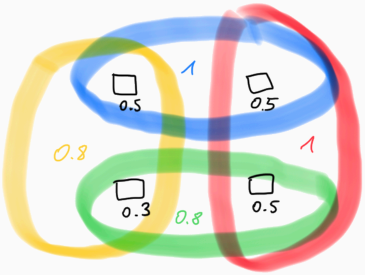
Decomposition of Probability Marginals for Security Games in Abstract Networks

Jannik Matuschke

KU Leuven

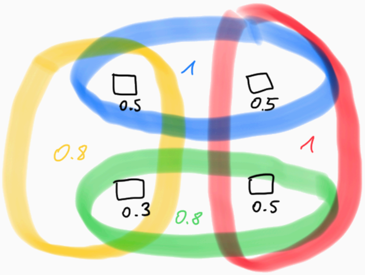
Input:

- ground set E
- set system $\mathcal{P} \subseteq 2^E$
- requirements $\pi \in [0,1]^{\mathcal{P}}$
- marginals $\rho \in [0,1]^E$



Input:

- ground set E
- set system $\mathcal{P} \subseteq 2^E$
- requirements $\pi \in [0,1]^{\mathcal{P}}$
- marginals $\rho \in [0,1]^E$



Goal: Find distribution for random set $S \subseteq E$ such that

$$\Pr\left[e \in S\right] = \rho_e \qquad \forall\, e \in E,$$
$$\Pr\left[P \cap S \neq \emptyset\right] \geq \pi_P \qquad \forall\, P \in \mathcal{P}.$$

Feasible Decompositions

We call a random set S with

$$\begin{aligned} \Pr\Big[e \in S\Big] \;&=\; \rho_e \qquad && \forall\, e \in E, \\ \Pr\Big[P \cap S \neq \emptyset\Big] \;&\geq\; \pi_P \qquad && \forall\, P \in \mathcal{P}, \end{aligned}$$

a **feasible decomposition** of ρ w.r.t. $(\mathcal{P}, \pi)$.

Feasible Decompositions

We call a random set S with

$$\begin{aligned} \Pr\left[e \in S\right] &\leq \rho_e && \forall\, e \in E, \\ \Pr\left[P \cap S \neq \emptyset\right] &\geq \pi_P && \forall\, P \in \mathcal{P}, \end{aligned}$$

a **feasible decomposition** of ρ w.r.t. $(\mathcal{P}, \pi)$.

Feasible Decompositions

We call a random set S with

$$\begin{aligned} \Pr\left[e \in S\right] &\leq \rho_e & \forall\, e \in E, \\ \Pr\left[P \cap S \neq \emptyset\right] &\geq \pi_P & \forall\, P \in \mathcal{P}, \end{aligned}$$

a **feasible decomposition** of ρ w.r.t. $(\mathcal{P}, \pi)$.

We say ρ is **feasible** if it has a feasible decomposition.

Feasible Decompositions

We call a random set S with

$$\Pr\left[e \in S\right] \leq \rho_e \qquad \forall\, e \in E,$$
$$\Pr\left[P \cap S \neq \emptyset\right] \geq \pi_P \qquad \forall\, P \in \mathcal{P},$$

a **feasible decomposition** of ρ w.r.t. $(\mathcal{P}, \pi)$.

We say ρ is **feasible** if it has a feasible decomposition.

Can we describe the set of feasible ρ?

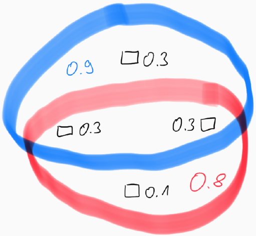
Feasible Decompositions

We call a random set S with

$$\Pr\left[e \in S\right] \leq \rho_e \qquad \forall\, e \in E,$$
$$\Pr\left[P \cap S \neq \emptyset\right] \geq \pi_P \qquad \forall\, P \in \mathcal{P},$$

a **feasible decomposition** of ρ w.r.t. $(\mathcal{P}, \pi)$.

We say ρ is **feasible** if it has a feasible decomposition.



feasible?

Can we describe the set of feasible ρ?

A Necessary Condition

$$\sum_{e \in P} \rho_e \geq \pi_P \quad \forall\, P \in \mathcal{P} \quad (\star)$$

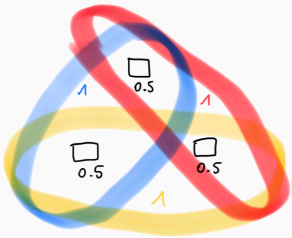
Observation: If ρ is feasible, then it must fulfil $(\star)$.

A Necessary Condition

$$\sum_{e \in P} \rho_e \geq \pi_P \quad \forall P \in \mathcal{P} \quad (\star)$$

Observation: If ρ is feasible, then it must fulfil $(\star)$.

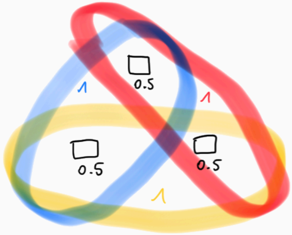
But $(\star)$ is not always sufficient:



$$\sum_{e \in P} \rho_e \geq \pi_P \quad \forall\, P \in \mathcal{P} \quad (\star)$$

Observation: If ρ is feasible, then it must fulfil $(\star)$.

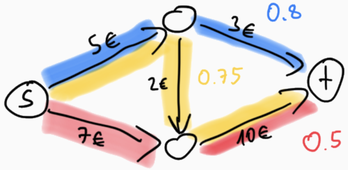
But $(\star)$ is not always sufficient:



A system $(\mathcal{P}, \pi)$ is **$(\star)$-sufficient** if for all $\rho \in [0,1]^E$:

ρ is feasible. $\quad\Leftrightarrow\quad$ ρ fulfils $(\star)$.

Given: set system $(E, \mathcal{P})$,
costs $c \in \mathbb{R}_+^E$,
requirements $\pi \in [0,1]^{\mathcal{P}}$

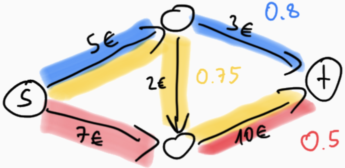


Motivation: Security Games

Given: set system $(E, \mathcal{P})$,
costs $c \in \mathbb{R}_+^E$,
requirements $\pi \in [0,1]^{\mathcal{P}}$

Defender

selects random set $S \subseteq E$ at cost $\sum_{e \in S} c_e$
wants to deter any attack at minimum cost



