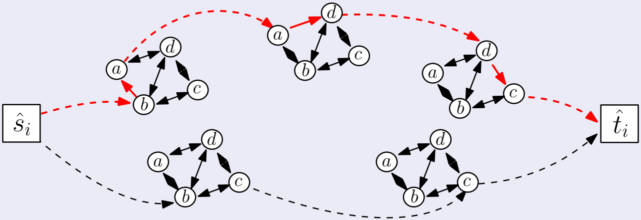
Formal Definition of PCFP

Valid Realizations: Example 2





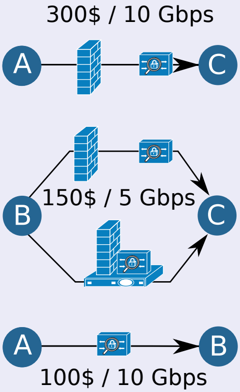
Valid Realizations in Product Network $pn(N, r_i)$



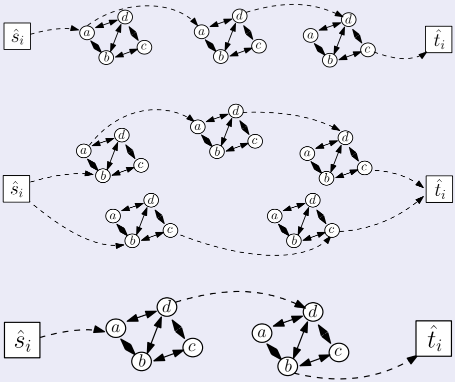
Approximating PCFP

PCFP as a Flow Problem

Requests



Product Networks

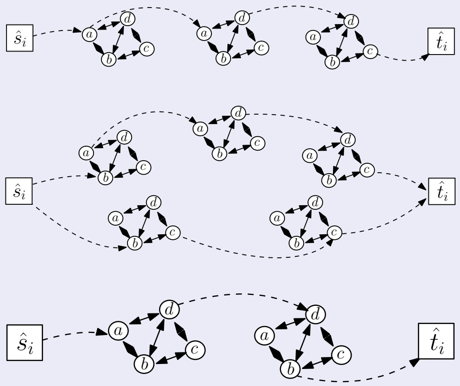


PCFP as a Flow Problem

Flow Formulation

- Compute *unsplittable* flows $\bar{f}_i : E(pn(N, r_i)) \rightarrow \{0, d_i\}$
- Flow preservation within each product network (except at $\hat{s}_i$ and $\hat{t}_i$)
- $\max \sum_i b_i \cdot |\bar{f}_i| / d_i$
- s.t. node and edge capacities are not violated

Product Networks

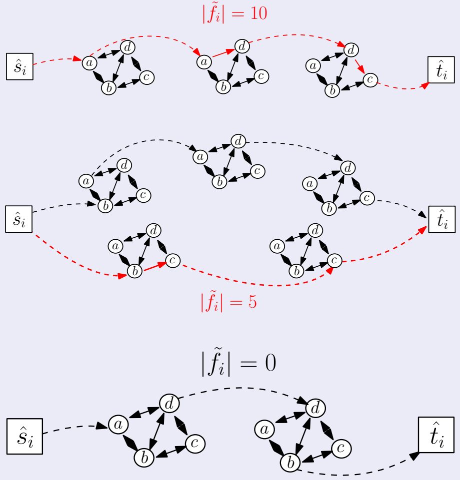


PCFP as a Flow Problem

Flow Formulation

- Compute *unsplittable* flows $\bar{f}_i : E(pn(N, r_i)) \to \{0, d_i\}$
- Flow preservation within each product network (except at $\hat{s}_i$ and $\hat{t}_i$)
- $\max \sum_i b_i \cdot |\bar{f}_i|/d_i$
- s.t. node and edge capacities are not violated

