Linear Programming (LP) (Chap.29)

- Suppose all the numbers below are real numbers.
- Given a linear function (called objective function)
  - $f(x_1, x_2, \ldots, x_n) = c_1x_1 + c_2x_2 + \ldots + c_nx_n = \sum_{j=1}^{n} c_jx_j$.
- With constraints:
  - $\sum_{j=1}^{n} a_{ij}x_j \leq b_i$ for $i=1,2,\ldots,m$ and
  - $x_j \geq 0$ for $j=1,2,\ldots,n$. (nonnegativity)
- Question: find values for $x_1, x_2, \ldots, x_n$, which maximize $f(x_1, x_2, \ldots, x_n)$.
- Or if change $\leq b_i$ to $\geq b_i$, then minimize $f(x_1, x_2, \ldots, x_n)$.

LP examples

- Political election problem:
  - Certain issues: building roads, gun control, farm subsidies, and gasoline tax.
  - Advertisement on different issues
  - Win the votes on different areas: urban, suburban, and rural.
  - Goal: minimize advertisement cost to win the majority of each area. (See page 772, (29.6) –(29.10))
- Flight crew schedule, minimize the number of crews:
  - Limitation on number of consecutive hours,
  - Limited to one model each month,…
- Oil well location decision with maximum of oil output:
  - A location is associated a cost and payoff of barrels of oil.
  - Limited budget.

Change a LP problem to standard format

- If some variables, such as x_i,may not have nonnegativity constraints:
  - delete x_i but introduce two variables x_i' and x_i'',
  - Replace each occurrence of x_i with x_i'-x_i''.
  - Add constraints: $x_i' \geq 0$ and $x_i'' \geq 0$.
- There may be equality constraints:
  - Replace $\sum_{j=1}^{n} a_{ij}x_j = b_i$ with $\sum_{j=1}^{n} a_{ij}x_j \leq b_i$ and $\sum_{j=1}^{n} a_{ij}x_j \geq b_i$.
  - Then change $\sum_{j=1}^{n} a_{ij}x_j \leq b_i$ to $\sum_{j=1}^{n} -a_{ij}x_j \geq -b_i$ for minimization or $\sum_{j=1}^{n} a_{ij}x_j \geq b_i$ to $\sum_{j=1}^{n} -a_{ij}x_j \leq -b_i$ for maximization.

Linear program in slack form

- Except nonnegativity constraints, all other constraints are equalities.
- Change standard form to slack form:
  – If $\sum_{j=1}^{n} a_{ij}x_j \le b_i$, then introduce new variable s, and set:
  – $s_i = b_i - \sum_{j=1}^{n} a_{ij}x_j$ and $s_i \ge 0$. (i=1,2,…,m).
  – If $\sum_{j=1}^{n} a_{ij}x_j \ge b_i$, then introduce new variable s, and set:
  – $s_i = \sum_{j=1}^{n} a_{ij}x_j - b_i$ and $s_i \ge 0$.
- All the left-hand side variables are called basic variables, whereas all the right-hand side variables are called nonbasic variables.
- Initially, s_1, s_1,…, s_m basic variables, x_1, x_1,…, x_n non-basic variables.

Formatting problems as LPs

- (Single pair) Shortest path :
  - A weighted direct graph G=<V,E> with weighted function w: E→R, a source s and a destination t, compute d which is the weight of the shortest path from s to t.
  - Change to LP:
    - For each vertex v, introduce a variable x_v: the weight of the shortest path from s to v.
    - Maximize x_t with the constraints:
    - $x_v \leq x_u + w(u,v)$ for each edge $(u,v) \in$ E, and $x_s = 0$.

Formatting problems as LPs

- Max-flow problem:
  - A directed graph G=<V,E>, a capacity function on each edge $c(u,v) \geq 0$ and a source s and a sink t. A flow is a function $f : V \times V \rightarrow R$ that satisfies:
    - Capacity constraints: for all $u,v \in V$, $f(u,v) \leq c(u,v)$.
    - Skew symmetry: for all $u,v \in V$, $f(u,v) = -f(v,u)$.
    - Flow conservation: for all $u \in V - \{s,t\}$, $\sum_{v \in V} f(u,v) = 0$, or to say, total flow out of a vertex other s or t is 0, or to say, how much comes in, also that much comes out.
  - Find a maximum flow from s to t.

