Minimum Cost Flow Problem

CO 351: Network Flow Theory

David Duan, 2019 Fall

Contents

1 Minimum Cost Flow Problem

1.1 Overview

1.1.1 Problem

Given digraph $D = (N, A)$, node demands $b \in \mathbb{R}^N$, arc costs $w \in \mathbb{R}^A$, arc capacities $c \in \mathbb{R}^A_+$, we want to find a feasible flow x satisfying node demands and capacity constraints while minimizing the total cost.

We assume $c_e > 0$ for all $e \in A$. Otherwise, we can just remove the arc with negative capacity without affecting the problem.

1.1.2 LP Formulation

We just need to take capacity constraints into consideration (on top of the TP LP):

$$\begin{aligned} \min \quad & w^T x \\ s.t. \quad & x(\delta(\bar{v})) - x(\delta(v)) = b_v \quad (\forall v \in N) \\ & x_e \leq c_e \quad (\forall e \in A) \\ & x \geq \mathbf{0} \end{aligned}$$

Adding slack variables s_e for each $e \in A$, we turn the LP into SEF:

$$\begin{aligned} \min \quad & w^T x \\ s.t. \quad & x(\delta(\bar{v})) - x(\delta(v)) = b_v \quad (\forall v \in N) \\ & x_e + s_e = c_e \quad (\forall e \in A) \\ & x, s \geq \mathbf{0} \end{aligned}$$

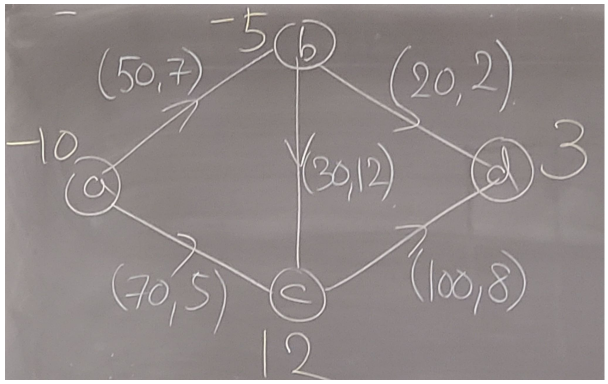
Expressing the constraints in matrix form, we have

$$\begin{array}{c} \\ |N| \\ |A| \end{array} \begin{array}{c} \begin{array}{cc} |A| & |A| \end{array} \\ \begin{pmatrix} M & O \\ I & I \end{pmatrix} \end{array} x = \begin{pmatrix} b \\ c \end{pmatrix},$$

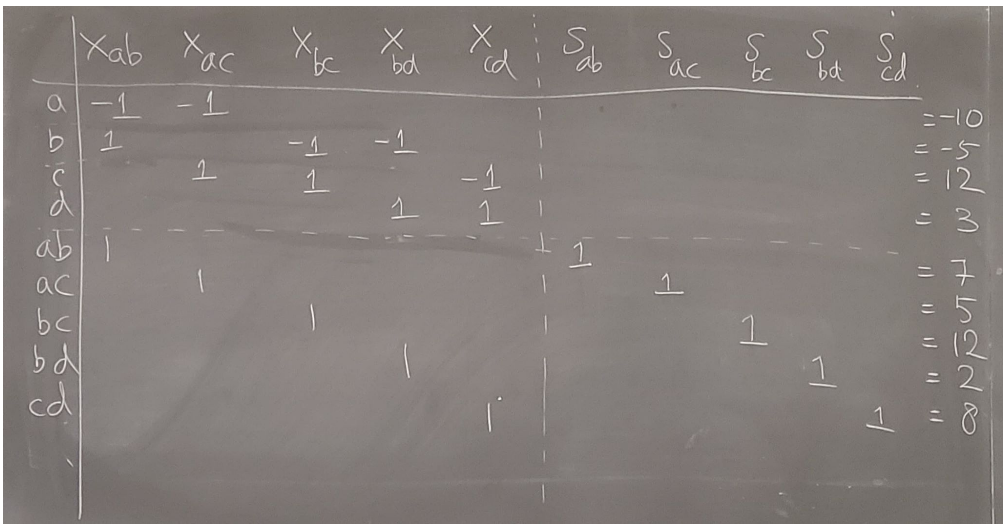
See the following example for more information.

