# Lecture 2:
# Linear Programming and Revised Simplex

Prof. Krishna R. Pattipati
Dept. of Electrical and Computer Engineering
University of Connecticut
Contact: krishna@engr.uconn.edu; (860) 486-2890

January 28, 2016

# Reading List

- Bertsimas and Tsitsiklis, secs. 2.3-2.6, 3.1
- Luenberger, chapters 2 and 3

# Outline

- Simple Examples
- Historical Perspective Revisited
- Various Versions of LP
- Why do we need to solve linear programming problems ?
  - $L_1$ and $L_\infty$ curve fitting (i.e., parameter estimation using 1−norm and ∞−norm of error as minimization objective)
    - Application to FIR filter design
  - Diet problem
  - Portfolio optimization
  - Optimal control
  - Transportation problem
  - Shortest path problems
- Revised Simplex method
  - Fundamental theorem of LP
  - Geometric interpretation
  - Optimality conditions
  - Simplex iteration

# Simple Example to Illustrate the Geometry of LP

- Advertising problem
  - Dorian manufacturing Co. makes cars and trucks
  - Customers: High-income men and women
  - Want to advertise on comedy shows and football games
  - Each ***comedy*** commercial is seen by 7 million high-income women and 2 million high-income men
  - Each ***football*** commercial is seen by 2 million high-income women and 12 million high-income men
  - Cost:
    - 1-minute comedy commercial cost : $50K
    - 1-minute football commercial cost : $100K
  - Want to reach at least 28 million high-income women and 24 million high-income men
  - **Q**: How much advertising to buy to minimize cost?

    $x_1$ = Number of minutes of commercial bought on comedy shows

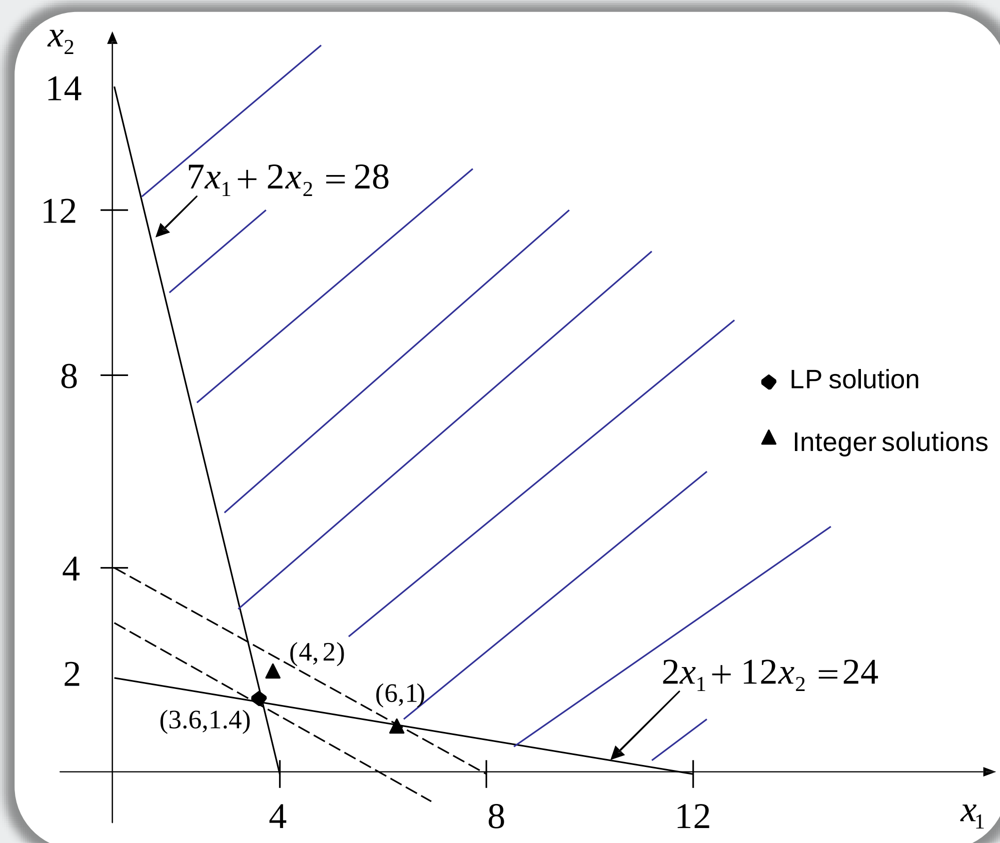
    $x_2$ = Number of minutes of commercial bought on football games
  - $x_1, x_2$ are integers $\Rightarrow$ Linear Integer Programming (LIP) problem

# Graphical Solution

$$\begin{aligned} \min\ f &= 50x_1 + 100x_2 \\ s.t.\quad 7x_1 + 2x_2 &\geq 28 \\ 2x_1 + 12x_2 &\geq 24 \end{aligned}$$

[x,fval,exitflag,output,lambda]=linprog(f,A,b)
MATLAB uses $A\,\underline{x} \leq \underline{b}$ !!!
[x,fval,exitflag,output]=intlinprog(f, intcon,A,b)
You can also use **solver** in Excel

- Relax integrality constraints $\Rightarrow x_1 \geq 0,\ x_2 \geq 0 \Rightarrow$ LP




Optimal Integer solutions:

$x_1 = 6, \quad x_2 = 1 \Rightarrow f = \$400K$

$x_1 = 4, \quad x_2 = 2 \Rightarrow f = \$400K$

LP Solution:

$x_1 = 3.6, x_2 = 1.4 \Rightarrow f = \$320K$

Note: Relaxed LP solution is a lower bound on the optimal LIP solution

UCONN

# Can LP problem have multiple solutions?

1. **An LP can have multiple solutions**
   - Automobile manufacturing process that makes cars and trucks
   - Must go through paint and body shops
   - Paint shop capacity
     - 40 trucks per day (or)
     - 60 cars per day
   - Body shop capacity
     - 50 trucks per day (or)
     - 50 cars per day
   - Profits
     - $300/truck
     - $200/car
   - Variables:
     - $x_1$ = # of trucks produced/day
     - $x_2$ = # of cars produced/day

# Is LP problem always feasible? No!!!

- Problem:

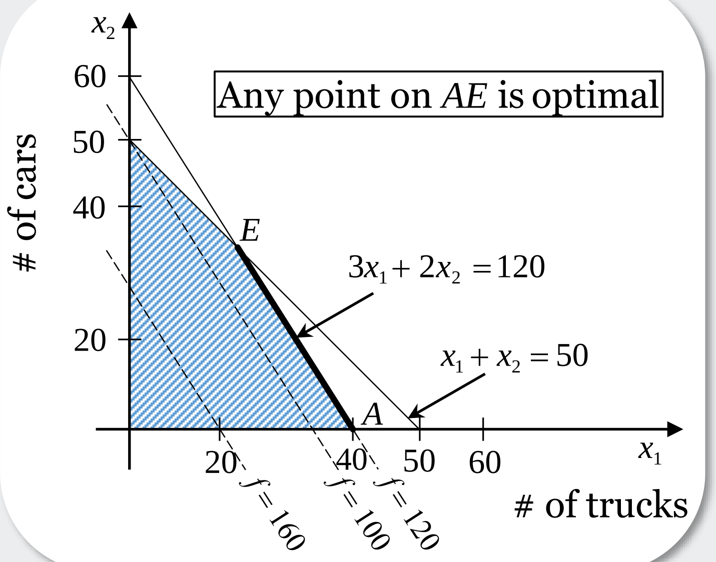
$$\max f = 3x_1 + 2x_2$$
$$s.t. \quad \frac{x_1}{40} + \frac{x_2}{60} \le 1 \quad \text{(paint shop)}$$
$$\frac{x_1}{50} + \frac{x_2}{50} \le 1 \quad \text{(body shop)}$$
$$x_1 \ge 0, x_2 \ge 0$$

$\Rightarrow$

$$\max f = 3x_1 + 2x_2$$
$$s.t.\ 3x_1 + 2x_2 \le 120 \quad \text{(paint shop)}$$
$$x_1 + x_2 \le 50 \quad \text{(body shop)}$$

- Multiple solutions:

$f = 120$;
$x_1 = 40, x_2 = 0$
$x_1 = 30, x_2 = 15$
$x_1 = 20, x_2 = 30$, etc.




## 2. An LP may be infeasible

$$\max f = 3x_1 + 2x_2$$
$$s.t.\ 3x_1 + 2x_2 \le 120 \quad \text{(paint shop)}$$
$$x_1 + x_2 \le 50 \quad \text{(body shop)}$$
$$x_1 \ge 30 \quad \text{(\# of trucks)}$$
$$x_2 \ge 20 \quad \text{(\# of cars)}$$

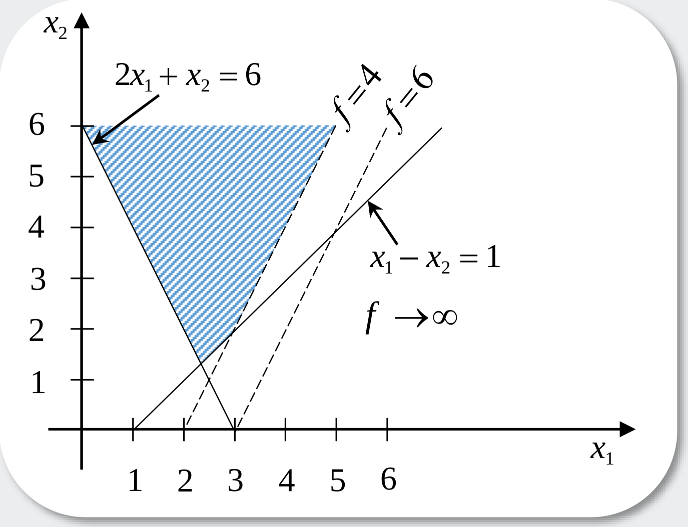
**Note:** at $x_1$= 30 and $x_2$= 20, 3 $x_1$ +2 $x_2$ =130
$\Rightarrow$ paint shop can't handle it

- Feasible space is empty

# Is the optimal solution always finite? No!!!

3. An LP can have an unbounded solution

$$\begin{aligned} \max\ f &= 2x_1 - x_2 \\ s.t.\ x_1 - x_2 &\le 1 \\ 2x_1 + x_2 &\ge 6 \\ x_1 &\ge 0 \\ x_2 &\ge 0 \end{aligned}$$




- Thus, an LP can have:
  - A unique solution
  - Multiple solutions (but with the same function value)
  - Infeasible solution space
  - Unbounded solutions ⇒

    $f \to \infty$ for max or

    $f \to -\infty$ for min

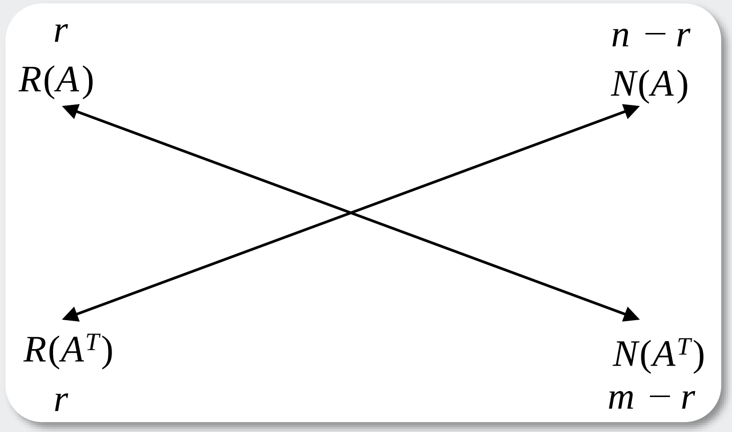
It will be nice if the algorithm detects these conditions