Flow Solution in Product Networks



PCFP as a Flow Problem

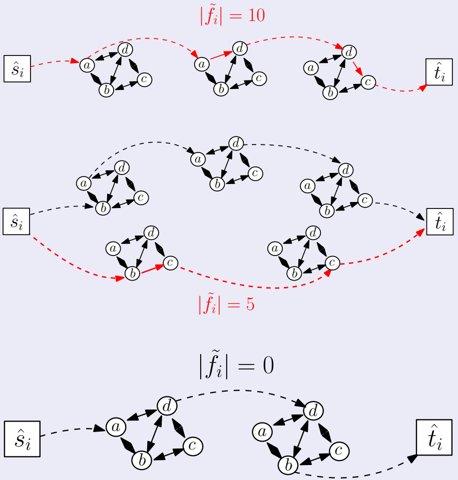
Flow Formulation

- Compute *unsplittable* flows $\bar{f}_i : E(pn(N, r_i)) \to \{0, d_i\}$
- Flow preservation within each product network (except at $\hat{s}_i$ and $\hat{t}_i$)
- $\max \sum_i b_i \cdot |\bar{f}_i| / d_i$
- s.t. node and edge capacities are not violated

NP-Hardness follows from ...

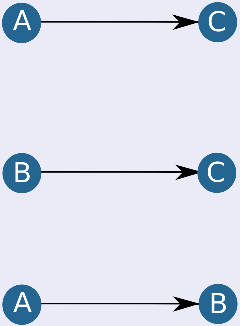
the Unsplittable Flow Problem.

Flow Solution in Product Networks

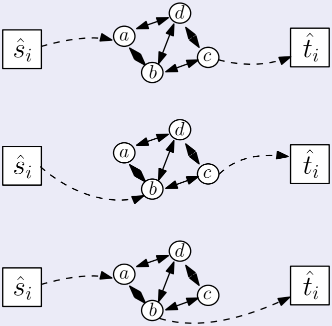


PCFP as a Flow Problem

USF Requests



Product Networks



NP-Hardness follows from . . .

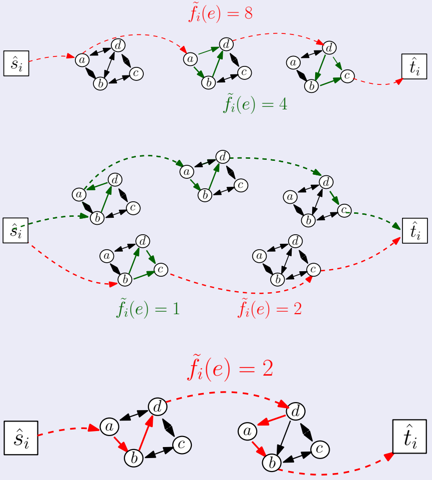
the Unsplittable Flow Problem.

Approximating PCFP using Randomized Rounding: Idea

Flow Formulation

- Compute flows as above, but relax integrality: $\bar{f}_i : E(pn(N, r_i)) \rightarrow [0, d_i]$

Product Networks



Approximating PCFP using Randomized Rounding: Idea

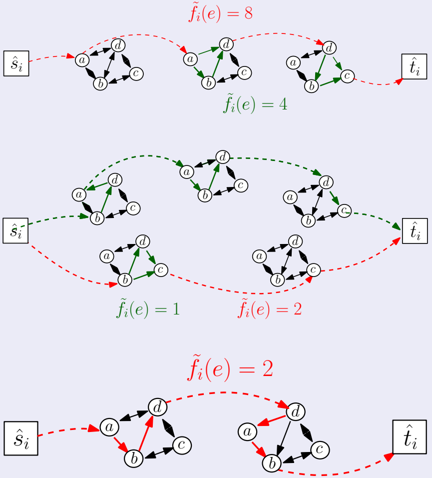
Flow Formulation

- Compute flows as above, but relax integrality: $\bar{f}_i : E(pn(N, r_i)) \rightarrow [0, d_i]$

Algorithm

1. Scale capacities by $1/(1+\varepsilon)$
2. Compute fractional flows
3. Place request $i \in I$ into set $I' \subseteq I$ with probability $|\bar{f}_i|/d_i$
4. Perform random walks to obtain $\bar{p}_i$ for $i' \in I'$

Product Networks



Approximating PCFP using Randomized Rounding: Idea

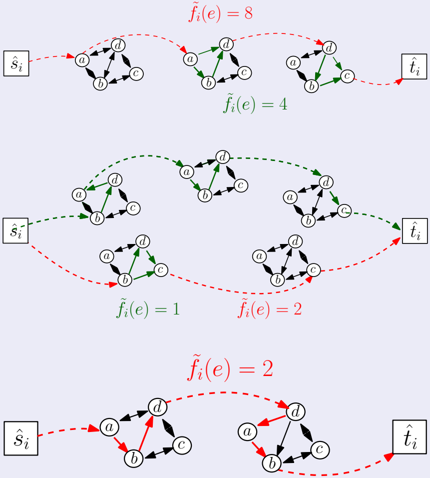
Algorithm

1. Scale capacities by $1/(1+\varepsilon)$
2. Compute fractional flows
3. Place request $i \in I$ into set $I' \subseteq I$ with probability $|\bar{f}_i|/d_i$
4. Perform random walks to obtain $\bar{p}_i$ for $i' \in I'$