1.1.3 Example

Consider the following MCFP ((cost, capacity) for each arc):



The corresponding constraint matrix:



	x_{ab}	x_{ac}	x_{bc}	x_{bd}	x_{cd}	s_{ab}	s_{ac}	s_{bc}	s_{bd}	s_{cd}	
a	-1	-1									= -10
b	1		-1	-1							= -5
c		1	1		-1						= 12
d				1	1						= 3
ab	1					1					= 7
ac		1					1				= 5
bc			1					1			= 12
bd				1					1		= 2
cd					1					1	= 8

We can divide the $(|N| + |A|) \times (|A| + |A|)$ constraint matrix into four parts:

1. Top-left $|N| \times |A|$: incidence matrix. This is what we've been dealing with in TP for the past two weeks. You should be familiar with this already.
2. Top-right $|N| \times |A|$: zero matrix. The columns in this sub-matrix correspond to the slack variables for capacity constraints. They have nothing to do with the nodes and thus it's a zero matrix.
3. Bottom-left $|A| \times |A|$: identity matrix. This is the "x_e" part of the capacity constraint $x_e + s_e = c_e$ for $e \in A$.
4. Bottom-right $|A| \times |A|$: identity matrix. This is the "s_e" part of the capacity constraint $x_e + s_e = c_e$ for $e \in A$.

1.1.4 Dual LP

Recall each column in the original LP corresponds to a constraint in the dual LP.

$$
\begin{aligned}
\min \quad & w^T x \\
s.t. \quad & x(\delta(\bar{v})) - x(\delta(v)) = b_v \quad (\forall v \in N) \\
& x_e + s_e = c_e \quad (\forall e \in A) \\
& x, s \geq \mathbf{0}
\end{aligned}
$$

We write the constraints in matrix form:

$$L := \begin{matrix} & |A| & |A| \\ |N| & M & O \\ |A| & I & I \end{matrix} \quad \begin{pmatrix} M & O \\ I & I \end{pmatrix} x = \begin{pmatrix} b \\ c \end{pmatrix}$$

and call the constraint matrix L. The primal LP and dual LP are thus: (let $b' = (b, c)^T$)

$$\begin{array}{lll} \min & w^T x \\ s.t. & Lx = b' \\ & x \geq \mathbf{0} \end{array} \iff \begin{array}{lll} \max & b'^T y' \\ s.t. & L^T y' \leq w \\ & y' \text{ free} \end{array}$$

Since there are two categories of primal constraints: demand constraints (the first $|N|$ of them) and capacity constraints (the last $|A|$ of them), we write $y' = (y, z)^T$ where y correspond to the demand constraints and z to the capacity constraints.

Next, observe the dual constraints can be written as

$$L^T := \begin{matrix} & |N| & |A| \\ |A| & M^T & I_1 \\ |A| & O & I_2 \end{matrix} \quad \begin{pmatrix} M^T & I_1 \\ O & I_2 \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} \leq \begin{pmatrix} w \\ \mathbf{0} \end{pmatrix}$$

Recall the RHS of dual constraints comes from the objective function, and z-constraints correspond to slack variables. Since slack variables do not contribute to the objective function, the RHS corresponding to them is a zero vector.

We now write constraints from matrix form back to explicit equations.

We deal with the first $|A|$ rows/constraints of L^T first. Suppose a row corresponds to an arc uv. The entry corresponding to u and v in M^T equal -1 and 1, respectively, and the entry in I_1 corresponding to uv equals 1, so we have $-y_u + y_v + z_{uv} \leq w_{uv}$ for each $uv \in A$.