# Historical Perspective Revisited

- One of the most celebrated problems since 1951
- Major breakthroughs
  - Dantzig: Simplex method (1947-1949)
  - Khachian: Ellipsoid method (1979)
    - Polynomial complexity of LP, but not competitive with the Simplex method ⇒ not practical
  - Karmarker: Projective interior point algorithm (1984)
    - Polynomial complexity of LP and a competitive algorithm (especially for large problems)
- LP Problem definition
  - Given
    - An $m \times n$ matrix $A$, $m < n$ or $A \in R^{mn}$, $m < n$
      assume $\text{rank}(A) = m$
    - A column vector $\underline{b}$ with $m$ components: $\underline{b} \in R^m$
    - A row vector $c^T$ with $n$ components: $\underline{c} \in R^n$
  - $m < n \Rightarrow A\underline{x} = \underline{b}$ has infinitely many solutions $\Rightarrow \underline{b} = \sum_{i=1}^{n} \underline{a}_i x_i$

# What is Linear Programming (LP)

- Recall




- $r = m \Rightarrow N(A^T) = \varphi(origin)$
- Consider $\underline{x}_r \in R(A^T) \ni A\underline{x}_r = \underline{b} \Rightarrow A(\underline{x}_r + \underline{x}_n) = \underline{b}$ where
  $\underline{x}_n \in N(A) \Rightarrow (\underline{x}_n : A\underline{x}_n = 0)$
- We impose two restrictions on $\underline{x}$:
  - Want nonnegative solutions of $A\underline{x} = \underline{b} \Rightarrow x_i \geq 0$ (or) $\underline{x} \geq \underline{0}$

    $\underline{x} \ni A\underline{x} = \underline{b}$ & $\underline{x} \geq \underline{0}$ are said to be **feasible**

  - Among all those feasible $\underline{x}$'s, want $\underline{x}^* \ni c^T\underline{x} = c_1x_1 + c_2x_2 + \ldots + c_nx_n$ is a minimum

# Any LP problem can be converted to SLP

- This leads to the so-called "standard form of LP"

$$\text{(SLP)}: \quad \begin{aligned} \min \quad & \underline{c}^T \underline{x} \\ \text{s.t.} \quad & A\underline{x} = \underline{b} \\ & \underline{x} \geq \underline{0} \end{aligned}$$

convex programming problem. If a bounded solution exists, then $\underline{x}^*$ is unique ⇒ a single minimum.

- **Claim**: Any LP problem can be converted into standard form
- Inequality constraints

a) $\underline{a}_i^T \underline{x} \leq b_i \Rightarrow \begin{bmatrix} \underline{a}_i^T & 1 \end{bmatrix} \begin{bmatrix} \underline{x} \\ x_{n+1} \end{bmatrix} = b_i;\ x_{n+1} \geq 0$

$x_{n+1}$ ~ slack variable

In general, $A\underline{x} \leq \underline{b} \Rightarrow A\underline{x} + \underline{y} = \underline{b} \Rightarrow \overbrace{\begin{bmatrix} A & I \end{bmatrix}}^{A_a} \begin{bmatrix} \underline{x} \\ \underline{y} \end{bmatrix} = \underline{b},\ \underline{x}, \underline{y} \geq \underline{0}$

Increase number of variables by $m$ and $A_a$ is an $m \times (n+m)$ matrix

b) $\underline{a}_i^T \underline{x} \geq b_i \Rightarrow \underline{a}_i^T \underline{x} - x_{n+1} = b_i;\ x_{n+1} \geq 0$

$x_{n+1}$ ~ surplus variable

In general, $A\underline{x} \geq \underline{b} \Rightarrow \begin{bmatrix} A & -I \end{bmatrix} \begin{bmatrix} \underline{x} \\ \underline{y} \end{bmatrix} = \underline{b},\ \underline{y} \geq 0$

c) $d_i \leq x_i \Rightarrow$ define $\hat{x}_i = x_i - d_i,\ \hat{x}_i \geq 0$

d) $d_i \geq x_i \Rightarrow$ define $\hat{x}_i = d_i - x_i,\ \hat{x}_i \geq 0$

# Converting to standard LP

*e)* $d_{i1} \le x_i \le d_{i2} \Rightarrow 0 \le x_i - d_{i1} \le d_{i2} - d_{i1}$

Define $\hat{x}_i = x_i - d_{i1}$

& $\hat{x}_i + y_i = d_{i2} - d_{i1}$; slack $y_i \ge 0$

*f)* $b_{1i} \le \underline{a}_i^T \underline{x} \le b_{2i} \Rightarrow$ use a slack and a surplus

$$\left.\begin{aligned} \underline{a}_i^T \underline{x} - y_{i1} &= b_{1i} \\ \underline{a}_i^T \underline{x} + y_{i2} &= b_{2i} \end{aligned}\right\} y_{i1}, y_{i2} \ge 0$$

*g)* $|\underline{a}_i^T \underline{x}| \le b_i \Rightarrow -b_i \le \underline{a}_i^T \underline{x} \le b_i$

$\Rightarrow \underline{a}_i^T \underline{x} - y_{i1} = -b_i$

$\underline{a}_i^T \underline{x} + y_{i2} = b_i$

- $x_i$ is a free variable
  - Define $x_i = \bar{x}_i - \hat{x}_i$, with $\bar{x}_i,\ \hat{x}_i \ge 0$
- Maximization: change $\underline{c}^T \underline{x}$ to $-\underline{c}^T \underline{x}$
- $L_1$-minimization: min $\sum_{i=1}^{n} |x_i|$ *s.t.* $A\underline{x} \le \underline{b}$

$\Rightarrow A\underline{x} + \underline{y} = \underline{b}$

Write $x_i = \bar{x}_i - \hat{x}_i$

$$\Rightarrow \min \sum_{i=1}^{n} (\bar{x}_i + \hat{x}_i) \text{ s.t. } \begin{bmatrix} A & -A & I \end{bmatrix} \begin{bmatrix} \bar{\underline{x}} \\ \hat{\underline{x}} \\ \underline{y} \end{bmatrix} = \underline{b}$$

} Optimal solution of this problem solves the original problem. Also, if $\bar{x}_i > 0,\ \hat{x}_i = 0$ and vice versa.

# $L_1$ - curve fitting

1. $L_1$ - curve fitting
   - Recall that given a set of scalars $(b_1, b_2, \ldots, b_m)$, the estimate that minimizes $\sum_{i=1}^{m}|x - b_i|$ is the **median** and that this estimate is insensitive to outliers in the data $\{b_i\}$.
   - In vector case, want

$$\underline{x} \ni \min_{\underline{x}} \sum_{i=1}^{m} \left| \underline{a}_i^T \underline{x} - b_i \right| = \min_{\underline{x}} \left\| A\underline{x} - \underline{b} \right\|_1$$

   - $L_1$ - curve fitting ⇒ an LP
     - Write $x_i = \tilde{x}_i - \hat{x}_i, i=1,2,..,n$; $|\underline{a}_i^T\underline{x} - b_i| = u_i + v_i$
     - Then, the LP problem is:

$$\begin{aligned} \min_{\underline{x},\underline{u},\underline{v}} \sum_{i=1}^{n} (u_i + v_i) &= \min_{\underline{x},\underline{u},\underline{v}} \underline{e}^T (\underline{u} + \underline{v}) \\ s.t.\ \ A(\underline{\tilde{x}} - \underline{\hat{x}}) - \underline{u} + \underline{v} &= \underline{b} \\ \underline{\tilde{x}}, \underline{\hat{x}}, \underline{u}, \underline{v} &\geq \underline{0} \end{aligned}$$

# $L_\infty$ - curve fitting

2. $L_\infty$ - curve fitting
   - Want $\underline{x}$ such that

$$\min_{\underline{x}} \max_{1 \le i \le m} \left| \underline{a}_i^T \underline{x} - b_i \right| = \min_{\underline{x}} \left\| A\underline{x} - \underline{b} \right\|_\infty$$

   - $L_\infty$ - curve fitting $\Rightarrow$ an LP
     - Let $\max_{1 \le i \le m} |\underline{a}_i^T \underline{x} - b_i| = w$
     - Then, the problem is equivalent to:

$$\begin{aligned} &\min_{\underline{x}, w} w \\ &s.t. -w \le \underline{a}_i^T \underline{x} - b_i \le w, \text{for } i = 1, 2, \ldots, m \\ &\Rightarrow \min w \\ &\quad s.t. \begin{bmatrix} A & \underline{e} \\ -A & \underline{e} \end{bmatrix} \begin{bmatrix} \underline{x} \\ w \end{bmatrix} \ge \begin{bmatrix} \underline{b} \\ -\underline{b} \end{bmatrix} \end{aligned}$$

$$\begin{aligned} &\max \underline{b}^T \left( \underline{\lambda} - \underline{\mu} \right) \\ &s.t. A^T \left( \underline{\lambda} - \underline{\mu} \right) = \underline{0} \\ &\quad e^T \left( \underline{\lambda} + \underline{\mu} \right) = 1 \\ &\quad \underline{\lambda}, \underline{\mu} \ge \underline{0} \end{aligned}$$

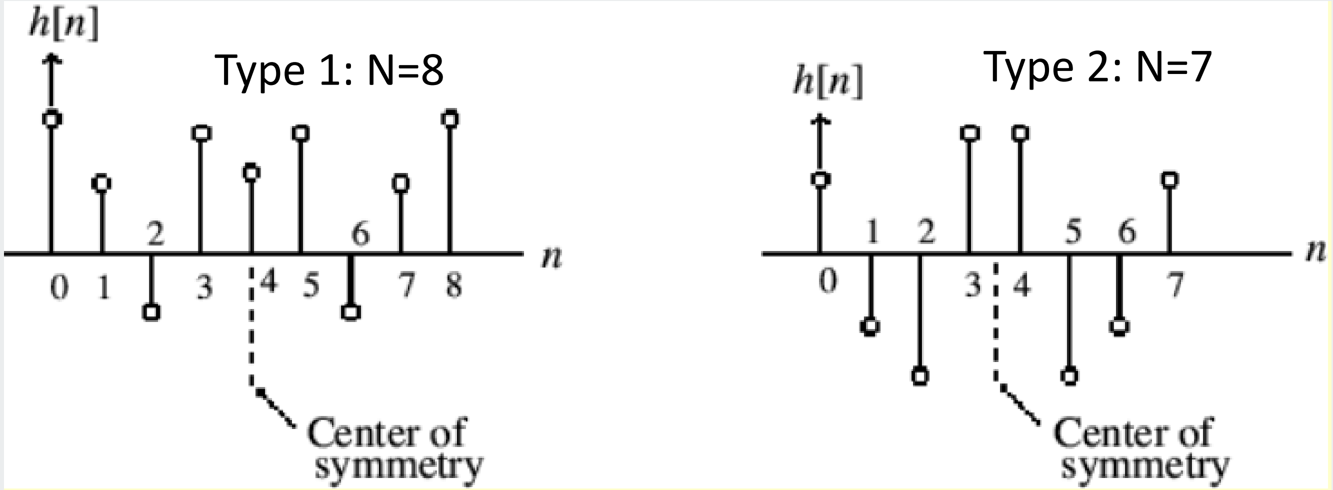
- Since the number of constraints is large (= $2m$) and the number of variables (= $n$) is small, typically the **dual** problem with ($n$ + 1) constraints and $2m$ variables is solved instead!!
- Dual is an LP
- We will discuss duality in Lecture 4