Questions

1. What is the expected profit?
2. How badly do we violate capacities?

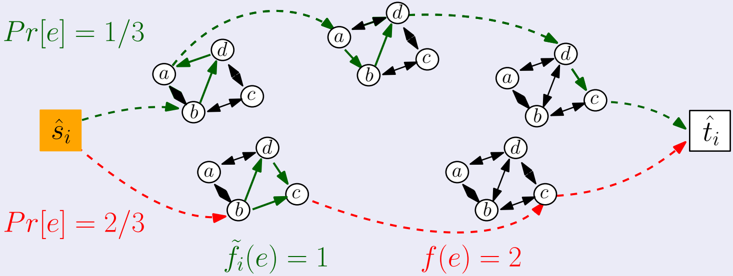
Product Networks



Performing Random Walks

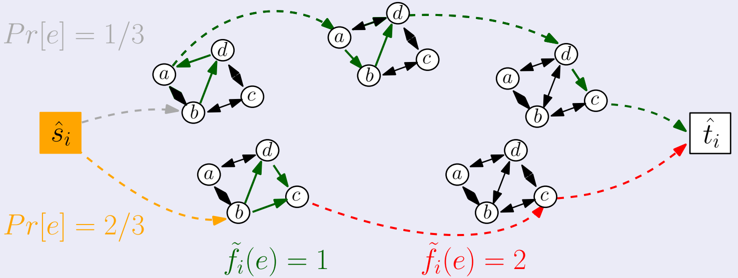
Approximating PCFP using Randomized Rounding: Random Walk

Random Walk at node v: $Pr[\text{choose } e] = \bar{f}_i(e) / \sum_{e' \in \delta^+(v)} \bar{f}_i(e)$



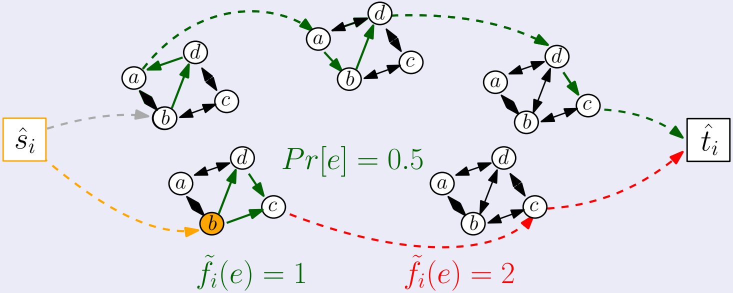
Approximating PCFP using Randomized Rounding: Random Walk

Random Walk at node v: $Pr[\text{choose } e] = \bar{f}_i(e) / \sum_{e' \in \delta^+(v)} \bar{f}_i(e)$



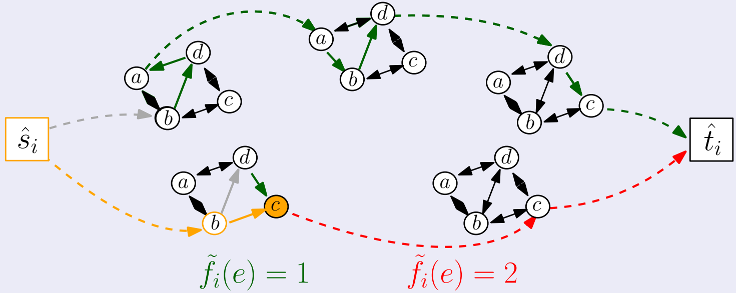
Approximating PCFP using Randomized Rounding: Random Walk

Random Walk at node v: $Pr[\text{choose } e] = \bar{f}_i(e) / \sum_{e' \in \delta^+(v)} \bar{f}_i(e)$