We now deal with the rest $|A|$ rows/constraints of L^T. Suppose a row corresponds to an arc uv. The only non-zero entry in this row is the entry in I_2 corresponding to uv. Thus, we have $z_{uv} \leq 0$ for each $uv \in A$. However, we want to keep all variables non-negative, so we flip the signs of z_{uv} for all z (we can do this because it corresponds to slack variables and only show up in the dual LP), so that the second constraint becomes $z_{uv} \geq 0$ for each $uv \in A$ and the first constraint becomes $-y_u + y_v - z_{uv} \leq w_{uv}$ for each $uv \in A$.

Finally, since we have flipped the signs of z_{uv}'s, the objective function is $b^T y - c^T z$. The dual is thus

$$\begin{array}{lll} \max & b^T y - c^T z & \\ s.t. & -y_u + y_v - z_{uv} \leq w_{uv} & \forall uv \in A \\ & z_{uv} \geq 0 & \forall uv \in A \end{array}$$

1.2 Basis

Lemma 1. The constraint matrix has rank $|N| - 1 + |A|$.

Recall from TP that the incidence matrix has rank $|N| - 1$. Observe the $|A|$ capacity constraints are linearly independent, so we can extend the basis by adding them in. The rank of this matrix is thus $|N| - 1 + |A|$. ■

Lemma 2. For each arc $e \in A$, at least one of x_e and s_e is in the basis.

Recall non-basic variables are set to zero. Suppose for some $e \in A$, x_e and s_e are both non-basic, so $x_e = s_e = 0$. Then there is no way to satisfy the capacity constraint of e. For example, if $x_{ab} = s_{ab} = 0$ in Example 1.1.4, then there is no way to satisfy $x_{ab} + s_{ab} = c_{ab} = 7$. ■

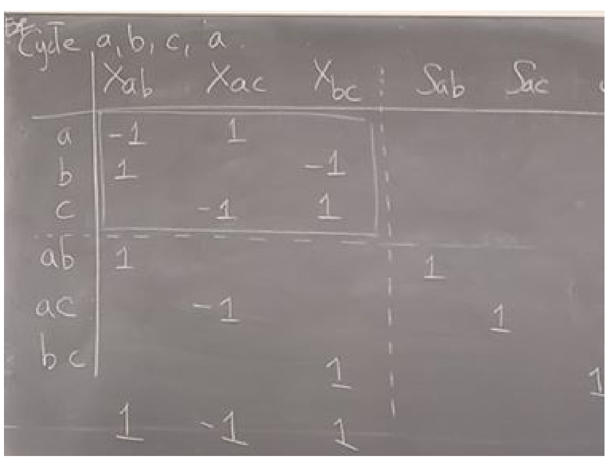
Therefore, for each arc $e \in A$, there are three possibilities:

1. Only x_e is in the basis. (The arc has flow equal to capacity.)
2. Only s_e is in the basis. (The arc does not have flow at all.)
3. Both x_e and s_e are in the basis. (The arc has positive flow, but less than its capacity.)

Lemma 3. There exist $|N| - 1$ case-3 arcs.

Let k denote the number of case-3 arcs. There are $|A|$ arcs in total, so $|A| - k$ arcs are case 1 and 2. Since both x_e and s_e are in the basis, each case-3 arc generates two basic variables. Recall the size of basis is $|N| + |A| - 1$, so there exist $|N| + |A| - 1$ basic variables. Counting the number of basic variables in two ways and solving for k, we get
$2k + |A| - k = |N| + |A| - 1 \implies k = |N| - 1$. ■

Lemma 4. Case-3 arcs cannot form a cycle.



Consider a cycle $C = \{ab, ac, bc\}$. Suppose all three arcs are case-3, i.e., columns corresponding to $x_{ab}, x_{ac}, x_{bc}, s_{ab}, s_{ac}, s_{bc}$ are all in the basis. However, the columns of the submatrix corresponding to these three arcs (shown above) are linearly dependent. A subset of a basis cannot be linearly dependent. Contradiction. ■

Since there are $|N|$ nodes in total and there are $|N| - 1$ case-3 arcs which contains no cycles, it follows that these case-3 arcs correspond to a spanning tree.