# $L_\infty$ - curve fitting in filter design

- Linear-phase Finite Impulse Response (FIR) filters
  - Impulse response coefficients: $\{h_n : n = 0,1,2,..,N\} \Rightarrow H(z) = \sum_{n=0}^{N} h_n z^{-n}$
  - Linear phase $\Rightarrow h_n = h_{N-n}$ *symmetric*
  - Frequency response ($0 \le \omega \le \pi/T$); $T$= sampling interval

$$H(z)|_{z=e^{j\omega T}} = H(e^{j\omega T}) = \sum_{n=0}^{N} h_n e^{-jn\omega T} = \begin{cases} e^{-j\frac{N\omega T}{2}} \left[ \underbrace{h_{N/2} + \sum_{n=0}^{N/2-1} 2h_n \cos[(N/2-n)\omega T]}_{real} \right]; N\ even; Type\ I\ \ filter \\ \underbrace{e^{-j\frac{N\omega T}{2}}}_{linear\ \ phase\ term} \left[ \sum_{n=0}^{(N-1)/2} \underbrace{2h_n \cos[(N/2-n)\omega T]}_{purely\ linear,\ linear\ in\{h_n\},real} \right]; N\ odd; Type\ II\ \ filter \end{cases}$$

$$|H(e^{j\omega T})| = |h_{N/2} + \sum_{n=0}^{N/2-1} 2h_n \cos[(N/2-n)\omega T]|\ for\ Type\ I\ or\ |\sum_{n=0}^{(N-1)/2} \underbrace{2h_n \cos[(N/2-n)\omega T]}_{purely\ linear,\ linear\ in\{h_n\},real}|\ for\ Type\ II$$



# $L_\infty$ - Type I FIR filter design problem

- $L_\infty$ - FIR filter design

$$Define\ \underline{x} = [x_0\ x_1 \ldots\ldots x_M];\ M = N/2$$

$$x_0 = h_{N/2} = h_M;\ x_j = 2h_{M-n} = 2h_{M+n};\ n = 1, 2, \ldots, M$$

$$H(e^{j\omega T}) = e^{-j\omega MT} \sum_{n=0}^{M} x_n \cos n\omega T = e^{-j\omega MT} x(\omega)$$

$$Desired\ response: d(\omega)\ and\ weighted\ error\ e(\omega) = f(\omega)[x(\omega) - d(\omega)]$$

$$f(\omega)\ strictly\ positive\ weighting\ function\ of\ \omega$$

$$\Pr oblem: \min_{\underline{x}} \max_{0 \le \omega \le \frac{\pi}{T}} |e(\omega)|$$

Minimize weighted Chebyshev error

$$\Rightarrow \min_{\underline{x},\delta} \delta\ s.t. -\delta \le f(\omega)[\sum_{n=0}^{M} x_n \cos n\omega T - d(\omega)] \le \delta\ \forall \omega \in [0, \frac{\pi}{T}]\ and$$

$$Discretize\ frequenecy: \{\omega_k : 1 \le k \le L\}.\ Let\ f_k = f(\omega_k)\ and\ d_k = d(\omega_k)$$

$$\min_{\underline{x},\delta} \delta\ s.t. -\delta \le f_k(\sum_{n=0}^{M} x_n \cos n\omega_k T - d_k) \le \delta\ \forall k = 1, 2, .., L$$

$$\Rightarrow \min_{x,\delta} \delta$$

$$s.t. -\frac{\delta}{f_k} \le \underline{a}_k^T \underline{x} - d_k \le \frac{\delta}{f_k}\ \forall k = 1, 2, .., L;\ \underline{a}_k^T = [1 \cos \omega_k \ldots\ldots \cos n\omega_k \ldots\ldots \cos M\omega_k]$$

# Matrix Formulation of FIR design problem

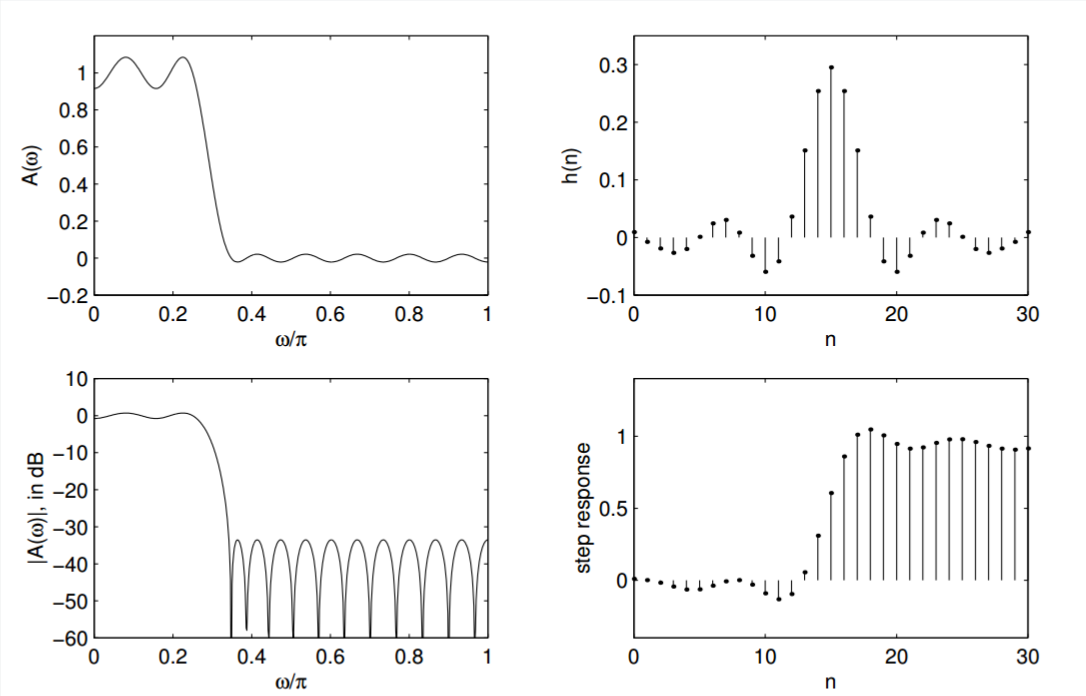
- Matrix Formulation

$$\Rightarrow \min_{\underline{x},\delta} \delta$$

$$s.t. \begin{bmatrix} A & [Diag(f_k)]^{-1}\underline{e} \\ -A & [Diag(f_k)]^{-1}\underline{e} \end{bmatrix} \begin{bmatrix} \underline{x} \\ \delta \end{bmatrix} \geq \begin{bmatrix} \underline{d} \\ -\underline{d} \end{bmatrix}$$

- Easy to include arbitrary linear constraints — including *time domain* constraints
- Sparse FIR coefficients

- Design a 30th order low-pass FIR filter




*Specs* :

$1. T = 1$

$$2. d(\omega) = \begin{cases} 1 \;\; for \;\; \omega \in [0, 0.26\pi] \\ 0 \;\; for \;\; \omega \in [0.34\pi, \pi] \end{cases}$$

$$3. f(\omega) = \begin{cases} 1 \;\; for \;\; \omega \in [0, 0.26\pi] \\ 0 \;\; for \;\; \omega \in [0.26\pi, 0.34\pi] \\ 2 \;\; for \;\; \omega \in [0.34\pi, \pi] \end{cases}$$

# Diet Problem

3. Diet problem

- A budget conscious Irish consumer wants to buy, at minimum cost, the following three basic foods: poultry, leafy spinach, and potatoes

| | **poultry** | **spinach** | **potatoes** |
|---|---|---|---|
| cost/100 *gms* | 40 | 15 | 10 |
| protein *gms* | 2 | 3 | 2 |
| carbohydrate *gms* | 0 | 3 | 18 |
| calcium *mgms* | 8 | 83 | 7 |
| iron *mgms* | 1.4 | 2 | 0.6 |
| vitamins *(IU)* | 80 | 7300 | 0 |

- He wants
  - 65 *gms* of protein
  - 90 *gms* of carbohydrate
  - 200 *mgms* of calcium
  - 10 *mgms* of iron
  - 5000 international units *(IU)* of vitamin A

- $x_1 \sim$ amount of poultry $(gms)$
- $x_2 \sim$ amount of spinach $(gms)$
- $x_3 \sim$ amount of potatoes $(gms)$

# LP formulation of Diet Problem

- Optimal solutions:

  $x_1 = 0$; $x_2 = 20.626$; $x_3 = 1{:}5625$, and $f = 325$ (solver)

  $x_1 = 0$; $x_2 = 0.7047$; $x_3 = 31.443$, and $f = 325$ (MATLAB)

$$
\begin{aligned}
\min\ & 40x_1 + 15x_2 + 10x_3 \\
s.t.\ & 2x_1 + 3x_2 + 2x_3 \geq 65 \\
& 3x_2 + 18x_3 \geq 90 \\
& 8x_1 + 83x_2 + 7x_3 \geq 200 \\
& 1.4x_1 + 2x_2 + 0.6x_3 \geq 10 \\
& 80x_1 + 7300x_2 \geq 5000 \\
& x_1, x_2, x_3 \geq 0
\end{aligned}
$$

- Show via **solver** in Excel or MATLAB
- More general diet problem can be formulated in a similar way
- Have $n$ different food items

  $c_j$ = cost of food item $j$

  $x_j$ = units of food item $j$ (in grams) included in our diet
- Have $m$ nutritional requirements

  $b_i$ = minimum daily requirement of $i^{th}$ nutrient

  $a_{ij}$ = amount of nutrient $i$ provided by a unit of food item $j$
- The problem is an LP

$$
\begin{aligned}
\min\ & \sum_{j=1}^{n} c_j x_j \\
s.t.\ & \sum_{j=1}^{n} a_{ij} x_j \geq b_i;\ i = 1, 2, \ldots, m \\
& x_j \geq 0;\ j = 1, 2, \ldots, n
\end{aligned}
\quad \Rightarrow \quad
\begin{aligned}
\min\ & \underline{c}^T \underline{x} \\
s.t.\ & A\underline{x} \geq \underline{b} \\
& \underline{x} \geq \underline{0}
\end{aligned}
$$

[x,fval,exitflag,output,lambda]
=linprog(f,A,b)
MATLAB uses $A\,\underline{x} \leq \underline{b}$

# Portfolio optimization problem

- 4. Portfolio Optimization
  - J investment options (Stocks, T-bills, Corporate Bonds, S&P, Gold,..)
  - Have historical data on returns
    - $r_j(t) =$ Return on investment $j$ in time period $t, t = 1, 2, .., T$
    - $x_j =$ Fraction of portfolio to be invested in $j; \sum_{j=1}^{J} x_j = 1; x_j \geq 0,\ j = 1, 2, .., J$
    - Portfolio's historical returns with this alloction in time period $t$:

$$r(t) = \sum_{j=1}^{J} x_j r_j(t)$$

    - Portfolio's average return over t=1,2,..,T

$$\bar{r} = \frac{1}{T}\sum_{t=1}^{T} r(t) = \frac{1}{T}\sum_{t=1}^{T}\sum_{j=1}^{J} x_j r_j(t)$$