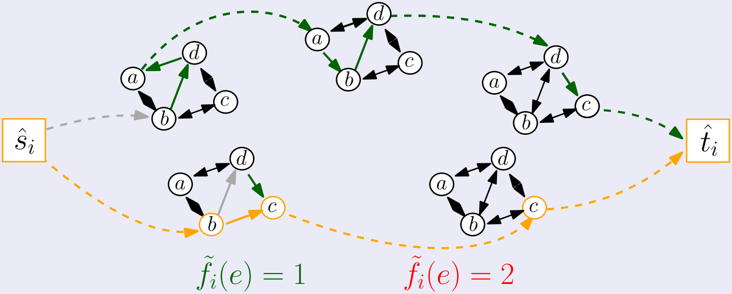
Approximating PCFP using Randomized Rounding: Random Walk

Random Walk at node v: $Pr[\text{choose } e] = \bar{f}_i(e) / \sum_{e' \in \delta^+(v)} \bar{f}_i(e)$



Approximating PCFP using Randomized Rounding: Random Walk

Random Walk at node v: $Pr[\text{choose } e] = \bar{f}_i(e) / \sum_{e' \in \delta^+(v)} \bar{f}_i(e)$

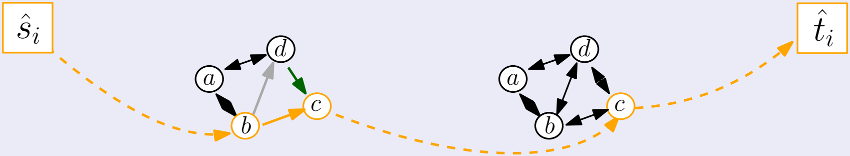


Theorem (by induction, cf. Motwani et al. [1996])

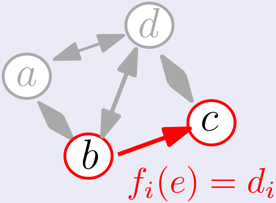
The probability that an edge $e \in E(pn(N, r_i))$ will be used equals $\bar{f}_i(e)/d_i$. Hence, the expected load on an edge $e \in E(pn(N, r_i))$ equals $\bar{f}_i(e)$.

Approximating PCFP using Randomized Rounding: Random Walk

Resulting realization $\bar{p}_i$

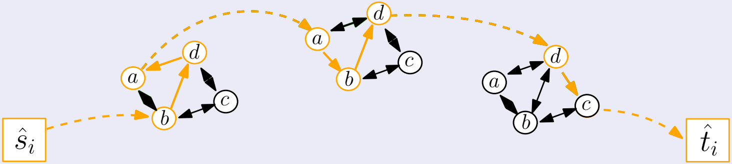


Projected flow f_i

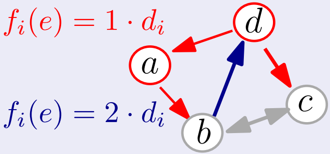


Approximating PCFP using Randomized Rounding: Random Walk

Other potential realization $\bar{p}_i$

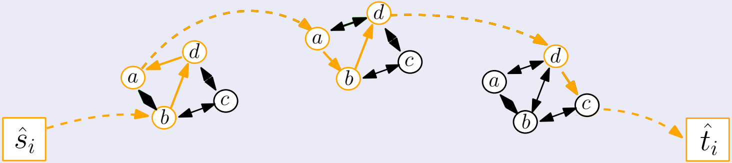


Projected flow f_i

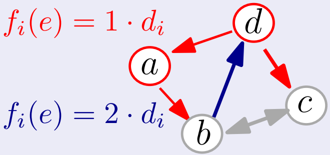


Approximating PCFP using Randomized Rounding: Random Walk

Other potential realization $\bar{p}_i$



Projected flow f_i



Notation

Let $E_i(e)$ denote all *copies* of edge $e \in E$ within $pn(N, r_i)$.

Important

$f_i(e) \leq |E_i(e)| \cdot d_i$.

Analysis of Randomized Rounding

Approximating PCFP using Randomized Rounding: Analysis of Edge Capacities

Notation

- Let $\Delta_{\text{max}} = \max_{i \in I} E_i(e)$ and $d_{\text{max}} = \max_{j \in I} d_i$

Approach: Fix single substrate edge $e \in E$

- Interpret $f_i(e)$ as random variable
- Define $X_i \in [0,1]$: $X_i \triangleq f_i(e)/(\Delta_{\text{max}} \cdot d_{\text{max}})$.