As a warning, do not confuse spanning tree with the basis; the basis is more complicated than the tree. We could have two different bases corresponding to the same spanning tree.

We summarize the above derivation into the following theorem:

Theorem A tree flow in MCFP consists of a spanning tree T and a feasible flow x where all arcs not in T satisfies $x_e \in \{0, c_e\}$.

- Case 1: x_e is in the basis $\implies s_e = 0 \implies x_e = c_e$.
- Case 2: s_e is in the basis $\implies x_e = 0 \implies s_e = c_e$.
- Case 3: x_e, s_e both in the basis $\implies 0 < x_e, s_e < c_e$.

1.3 Complementary Slackness Conditions

Recall the primal and dual LP for MCFP:

$$\begin{array}{ll} \min & w^T x \\ s.t. & x(\delta(\bar{v})) - x(\delta(v)) = b_v \quad (\forall v \in N) \\ & x_e \le c_e \quad (\forall e \in A) \\ & x \ge \mathbf{0} \end{array}$$

$$\begin{array}{lll} \max & b^T y - c^T z & \\ s.t. & -y_u + y_v - z_{uv} \le w_{uv} & \forall uv \in A \\ & z_{uv} \ge 0 & \forall uv \in A \end{array}$$

The CS conditions are thus:

1. $z_{uv} > 0 \implies x_{uv} = c_{uv}$.
2. $-y_u + y_v - z_{uv} < w_{uv} \implies x_{uv} = 0$.

But these conditions are too complicated to work with. We will simplify them a bit.

Recall from TP, the reduced cost $\bar{w}_{uv} = w_{uv} + y_u - y_v$. We can rewrite the dual constraint as $-z_{uv} \le \bar{w}_{uv}$, or $z_{uv} \ge -\bar{w}_{uv}$, with $z \ge \mathbf{0}$. The objective function contains $-c^T z$ where $c > 0$, so maximizing $b^T y - c^T z$ is equivalent to minimizing z. Therefore, we have $z_{uv} = \max\{0, -\bar{w}_{uv}\}$.

We can rewrite the CS conditions base on this fact. If $z_{uv} > 0$, we must have $-\bar{w}_{uv} > 0$, or $\bar{w}_{uv} < 0$. Thus, we can rewrite (1) as $\bar{w}_{uv} < 0 \implies x_{uv} = c_{uv}$.

Next, observe

$$\begin{aligned} -y_u + y_v - z_{uv} < w_{uv} &\iff -z_{uv} < w_{uv} + y_u - y_v = \bar{w}_{uv} \\ &\iff z_{uv} > -\bar{w}_{uv} \iff \bar{w}_{uv} > 0 \wedge z_{uv} = 0. \end{aligned}$$

Thus, (2) is equivalent to saying $\bar{w}_{uv} > 0 \implies x_{uv} = 0$.

Theorem The optimality conditions for MCFP are:

1. $\bar{w}_{uv} < 0 \implies x_{uv} = c_{uv}$.
2. $\bar{w}_{uv} > 0 \implies x_{uv} = 0$.
3. $0 < x_{uv} < c_{uv} \implies \bar{w}_{uv} = 0$.

In words, if the reduced cost for an arc is negative, we use up its capacity; if the reduced cost for an arc is positive, we will not use it at all; if an arc is used but not to its full capacity, its reduced cost must be zero.

1.3.1 Economic Interpretation

We could dual potentials as price of goods and reduced costs as the outcome for buying one unit at u, transporting through uv, then selling at v. If $\bar{w}_{uv} < 0$, we make a profit, so we send as much as we can, i..e, $x_{uv} = c_{uv}$. If $\bar{w}_{uv} > 0$, we suffer a loss, so we don't use it, i.e., $x_{uv} = 0$.

1.4 Network Simplex For MFCP

0. Find initial tree flow x with spanning tree T.
1. Find potentials y such that $\bar{w}_{uv} = 0$ for all $uv \in T$.
2. Find an non-basic arc uv where either
   a. $\bar{w}_{uv} < 0$ and $x_{uv} = 0$, or
   b. $\bar{w}_{uv} > 0$ and $x_{uv} = c_{uv}$.

