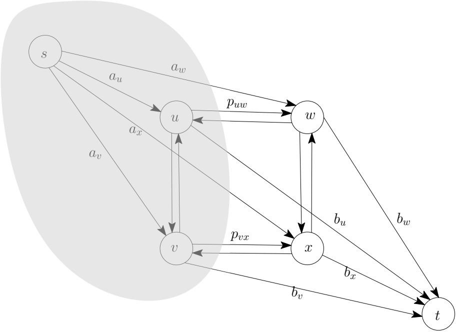
Review of Last Lecture

- Massaging mixed-signed objectives
- Recognizing a min cut problem



$$c(A', B') = \sum_{i \in A' \setminus \{s\}} b_i + \sum_{j \in B' \setminus \{t\}} a_j + \sum_{\{i,j\} \in E : |A' \cap \{i,j\}| = 1} p_{ij}.$$

Learning Goal

- Project selection problem and its reduction to min cut
- Turning hard constraints into punishments in the objective

Motivating Problem

- Let P be a set of n projects; doing project i yields a profit $p_i \in \mathbb{R}$. Note that p_i can be negative.

Motivating Problem

- Let P be a set of n projects; doing project i yields a profit $p_i \in \mathbb{R}$. Note that p_i can be negative.
- Each project i associated with a set $R_i \subset P$: project i can be done only if all projects in R_i are done first.

Motivating Problem

- Let P be a set of n projects; doing project i yields a profit $p_i \in \mathbb{R}$. Note that p_i can be negative.
- Each project i associated with a set $R_i \subset P$: project i can be done only if all projects in R_i are done first.
- Objective: Choose a set of projects to maximize total profits, subject to the prerequisite requirements;

Motivating Problem

- Let P be a set of n projects; doing project i yields a profit $p_i \in \mathbb{R}$. Note that p_i can be negative.
- Each project i associated with a set $R_i \subset P$: project i can be done only if all projects in R_i are done first.
- Objective: Choose a set of projects to maximize total profits, subject to the prerequisite requirements;
- Formally, choose a set $S \subseteq P$ that maximizes $\sum_{i \in S} p_i$, such that $R_i \subseteq S$ for every $i \in S$

Problem Formulation

- Input: an acyclic directed graph $G = (P, E)$, each node representing a project, with the associated profit;

Problem Formulation

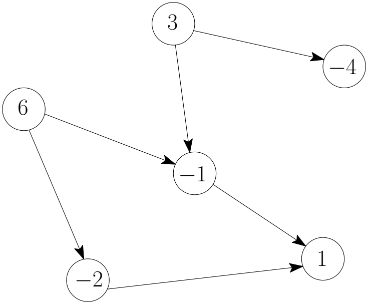
- Input: an acyclic directed graph $G = (P, E)$, each node representing a project, with the associated profit;
- Edge from i to j represents $j \in R_i$, i.e., project j is a prerequisite for i;

Problem Formulation

- Input: an acyclic directed graph $G = (P, E)$, each node representing a project, with the associated profit;
- Edge from i to j represents $j \in R_i$, i.e., project j is a prerequisite for i;
- Choosing a subset of projects corresponds to choosing a partition of the node set, which suggests a cut problem.

Problem Formulation

- Input: an acyclic directed graph $G = (P, E)$, each node representing a project, with the associated profit;
- Edge from i to j represents $j \in R_i$, i.e., project j is a prerequisite for i;
- Choosing a subset of projects corresponds to choosing a partition of the node set, which suggests a cut problem.



Adjusting the objective

- The objective is in fact mixed signed.
  - Let A be the set of projects with nonnegative profits, and B the set with negative profits, then the profit of any $S \subseteq P$ is

$$\sum_{i \in S \cap A} p_i - \sum_{i \in S \cap B} |p_i|.$$

Adjusting the objective

- The objective is in fact mixed signed.
  - Let A be the set of projects with nonnegative profits, and B the set with negative profits, then the profit of any $S \subseteq P$ is

$$\sum_{i \in S \cap A} p_i - \sum_{i \in S \cap B} |p_i|.$$

  - Adding a constant $\sum_{i \in B} |p_i|$, the objective becomes

$$\max_{S \subseteq P} \sum_{i \in S \cap A} p_i + \sum_{i \in B \setminus S} |p_i|.$$

Adjusting the objective

- The objective is in fact mixed signed.
  - Let A be the set of projects with nonnegative profits, and B the set with negative profits, then the profit of any $S \subseteq P$ is

$$\sum_{i \in S \cap A} p_i - \sum_{i \in S \cap B} |p_i|.$$

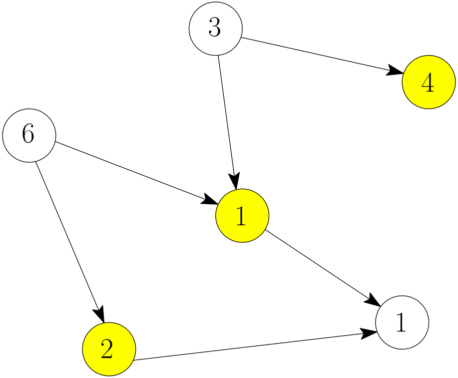
  - Adding a constant $\sum_{i \in B} |p_i|$, the objective becomes

$$\max_{S \subseteq P} \sum_{i \in S \cap A} p_i + \sum_{i \in B \setminus S} |p_i|.$$

  - To reduce to min cut, we need a minimization problem:

$$\min_{S \subseteq P} \sum_{i \in A \setminus S} p_i + \sum_{i \in S \cap B} |p_i|.$$

An illustration



Reduction to Min Cut

$$\min_{S \subseteq P} \sum_{i \in A \setminus S} p_i + \sum_{i \in S \cap B} |p_i|.$$

Reduction to Min Cut

$$\min_{S \subseteq P} \sum_{i \in A \setminus S} p_i + \sum_{i \in S \cap B} |p_i|.$$

- Add source s and sink t to G; add edge from s to each $i \in A$ with capacity p_i, and edge to t from each $i \in B$ with capacity $|p_i|$.