Approximating PCFP using Randomized Rounding: Analysis of Edge Capacities

Notation

- Let $\Delta_{\max} = \max_{i \in I} E_i(e)$ and $d_{\max} = \max_{j \in I} d_i$

Approach: Fix single substrate edge $e \in E$

- Interpret $f_i(e)$ as random variable
- Define $X_i \in [0, 1]$: $X_i \triangleq f_i(e)/(\Delta_{\max} \cdot d_{\max})$.
- Observe $\mathbf{E}[X_i] = \sum_{e' \in E_i(e)} \bar{f}_i(e')/(\Delta_{\max} \cdot d_{\max})$.

Approximating PCFP using Randomized Rounding: Analysis of Edge Capacities

Notation

- Let $\Delta_{\max} = \max_{i \in I} E_i(e)$ and $d_{\max} = \max_{j \in I} d_i$

Approach: Fix single substrate edge $e \in E$

- Interpret $f_i(e)$ as random variable
- Define $X_i \in [0,1]$: $X_i \triangleq f_i(e)/(\Delta_{\max} \cdot d_{\max})$.
- Observe $\mathbf{E}[X_i] = \sum_{e' \in E_i(e)} \bar{f}_i(e')/(\Delta_{\max} \cdot d_{\max})$.
- Note that $\mathbf{E}[X_i] \leq \mu_i$ holds for

$$\mu_i \triangleq \frac{\bar{c}(e)}{\Delta_{\max} \cdot d_{\max}} \cdot \frac{\sum_{e' \in E_i(e)} \bar{f}_i(e')}{\sum_{j \in I} \sum_{e' \in E_j(e)} \bar{f}_j(e')} \,,$$

as $\sum_{j \in I} \sum_{e' \in E_j(e)} \bar{f}_j(e') \leq \bar{c}(e)$ holds.

Approximating PCFP using Randomized Rounding: Analysis of Edge Capacities

Notation

- Let $\Delta_{\max} = \max_{i \in I} E_i(e)$ and $d_{\max} = \max_{j \in I} d_i$

Approach: Fix single substrate edge $e \in E$

- Interpret $f_i(e)$ as random variable
- Define $X_i \in [0,1]$: $X_i \triangleq f_i(e)/(\Delta_{\max} \cdot d_{\max})$.
- Observe $\mathbf{E}[X_i] = \sum_{e' \in E_i(e)} \bar{f}_i(e')/(\Delta_{\max} \cdot d_{\max})$.
- Note that $\mathbf{E}[X_i] \leq \mu_i$ holds for

$$\mu_i \triangleq \frac{\bar{c}(e)}{\Delta_{\max} \cdot d_{\max}} \cdot \frac{\sum_{e' \in E_i(e)} \bar{f}_i(e')}{\sum_{j \in I} \sum_{e' \in E_j(e)} \bar{f}_j(e')} ,$$

  as $\sum_{j \in I} \sum_{e' \in E_j(e)} \bar{f}_j(e') \leq \bar{c}(e)$ holds.
- Hence, $\mu \triangleq \sum_{i \in I} \mu_i = \bar{c}(e)/(\Delta_{\max} \cdot d_{\max})$.

Approximating PCFP using Randomized Rounding: Analysis of Edge Capacities

Approach: Fix single substrate edge $e \in E$

- Interpret $f_i(e)$ as random variable
- Define $X_i \in [0,1]$: $X_i \triangleq f_i(e)/(\Delta_{\max} \cdot d_{\max})$.
- Observe $\mathbf{E}[X_i] \leq \mu_i = \mu_i \triangleq \frac{\bar{c}(e)}{\Delta_{\max} \cdot d_{\max}} \cdot \frac{\sum_{e' \in E_i(e)} \bar{f}_i(e')}{\sum_{j \in I} \sum_{e' \in E_j(e)} \bar{f}_j(e')}$.
- Let $X = \sum_{i \in I} X_i$ with $\mathbf{E}[X] \leq \mu = \sum_{j \in I} \mu_i = \bar{c}(e)/(\Delta_{\max} \cdot d_{\max})$.
- The capacity along edge $e \in E$ is violated, if

$$X \geq (1+\varepsilon) \cdot \mu = \frac{c(e)}{\Delta_{\max} \cdot d_{\max})}$$