Example of max-flow problem

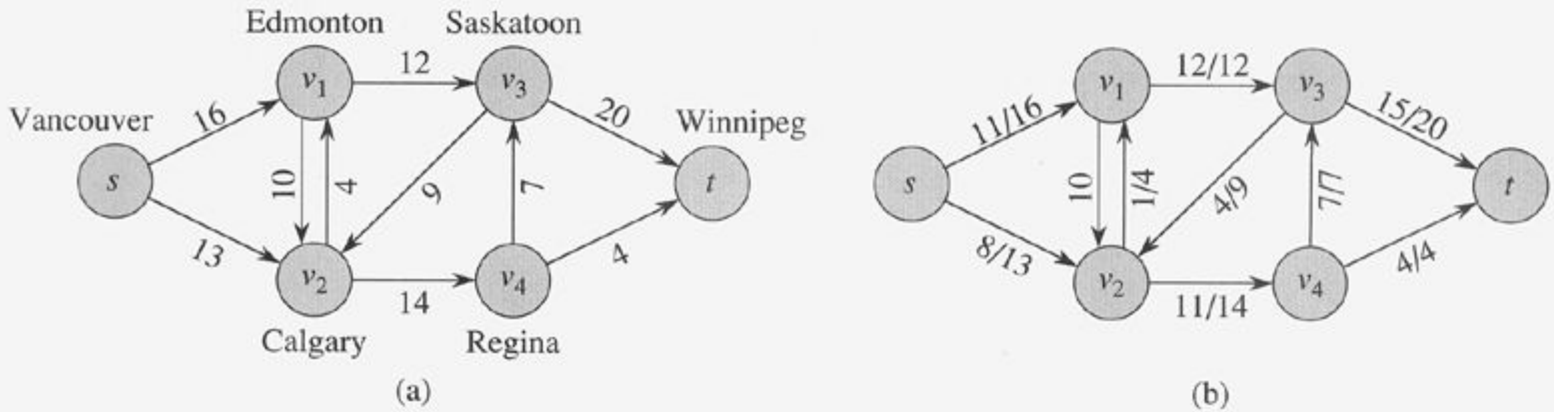

Figure 26.1 **(a)** A flow network $G = (V, E)$ for the Lucky Puck Company's trucking problem. The Vancouver factory is the source s, and the Winnipeg warehouse is the sink t. Pucks are shipped through intermediate cities, but only $c(u, v)$ crates per day can go from city u to city v. Each edge is labeled with its capacity. **(b)** A flow f in G with value $|f| = 19$. Only positive flows are shown. If $f(u, v) > 0$, edge (u, v) is labeled by $f(u, v)/c(u, v)$. (The slash notation is used merely to separate the flow and capacity; it does not indicate division.) If $f(u, v) \leq 0$, edge (u, v) is labeled only by its capacity.

Formatting Max-flow problem as LPs

- Maximize $\sum_{v \in V} f(s,v)$ with constraints:
    - for all $u,v \in V$, $f(u,v) \leq c(u,v)$.
    - for all $u,v \in V$, $f(u,v) = -f(v,u)$.
    - for all $u \in V - \{s,t\}$, $\sum_{v \in V} f(u,v) = 0$.

Linear Programming (LP) (Chap.29)

- Suppose all the numbers below are real numbers.
- Given a linear function (called objective function)
  - $f(x_1, x_2,\ldots, x_n)=c_1x_1+ c_2x_2+\ldots+ c_nx_n=\sum_{j=1}^{n} c_jx_j$.
- With constraints:
  - $\sum_{j=1}^{n} a_{ij}x_j \leq b_i$ for $i=1,2,\ldots,m$ and
  - $x_j \geq 0$ for $j=1,2,\ldots,n$. (nonnegativity)
- Question: find values for $x_1, x_2,\ldots, x_n$, which maximize $f(x_1, x_2,\ldots, x_n)$.
- Or if change $\leq b_i$ to $\geq b_i$, then minimize $f(x_1, x_2,\ldots, x_n)$.