    - Portfolio's risk (some measure of variability around mean)

$$q = risk(\underline{x}) = \frac{1}{T}\sum_{t=1}^{T} | r(t) - \bar{r} | = \frac{1}{T}\sum_{t=1}^{T} \left| \left[ \sum_{j=1}^{J} x_j r_j(t) - \frac{1}{T}\sum_{s=1}^{T}\sum_{j=1}^{J} x_j r_j(s) \right] \right|$$

$$= \frac{1}{T}\sum_{t=1}^{T} \left| \left[ \sum_{j=1}^{J} x_j [r_j(t) - \frac{1}{T}\sum_{s=1}^{T} r_j(s)] \right] \right| = \frac{1}{T}\sum_{t=1}^{T} \left| \left[ \sum_{j=1}^{J} x_j [r_j(t) - \bar{r}_j] \right] \right|$$

where $\bar{r}_j = \frac{1}{T}\sum_{t=1}^{T} r_j(t)$ which is *precomputable*

# $L_1$-version of Markowitz problem

- Maximize average return subject to a constraint on risk
- Problem

$$\underset{\underline{x}\geq\underline{0}}{Max}\ \bar{r} = \frac{1}{T}\sum_{t=1}^{T}\sum_{j=1}^{J} x_j r_j(t)$$

$$s.t.\quad q = \frac{1}{T}\sum_{t=1}^{T}\left|\left[\sum_{j=1}^{J} x_j[r_j(t)-\bar{r}_j]\right]\right| \leq \mu;\ \mu = \text{risk aversion parameter}$$

$$\sum_{j=1}^{J} x_j = 1$$

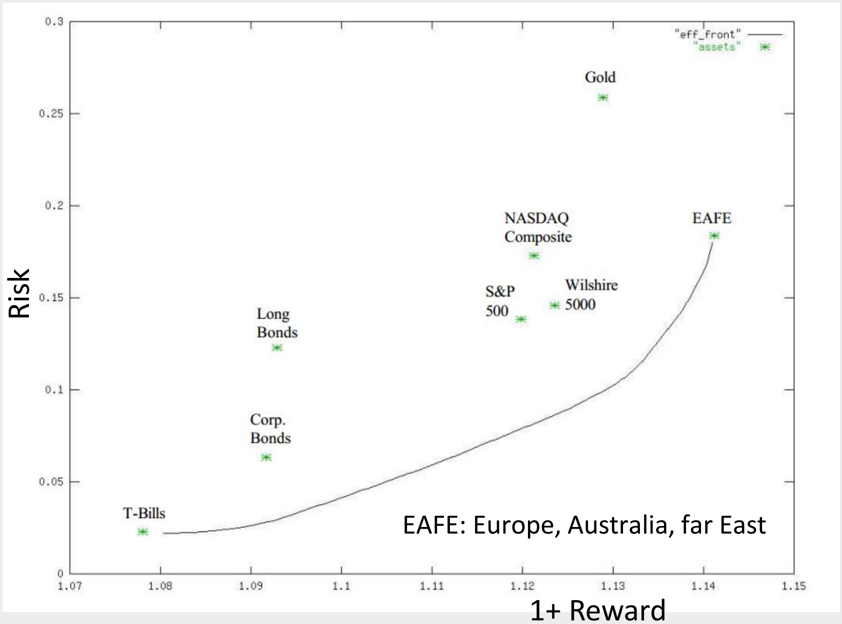
- LP formulation

$$\underset{\underline{x}\geq\underline{0},\underline{y}\geq\underline{0}}{Max}\ \bar{r} = \frac{1}{T}\sum_{t=1}^{T}\sum_{j=1}^{J} x_j r_j(t)$$

$$s.t.\quad -y_t \leq \left[\sum_{j=1}^{J} x_j[r_j(t)-\bar{r}_j]\right] \leq y_t;\ t=1,2,..,T$$

$$\frac{1}{T}\sum_{t=1}^{T} y_t \leq \mu$$

$$\sum_{j=1}^{J} x_j = 1$$




From Vanderbei

# Optimal Control

- 5. Optimal $L_1$ and $L_\infty$ control
  - Consider a linear time-invariant discrete-time system

    $\underline{x}_{k+1} = A\underline{x}_k + \underline{b}u_k$, $u_k \sim$ scalar for simplicity, $k = 0, 1, \ldots$

    $\underline{x}_k = A^k\underline{x}_0 + \sum_{l=0}^{k-1} A^{k-l-1}\underline{b}u_l$
  - Define terminal error: $e_N = \underline{x}_d - \underline{x}_N = \underline{x}_d - A^N\underline{x}_0 - \sum_{l=0}^{N-1} A^{N-l-1}\underline{b}u_l$
  - Given $x_0$, $\underline{x}_d$ and given the fact that $u_k$ is constrained by

    $u_{min} \leq u_k \leq u_{max}$, we can formulate various versions of LP

a)

$$\min \sum_{i=1}^{n} \left|e_{N_i}\right| = \sum_{i=1}^{n} \left| \left(\underline{x}_d - A^N \underline{x}_0\right)_i - \left(\sum_{l=0}^{N-1} A^{N-l-1} \underline{b} u_l\right)_i \right| \Rightarrow \text{1-norm of error}$$

$$= \sum_{i=1}^{n} \left|c_i + \underline{d}_i^T \underline{z}\right|, \underline{d}_i \sim N \text{ vector components}$$

$$-\left(A^{N-l-1}b\right)_i = d_{il}$$

$$\min \sum_{i=1}^{n} \left|c_i + \underline{d}_i^T \underline{z}\right|, \underline{z} = \left[u_0\, u_1 \cdots u_{N-1}\right]^T$$
$$s.t.\ u_{\min}\underline{1} \leq \underline{z} \leq u_{\max}\underline{1}$$

- Convert to standard form via:

  $v_i - u_i = c_i + \underline{d}_i^T\underline{z}$, $1 \leq i \leq n$

optimal solution:

$$\left.\begin{aligned} &\min \sum_{i=1}^{n} (v_i + u_i) \\ s.t.\ \ &u_{\min}\underline{1} \leq \underline{z} \leq u_{\max}\underline{1} \\ &v_i - u_i = c_i + \underline{d}_i^T \underline{z} \end{aligned}\right\} \begin{aligned} v_i^* &= \begin{cases} \underline{d}_i^T \underline{z} + c_i & \text{if } \underline{d}_i^T \underline{z} + c_i > 0 \\ 0 & \text{otherwise} \end{cases} \\ u_i^* &= \begin{cases} -\left(\underline{d}_i^T \underline{z} + c_i\right) & \text{if } \underline{d}_i^T \underline{z} + c_i < 0 \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

- Can also include constraints on state variables

# Properties of optimal control

b)

$$\min \max_{1\le i\le n} |e_{N_i}| = \min \max_{1\le i\le n} |c_i + \underline{d}_i^T \underline{z}| \Rightarrow \infty\text{-norm of error}$$

$$\text{define } v = \max_{1\le i\le n} |c_i + \underline{d}_i^T \underline{z}|$$

$$\min v$$

$$s.t.\ u_{\min}\underline{1} \le \underline{z} \le u_{\max}\underline{1}$$

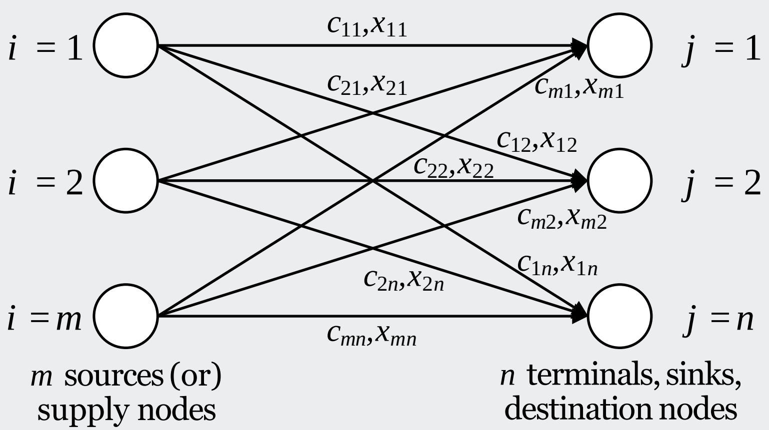
$$v + c_i + \underline{d}_i^T \underline{z} \ge 0$$

$$v - c_i - \underline{d}_i^T \underline{z} \ge 0$$

- Proof of equivalence for (a)
  - Suppose $v_i^*$, $u_i^*$, and $z^*$ are optimal solutions
  - $v_i^*$ & $u_i^*$ cannot simultaneously be non-zero
  - If they are, define $\hat{v}_i = v_i^* - u_i^*$ and $\hat{u}_i = 0 \Rightarrow$ feasible
    But, cost $\hat{v}_i + \hat{u}_i < v_i^* + u_i^*$ ....a contradiction
  
  $\Rightarrow$ only either of the two is nonzero
- Proof of equivalence for (b)
  - Let $z^*, v^*$ be optimal for the revised problem, but $z^*$ is not optimal for the original problem
  - Suppose $\hat{z}$ is the optimal solution of the original problem
  - Define $v = \max |c_i + \underline{d}_i^T \hat{z}| \Rightarrow$ feasible for the revised problem $\Rightarrow v < v_i^*$
    $\Rightarrow$ contradiction

# Transportation or Hitchcock Problem

- $m$ sources of a commodity or product and $n$ destinations
- Commodity amount to be shipped from source $i = a_i$; $1 \le i \le m$
- Commodity amount to be received at destination (sink, terminal node) $j = b_j$; $1 \le j \le n$
- Shipping cost from source $i$ to destination $j$ per unit commodity $= c_{ij}$ dollars/unit
- **Problem**: How much commodity should be shipped from source $i$ to destination $j$ to minimize transportation cost




$$\min \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$$

$$s.t. \quad \sum_{j=1}^{n} x_{ij} = a_i;\ \forall i = 1,2,\ldots,m$$

$$\sum_{i=1}^{m} x_{ij} = b_j;\ \forall j = 1,2,\ldots,n$$

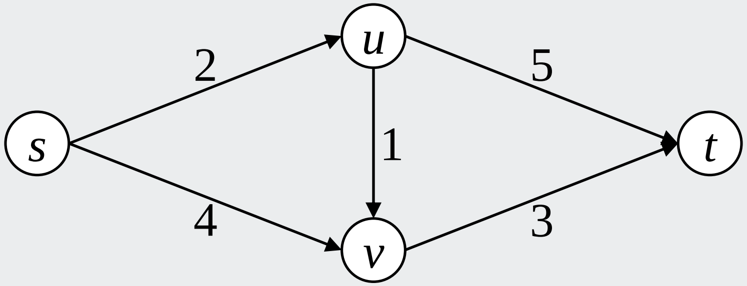
$$\text{also}: \sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

Conservation constraint

- Directed network or graph, $mn$ variables and $(m+n)$ constraints
- Note: arcs emanate from sources and terminate on sinks
- BIPARTITE GRAPHS ⇒ special LP problem ⇒ $a_i = b_i = 1$
  ⇒ Assignment problem or weighted bipartite matching problem

# Shortest Path Problem

- (For conceptual reasons only, but solved differently)




- $s$, $u$, $v$, $t$ are computers, edge lengths are costs of sending a message between them
- **Q**: what is the cheapest way to send a message from $s$ to $t$
- Shortest path $s \rightarrow u \rightarrow v \rightarrow t \Rightarrow x_{su} = x_{uv} = x_{vt} = 1$
- Shortest path length = 2 + 1 + 3 = 6
- Intuitively, $x_{sv} = x_{ut} = 0$ (i.e., no messages are sent from $s$ to $v$ and from $u$ to $t$)
- Let $x_{sv}$ be the fraction of messages sent from $s$ to $v$. Similarly, for arcs $(s, u)$, $(u, v)$, $(u, t)$, and $(v, t)$

# Shortest Path Problem

- Problem formulation

$$\begin{aligned} &\min\ 2x_{su} + 4x_{sv} + x_{uv} + 5x_{ut} + 3x_{vt} \\ &s.t.\quad x_{su}, x_{sv}, x_{uv}, x_{ut}, x_{vt} \ge 0 \\ &x_{su} - x_{uv} - x_{ut} = 0 \text{ (message not lost at } u) \\ &x_{sv} + x_{uv} - x_{vt} = 0 \text{ (message not lost at } v) \\ &\quad x_{ut} + x_{vt} = 1 \text{ (message received at } t) \end{aligned}$$

- Add all constraints

$\Rightarrow x_{su} + x_{sv} = 1$ which it must be!!