Motivation: Security Games

Given: set system $(E, \mathcal{P})$,

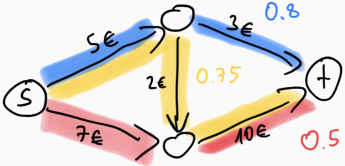
costs $c \in \mathbb{R}_+^E$,

requirements $\pi \in [0,1]^{\mathcal{P}}$

Defender

selects random set $S \subseteq E$ at cost $\sum_{e \in S} c_e$

wants to deter any attack at minimum cost



Attacker

selects $P \in \mathcal{P}$ or remains inactive

π_P: risk threshold for deterring attacker from P

Motivation: Security Games

Defender's problem:

$$
\begin{aligned}
\min \sum_{S \subseteq E} \sum_{e \in S} c_e\, x_S & \\
\text{s.t.} \sum_{S : P \cap S \neq \emptyset} x_S &\geq \pi_P \quad \forall\, P \in \mathcal{P} \\
\sum_{S \subseteq E} x_S &= 1 \\
x &\geq 0
\end{aligned}
$$

Motivation: Security Games

Defender's problem:

$$\begin{aligned}
\min \quad & \sum_{e \in E} c_e \sum_{S : e \in S} x_S \\
\text{s.t.} \sum_{S : P \cap S \neq \emptyset} x_S &\geq \pi_P \quad \forall P \in \mathcal{P} \\
\sum_{S \subseteq E} x_S &= 1 \\
x &\geq 0
\end{aligned}$$

Motivation: Security Games

Defender's problem:

$$\begin{aligned} \min \quad & \sum_{e \in E} c_e \sum_{S: e \in S} x_S \\ \text{s.t.} \quad & \sum_{S: P \cap S \neq \emptyset} x_S \geq \pi_P \quad \forall\, P \in \mathcal{P} \\ & \sum_{S \subseteq E} x_S = 1 \\ & x \geq 0 \end{aligned}$$

If $(\star)$ is sufficient:

$$\begin{aligned} \min \quad & \sum_{e \in E} c_e\, \rho_e \\ \text{s.t.} \quad & \sum_{e \in P} \rho_e \geq \pi_P \quad \forall\, P \in \mathcal{P} \\ & \rho \geq 0 \end{aligned}$$

Exponentially smaller dimension :)

Motivation: Security Games

Defender's problem:

$$\begin{aligned} \min \quad & \sum_{e \in E} c_e \sum_{S : e \in S} x_S \\ \text{s.t.} \sum_{S : P \cap S \neq \emptyset} x_S &\geq \pi_P \quad \forall\, P \in \mathcal{P} \\ \sum_{S \subseteq E} x_S &= 1 \\ x &\geq 0 \end{aligned}$$

If $(\star)$ is sufficient:

$$\begin{aligned} \min \quad & \sum_{e \in E} c_e \, \rho_e \\ \text{s.t.} \quad \sum_{e \in P} \rho_e &\geq \pi_P \quad \forall\, P \in \mathcal{P} \\ \rho &\geq 0 \end{aligned}$$

Exponentially smaller dimension :)

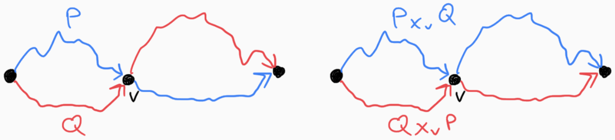
further applications: fairness/balance constraints in social choice, randomized algorithms

similar: Border's Theorem for auctions

$\mathcal{P} = \{s\text{-}t\text{-paths in a DAG}\}$, two settings for π:

(A) **Affine requirements:** $\pi_P = 1 - \sum_{e \in E} \mu_e$ for some $\mu \in [0,1]^E$

(C) **Conservation law:** $\pi_P + \pi_Q = \pi_{P \times_v Q} + \pi_{Q \times_v P}$ for $P, Q \in \mathcal{P}, v \in P \cap Q$

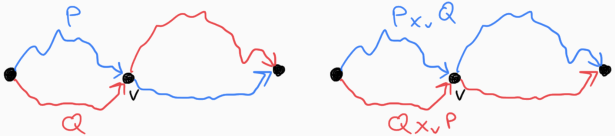


$\mathcal{P} = \{s\text{-}t\text{-paths in a DAG}\}$, two settings for π:

(A) **Affine requirements:** $\pi_P = 1 - \sum_{e \in E} \mu_e$ for some $\mu \in [0,1]^E$

(C) **Conservation law:** $\pi_P + \pi_Q = \pi_{P \times_v Q} + \pi_{Q \times_v P}$ for $P, Q \in \mathcal{P}, v \in P \cap Q$

Note: (A) $\Rightarrow$ (C).

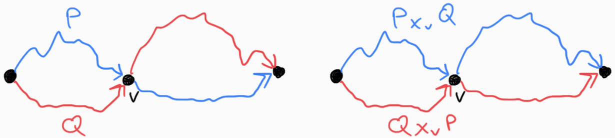


$\mathcal{P} = \{s\text{-}t\text{-paths in a DAG}\}$, two settings for π:

(A) **Affine requirements:** $\pi_P = 1 - \sum_{e \in E} \mu_e$ for some $\mu \in [0,1]^E$

(C) **Conservation law:** $\pi_P + \pi_Q = \pi_{P \times_v Q} + \pi_{Q \times_v P}$ for $P, Q \in \mathcal{P}, v \in P \cap Q$

Note: (A) $\Rightarrow$ (C).



Their results:

- For (C): $(\mathcal{P}, \pi)$ is $(\star)$-sufficient.
- For (A): Decomposition can be computed efficiently.
- Consequence: Computation of Nash equilibria in interdiction game on DAG.

New Results

	DAGs (Dahan et al.)
Affine	efficient algorithm
Conservation	$(\star)$-sufficient (exp.-time algorithm)

New Results

	DAGs (Dahan et al.)	Abstract Networks (incl. digraphs w. cycles)
Affine	efficient algorithm	efficient algorithm (explicit description) ⊕
Conservation	($\star$)-sufficient (exp.-time algorithm)	

⊕ combinatorial shortest-path algorithm for abstract networks

New Results

	DAGs (Dahan et al.)	Abstract Networks (incl. digraphs w. cycles)	Max-Flow/Min-Cut
Affine	efficient algorithm	efficient algorithm (explicit description) ⊕	characterize (⋆)-sufficiency (oracle-poly)
Conservation	(⋆)-sufficient (exp.-time algorithm)		

⊕ combinatorial shortest-path algorithm for abstract networks

New Results

	DAGs (Dahan et al.)	Abstract Networks (incl. digraphs w. cycles)	Max-Flow/Min-Cut
Affine	efficient algorithm	efficient algorithm (explicit description) ⊕	characterize (⋆)-sufficiency (oracle-poly)
	⇕		
Conservation	(⋆)-sufficient (exp.-time algorithm)		

⊕ combinatorial shortest-path algorithm for abstract networks

New Results

	DAGs (Dahan et al.)	Abstract Networks (incl. digraphs w. cycles)	Max-Flow/Min-Cut
Affine	efficient algorithm	efficient algorithm (explicit description) $\oplus$	characterize $(\star)$-sufficiency (oracle-poly)
	$\Updownarrow$		
Conservation	$(\star)$-sufficient (exp.-time algorithm)	$(\star)$-sufficient (oracle-poly)	