Reduction to Min Cut

$$\min_{S \subseteq P} \sum_{i \in A \setminus S} p_i + \sum_{i \in S \cap B} |p_i|.$$

- Add source s and sink t to G; add edge from s to each $i \in A$ with capacity p_i, and edge to t from each $i \in B$ with capacity $|p_i|$.
- Now for any $S \subseteq P$, $c(S \cup \{s\}, \overline{S} \cup \{t\}) = \sum_{i \in A \setminus S} p_i + \sum_{i \in S \cap B} |p_i|$, i.e., nodes on the side of s in the cut are the selected projects.

Reduction to Min Cut

$$\min_{S \subseteq P} \sum_{i \in A \setminus S} p_i + \sum_{i \in S \cap B} |p_i|.$$

- Add source s and sink t to G; add edge from s to each $i \in A$ with capacity p_i, and edge to t from each $i \in B$ with capacity $|p_i|$.
- Now for any $S \subseteq P$, $c(S \cup \{s\}, \overline{S} \cup \{t\}) = \sum_{i \in A \setminus S} p_i + \sum_{i \in S \cap B} |p_i|$, i.e., nodes on the side of s in the cut are the selected projects.
- How about the prerequisite requirements?

Reduction to Min Cut

$$\min_{S \subseteq P} \sum_{i \in A \setminus S} p_i + \sum_{i \in S \cap B} |p_i|.$$

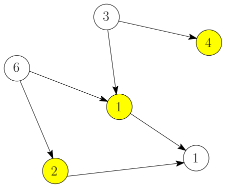
- Add source s and sink t to G; add edge from s to each $i \in A$ with capacity p_i, and edge to t from each $i \in B$ with capacity $|p_i|$.
- Now for any $S \subseteq P$, $c(S \cup \{s\}, \overline{S} \cup \{t\}) = \sum_{i \in A \setminus S} p_i + \sum_{i \in S \cap B} |p_i|$, i.e., nodes on the side of s in the cut are the selected projects.
- How about the prerequisite requirements?
- If no edge in G is in the cut we output, all prerequisite requirements are satisfied!

Reduction to Min Cut

$$\min_{S\subseteq P} \sum_{i\in A\setminus S} p_i + \sum_{i\in S\cap B} |p_i|.$$

- Add source s and sink t to G; add edge from s to each $i \in A$ with capacity p_i, and edge to t from each $i \in B$ with capacity $|p_i|$.
- Now for any $S \subseteq P$, $c(S \cup \{s\}, \overline{S} \cup \{t\}) = \sum_{i\in A\setminus S} p_i + \sum_{i\in S\cap B} |p_i|$, i.e., nodes on the side of s in the cut are the selected projects.
- How about the prerequisite requirements?
- If no edge in G is in the cut we output, all prerequisite requirements are satisfied!
- Let all the original edges in G carry infinite (or large enough) capacity.

Illustration of flow network



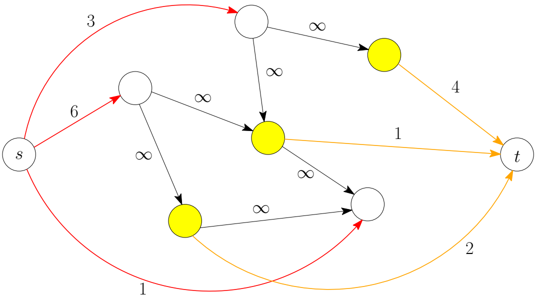
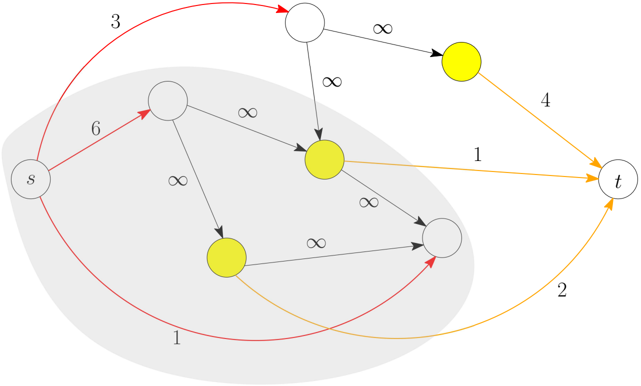


Illustration of a cut



Last remark

- A commonly used idea when doing reductions among problems: convert “hard” constraints to “soft” ones. I.e., if in one problem, certain patterns are forbidden, then in the other problem, punish such patterns in the objective — when the punishment is high enough, such patterns are forbidden from the solution.

Last remark

- A commonly used idea when doing reductions among problems: convert "hard" constraints to "soft" ones. I.e., if in one problem, certain patterns are forbidden, then in the other problem, punish such patterns in the objective — when the punishment is high enough, such patterns are forbidden from the solution.
- (Outside this course:) Recall Lagrangian multipliers from calculus. Hard constraints are softened into punishment in the objective, and the multipliers adjust how heavy the punishment is.