$\Rightarrow$ only 3 independent constraints (although 4 nodes)

- In matrix notation

$$A\underline{x} = \begin{bmatrix} 1 & 0 & -1 & -1 & 0 \\ 0 & 1 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_{su} \\ x_{sv} \\ x_{uv} \\ x_{ut} \\ x_{vt} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \underline{b}$$

- $n$ nodes $\Rightarrow n - 1$ independent equations

$\Rightarrow$ Similar to Kirchoff's laws

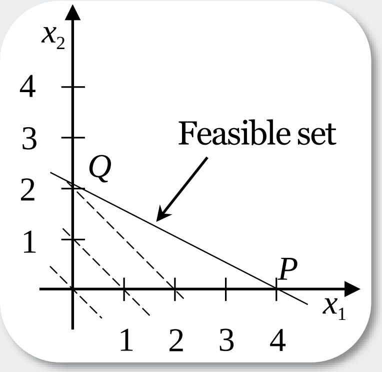
$\Rightarrow A$ is called the incidence matrix

$$\begin{aligned} &\min\ \underline{e}^T \underline{x} \\ &\text{s.t.}\quad A\underline{x} = \underline{b} \end{aligned}$$

- Note: $\underline{b}$ is a special vector *s.t.* $A\underline{x} = \underline{b}$, $\underline{x} \ge 0$. $A$ is a **unimodular** matrix and so are all invertible submatrices $\tilde{A}$ of $A \Rightarrow det\ \tilde{A} = 1$ or $-1$.

  $\Rightarrow$ Inverses will have integer elements $\Rightarrow$ Solutions are integers if $\underline{b}$ is integer.

# Standard Linear Program




- **Let us return to the solution of SLP**

$\Rightarrow$ min $\underline{c}^T \underline{x}$ s.t. $A\underline{x} = \underline{b}$, $\underline{x} \geq 0$

$A$ is an $m \times n$ matrix of rank $m$

- **Example**

$$\min\ x_1 + x_2$$
$$s.t.\ x_1 + 2x_2 = 4$$

- First contact of $x_1 + x_2 = a$ occurs at $a = 2$, $x_1 = 0$, $x_2 = 2$

$\Rightarrow$ optimal solution: $x_2 = 2$, $x_1 = 0$

- In general, the optimal solution $\underline{x}^*$ is such that $(n - m)$ of its components are zero. If we knew which of the $n - m$ components are zero, we can immediately compute the optimal solution (i.e., the remaining $m$ nonzero components) from $A\underline{x} = \underline{b}$. Since we don't know the zeros *a priori*, the chief task of every algorithm is to discover where they belong.

- Need to look at only extreme points of the feasible set.

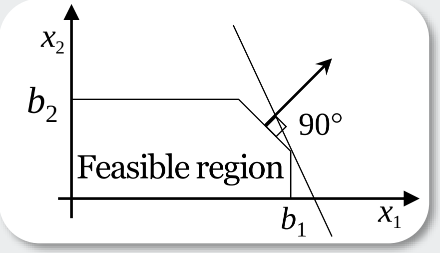
# 2-Phase Simplex Algorithm

- How does **Simplex algorithm** work?
  - **Phase 1:** Find a vector $\underline{x}$ that has $(n - m)$ zero components, with $A\underline{x} = \underline{b}$ and $\underline{x} \geq \underline{0}$. This is a feasible $\underline{x}$, not necessarily optimal
  - **Phase 2:** Allow one of the zero components to become positive and force one of the positive components to become zero
    - **Q**: How to pick “entering” and “leaving” variables
    - **A**: Cost $\underline{c}^T \underline{x} \downarrow$ and $A\underline{x} = \underline{b}$, $\underline{x} \geq 0$ must be satisfied
  - Inequality constraints:
    - $x_1$ : invest in stock
    - $x_2$ : invest in real estate
    - $0 \leq x_1 \leq b_1;\ 0 \leq x_2 \leq b_2;\ 0 \leq x_1 + x_2 \leq b_3$

    $\Rightarrow$ can also look at it as a 5 dimensional problem with slacks

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} \underline{x} \leq \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Assume $b_3 > b_1$ and $b_3 > b_2$

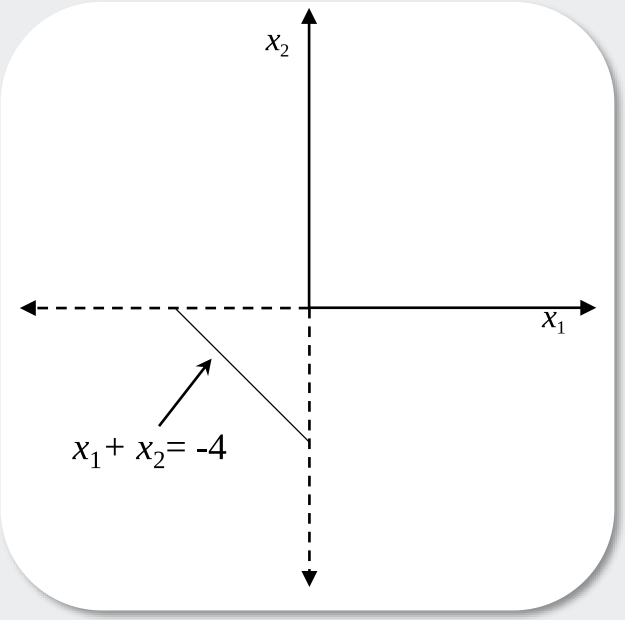



$$\begin{aligned} &\min \ -4x_1 - 2x_2 \\ &s.t.\ A\underline{x} \leq \underline{b} \\ &\Rightarrow x_1^* = b_1;\ x_2^* = b_3 - b_1 \\ &\text{max. profit}:\ 2b_1 + 2b_3 \end{aligned}$$

- In $n$ dimensions
  - $\underline{a}_i^T \underline{x} = b_i$ define hyperplanes
  - $\underline{a}_i^T \underline{x} \leq b_i$ define half spaces
  - $\underline{x} \geq 0$ positive cone

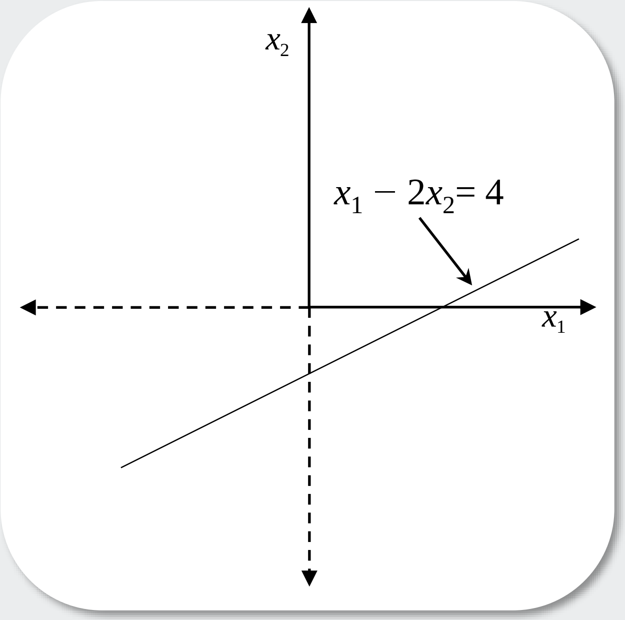
# Infeasibility

- Feasible set is a convex polytope. If bounded, a convex polyhedron. Need to consider only extreme points of this set.
- Some other nuances
  - An LP may not have a solution
  - e.g., $$\begin{aligned} \min\ & x_1 + x_2 \\ s.t.\ & x_1 + x_2 = -4 \\ & x_1, x_2 \geq 0 \end{aligned}$$ $\Rightarrow$ Feasible set is empty $\Rightarrow$ inconsistent constraints



# Unboundedness

- An LP may have an unbounded solution
  - e.g., $\min \; -(x_1 + x_2)$
    $s.t. \; x_1 - 2x_2 \geq 4$
    $\Rightarrow \text{opt. } x_1, x_2 = (\infty, \infty)$
- So, an algorithm must decide
  - Whether there exists an optimal solution
  - If it does, find the corner where optimum occurs



# Basic Feasible Solution (BFS)

- Assume rank($A$) = $m$, then we can partition $A$=[$B$ $N$], where $B$ has $m$ linearly independent columns
- Assume first $m$ columns for convenience

$$\begin{bmatrix} B & N \end{bmatrix} \begin{bmatrix} \underline{x}_B \\ \vdots \\ \underline{x}_N \end{bmatrix} = \underline{b}, \quad \underline{x}_B \in R^m; \; \underline{x}_N \in R^{n-m}$$

$$\sum_{i=1}^{m} \underline{a}_i x_i + \sum_{i=m+1}^{n} \underline{a}_i x_i = \underline{b}$$

$$\text{or}, B\underline{x}_B + N\underline{x}_N = \underline{b} \Rightarrow B\underline{x}_B = \underline{b} - N\underline{x}_N$$

- Since rank($B$) = $m$ ⇒ $B^{-1}$ exists
- $\underline{x}_B = B^{-1}\underline{b} - B^{-1}N\underline{x}_N$
    - $\underline{x}_B$ = vector of **basic** variables
    - $\underline{x}_N$ = vector of **non-basic** variables
- **Basic solution:** set non-basics to their lower bound (i.e., $\underline{x}_N = \underline{0}$)

⇒ $\underline{x}_B = B^{-1}\underline{b}$; $B$ is called the **basis matrix**

- **Basic feasible solution (bfs):** $\underline{x}_B \geq \underline{0}$ ⇒ $\underline{x}$ is feasible

$$\underline{x} = [\underline{x}_B \geq \underline{0} \;\; \underline{x}_N = \underline{0}\;]^T$$

# Fundamental Theorem of LP

- <u>Theorem</u>
  a) Existence of a feasible $\underline{x} \Rightarrow$ existence of $\underline{x}_B \geq \underline{0}$ , a basic feasible solution
  b) Existence of a optimal $\underline{x}^* \Rightarrow$ existence of $\underline{x}_B{}^* \geq \underline{0}$ , an optimal basic feasible solution
  - <u>Proof of *a*:</u>
    - Feasible $\underline{x} \Rightarrow \sum_{i=1}^{n} \underline{a}_i x_i = \underline{b}$
    - Suppose $x_1, x_2, \ldots, x_p > 0$ and the rest are zero

$$\Rightarrow \sum_{i=1}^{p} \underline{a}_i x_i = \underline{b}$$

    - **Case 1**: linearly independent $(a_1, a_2, \ldots, a_p) \Rightarrow p \leq m$

      If $p = m$, $\underline{x}_p = \underline{x}_B$, where $\underline{x}_B = B^{-1}\underline{b}$

      If $p < m$, can find $(m - p)$ dependent vectors

      Set $x_i = 0$, $i = p + 1, \ldots, m$

      $\Rightarrow \underline{x}_B$ is (degenerate) basic feasible

# Feasible ⇒ Basic Feasible Solution (BFS)

- **Case 2**: $(a_1, a_2, \ldots, a_p)$ are linearly dependent

  ⇒ Can find $y_1, y_2, \ldots, y_p$, such that

$$\sum_{i=1}^{p} \underline{a}_i y_i = \underline{0}$$

Assume, without loss of generality, at least one $y_i > 0$

$$\sum_{i=1}^{p} \underline{a}_i \left(x_i - \varepsilon y_i\right) = \underline{b}, \forall \varepsilon$$

Assume $\varepsilon \geq 0$ without loss of generality

Note that as $\varepsilon \uparrow$:

$$x_i - \varepsilon y_i \uparrow \text{ if } y_i < 0$$

We have

$$x_i \text{ if } y_i = 0$$

$$x_i - \varepsilon y_i \downarrow \text{ if } y_i > 0$$

Set $\varepsilon = \min \{x_i/y_i : y_i > 0\}$

- For this $\varepsilon$, we have an $\underline{x}$ with $(p - 1)$ positive values
- The equation for $\varepsilon$ is simply that for the simplex step
- Continue this process until all vectors are independent, then **case 1** applies