$\oplus$ combinatorial shortest-path algorithm for abstract networks

New Results

	DAGs (Dahan et al.)	Abstract Networks (incl. digraphs w. cycles)	Max-Flow/Min-Cut
Affine	efficient algorithm $\Updownarrow$	efficient algorithm (explicit description) $\bigoplus$	characterize $(\star)$-sufficiency (oracle-poly)
Conservation	$(\star)$-sufficient (exp.-time algorithm)	$(\star)$-sufficient (oracle-poly)	

$\bigoplus$ combinatorial shortest-path algorithm for abstract networks

Also: NP-hard to decide feasibility of given ρ in general systems

Feasible Decompositions for Max-Flow/Min-Cut Systems

The Max-Flow/Min-Cut Property

$\mathcal{P}$ has the max-flow/min-cut property if

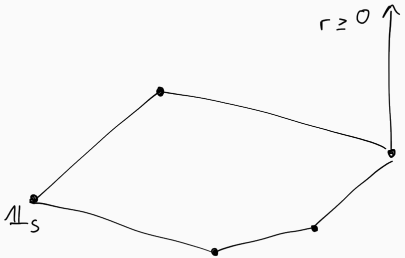
$$Q_{\mathcal{P}} := \left\{ \begin{array}{rlr} \sum_{e \in P} y_e & \geq 1 & \forall\, P \in \mathcal{P} \\ y_e & \geq 0 & \forall\, e \in E \end{array} \right\}$$ is integral.

$\mathcal{P}$ has the max-flow/min-cut property if

$$Q_{\mathcal{P}} := \left\{ \begin{array}{rl} \sum_{e \in P} y_e \geq 1 & \forall\, P \in \mathcal{P} \\ y_e \geq 0 & \forall\, e \in E \end{array} \right\}$$ is integral.

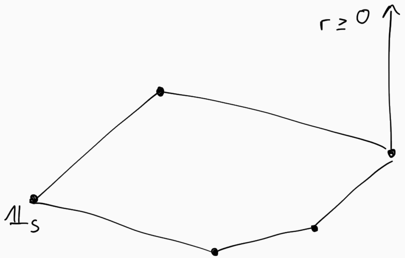
Note: $Q_{\mathcal{P}}$ is integral iff every vertex is of the form $\mathbb{1}_S$ for some $S \in \mathcal{S}$.

$$\mathcal{S} := \{S \subseteq E \;:\; P \cap S \neq \emptyset \quad \forall\, P \in \mathcal{P}\}$$



$\mathcal{P}$ has the max-flow/min-cut property if

$$Q_{\mathcal{P}} := \left\{ \begin{array}{rl} \sum_{e \in P} y_e \geq 1 & \forall\, P \in \mathcal{P} \\ y_e \geq 0 & \forall\, e \in E \end{array} \right\}$$ is integral.



Note: $Q_{\mathcal{P}}$ is integral iff every vertex is of the form $\mathbb{1}_S$ for some $S \in \mathcal{S}$.

$$\mathcal{S} := \{S \subseteq E \;:\; P \cap S \neq \emptyset \quad \forall\, P \in \mathcal{P}\}$$

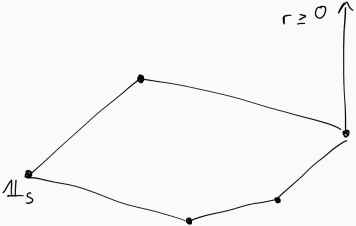
Theorem. The following two statements are equivalent:

1. $\mathcal{P}$ has the MF/MC property.
2. $(\mathcal{P}, \pi)$ is $(\star)$-sufficient for all π fulfiling (A).

Proof: MF/MC implies $(\star)$-sufficiency for affine π

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$



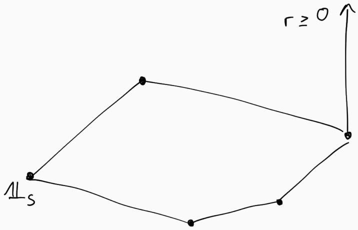
$Q_{\mathcal{P}} = \{ y \in \mathbb{R}^E_+ : \sum_{e \in P} y_e \geq 1 \;\; \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \;\; \forall P \in \mathcal{P} \}$

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$y := \rho + \mu \in Q_{\mathcal{P}}$



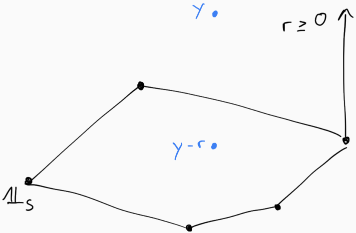
$$Q_{\mathcal{P}} = \{ y \in \mathbb{R}^E_+ : \sum_{e \in P} y_e \geq 1 \quad \forall P \in \mathcal{P} \}$$

$$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \quad \forall P \in \mathcal{P} \}$$

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \;\; \forall P \in \mathcal{P} \}$$

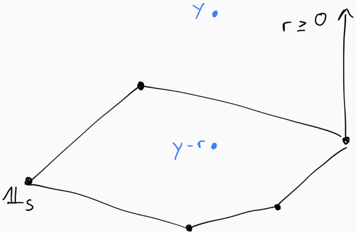
$$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \;\; \forall P \in \mathcal{P} \}$$

Proof: MF/MC implies $(\star)$-sufficiency for affine π

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \ \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \ \forall P \in \mathcal{P} \}$

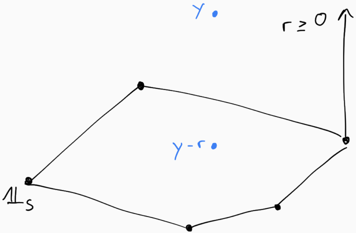
	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$

Proof: MF/MC implies $(\star)$-sufficiency for affine π

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \;\; \forall P \in \mathcal{P} \}$

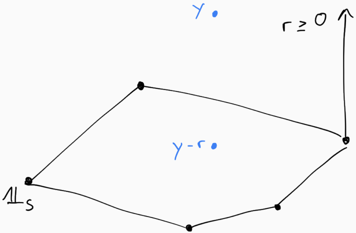
$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \;\; \forall P \in \mathcal{P} \}$

	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \ \ \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \ \ \forall P \in \mathcal{P} \}$

	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$		$\leq \rho_e$

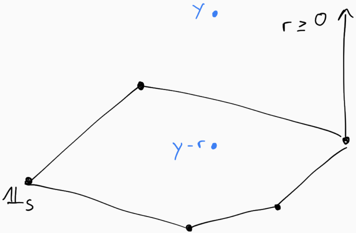
Proof: MF/MC implies $(\star)$-sufficiency for affine π