Linear program in slack form

- Except nonnegativity constraints, all other constraints are equalities.
- Change standard form to slack form:
  - If $\sum_{j=1}^{n} a_{ij}x_j \leq b_i$, then introduce new variable s, and set:
  - $s_i = b_i - \sum_{j=1}^{n} a_{ij}x_j$ and $s_i \geq 0$. ($i=1,2,\ldots,m$).
  - If $\sum_{j=1}^{n} a_{ij}x_j \geq b_i$, then introduce new variable s, and set:
  - $s_i = \sum_{j=1}^{n} a_{ij}x_j - b_i$ and $s_i \geq 0$.
- All the left-hand side variables are called basic variables, whereas all the right-hand side variables are called nonbasic variables.
- Initially, s_1, s_1,…, s_m basic variables, x_1, x_1,…, x_n non-basic variables.

The Simplex algorithm for LP

- It is very simple
- It is often very fast in practice, even its worst running time is not poly.
- Illustrate by an example.

An example of Simplex algorithm

- Maximize $3x_1+x_2+2x_3$ (29.56)
- Subject to:
  - $x_1+x_2+3x_3 \leq 30$ (29.57)
  - $2x_1+2x_2+5x_3 \leq 24$ (29.58)
  - $4x_1+x_2+2x_3 \leq 36$ (29.59)
  - $x_1, x_2, x_3 \geq 0$ (29.60)
- Change to slack form:
  - $z= 3x_1+x_2+2x_3$ (29.61)
  - $x_4=30- x_1-x_2-3x_3$ (29.62)
  - $x_5=24- 2x_1-2x_2-5x_3$ (29.63)
  - $x_6=36- 4x_1-x_2-2x_3$ (29.64)
  - $x_1, x_2, x_3, x_4, x_5, x_6 \geq 0$

Simplex algorithm steps

- Feasible solutions (infinite number of them):
  - A solution whose values satisfy constraints, in this example,
  - as long as all of x_1, x_2, x_3, x_4, x_5, x_6 are nonnegative.
- basic solution:
  - set all nonbasic variables to 0 and compute all basic variables
- Iteratively rewrite the set of equations such that
  - No change to the underlying LP problem.
  - The feasible solutions keep the same.
  - However the basic solution changes, resulting in a greater objective value each time:
    - Select a nonbasic variable x_e whose coefficient in objective function is positive,
    - increase value of x_e as much as possible without violating any of constraints,
    - x_e is changed to basic and some other variable to nonbasic.

Simplex algorithm example

- Basic solution: $(x_1,x_2,x_3,x_4,x_5,x_6)$ =(0,0,0,30,24,36).
  - The result is $z=3 \cdot 0+0+2 \cdot 0=0$. Not maximum.
- Try to increase the value of x_1:
  - $z= 3x_1+x_2+2x_3$ (29.61)
  - $x_4=30- x_1-x_2-3x_3$ (29.62)
  - $x_5=24- 2x_1-2x_2-5x_3$ (29.63)
  - $x_6=36- 4x_1-x_2-2x_3$ (29.64)
    - 30: x_4 will be OK; 12: x_5; 9: x_6. So only to 9.
  - Change x_1 to basic variable by rewriting (29.64) to:
    - $x_1=9-x_2/4 - x_3/2 - x_6/4$
  - Note: x_6 becomes nonbasic.
  - Replace x_1 with above formula in all equations to get:

Simplex algorithm example

- $z=27+x_2/4+x_3/2-3x_6/4$ (29.67)
- $x_1=9-x_2/4-x_3/2-x_6/4$ (29.68)
- $x_4=21-3x_2/4-5x_3/2+x_6/4$ (29.69)
- $x_5=6-3x_2/2-4x_3+x_6/2$ (29.70)
- This operation is called pivot.
  - A pivot chooses a nonbasic variable, called entering variable, and a basic variable, called leaving variable, and changes their roles.
  - It can be seen the pivot operation is equivalent.
  - It can be checked the original solution (0,0,0,30,24,36) satisfies the new equations.
- In the example,
  - x_1 is entering variable, and x_6 is leaving variable.
  - x_2, x_3, x_6 are nonbasic, and x_1, x_4, x_5 becomes basic.
  - The basic solution for this is (9,0,0,21,6,0), with z=27.