# Optimal Solution ⇒ Optimal BFS

- Proof of $b$:
  - **Case 1**: linearly independent $(a_1, a_2, \ldots, a_p) \Rightarrow p \leq m$

    If $p = m$, $\underline{x}_p = \underline{x}_B^*$, where $\underline{x}_B^* = B^{-1}\underline{b}$

    If $p < m$, can find $(m - p)$ dependent vectors

    Set $x_i = 0$, $i = p + 1, \ldots, m$

    $\Rightarrow \underline{x}_B^*$ is (degenerate) optimal basic feasible
  - **Case 2**: $(a_1, a_2, \ldots, a_p)$ are linearly dependent

    $\Rightarrow$ Can find $y_1, y_2, \ldots, y_p$, such that

    $$\sum_{i=1}^{p} \underline{a}_i y_i = \underline{0}$$

    Assume, without loss of generality, at least one $y_i > 0$

    $$\sum_{i=1}^{p} \underline{a}_i \left( x_i^* - \varepsilon y_i \right) = \underline{b}, \forall \varepsilon$$

    Assume $\varepsilon \geq 0$ without loss of generality

    Note that as $\varepsilon \uparrow$:

    $x_i - \varepsilon y_i \uparrow$ if $y_i < 0 \Rightarrow$ feasibility is maintained

    We have $x_i$ if $y_i = 0 \Rightarrow$ feasibility is maintained

# Finite search space of LP

$$x_i - \varepsilon y_i \downarrow \text{ if } y_i > 0 \Rightarrow \text{feasibility for some } \varepsilon$$

$$\text{Set } \varepsilon = \min \{x_i/y_i : y_i > 0\}$$

- ❖ For this $\varepsilon$, we have an $\underline{x}^*$ with $(p-1)$ positive values
- ❖ But, what is the cost at $(\underline{x}^* - \varepsilon \underline{y})$?
- ❖ The cost is $\underline{c}^T(\underline{x}^* - \varepsilon \underline{y})$
- ❖ Since $\underline{x}^*$ is optimal, $\underline{c}^T\underline{y} = 0$. Otherwise, we can find a small $\varepsilon$ such that $\underline{c}^T(\underline{x}^* - \varepsilon \underline{y}) < \underline{c}^T\underline{x}^*$
- ❖ A solution with $(p-1)$ positive values is also optimal!
- ❖ Continue this process until all vectors are independent, then **case 1** applies

- What this theorem says is that we need to find $(n - m)$ zero variables among $n$ nonnegative variables

$$\binom{n}{n-m} = \binom{n}{m} = \frac{n!}{(n-m)!m!}$$

$\Rightarrow$LP is a finite search problem (fortunately, we never have to solve it that way!)

# Two views of convex polyhedron

- Basic feasible solutions of LP ≡ extreme (corner) points of a convex polytope
- Recall from lecture 1:
  - $A\underline{x} = \underline{b}$ is the intersection of $m$ hyperplanes in $R^m$
  - $\underline{x} \geq 0 \Rightarrow$ convex cone in $R^n$
  - Feasible set is a convex polytope; if bounded, it is called a convex polyhedron

**Convex polyhedron**

→ **Intersection of a finite number of half spaces**

→ **Convex hull of a finite number of extreme points**

$\Rightarrow$ Any point $\underline{x} = \Sigma_i \alpha_i \underline{x}_i$; $\Sigma_i \alpha_i = 1$; $\alpha_i \geq 0$

$\{\underline{x}_i\}$ are extreme (corner) points of the feasible set

# BFS ≡ corner points of convex polyhedron

- <u>Theorem</u>: **extreme points of convex polytope (polyhedron)** $K = \{\underline{x} : A\underline{x} = \underline{b},\ \underline{x} \geq \underline{0}\} \Leftrightarrow$ **basic feasible solutions of LP**
  - <u>Proof of ⇐ part</u>:
    - ❖ Suppose we have a bfs $\underline{x} = [\underline{x}_B\ \ \underline{0}\ ]^T \Rightarrow A\underline{x} = \underline{b}$
    - ❖ Suppose $\underline{x}$ is not an extreme point $\Rightarrow \underline{x} = \alpha\underline{y} + (1 - \alpha)\underline{z}$,
      $0 < \alpha < 1, \Rightarrow A\underline{x} = \underline{b}$
      $A\underline{z} = \underline{b}$ and $\underline{y}, \underline{z}$ are bfs
    - ❖ Suppose $\underline{y}, \underline{z} \geq \underline{0}$, and $\underline{x}_N = \underline{0} \Rightarrow \underline{y}_N = \underline{z}_N = \underline{0}$, $A\underline{y} = A\underline{z} = \underline{b}$
    - ❖ Since $m$ columns of $A$ are independent $\Rightarrow \underline{x} = \underline{y} = \underline{z}$
    - $\Rightarrow$ a contradiction $\Rightarrow \underline{x}$ is an extreme point of $K$
  - <u>Proof of ⇒ part</u>:
    - ❖ Suppose we have an extreme point of $\underline{x}$ of $K$ with components:
      $x_1, x_2, \ldots, x_p > 0$
    - ❖ To show that $\underline{x}$ is a bfs, we must show that $\underline{a}_1, \underline{a}_2, \ldots, \underline{a}_p$ are linearly independent
      Suppose $\underline{a}_1, \underline{a}_2, \ldots, \underline{a}_p$ are linearly dependent

$$\Rightarrow \sum_{i=1}^{p} y_i \underline{a}_i = \underline{0} \Rightarrow A\underline{y} = \underline{0}$$

# Development of Simplex Algorithm

❖ Since $\underline{x} \geq \underline{0}$, we can pick $\varepsilon$ such that

$$(\underline{x} + \varepsilon\underline{y}) \geq \underline{0} \text{ and } (\underline{x} - \varepsilon\underline{y}) \geq \underline{0}$$

then $\underline{x} = ½(\underline{x} + \varepsilon\underline{y}) + ½(\underline{x} - \varepsilon\underline{y})$ … contradiction

$\Rightarrow \underline{x}$ is a bfs (degenerate if $p < m$)

- Simplex: partition $\underline{c}$ as follows

$$\underline{c} = \begin{bmatrix} \underline{c}_B \\ \underline{c}_N \end{bmatrix} \quad \text{then} \quad \begin{aligned} f &= \underline{c}^T \underline{x} = \underline{c}_B^T \underline{x}_B + \underline{c}_N^T \underline{x}_N ; \underline{x}_B = B^{-1}\underline{b} - B^{-1}N\underline{x}_N \\ f &= \underline{c}_B^T \left( B^{-1}\underline{b} - B^{-1}N\underline{x}_N \right) + \underline{c}_N^T \underline{x}_N = \underline{c}_B^T B^{-1}\underline{b} + \left( \underline{c}_N^T - \underline{c}_B^T B^{-1} N \right) \underline{x}_N \end{aligned}$$

| <u>original</u> | | <u>transformed problem</u> |
|---|---|---|
| $\min f = \underline{c}^T \underline{x}$ | $\Rightarrow$ | $\underline{c}_B^T B^{-1}\underline{b} + \left( \underline{c}_N^T - \underline{c}_B^T B^{-1} N \right) \underline{x}_N$ |
| $s.t.\ A\underline{x} = \underline{b}$ | $\Rightarrow$ | $\underline{x}_B = B^{-1}\underline{b} - B^{-1}N\underline{x}_N$ |
| $\underline{x} \geq \underline{0}$ | | $\underline{x} \geq \underline{0}$ |

- ○ Let $\underline{\beta} = B^{-1}\underline{b}$; $\underline{\lambda}^T = \underline{c}_B{}^T B^{-1}$
- ○ $B\underline{\beta} = \underline{b}$; $B^T\underline{\lambda} = \underline{c}_B$

# Basic and non-basic aspects of simplex

- Transformed problem is:

$$\min f = \underline{c}_B^T \underline{\beta} + \left(\underline{c}_N^T - \underline{\lambda}^T N\right) \underline{x}_N$$
$$s.t.\ \underline{x}_B = \underline{\beta} - B^{-1} N \underline{x}_N$$

$\Rightarrow f = \underline{c}_B{}^T \underline{\beta} + p_1 x_{N_1} + \ldots + p_{n-m} x_{N_{n-m}}$

where $p_j = c_{N_j} - \underline{\lambda}^T \underline{a}_j$; $\underline{a}_j$ = column $j$ of $N$

$p_j$ = *reduced cost of j*

$\Rightarrow$ also $\underline{x}_B = \underline{\beta} - \underline{\alpha}_1 x_{N_1} - \underline{\alpha}_2 x_{N_2} - \ldots - \underline{\alpha}_{n-m} x_{N_{n-m}}$

where $\underline{\alpha}_j = B^{-1} \underline{a}_j$; $\underline{a}_j$ = column $j$ of $N$

<u>Note</u>: when $\underline{x}_N = \underline{0}$, $\underline{x}_B = \underline{\beta}$ and $f = \underline{c}_B{}^T \underline{\beta} = f_0$

# Optimality Conditions

- $\underline{p}^T = \underline{c}_N{}^T - \underline{\lambda}^T \underline{N}$ is called the vector of **reduced costs**
- This vector indicates how $f$ changes as $\underline{c}_N$ changes
- What is $p_j$, the $j^{th}$ component of the $\underline{p}$ vector?

$$p_j = c_j - \left(\underline{\lambda}^T \underline{N}\right)_j = c_j - \underline{\lambda}^T \underline{a}_j$$

  - Note: need only column $\underline{a}_j$ to compute $p_j$
- If $\underline{x}_B = \underline{\beta} \geq 0$ and $\underline{x}_N = \underline{0}$, we need $p_j \geq 0$ for optimality $\forall\, j \Rightarrow$ it doesn't pay to increase $\underline{x}_N$
- So,
  - Feasibility: $\beta_i \geq 0,\ i = 1, 2, \ldots, m$
  - Optimality: $p_j \geq 0,\ j = 1, 2, \ldots, n - m$

# Illustration of Optimality Conditions

- Example:

$$\min\ f = 30 + 4x_4 + 5x_5 + 3x_6 + 4x_7$$
$$s.t.\ x_1 = 5 + 3x_4 - 3x_5 + x_6 - x_7$$
$$x_2 = 6 - 7x_4 + 2x_5 - 2x_6 - 2x_7$$
$$x_3 = 7 - x_4 - 3x_5 + 3x_6 + 3x_7$$

$$\underline{\beta} = \begin{bmatrix} 5 \\ 6 \\ 7 \end{bmatrix}; \underline{c}_B^T \underline{\beta} = 30; \underline{x}_N = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix}; \underline{x}_B = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$$-B^{-1}N = \begin{bmatrix} 3 & -1 & 1 & -1 \\ -7 & 2 & -2 & -2 \\ 1 & -3 & 3 & -3 \end{bmatrix};$$

$$\underline{p}^T = \underline{c}_N^T - \underline{\lambda}^T N = \underline{c}_N^T - \underline{c}_B^T B^{-1} N = \begin{bmatrix} 4 & 5 & 3 & 4 \end{bmatrix} > \underline{0}$$

$$\Rightarrow \begin{bmatrix} 5 & 6 & 7 & 0 & 0 & 0 & 0 \end{bmatrix} \text{is optimal}$$

# Proof of Optimality Conditions

- <u>Proof of sufficiency</u>:
  - Since $p_j \geq 0$, an increase in $x_{Nj}$ results in an increase in cost. Thus, if we have a basic feasible solution such that $p_j \geq 0$, then it is optimal
- <u>Proof of necessity</u>:
  - Suppose $p_j < 0$, for some $j = 1, 2, \ldots, n - m$
  - Two cases can occur
    1. $\underline{\alpha}_j = B^{-1}\underline{a}_j \leq \underline{0} \Rightarrow x_{N_j} \geq 0$ can be increased to any positive value and $\underline{x}_B$ remains feasible