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$

	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$?	$\leq \rho_e$



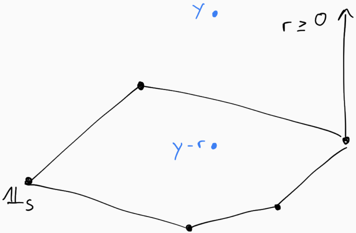
$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \;\; \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \;\; \forall P \in \mathcal{P} \}$

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \ \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \ \forall P \in \mathcal{P} \}$

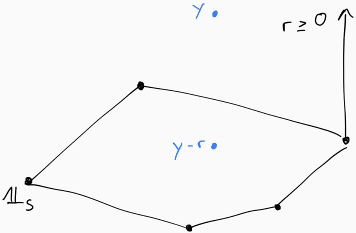
	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$?	$\leq \rho_e$

$$\Pr\left[R \cap P = \emptyset\right] \leq \Pr\left[\exists e \in P \cap R_1 \setminus R_2\right]$$

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \ \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \ \forall P \in \mathcal{P} \}$

	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$?	$\leq \rho_e$

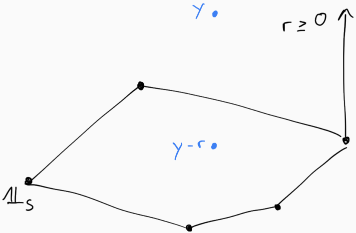
$$\Pr[R \cap P = \emptyset] \leq \sum_{e \in P} \Pr[e \in R_1] \cdot \Pr[e \notin R_2]$$

Proof: MF/MC implies $(\star)$-sufficiency for affine π

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \;\; \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \;\; \forall P \in \mathcal{P} \}$

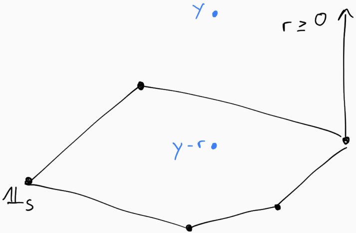
	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$?	$\leq \rho_e$

$$\Pr\left[R \cap P = \emptyset\right] \;\leq\; \sum_{e \in P} (y_e - r_e) \cdot \left(1 - \min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}\right)$$

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \;\; \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \;\; \forall P \in \mathcal{P} \}$

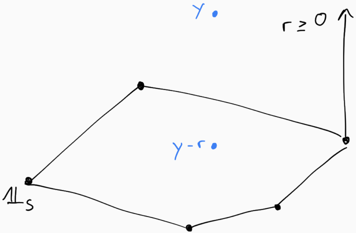
	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$?	$\leq \rho_e$

$$\Pr\left[R \cap P = \emptyset\right] \leq \sum_{e \in P} (y_e - r_e) \cdot \max\left\{1 - \frac{\rho_e}{y_e - r_e}, 0\right\}$$

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \;\; \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \;\; \forall P \in \mathcal{P} \}$

	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$?	$\leq \rho_e$

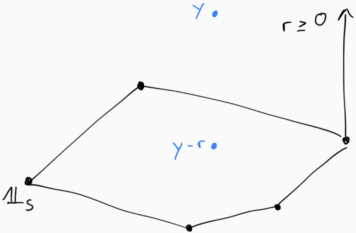
$$\Pr\left[R \cap P = \emptyset\right] \leq \sum_{e \in P} \max\left\{y_e - r_e - \rho_e,\; 0\right\}$$

Proof: MF/MC implies $(\star)$-sufficiency for affine π

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}^E_+ : \sum_{e \in P} y_e \geq 1 \quad \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \quad \forall P \in \mathcal{P} \}$

	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$?	$\leq \rho_e$

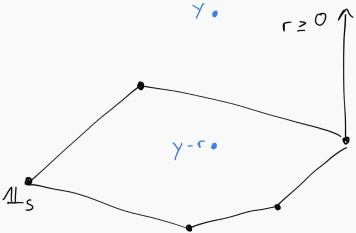
$$\Pr[R \cap P = \emptyset] \leq \sum_{e \in P} \max\{y_e - \rho_e,\ 0\}$$

Proof: MF/MC implies $(\star)$-sufficiency for affine π

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}^E_+ : \sum_{e \in P} y_e \geq 1 \ \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \ \forall P \in \mathcal{P} \}$

	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$?	$\leq \rho_e$

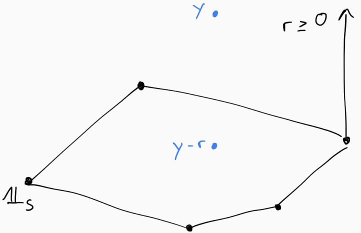
$$\Pr\left[R \cap P = \emptyset\right] \leq \sum_{e \in P} \mu_e$$

Proof: MF/MC implies $(\star)$-sufficiency for affine π

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \;\; \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \;\; \forall P \in \mathcal{P} \}$

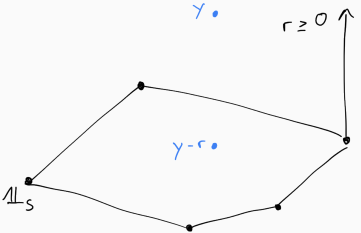
	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$	$\leq \rho_e$

$$\Pr\left[R \cap P = \emptyset\right] \leq \sum_{e \in P} \mu_e \quad \Rightarrow \quad R \text{ is a feasible decomposition of } \rho.$$

(A) $\pi_P = 1 - \sum_{e \in P} \mu_e$ for some $\mu \in [0,1]^E$

$(\star)$ $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$

$$y := \rho + \mu \in Q_{\mathcal{P}} \quad \Rightarrow \quad y = \underbrace{\sum_{S \in \mathcal{S}} \lambda_S \mathbb{1}_S}_{\text{convex combination}} + \underbrace{r}_{\geq 0}$$



$Q_{\mathcal{P}} = \{ y \in \mathbb{R}_+^E : \sum_{e \in P} y_e \geq 1 \;\; \forall P \in \mathcal{P} \}$

$\mathcal{S} = \{ S \subseteq E : P \cap S \neq \emptyset \;\; \forall P \in \mathcal{P} \}$

	Definition	$\Pr[S \cap P \neq \emptyset]$	$\Pr[e \in S]$
R_1	$\Pr[R_1 = S] = \lambda_S$	1	$y_e - r_e$
R_2	indep. from R_1		$\min\left\{\frac{\rho_e}{y_e - r_e}, 1\right\}$
R	$= R_1 \cap R_2$	$\geq \pi_P$	$\leq \rho_e$

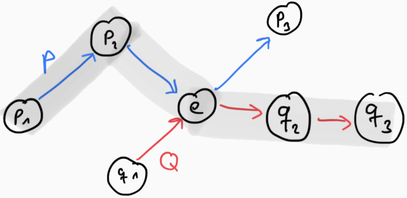
Note: We can construct R if we can solve SEPARATION for $Q_{\mathcal{P}}$.