Simplex algorithm example

- We continue to try to find a new variable whose value may increase.
  - x_6 will not work, since z will decrease.
  - x_2 and x_3 are OK. Suppose we select x_3.
- How far can we increase x_3 :
  - (29.68) limits it to 18, (29.69) to 42/5, (29.70) to 3/2. So rewrite (29.70) to:
    - x_3=3/2-3x_2/8 –x_5/4+x_6/8
  - Replace x_3 with this in all the equations to get:

Simplex algorithm example

- The LP equations:
  - $z=111/4+x_2/16 - x_5/8 - 11x_6/16$ (29.71)
  - $x_1=33/2- x_2/16 +x_5/8 - 5x_6/16$ (29.72)
  - $x_3=3/2-3x_2/8 - x_5/4+x_6/8$ (29.73)
  - $x_4=69/4+3x_2/16 +5x_5/8-x_6/16$ (29.74)
- The basic solution is (33/4,0,3/2,69/4,0,0) with $z=111/4$.
- Now we can only increase x_2. (29,72), (29.73), (29.74) limits x_2 to 132,4, ∞ respectively. So rewrite (29.73) to $x_2=4-8x_3/3 - 2x_5/3+x_6/3$
- Replace in all equations to get:

Simplex algorithm example

- LP equations:
  - $z=28-x_3/6 - x_5/6-2x_6/3$
  - $x_1=8+x_3/6+x_5/6-x_6/3$
  - $x_2=4-8x_3/3-2x_5/3+x_6/3$
  - $x_4=18-x_3/2+x_5/2$.
- At this point, all coefficients in objective functions are negative. So no further rewrite can be done. In fact, this is the state with optimal solution.
- The basic solution is (8,4,0,18,0,0) with objective value $z=28$.
- The original variables are x_1, x_2, x_3 , with values (8,4,0), the objective value of (29.58) $=3 \cdot 8+4+2 \cdot 0=28$.

Simplex algorithm --Pivot

```
PIVOT(N, B, A, b, c, v, l, e)
▷ Compute the coefficients of the equation for new basic variable x_e.
b̂_e ← b_l / a_le
for each j ∈ N − {e}
      do â_ej ← a_lj / a_le
â_el ← 1 / a_le
▷ Compute the coefficients of the remaining constraints.
for each i ∈ B − {l}
      do b̂_i ← b_i − a_ie b̂_e
         for each j ∈ N − {e}
               do â_ij ← a_ij − a_ie â_ej
         â_il ← −a_ie â_el
▷ Compute the objective function.
v̂ ← v + c_e b̂_e
for each j ∈ N − {e}
      do ĉ_j ← c_j − c_e â_ej
ĉ_l ← −c_e â_el
▷ Compute new sets of basic and nonbasic variables.
N̂ = N − {e} ∪ {l}
B̂ = B − {l} ∪ {e}
return (N̂, B̂, Â, b̂, ĉ, v̂)
```

N: indices set of nonbasic variables
B: indices set of basic variables
A: a_{ij}
b: b_i
c: c_i
v: constant coefficient.
e: index of entering variable
l: index of leaving variable

$z=v+\sum_{j\in N} c_j x_j$
$x_i=b_i - \sum_{j\in N} a_{ij} x_j$ for $i \in B$

Formal Simplex algorithm

SIMPLEX(A, b, c)

$(N, B, A, b, c, v) \leftarrow$ INITIALIZE-SIMPLEX(A, b, c)
while some index $j \in N$ has $c_j > 0$
    do choose an index $e \in N$ for which $c_e > 0$
        for each index $i \in B$
            do if $a_{ie} > 0$
                then $\Delta_i \leftarrow b_i / a_{ie}$
                else $\Delta_i \leftarrow \infty$
        choose an index $l \in B$ that minimizes Δ_i
        if $\Delta_l = \infty$
          then return "unbounded"
          else $(N, B, A, b, c, v) \leftarrow$ PIVOT(N, B, A, b, c, v, l, e)
for $i \leftarrow 1$ **to** $n + m$
    do if $i \in B$
      then $\bar{x}_i \leftarrow b_i$
      else $\bar{x}_i \leftarrow 0$
return $(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n)$