   If no such arc exists, the current solution is optimal.
3. Let C be the unique cycle in $T + uv$.
   a. If (2a) occurs, orient C in the direction of uv.
   b. If (2b) occurs, orient C in the opposite direction from uv.
4. Find $t := \min(\{c_f - x_f : f \text{ is a forward arc of } C\} \cup \{x_f : f \text{ is a backward arc of } C\})$.
5. Push flow t along C.
6. Update tree and go back.

1.5 Feasibility Characterization

Theorem An MCFP is infeasible if and only if there exists $S \subseteq N$ such that $b(S) > c(\delta(\bar{S}))$ or $b(S) < -c(\delta(S))$.

Intuition.

- $b(S) > c(\delta(\bar{S}))$: Total demand is more than total capacity of in-arcs; too much demand, not enough transporting power.
- $b(S) < -c(\delta(S))$: Total supply is more than total capacity of out-arcs; too much supply, not enough transporting power.

Proof.

$\Longrightarrow$: We know from linear programming that an MCFP is feasible if and only if its corresponding auxiliary MCFP has optimal value 0.

We use the same auxiliary digraph as TP. Suppose our MCFP is infeasible. Then the auxiliary MCFP has optimal value greater than 0. Take x to be a feasible tree flow and set potentials y with $y_z = 0$ and all other potentials are either 1 or -1. Define S_+ and S_- as before. We will show that S_- satisfies $b(S_-) < -c(\delta(S_-))$.

Consider e from S_- to S_+ and f from S_+ to S_-. By CS conditions,

- $\bar{w}_e = 0 - 1 - 1 = -2 < 0 \implies x_e = c_e$, so all arcs going from S_- to S_+ are at capacity.
- $\bar{w}_f = 0 + 1 + 1 = 2 > 0 \implies x_f = 0$, so there is no in-flow from S_+ to S_-. Since there is also no arc from z to S_-, S_- has no in-flow at all.

Since the AUX MCFP has optimal value greater than 0 and $b_z = 0$, there must also be some flow leaving S_- for z. Then

$$\begin{aligned} b(S_-) &= \overbrace{x(\delta(\bar{S}_-))}^{0} - x(\delta(S_-)) \\ &= \sum_{u \in S_-, v \in S_+} (-c_{uv}) + \underbrace{\sum_{w \in S_-} (-x_{wz})}_{\text{non-zero by assumption}} < -c(\delta(S_-)). \end{aligned}$$

$\Longleftarrow$: Suppose there exists $S \subseteq N$ st $b(S) > c(\delta(\bar{S}))$. Suppose for a contradiction that there is a feasible flow x, then (the middle equality comes from A2)

$$b(S) = \sum_{v \in S} b_v = \sum_{v \in S} [x(\delta(\bar{v})) - x(\delta(v))] = x(\delta(\bar{S})) - x(\delta(S)) \leq c(\delta(\bar{S})) - 0 = c(\delta(\bar{S})),$$

which contradicts the hypothesis $b(S) > c(\delta(\bar{S}))$. $\square$

1.6 Applications for TP/MCFP

1.6.1 Minimum Cost Perfect Matching (MCPM)

Recall the following from graph theory: An undirected graph $G = (V, E)$ is *bipartite* if $V = A \cup B$, $A \cap B = \varnothing$, and all edges join A to B. A *matching* is a subset of edges with no common endpoints. A matching is *perfect* if it uses all vertices.

Given a bipartite graph $G = (V, E)$, $V = A \cup B$, $|A| = |B|$, and edge costs $w \in \mathbb{R}^E$, we want to find a perfect matching in G of minimum total cost.

We will formulate this as a MCFP:

- Direct each edge from A to B.
- Set the capacity for each arc to be 1.

- Set $v \in A$ as supply nodes with $b_v = -1$ and $v \in B$ as demand nodes with $b_v = 1$.
- Arc costs stay the same.

We want to show a correspondence between optimal solutions to MCFP and MCPM.