Abstract Networks

An **abstract network** is a set system $(E, \mathcal{P})$ such that

- for every $P \in \mathcal{P}$, there is an order $\preceq_P$ on P,
- for every $P, Q \in \mathcal{P}$ and $e \in P \cap Q$, there is $P \times_e Q \in \mathcal{P}$ with

$$P \times_e Q \subseteq \{p \in P \ : \ p \preceq_P e\} \cup \{q \in Q \ : \ e \preceq q\}.$$

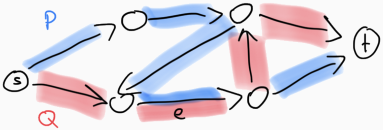


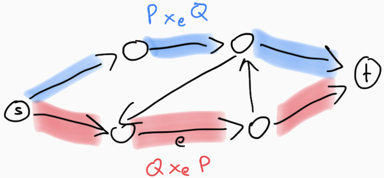
$$P \times_e Q = q_3 \to p_2 \to q_2$$

Example: s-t-Paths in a Digraph

Digraph $D = (V, A)$ $\quad\rightarrow\quad$ $E = A$ (or $E = V$, or $E = V \cup A$)

$\mathcal{P} = \{s\text{-}t\text{-paths in } D\}$

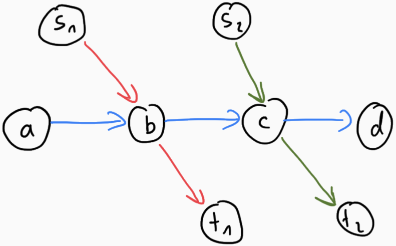




Another Example

$$E = \{a, b, c, d, s_1, s_2, t_1, t_2\}$$

$$\mathcal{P} = \{\{a, b, c, d\},\ \{s_1, b, t_1\},\ \{s_2, c, t_3\},$$
$$\{s_1, d\},\ \{s_2, d\},\ \{a, t_1\},\ \{a, t_2\}\}$$



Note: No path starting with s_1 and ending with t_2.

Abstract Networks and Max-Flow/Min-Cut

Abstract Max Flow

$$\begin{aligned} \max \sum_{P \in \mathcal{P}} x_P & \\ \text{s.t.} \sum_{P : e \in P} x_P &\leq u_e \quad \forall\, e \in E \\ x &\geq 0 \end{aligned}$$

Abstract Min Cut

$$\begin{aligned} \min \sum_{e \in E} u_e y_e & \\ \text{s.t.} \quad \sum_{e \in P} y_e &\geq 1 \quad \forall\, P \in \mathcal{P} \\ y &\geq 0 \end{aligned}$$

Hoffman (Math. Prog. 1974): Abstract Min Cut is TDI.

Abstract Networks and Max-Flow/Min-Cut

Abstract Max Flow

$$\begin{aligned} \max \sum_{P \in \mathcal{P}} x_P & \\ \text{s.t.} \sum_{P : e \in P} x_P &\le u_e \quad \forall\, e \in E \\ x &\ge 0 \end{aligned}$$

Abstract Min Cut

$$\begin{aligned} \min \sum_{e \in E} u_e y_e & \\ \text{s.t.} \quad \sum_{e \in P} y_e &\ge 1 \quad \forall\, P \in \mathcal{P} \\ y &\ge 0 \end{aligned}$$

Hoffman (Math. Prog. 1974): Abstract Min Cut is TDI.

McCormick (SODA 1996): combinatorial algorithm

Abstract Networks and Max-Flow/Min-Cut

Abstract Max Flow

$$\begin{aligned} \max \sum_{P \in \mathcal{P}} r_P x_P & \\ \text{s.t.} \quad \sum_{P: e \in P} x_P &\le u_e \quad \forall\, e \in E \\ x &\ge 0 \end{aligned}$$

Abstract Min Cut

$$\begin{aligned} \min \sum_{e \in E} u_e y_e & \\ \text{s.t.} \quad \sum_{e \in P} y_e &\ge r_P \quad \forall\, P \in \mathcal{P} \\ y &\ge 0 \end{aligned}$$

Hoffman (Math. Prog. 1974): Abstract Min Cut is TDI, even with weights fulfilling "weak conservation law": $r_{P \times_e Q} + r_{Q \times_e P} \ge r_P + r_Q$

McCormick (SODA 1996): combinatorial algorithm (unweighted version)

Abstract Networks and Max-Flow/Min-Cut

Abstract Max Flow

$$\begin{aligned} \max \sum_{P \in \mathcal{P}} r_P x_P & \\ \text{s.t.} \quad \sum_{P : e \in P} x_P &\leq u_e \quad \forall\, e \in E \\ x &\geq 0 \end{aligned}$$

Abstract Min Cut

$$\begin{aligned} \min \sum_{e \in E} u_e y_e & \\ \text{s.t.} \quad \sum_{e \in P} y_e &\geq r_P \quad \forall\, P \in \mathcal{P} \\ y &\geq 0 \end{aligned}$$

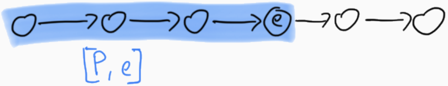
Hoffman (Math. Prog. 1974): Abstract Min Cut is TDI, even with weights fulfilling "weak conservation law": $r_{P \times_e Q} + r_{Q \times_e P} \geq r_P + r_Q$

McCormick (SODA 1996): combinatorial algorithm (unweighted version)

Martens & McCormick (IPCO 2008): combinatorial algorithm for weighted version

Feasible Decompositions in Abstract Networks

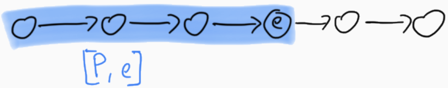
$$\alpha_e := \min_{P \in \mathcal{P}} \sum_{f \in [P,e]} \rho_f + \mu_f$$



Feasible Decompositions in Abstract Networks $(\pi_P = 1 - \sum_{e \in P} \mu_e)$

$$S_\tau := \left\{ e \in E \,:\, \alpha_e - \rho_e \leq \tau \leq \alpha_e \right\}$$

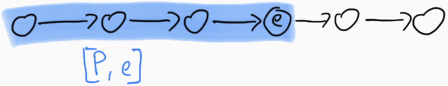
with $\alpha_e := \min_{P \in \mathcal{P}} \sum_{f \in [P,e]} \rho_f + \mu_f$ and $\tau \sim U[0,1]$



Theorem. S_τ is a feasible decomposition of ρ.

$$S_\tau := \left\{ e \in E \,:\, \alpha_e - \rho_e \leq \tau \leq \alpha_e \right\}$$

$$\text{with } \alpha_e := \min_{P \in \mathcal{P}} \sum_{f \in [P,e]} \rho_f + \mu_f \quad \text{and} \quad \tau \sim U[0,1]$$



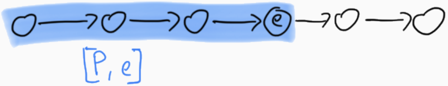
Theorem. S_τ is a feasible decomposition of ρ.