Excursion: A Chernoff-Bound

Chernoff

Let $\{X_i\}_i$ denote a sequence of independent random variables attaining values in $[0, 1]$. Assume that $\mathbf{E}[X_i] \leq \mu_i$. Let $X \triangleq \sum_i X_i$ and $\mu \triangleq \sum_i \mu_i$. Then, for $\varepsilon > 0$,

$$\mathbf{Pr}[X \geq (1+\varepsilon) \cdot \mu] \leq e^{-\beta(\varepsilon) \cdot \mu}.$$

Excursion: A Chernoff-Bound

Chernoff

Let $\{X_i\}_i$ denote a sequence of independent random variables attaining values in $[0,1]$. Assume that $\mathbf{E}[X_i] \leq \mu_i$. Let $X \triangleq \sum_i X_i$ and $\mu \triangleq \sum_i \mu_i$. Then, for $\varepsilon > 0$,

$$\mathbf{Pr}[X \geq (1+\varepsilon)\cdot\mu] \leq e^{-\beta(\varepsilon)\cdot\mu}.$$

Definition of β

The function $\beta : (-1,\infty) \to \mathbb{R}$ is defined by $\beta(\varepsilon) \triangleq (1+\varepsilon)\ln(1+\varepsilon) - \varepsilon$.

Observation

For $0 < \varepsilon < 1$ we have $\beta(\varepsilon) \geq \frac{2\varepsilon^2}{4.2+\varepsilon}$ and hence $\beta(\varepsilon) = \Theta(\varepsilon^2)$.

Approximating PCFP using Randomized Rounding: Analysis of Edge Capacities

Approach: Fix single substrate edge $e \in E$

- Define $X_i \in [0,1]$: $X_i \triangleq f_i(e)/(\Delta_{\max} \cdot d_{\max})$, with $\mathbf{E}[X_i] \leq \mu_i$.
- Let $X = \sum_{i \in I} X_i$ with $\mathbf{E}[X] \leq \mu = \bar{c}(e)/(\Delta_{\max} \cdot d_{\max})$.
- The capacity along edge $e \in E$ is violated, if $X \geq (1+\varepsilon) \cdot \mu$

Application of Chernoff-Bound

$$\mathbf{Pr}\left[\sum_{i \in I} X_i \geq (1+\varepsilon) \cdot \mu\right] \leq e^{-\beta(\varepsilon) \cdot \mu} = e^{-\beta(\varepsilon) \cdot \bar{c}(e)/(\Delta_{\max} \cdot d_{\max})}$$

Under small demands, i.e. assuming $\frac{\bar{c}(e)}{\Delta_{\max} d_{\max}} \geq \frac{4.2+\varepsilon}{\varepsilon^2} \cdot \ln |E|$

As $\beta(\varepsilon) \geq \dfrac{2\varepsilon^2}{4.2+\varepsilon}$ holds, $\mathbf{Pr}\left[\sum_{i \in I} X_i \geq (1+\varepsilon) \cdot \mu\right] \leq 1/|E|^2$ follows.

Main Results

Approximating PCFP using Randomized Rounding: Main Results

Main Theorem

Assume that $\frac{c_{\min}}{\Delta_{\max} \cdot d_{\max}} \geq \frac{4.2+\varepsilon}{\varepsilon^2} \cdot (1+\varepsilon) \cdot \ln |E|$ for $\varepsilon \in (0,1)$. The rounding scheme – under scaling capacities by $1/(1+\varepsilon)$ – yields

$$\mathbf{Pr}\left[\text{original edge capacity is violated}\right] \leq \frac{1}{|E|}$$

$$\mathbf{Pr}\left[B(\text{alg}) < \frac{1-\varepsilon}{1+\varepsilon} \cdot B(\text{opt}^*)\right] \leq e^{-\beta(-\varepsilon) \cdot B(\text{opt}^*)/((1+\varepsilon) \cdot b_{\max} \cdot d_{\max})}.$$

Approximating PCFP using Randomized Rounding: Main Results

Main Theorem

Assume that $\frac{c_{\min}}{\Delta_{\max} \cdot d_{\max}} \geq \frac{4.2+\varepsilon}{\varepsilon^2} \cdot (1+\varepsilon) \cdot \ln |E|$ for $\varepsilon \in (0,1)$. The rounding scheme – under scaling capacities by $1/(1+\varepsilon)$ – yields

$$\mathbf{Pr}\left[\text{original edge capacity is violated}\right] \leq \frac{1}{|E|}$$

$$\mathbf{Pr}\left[B(\text{alg}) < \frac{1-\varepsilon}{1+\varepsilon} \cdot B(\text{opt}^*)\right] \leq e^{-\beta(-\varepsilon) \cdot B(\text{opt}^*)/((1+\varepsilon) \cdot b_{\max} \cdot d_{\max})}.$$

Monte Carlo

By repeating the rounding finitely many times, a high quality solution can be found with high probability.