       ⇒ **set of solutions to $A\underline{x} = \underline{b}$, $\underline{x} \geq \underline{0}$ is unbounded and $f$ can be made an arbitrarily large negative number ($-\infty$)**

       ⇒ **This is the way to detect unboundedness**

       ⇒ in practice, what it means is that some constraints were over-looked!!
    2. $p_j < 0$ and $\alpha_j > 0$ for at least one $k = 1, 2, \ldots, n - m$

       ⇒ $x_{N_k}$ can increase from zero to reduce cost

       ⇒ $\underline{x}_B$ is not optimal… contradiction

       ⇒ **This is the way to go from corner to another corner**

# Picking the Entering Variable and Step Size

- When $p_j < 0$, we can increase $x_{N_j}$ from zero to reduce cost
- Two questions:
  - If several $p_j < 0$, which one should we pick to enter the basis?
  - How far to go? ⇒ Which one should leave the basis?
- Which one to pick?
  - Most widely used: pick $k = \arg\min p_j$

  ⇒ "**steepest coordinate descent**" or "**nonbasic gradient method**"
  - All variable gradient method

$$k = \arg\min_j \frac{p_j}{\sqrt{1 + \sum_{k=1}^{m} \alpha_{ij}^2}}$$

  - $k = \min \{j: p_j < 0\}$ (i.e., choose the lowest numbered column that gives $p_j < 0 \Rightarrow$ 1st $j$ with negative $p_j$) … **Bland's method avoids cycling**
- How far to go?
  - Suppose $k$ is the entering variable
  - Recall $\underline{x}_B = \underline{\beta} - \underline{\alpha}_k x_{N_k}$
    - ❖ As $x_{N_k}$ increases, $\underline{x}_B$ changes
    - ❖ If $\alpha_{ik} > 0$, then $x_{B_i}$ decreases and goes through zero

# Updating the Basis

- **Must not go below zero, since this would ruin feasibility**
  - So, increase $x_{N_k}$ from zero until one of the basic variables goes to zero

    $x_{N_k} = \theta = \min\{\beta/\alpha_{ik} : \alpha_{ik} > 0\}$

    $\Rightarrow$ if $i = l$ is the minimizing index, then $x_{N_k} = \beta_l/\alpha_{ik}$ and $x_{B_l} = 0 \Rightarrow$ the basic variable $l$ will leave the basis
  - **If more than one hits zero at the same time, pick one arbitrarily $\Rightarrow$ degenerate basic feasible solution**
- What happens to $B$?
  - $x_{N_k}$ goes from zero to $\beta_l/\alpha_{ik}$ and $x_{B_l}$ goes from $\beta_l$ to zero

    $\Rightarrow$ replace $l^{\text{th}}$ column of original $B$ with $k^{\text{th}}$ column of $N$

$$\overline{B} = B - B\underline{e}_l\underline{e}_l^T + \underline{a}_k\underline{e}_l^T = B\left(I - \underline{e}_l\underline{e}_l^T + B^{-1}\underline{a}_k\underline{e}_l^T\right)$$
$$= \begin{bmatrix}\underline{a}_1 & \underline{a}_2 & \cdots & \underline{a}_{l-1} & \underline{a}_k & \underline{a}_{l+1} & \cdots & \underline{a}_m\end{bmatrix}$$

  - We will have more to say about this in lecture 3

# One Iteration of Revised Simplex Algorithm

- **Step 1:** Given the basis $B$ such that $\underline{x}_B = B^{-1}\underline{b} \geq 0$
- **Step 2:** Solve $B^T\underline{\lambda} = \underline{c}_B$ for the vector of simplex multipliers $\underline{\lambda}$
- **Step 3:** Select a column $\underline{a}_k$ of $N$ such that $p_k = c_{N_k} - \underline{\lambda}^T\underline{a}_k < 0$

  Note: we may select the $\underline{a}_k$ which gives the largest negative value of $p_k$ or the first $k$ with negative $p_k$

  if $\underline{p}^T = \underline{c}_N - \underline{\lambda}^T N \geq 0$, stop $\Rightarrow$ current solution is optimal
- **Step 4:** Solve $\underline{\alpha} : B\underline{\alpha} = \underline{a}_k$
- **Step 5:** Find $\theta = x_{B_l}/\alpha_l = \min x_{B_i}/\alpha_i,\ 1 \leq i \leq m,\ \alpha_i > 0$
  - **If none of the $\{\alpha_i\}$ is positive**, then the set of solutions to $A\underline{x} = \underline{b}$, $\underline{x} \geq \underline{0}$ is **unbounded** and the cost $f$ can be made an arbitrarily large negative number

  $\Rightarrow$ Terminate computation, since an unbounded solution
- **Step 6:** Update the basic solution, $\bar{x}_i = x_i - \theta\alpha_i, i \neq k; \bar{x}_k = \theta$
- **Step 7:** Update the basis and return to **Step 1**

# Remarks

- Typically, the # of simplex iterations, $k \in \{2m, 4m\}$
- Computation time is $\propto k$
- Round-off errors
  - Inability to store numbers and perform computations exactly gives rise to round-off errors
  - Rounding error accumulates with floating point operations (flops)
  - To reduce round-off errors:
    - Balance matrix $A \Rightarrow$ try to make $\|A\|_1 = \|A\|_\infty$
    - Monitor residuals: $\|A\underline{x} - \underline{b}\|_\infty$ and $\|\underline{c}_B - B^T\underline{\lambda}\|_\infty$
    - Use error tolerances,
      - $p_j > -10^{-5} \Rightarrow$ optimal
      - $a_{ij} < 10^{-10} \Rightarrow a_{ij} = 0$
      - If $x_{N_k} > 10^{-8} \Rightarrow$ reinvert basis
      - If $\|A\underline{x} - \underline{b}\|_\infty$ or $\|\underline{c}_B - B^T\underline{\lambda}\|_\infty > 10^{-6} \Rightarrow$ reinvert basis

# How to get initial feasible solution – Phase I of LP

- Method 1
  - An initial basic feasible solution can be obtained by solving the following LP problem

$$\min \sum_{i=1}^{m} \hat{y}_i$$
$$s.t.\; A\underline{x} + I\,\hat{\underline{y}} = \underline{b}, \quad \hat{\underline{y}} \sim \text{artificial variable}$$
$$\underline{x}, \hat{\underline{y}} \geq \underline{0}$$

  - If we can find an optimal solution $\ni \sum_{i=1}^{m} \hat{y}_i = 0$, then we have $\underline{x}_B$
  - **If $\sum_{i=1}^{m} \hat{y}_i > 0$, then there is no feasible solution to $A\underline{x} = \underline{b}$, $\underline{x} \geq \underline{0}$ $\Rightarrow$ an infeasible problem**
  - Solve via the revised simplex starting with $\underline{x} = \underline{0}$, $\hat{y}_i = \underline{b}$ and $B = I_m$
  - Note: we have assumed $\underline{b} \geq \underline{0}$ . Is it OK? Yes!!
    - ❖ If $b_i < 0$, scale the corresponding equation by $-1$
- Method 2
  - Another approach is to combine both phases I and II by solving:

$$\min_{\underline{x},\underline{y}} \underline{c}^T \underline{x} + M\underline{e}^T \underline{y};$$
$$s.t.\; A\underline{x} + \underline{y} = \underline{b}$$
$$\underline{x}, \underline{y} \geq \underline{0}$$

$M$ is a large number > 100 $||\underline{c}||_\infty$

  - This is called the "big-M" method

# Example: Detecting unboundedness (Phase II)

- Consider

$$\max\ x_1 + 4x_2 + x_3$$
$$s.t.\ 2x_1 - 2x_2 + x_3 = 4$$
$$x_1 - x_3 = 1$$
$$x_2 \geq 0; x_3 \geq 0$$

$$\max\ 4x_2 + 2x_3$$
$$s.t. -2x_2 + 3x_3 = 2$$
$$x_2 \geq 0; x_3 \geq 0$$

$$\max\ 8x_3 \quad \text{or} \quad \max\ \frac{16}{3}x_2$$
$$s.t.\ x_3 \geq 0 \quad \text{s.t.}\ \ x_2 \geq 0$$

*Unbounded*

$$Phase\ I: \min\ y_1 + y_2 = [\underline{c}_1\ \underline{c}_2]\begin{bmatrix}\underline{x}\\ \underline{y}\end{bmatrix}; \underline{c}_1^T = [0\ 0\ 0]; \underline{c}_2^T = [1 1]$$

$$A = \begin{bmatrix} 2 & -2 & 1 & 1 & 0 \\ 1 & 0 & -1 & 0 & 1 \end{bmatrix}$$

$$Iteration 1: B = I = B^{-1}; \underline{x}_B = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \end{bmatrix}; \underline{\lambda}^T = [1 1]B^{-1} = [1 1]; \cos t = 5$$

$$\text{Re}\,duced\ \cos ts: p_1 = 0 - \underline{\lambda}^T \underline{a}_1 = -3; p_2 = 0 - \underline{\lambda}^T \underline{a}_2 = 2;$$
$$p_3 = 0 - \underline{\lambda}^T \underline{a}_3 = 0; p_4 = 1 - \underline{\lambda}^T \underline{a}_4 = 0; p_5 = 1 - \underline{\lambda}^T \underline{a}_5 = 0$$

$$x_1 \text{ comes into basis} \Rightarrow \underline{\alpha} = B^{-1}\underline{a}_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \Rightarrow \theta = \min\{4/2\ 1/1\} = 1 \Rightarrow y_2 \text{ should go out}$$

Iteration 2:

$$B = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} \Rightarrow B^{-1} = \begin{bmatrix} 1 & -2 \\ 0 & 1 \end{bmatrix} \Rightarrow \underline{\lambda}^T = [1\ 0]B^{-1} = [1 -2]; \underline{x}_B = \begin{bmatrix} y_1 \\ x_1 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}; \cos t = 2$$

$$\text{Re}\,duced\ \cos ts: p_1 = 0 - \underline{\lambda}^T \underline{a}_1 = 0; p_2 = 0 - \underline{\lambda}^T \underline{a}_2 = 2;$$
$$p_3 = 0 - \underline{\lambda}^T \underline{a}_3 = -3; p_4 = 1 - \underline{\lambda}^T \underline{a}_4 = 0; p_5 = 1 - \underline{\lambda}^T \underline{a}_5 = 3$$

$$x_3 \text{ comes into basis} \Rightarrow \underline{\alpha} = B^{-1}\underline{a}_3 = \begin{bmatrix} 3 \\ -1 \end{bmatrix} \Rightarrow \theta = 2/3 \Rightarrow y_1 \text{ should go out}$$

Iteration 3:

$$B = \begin{bmatrix} 1 & 2 \\ -1 & 1 \end{bmatrix} \Rightarrow B^{-1} = \begin{bmatrix} 1/3 & -2/3 \\ 1/3 & 1/3 \end{bmatrix} \Rightarrow \underline{\lambda}^T = [0\ 0]B^{-1} = [0\ 0]; \underline{x}_B = \begin{bmatrix} x_3 \\ x_1 \end{bmatrix} = \begin{bmatrix} 2/3 \\ 5/3 \end{bmatrix}; \cos t = 0$$

$$\text{Re}\,duced\ \cos ts: p_1 = 0 - \underline{\lambda}^T \underline{a}_1 = 0; p_2 = 0 - \underline{\lambda}^T \underline{a}_2 = 0;$$
$$p_3 = 0 - \underline{\lambda}^T \underline{a}_3 = 0; p_4 = 1 - \underline{\lambda}^T \underline{a}_4 = 1; p_5 = 1 - \underline{\lambda}^T \underline{a}_5 = 1$$

$$\Rightarrow optimal\ \ and\ \ \cos t = 0 \Rightarrow \underline{x}_B = \begin{bmatrix} x_3 \\ x_1 \end{bmatrix} = \begin{bmatrix} 2/3 \\ 5/3 \end{bmatrix} \text{ is feasible; You can get this by setting } x_2 = 0.$$