Proof sketch. Want to show: $\Pr\left[S_\tau \cap P \neq \emptyset\right] + \sum_{e \in P} \mu_e \geq 1$

Feasible Decompositions in Abstract Networks $\left(\pi_P = 1 - \sum_{e \in P} \mu_e\right)$

$$S_\tau := \left\{ e \in E \,:\, \alpha_e - \rho_e \leq \tau \leq \alpha_e \right\}$$

with $\alpha_e := \min_{P \in \mathcal{P}} \sum_{f \in [P,e]} \rho_f + \mu_f$ and $\tau \sim U[0,1]$



Theorem. S_τ is a feasible decomposition of ρ.

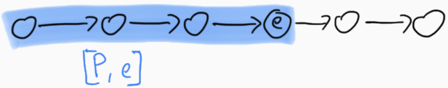
Proof sketch.

Want to show: $\Pr\left[S_\tau \cap P \neq \emptyset\right] + \sum_{e \in P} \mu_e \geq 1$

By induction: $\Pr\left[S_\tau \cap [P,e] \neq \emptyset \,\wedge\, \tau \leq \alpha_e\right] + \sum_{f \in [P,e]} \mu_f \geq \alpha_e$

$$S_\tau := \left\{ e \in E \,:\, \alpha_e - \rho_e \le \tau \le \alpha_e \right\}$$

with $\alpha_e := \min_{P \in \mathcal{P}} \sum_{f \in [P,e]} \rho_f + \mu_f$ and $\tau \sim U[0,1]$



Theorem. S_τ is a feasible decomposition of ρ.

Proof sketch.

Want to show: $\Pr\left[S_\tau \cap P \neq \emptyset\right] + \sum_{e \in P} \mu_e \ge 1$

By induction: $\Pr\left[S_\tau \cap [P,e] \neq \emptyset \,\wedge\, \tau \le \alpha_e\right] + \sum_{f \in [P,e]} \mu_f \ge \alpha_e$

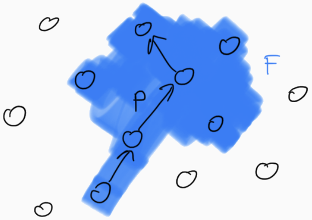
By $(\star)$: $\alpha_t \ge 1$ for last element t of P

Shortest Paths in Abstract Networks

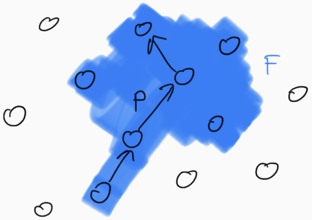
How Do We Access Abstract Networks?

Membership oracle for an abstract network:
Given $F \subseteq E$, either

- return $P \in \mathcal{P}$ (and $\preceq_P$) with $P \subseteq F$,
- or assert that no such P exists.



How Do We Access Abstract Networks?



Membership oracle for an abstract network:
Given $F \subseteq E$, either

- return $P \in \mathcal{P}$ (and $\preceq_P$) with $P \subseteq F$,
- or assert that no such P exists.

McCormick's Max Abstract Flow algorithm uses membership oracle.

Question (McCormick 1996): Can a stronger oracle (e.g., shortest paths) yield a strongly poly-time algorithm for Max Abstract Flow?

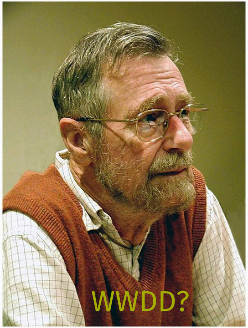
Shortest Paths in Abstract Networks

Given: abstract network $(E, \mathcal{P})$, costs $c \in \mathbb{R}_+^E$

Task: find $P \in \mathcal{P}$ minmizing $c(P) := \sum_{e \in P} c_e$

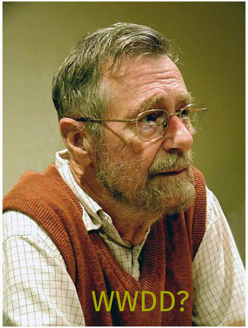
Given: abstract network $(E, \mathcal{P})$, costs $c \in \mathbb{R}_+^E$

Task: find $P \in \mathcal{P}$ minmizing $c(P) := \sum_{e \in P} c_e$



Given: abstract network $(E, \mathcal{P})$, costs $c \in \mathbb{R}_+^E$

Task: find $P \in \mathcal{P}$ minmizing $c(P) := \sum_{e \in P} c_e$

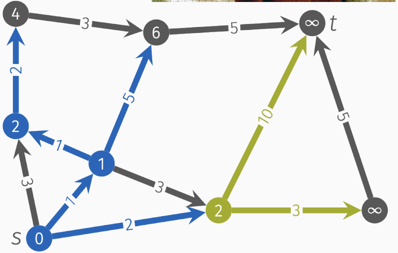


Dijkstra's Algorithm

- initialize:
  - label ϕ_v for $v \in V$
  - s-v-path Q_v with $\phi_v = c(Q_v)$
  - set of processed nodes T
- while $\min_{v \in V \setminus T} \phi_v < \phi_t$

  pick $v \in \operatorname{argmin}_{w \in V \setminus T} \phi_w$

  $\texttt{process}(v)$



Shortest Paths in Abstract Networks

Given: abstract network $(E, \mathcal{P})$, costs $c \in \mathbb{R}_+^E$

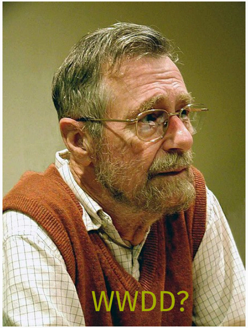
Task: find $P \in \mathcal{P}$ minmizing $c(P) := \sum_{e \in P} c_e$

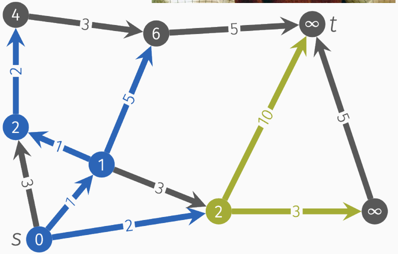
Dijkstra's Algorithm (adapted)

- initialize:
  - label ϕ_e for $e \in E$
  - $Q_e \in \mathcal{P}$ with $\phi_e = c([Q_e, e])$
  - set of processed elements T
- while $\min_{v \in V \setminus T} \phi_v < \phi_t$

  pick $v \in \operatorname{argmin}_{w \in V \setminus T} \phi_w$

  $\texttt{process}(v)$





Given: abstract network $(E, \mathcal{P})$, costs $c \in \mathbb{R}_+^E$

Task: find $P \in \mathcal{P}$ minmizing $c(P) := \sum_{e \in P} c_e$

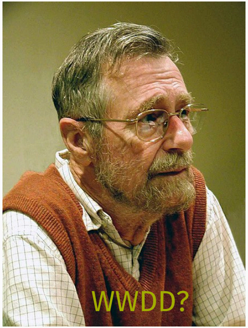
Dijkstra's Algorithm (adapted)