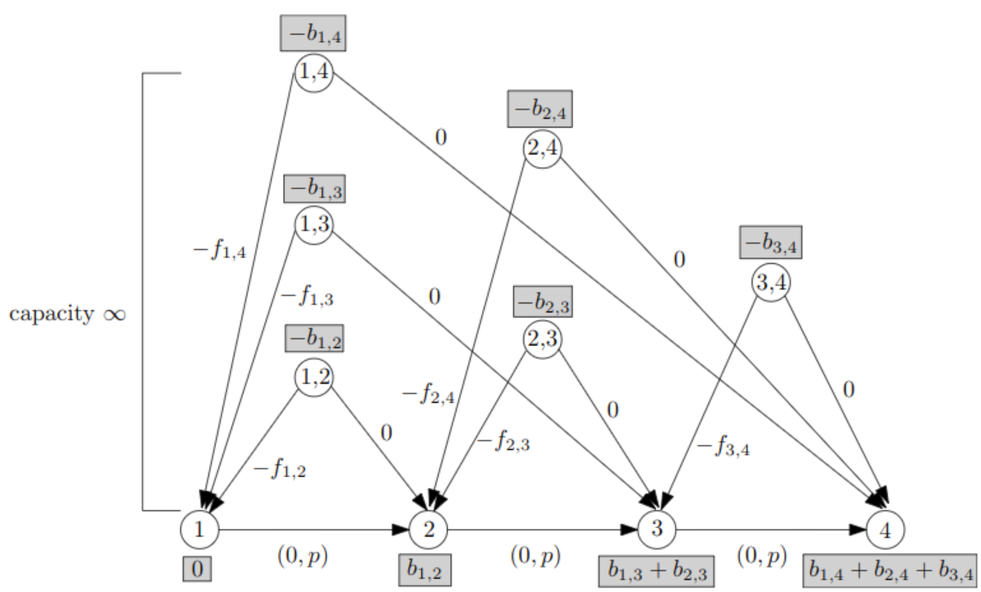
Lemma. If a MCFP has an optimal solution, and all capacities and node demands are integers, then there exists an integral optimal solution.

Proof. Check network simplex. Since the node demands are integers, we have an integral basic feasible solution. But the capacities are also integers, so at each iteration, we remain integral. ☐

For our MCFP formulation, by lemma, there exists an integer-valued optimal solution x. Then for each arc $e \in A$, $x_e = 0$ or $x_e = 1$. The set of active arcs $M = \{e : x_e = 1\}$ is a perfect matching because each node is incident with exactly one edge (supply = 1 so only one arc can be chosen). Also, any perfect matching corresponds to an integral flow. Hence, our MCFP solves the MCPM problem.

1.6.2 Airline Scheduling

- A plane visits cities $1, 2, \ldots, n$ in this order.
- There are $b_{i,j}$ passengers from city i to city j $(i < j)$.
- The ticket costs are $f_{i,j}$ $(i < j)$.
- The plane has capacity P.
- Our goal is to maximize ticket costs subject to plane capacity.



The arcs between nodes represent the path of the plane with capacity p, cost 0.

- Cost 0: We need to make the trip anyway, so we consider the cost as 0.
- Capacity p: the plane has capacity p, so each trip has p people at most.

Each node (i, j) takes passengers from i to j, either through the plane, or through other means.