Running time of Simplex

- *Lemma 29.7* (page 803):
  - Assuming that INITIALIZE-SIMPLEX returns a slack form for which the basic solution is feasible, SIMPLEX either reports that a linear program is unbounded, or it terminates with a feasible solution in at most $\binom{m+n}{m}$ iterations.
- Feasible solution: a set of values for x_i*'s* which satisfy all constraints.
- Unbound: has feasible solutions but does not have a finite optimal objective value.

Two variable LP problems

- Example:
  - Maximize x_1+x_2 (29.11)
  - Subject to:
  - $4x_1 - x_2 \leq 8$ (29.12)
  - $2x_1+x_2 \leq 10$ (29.13)
  - $5x_1 - 2x_2 \geq -2$ (29.14)
  - $x_1, x_2 \geq 0$ (29.15)
- Can be solved by graphical approach.
- By prune-and-search approach.

Find maximum via graph

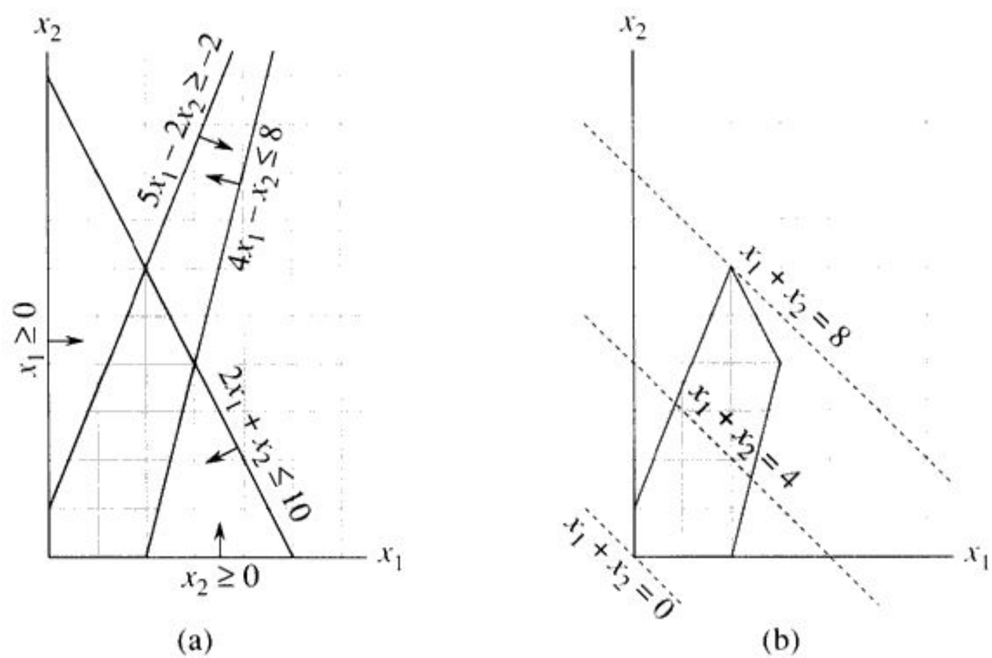

Figure 29.2 **(a)** The linear program given in (29.12)–(29.15). Each constraint is represented by a line and a direction. The intersection of the constraints, which is the feasible region, is shaded. **(b)** The dotted lines show, respectively, the points for which the objective value is 0, 4, and 8. The optimal solution to the linear program is $x_1 = 2$ and $x_2 = 6$ with objective value 8.

Integer linear program

- A linear program problem with additional constraint that all variables must take integral values.
  - Given an integer mxn matrix A and an integer m-vector b, whether there is an integer n-vector x such that $Ax<=b$.
  - this problem is NP-complete. (Prove it??)
- However the general linear program problem is poly time solvable.