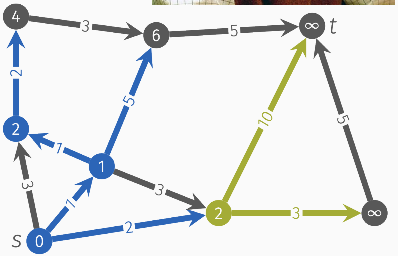
- initialize:
  - label ϕ_e for $e \in E$
  - $Q_e \in \mathcal{P}$ with $\phi_e = c([Q_e, e])$
  - set of processed elements T
- while $\min_{e \in E \setminus T} \phi_e < \phi_t$

  pick $e \in \operatorname{argmin}_{f \in E \setminus T} \phi_f$

  `process`(e)

(w.l.o.g.: every $P \in \mathcal{P}$ starts with s and ends with t)





Shortest Paths in Abstract Networks

Given: abstract network $(E, \mathcal{P})$, costs $c \in \mathbb{R}_+^E$

Task: find $P \in \mathcal{P}$ minmizing $c(P) := \sum_{e \in P} c_e$

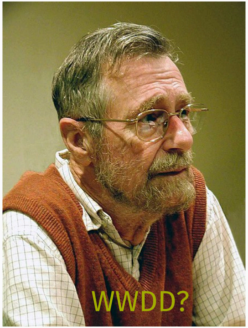
Dijkstra's Algorithm (adapted)

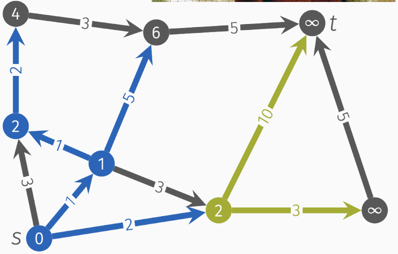
- initialize:
  - label ϕ_e for $e \in E$
  - $Q_e \in \mathcal{P}$ with $\phi_e = c([Q_e, e])$
  - set of processed elements T
- while $\min_{e \in E \setminus T} \phi_e < \phi_t$

  pick $e \in \operatorname{argmin}_{f \in E \setminus T} \phi_f$

  `process`(e)

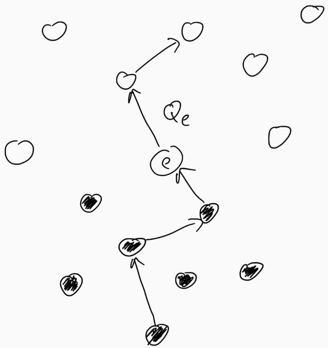
(w.l.o.g.: every $P \in \mathcal{P}$ starts with s and ends with t)





Processing Elements

How to find all relevant ways to continue $[Q_e, e]$?

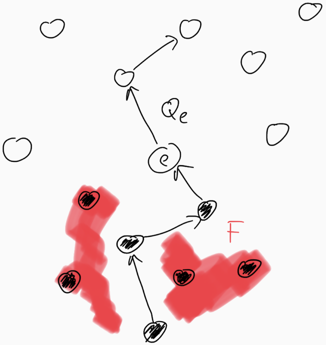


Processing Elements

How to find all relevant ways to continue $[Q_e, e]$?

`process`(e)

- $F := T \backslash [Q_e, e]$



Processing Elements

How to find all relevant ways to continue $[Q_e, e]$?

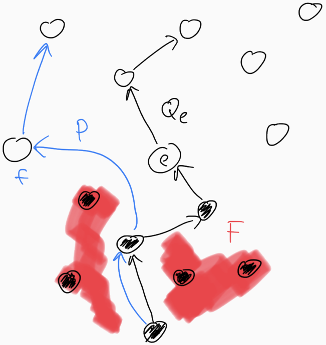
$\texttt{process}(e)$

- $F := T \setminus [Q_e, e]$
- while $\exists P \in \mathcal{P}$ with $P \subseteq E \setminus F$:

  $f := \min_{\preceq_P} P \setminus [Q_e, e]$

  $F := F \cup \{f\}$

  if $c([P, f]) < \phi_f$ then update ϕ_f and Q_f



How to find all relevant ways to continue $[Q_e, e]$?

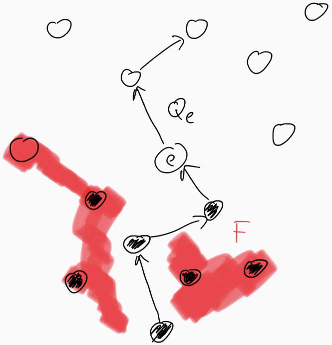
`process`(e)

- $F := T \setminus [Q_e, e]$
- while $\exists P \in \mathcal{P}$ with $P \subseteq E \setminus F$:

  $f := \min_{\preceq_P} P \setminus [Q_e, e]$

  $F := F \cup \{f\}$

  if $c([P, f]) < \phi_f$ then update ϕ_f and Q_f



Processing Elements

How to find all relevant ways to continue $[Q_e, e]$?

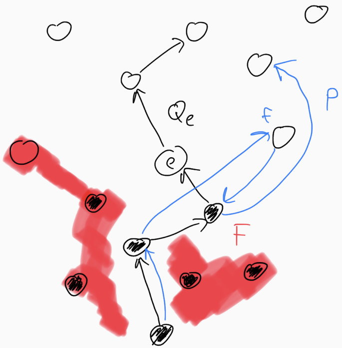
$\texttt{process}(e)$

- $F := T \backslash [Q_e, e]$
- while $\exists P \in \mathcal{P}$ with $P \subseteq E \backslash F$:

  $f := \min_{\preceq_P} P \backslash [Q_e, e]$

  $F := F \cup \{f\}$

  if $c([P, f]) < \phi_f$ then update ϕ_f and Q_f



Processing Elements

How to find all relevant ways to continue $[Q_e, e]$?

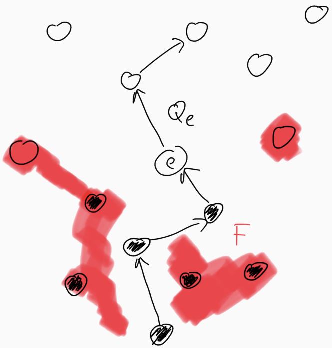
$\texttt{process}(e)$

- $F := T \backslash [Q_e, e]$
- while $\exists P \in \mathcal{P}$ with $P \subseteq E \backslash F$:

  $f := \min_{\preceq_P} P \backslash [Q_e, e]$

  $F := F \cup \{f\}$

  if $c([P, f]) < \phi_f$ then update ϕ_f and Q_f



How to find all relevant ways to continue $[Q_e, e]$?