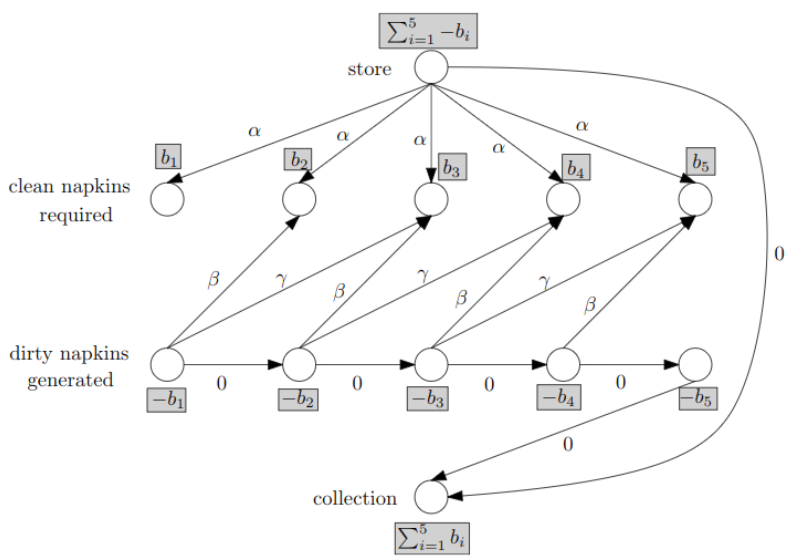
- $-f_{i,j}$ cost from node (i, j) to node (i): make money (negative since minimization).
- 0 cost from node (i, j) to node (j): passengers not taking the plane, no gain.

For example,

- $-b_{1,4}$: The number of people going from city 1 to 4.
- $b_{1,2}$: The number of people city 2 gets from city 1.
- The arc from node $(1, 2)$ to node (1): the amount of people going from city 1 to city 2.
    - We don't care about the capacity for this arc, e.g., we can set it to ∞, because at most there are $b_{1,2}$ people will be using this path.

1.6.3 Catering

- A caterer requires b_i clean napkins for each day $i = 1, \ldots, n$.
- They can buy new ones from the store for a cost of α.
- Used napkins can be washed in two ways:
    - 1-day service for a cost of β each.
    - 2-day service for a cost of γ each.
    - Used napkins can be kept in storage for free.
- We want to minimize the total cost of napkins.



For example, napkins required for day 3 can come from the following sources:

1. Dirty napkins from day 1 after a 2-day wash: $-b_1 \xrightarrow{\gamma} b_3$.
2. Dirty napkins from day 2 after an 1-day wash: $-b_2 \xrightarrow{\beta} b_3$.
3. Buy clean napkins from store: $(\sum_{i=1}^{5} -b_i) \xrightarrow{\alpha} b_3$.

1.6.4 Matrix with Consecutive 1's

Suppose we have the following LP

$$\begin{aligned} \min \quad & w^T x \\ s.t. \quad & Ax \geq b \\ & x \geq \mathbf{0} \end{aligned}$$

where the matrix has a special property: each column has consecutive 1's and all other entries are 0's. For example, consider

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 5 \\ 12 \\ 10 \\ 6 \end{pmatrix}$$

Let's say A has p rows. Add slack variables:

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & -1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}$$

Add a redundant row of all zeros:

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & -1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

In the order $p, p-1, p-2, \ldots, 1$, subtract the i-th constraint. (Subtract 4th from 5th, then 3rd from 4th, 2nd from 3rd, etc.) We get

$$A' = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 5 \\ 7 \\ -2 \\ -4 \\ -6 \end{pmatrix}$$

Observe A' becomes an incidence matrix! The entry with "1" in A' corresponds to the topmost "1" in A and "-1" in A' corresponds to the entry one below the bottommost "1".

Also, the demands add up to zero:

$$
\begin{aligned}
b'_{p+1} &= 0 - b_p, \\
b'_p &= b_p - b_{p-1}, \\
b'_{p-1} &= b_{p-1} - b_{p-2}, \\
&\cdots \\
b'_2 &= b_2 - b_1, \\
b'_1 &= b_1 - b_{p+1} = b_1,
\end{aligned}
\qquad \Longrightarrow \sum_{i=1}^{p+1} b'_i = (0 - b_p) + (b_p - b_{p-1}) + \cdots + (b_2 - b_1) + b_1 = 0.
$$

We now provide an scenario for this matrix to appear. Suppose you are running a manufacturing company. You must contract a storage company for $d(i)$ units of storage for the periods $i = 1, \ldots, n$. Let w_{ij} = cost of 1 unit of storage from period i to period j. You want to know how much capacity to acquire at what times and for how many periods.