Approximating PCFP using Randomized Rounding: Main Results

Main Theorem

Assume that $\frac{c_{\min}}{\Delta_{\max} \cdot d_{\max}} \geq \frac{4.2+\varepsilon}{\varepsilon^2} \cdot (1+\varepsilon) \cdot \ln |E|$ for $\varepsilon \in (0,1)$. The rounding scheme – under scaling capacities by $1/(1+\varepsilon)$ – yields

$$\mathbf{Pr}\left[\text{original edge capacity is violated}\right] \leq \frac{1}{|E|}$$

$$\mathbf{Pr}\left[B(\text{alg}) < \frac{1-\varepsilon}{1+\varepsilon} \cdot B(\text{opt}^*)\right] \leq e^{-\beta(-\varepsilon) \cdot B(\text{opt}^*)/((1+\varepsilon) \cdot b_{\max} \cdot d_{\max})}.$$

Corollary

If additionally, $b_i = 1$ holds for all $i \in I$, then with probability $1 - O(1/Poly(|E|)$, the algorithm returns a solution with at least $1 - O(\varepsilon)$ times the optimal benefit with high probability.

Conclusion

Conclusion

Summary

- PCFP considers the placement of functions and the routing between these for multiple requests to maximize the profit.
- Apply randomized rounding (cf. Raghavan and Tompson [1987]) and obtain approximation under certain assumptions:
    - Small demands $\frac{\bar{c}(e)}{\Delta_{\max} d_{\max}} \geq \frac{4.2+\varepsilon}{\varepsilon^2} \cdot \ln |E|$ to not violate capacites
    - Small demands and unit benefits yield $1 - \mathcal{O}(\varepsilon)$ approximation.

Conclusion

Summary

- PCFP considers the placement of functions and the routing between these for multiple requests to maximize the profit.
- Apply randomized rounding (cf. Raghavan and Tompson [1987]) and obtain approximation under certain assumptions:
  - Small demands $\frac{\bar{c}(e)}{\Delta_{\mathbf{max}} d_{\mathbf{max}}} \geq \frac{4.2+\varepsilon}{\varepsilon^2} \cdot \ln |E|$ to not violate capacites
  - Small demands and unit benefits yield $1 - \mathcal{O}(\varepsilon)$ approximation.

Contribution: "Rediscovery" of randomized rounding

- Consider several (virtual) embedding options for requests (DAGs).
- Show applicability of randomized rounding to exert admission control.
- Perform concise mathematical analysis.
- *First non-trivial approximation for embeddings multiple graphs.*

Related Work

Randomized Rounding

- VLSI design to minimize width [Raghavan and Tompson, 1987]
- Analysis of the approximation for PCFP without requiring assumptions and generalization to 'cyclic' requests [Rost and Schmid, 2016]

Modeling and Embedding Requests

- Product Network and Online Approximation [Even et al., 2016]
- Heuristics for choosing virtual embedding options and embedding services [Sahhaf et al., 2015]

References I

Guy Even, Moti Medina, and Boaz Patt-Shamir. Competitive path computation and function placement in sdns. *CoRR*, abs/1602.06169, 2016. URL http://arxiv.org/abs/1602.06169.

D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1):14–76, 2015. ISSN 0018-9219. doi: 10.1109/JPROC.2014.2371999.

Rajeev Motwani, Joseph Seffi Naor, and Prabhakar Raghavan. Randomized approximation algorithms in combinatorial optimization. In *Approximation algorithms for NP-hard problems*, pages 447–481. PWS Publishing Co., 1996.

Prabhakar Raghavan and Clark D Tompson. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7(4):365–374, 1987.

References II

Matthias Rost and Stefan Schmid. Service chain and virtual network embeddings: Approximations using randomized rounding. *CoRR*, abs/1604.02180, 2016.

Sahel Sahhaf, Wouter Tavernier, Matthias Rost, Stefan Schmid, Didier Colle, Mario Pickavet, and Piet Demeester. Network service chaining with optimized network function embedding supporting service decompositions. In *Journal Computer Networks (COMNET), Elsevier*, 2015.