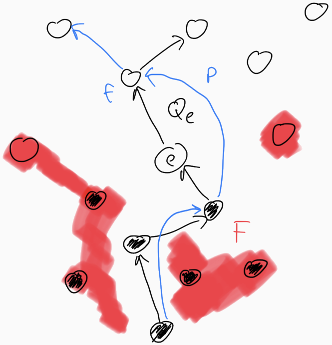
`process`(e)

- $F := T \backslash [Q_e, e]$
- while $\exists P \in \mathcal{P}$ with $P \subseteq E \backslash F$:

  $f := \min_{\preceq_P} P \backslash [Q_e, e]$

  $F := F \cup \{f\}$

  if $c([P, f]) < \phi_f$ then update ϕ_f and Q_f



Processing Elements

How to find all relevant ways to continue $[Q_e, e]$?

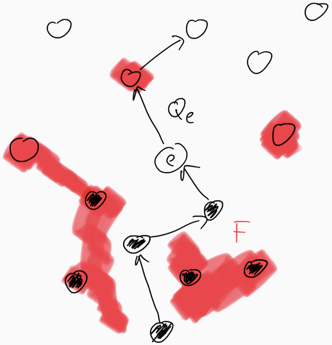
`process`(e)

- $F := T \setminus [Q_e, e]$
- while $\exists P \in \mathcal{P}$ with $P \subseteq E \setminus F$:

  $f := \min_{\preceq_P} P \setminus [Q_e, e]$

  $F := F \cup \{f\}$

  if $c([P, f]) < \phi_f$ then update ϕ_f and Q_f
- $T := T \cup \{e\}$



Processing Elements

How to find all relevant ways to continue $[Q_e, e]$?

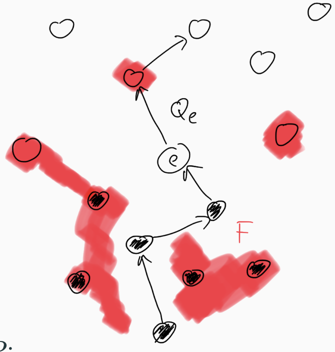
$\texttt{process}(e)$

- $F := T \setminus [Q_e, e]$
- while $\exists P \in \mathcal{P}$ with $P \subseteq E \setminus F$:

  $f := \min_{\preceq_P} P \setminus [Q_e, e]$

  $F := F \cup \{f\}$

  if $c([P, f]) < \phi_f$ then update ϕ_f and Q_f
- $T := T \cup \{e\}$



Lemma. After $\texttt{process}(e)$, for every $P \in \mathcal{P}$ with $e \in P$:

- there is $f \in P \setminus T$ with $\phi_f \leq \phi_e + c_f$.

Processing Elements

How to find all relevant ways to continue $[Q_e, e]$?

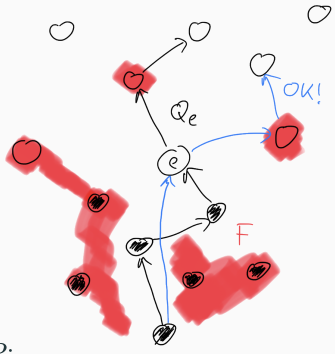
$\texttt{process}(e)$

- $F := T \backslash [Q_e, e]$
- while $\exists P \in \mathcal{P}$ with $P \subseteq E \backslash F$:

  $f := \min_{\preceq_P} P \backslash [Q_e, e]$

  $F := F \cup \{f\}$

  if $c([P, f]) < \phi_f$ then update ϕ_f and Q_f
- $T := T \cup \{e\}$



Lemma. After $\texttt{process}(e)$, for every $P \in \mathcal{P}$ with $e \in P$:

- there is $f \in P \backslash T$ with $\phi_f \leq \phi_e + c_f$.

How to find all relevant ways to continue $[Q_e, e]$?

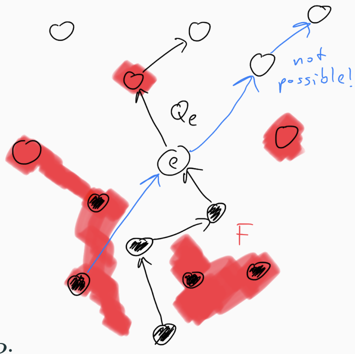
`process`(e)

- $F := T \setminus [Q_e, e]$
- while $\exists P \in \mathcal{P}$ with $P \subseteq E \setminus F$:

  $f := \min_{\preceq_P} P \setminus [Q_e, e]$

  $F := F \cup \{f\}$

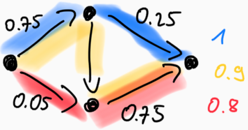
  if $c([P, f]) < \phi_f$ then update ϕ_f and Q_f
- $T := T \cup \{e\}$

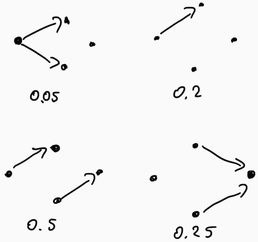


Lemma. After `process`(e), for every $P \in \mathcal{P}$ with $e \in P$:

- there is $f \in P \setminus T$ with $\phi_f \leq \phi_e + c_f$.

Conclusion





- $(\star)$-**sufficiency** allows formulating problems via their marginals:

$$\sum_{e \in P} \rho_e \geq \pi_P \quad \forall\, P \in \mathcal{P} \quad (\star)$$

- many systems are $(\star)$-sufficient, including **abstract networks**
- feasible decompositions can be computed via a **shortest-path algorithm**

Overview & Open Questions

	DAGs (Dahan et al.)	Abstract Networks (incl. digraphs w. cycles)	Max-Flow/Min-Cut
Affine	efficient algorithm	efficient algorithm (explicit description) $\oplus$	characterize $(\star)$-sufficiency
	$\Updownarrow$		
Conservation	$(\star)$-sufficient (exp.-time algorithm)	$(\star)$-sufficient (oracle-poly)	TDI systems?

$\oplus$ combinatorial shortest-path algorithm for abstract networks

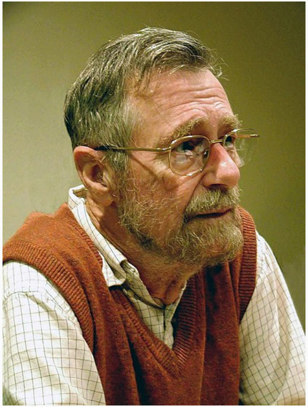
Strongly poly-time algorithm for Abstract Max Flow?

Also: NP-hard to decide feasibility of given ρ in general systems

Poly-time algorithms for some non-$(\star)$-sufficient systems?

$(\star)$-sufficiency under additional constraints on decomposition?

Copyright note



“Portrait of Edsger W. Dijkstra, one of the greatest mathematicans in history of modern mathematics.”