# Example: $\underline{\alpha}$ vector has all negative (non-positive) elements

- Consider

$$\max\ x_1 + 4x_2 + x_3$$
$$s.t.\ 2x_1 - 2x_2 + x_3 = 4$$
$$x_1 - x_3 = 1$$
$$x_2 \geq 0;\ x_3 \geq 0$$

⬇

$$\max\ 4x_2 + 2x_3$$
$$s.t. -2x_2 + 3x_3 = 2$$
$$x_2 \geq 0;\ x_3 \geq 0$$

⬇

$$\max\ 8x_3 \quad \text{or} \quad \max\ \frac{16}{3}x_2$$
$$s.t.\ x_3 \geq 0 \quad \text{s.t.}\ x_2 \geq 0$$

*Unbounded*

Let us continue with Phase II

$$\min -x_1 - 4x_2 - x_3$$

$$\underline{x}_B = \begin{bmatrix} x_3 \\ x_1 \end{bmatrix} = \begin{bmatrix} 2/3 \\ 5/3 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 2 \\ -1 & 1 \end{bmatrix}; B^{-1} = \begin{bmatrix} 1/3 & -2/3 \\ 1/3 & 1/3 \end{bmatrix}$$

$$\underline{\lambda}^T = \underline{c}_B^T B^{-1} = [-1 -1]\begin{bmatrix} 1/3 & -2/3 \\ 1/3 & 1/3 \end{bmatrix} = [-2/3\ 1/3]$$

Reduced costs:

$$p_1 = c_1 - \underline{\lambda}^T \underline{a}_1 = -1 - [-2/3\ 1/3]\begin{bmatrix} 2 \\ 1 \end{bmatrix} = 0$$

$$p_2 = c_2 - \underline{\lambda}^T \underline{a}_2 = -4 - [-2/3\ 1/3]\begin{bmatrix} -2 \\ 0 \end{bmatrix} = -8/3$$

$$p_3 = c_3 - \underline{\lambda}^T \underline{a}_3 = -1 - [-2/3\ 1/3]\begin{bmatrix} 1 \\ -1 \end{bmatrix} = 0$$

*Bring* $\underline{a}_2$ *into the basis*

$$\underline{\alpha} = B^{-1}\underline{a}_2 = \begin{bmatrix} 1/3 & -2/3 \\ 1/3 & 1/3 \end{bmatrix}\begin{bmatrix} -2 \\ 0 \end{bmatrix} = \begin{bmatrix} -2/3 \\ -2/3 \end{bmatrix} < \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$\Rightarrow$ *unbounded* because $x_1$ and $x_3$ can be increased to $\infty$.

All you need to do, for example, is to put an upper bound on $x_2$

# Example: Regular termination

$$\min -60x_1 - 30x_2 - 20x_3$$
$$\begin{aligned} s.t.\ 8x_1 + 6x_2 + x_3 + s_1 &= 48 \\ 4x_1 + 2x_2 + 1.5x_3 + s_2 &= 20 \\ 2x_1 + 1.5x_2 + 0.5x_3 + s_3 &= 8 \end{aligned}$$

- *Iteration 1*
  - Initially $B = I$,

$$\Rightarrow x_B = (s_1 \quad s_2 \quad s_3)^T = (48, 20, 8)^T$$
$$\Rightarrow x_N = (x_1 \quad x_2 \quad x_3)^T$$
$$\lambda^T = [0 \quad 0 \quad 0]$$
$$p_1 = c_1 - \underline{\lambda}^T \underline{a}_1 = -60 - \lambda^T \begin{bmatrix} 8 \\ 4 \\ 2 \end{bmatrix} = -60$$
$$p_2 = c_2 - \underline{\lambda}^T \underline{a}_2 = -30 - \lambda^T \begin{bmatrix} 6 \\ 2 \\ 1.5 \end{bmatrix} = -30$$
$$p_3 = c_3 - \underline{\lambda}^T \underline{a}_3 = -20 - \lambda^T \begin{bmatrix} 1 \\ 1.5 \\ 0.5 \end{bmatrix} = -20$$

  - $p_1$ is the most negative. Bring $x_1$ into the basis

# Example: Iteration 2

$$\min -60x_1 - 30x_2 - 20x_3$$
$$s.t.\ 8x_1 + 6x_2 + x_3 + s_1 = 48$$
$$4x_1 + 2x_2 + 1.5x_3 + s_2 = 20$$
$$2x_1 + 1.5x_2 + 0.5x_3 + s_3 = 8$$

- Solve for $\underline{\alpha} \Rightarrow \underline{\alpha} = B^{-1}a_1$

$$\underline{\alpha} = \begin{bmatrix} 8 \\ 4 \\ 2 \end{bmatrix}$$

$$\theta = \min\left\{\frac{48}{8} \quad \frac{20}{4} \quad \frac{8}{2}\right\} = \min\{6 \quad 5 \quad 4\} = 4$$

$\Rightarrow s_3$ goes out of the basis

New basic solution:

$$\begin{pmatrix} s_1 \\ s_2 \\ x_1 \end{pmatrix} = \begin{pmatrix} 48-32=16 \\ 20-16=4 \\ 4 \end{pmatrix} = \begin{pmatrix} 16 \\ 4 \\ 4 \end{pmatrix}$$

- *Iteration 2*

$$\text{New } B = \begin{bmatrix} 1 & 0 & 8 \\ 0 & 1 & 4 \\ 0 & 0 & 2 \end{bmatrix}$$

$$B^{-1} = \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & 0.5 \end{bmatrix}$$

$$\lambda^T = c_B^T B^{-1} = \begin{pmatrix} 0 & 0 & -60 \end{pmatrix} \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & 0.5 \end{bmatrix} = \begin{pmatrix} 0 & 0 & -30 \end{pmatrix}$$

# Example: Iteration 2 (cont'd)

$$\min -60x_1 - 30x_2 - 20x_3$$
$$s.t. \; 8x_1 + 6x_2 + x_3 + s_1 = 48$$
$$4x_1 + 2x_2 + 1.5x_3 + s_2 = 20$$
$$2x_1 + 1.5x_2 + 0.5x_3 + s_3 = 8$$

$$p_1 = -\begin{pmatrix}0 & 0 & -30\end{pmatrix}\begin{bmatrix}8\\4\\2\end{bmatrix} + -60 = 0$$

$$p_2 = -30 - \begin{pmatrix}0 & 0 & -30\end{pmatrix}\begin{bmatrix}6\\2\\1.5\end{bmatrix} = 15$$

$$p_3 = -20 - \begin{pmatrix}0 & 0 & -30\end{pmatrix}\begin{bmatrix}1\\1.5\\0.5\end{bmatrix} = -5$$

$$p_{s_3} = 0 - \begin{pmatrix}0 & 0 & -30\end{pmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix} = 30$$

- $\Rightarrow$ bring $x_3$ into the basis

$$\underline{\alpha} = B^{-1}a_3 = \begin{bmatrix}1 & 0 & -4\\ 0 & 1 & -2\\ 0 & 0 & \frac{1}{2}\end{bmatrix}\begin{bmatrix}1\\1.5\\0.5\end{bmatrix} = \begin{bmatrix}-1\\ \frac{1}{2}\\ \frac{1}{4}\end{bmatrix}$$

$$\Rightarrow \theta = \min \begin{pmatrix} \overset{s_1}{*} & \overset{s_2}{8} & \overset{x_1}{16} \end{pmatrix}$$

$\Rightarrow s_2$ goes out and $x_3$ comes in

New basic solution:

$$\begin{pmatrix}s_1\\x_3\\x_1\end{pmatrix} = \begin{pmatrix}16-8(-1)\\8\\4-8(0.25)\end{pmatrix} = \begin{pmatrix}24\\8\\2\end{pmatrix}$$

# Example: Iteration 3

$$\min -60x_1 - 30x_2 - 20x_3$$
$$s.t.\ 8x_1 + 6x_2 + x_3 + s_1 = 48$$
$$4x_1 + 2x_2 + 1.5x_3 + s_2 = 20$$
$$2x_1 + 1.5x_2 + 0.5x_3 + s_3 = 8$$

- *Iteration 3*

$$\text{New } B = \begin{bmatrix} 1 & 1 & 8 \\ 0 & \frac{3}{2} & 4 \\ 0 & \frac{1}{2} & 2 \end{bmatrix}$$

$$B^{-1} = \begin{bmatrix} 1 & 2 & -8 \\ 0 & 2 & -4 \\ 0 & \frac{-1}{2} & \frac{3}{2} \end{bmatrix}$$

$$\lambda^T = c_B^T B^{-1}$$

$$= \begin{pmatrix} 0 & -20 & -60 \end{pmatrix} \begin{bmatrix} 1 & 2 & -8 \\ 0 & 2 & -4 \\ 0 & -0.5 & 1.5 \end{bmatrix} = \begin{pmatrix} 0 & -10 & -10 \end{pmatrix}$$

  - Reduced costs for non-basic variables

$$p_{s_2} = 0 - \lambda^T \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = 0 - \begin{pmatrix} 0 & -10 & -10 \end{pmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = 10$$

$$p_{x_2} = -30 - \lambda^T \begin{pmatrix} 6 \\ 2 \\ 1.5 \end{pmatrix} = -30 - \begin{pmatrix} 0 & -10 & -10 \end{pmatrix} \begin{bmatrix} 6 \\ 2 \\ 1.5 \end{bmatrix} = 5$$

$$p_{s_3} = 0 - \begin{pmatrix} 0 & -10 & -10 \end{pmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 10$$

$\Rightarrow$ all $p_i > 0 \Rightarrow$ optimal

$$\begin{pmatrix} s_1^* \\ x_3^* \\ x_1^* \end{pmatrix} = \begin{pmatrix} 24 \\ 8 \\ 2 \end{pmatrix}$$

Optimal solution:

$x_1^* = 2,\ x_2^* = 0,\ x_3^* = 8$

Optimal cost:

$(-60)*2 - 30*0 - 20*8$

$= -280$

For $\leq$ constraints with non-negative $\underline{b}$ (as in this problem), feasible solution is easy; set slacks = $\underline{b}$. For $\underline{b}$ with negative elements, need both a slack and a y to initiate Phase I or big *M* method with $\leq$ constraints. Complementary comments apply to $\geq$ constraints.

# Illustration of Big M method for ≥ constraints

Crude oil problem:

$x_1$ = number of barrels of light crude

$x_2$ = number of barrels of heavy crude

$$\min 56x_1 + 50x_2$$
$$s.t.\ 0.3x_1 + 0.3x_2 \geq 900{,}000$$
$$0.2x_1 + 0.4x_2 \geq 800{,}000$$
$$0.3x_1 + 0.2x_2 \geq 500{,}000$$
$$x_1 \geq 0; x_2 \geq 0$$

Big-M SLP: $\min 56x_1 + 50x_2 + 10000y_1 + 10000y_2 + 10000y_3$

$$s.t.\ 0.3x_1 + 0.3x_2 - s_1 + y_1 = 900{,}000$$
$$0.2x_1 + 0.4x_2 - s_2 + y_2 = 800{,}000$$
$$0.3x_1 + 0.2x_2 - s_3 + y_3 = 500{,}000$$
$$x_1 \geq 0; x_2 \geq 0; s_i \geq 0; y_i \geq 0, i = 1,2,3$$

Lot easier to do this problem geometrically as in HW, but Illustrates the big M method nicely

*Iteration* 1:

$$\underline{x}_B = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 900{,}000 \\ 800{,}000 \\ 500{,}000 \end{bmatrix}; B = I = B^{-1}; \underline{\lambda}^T = 10^4 [1 \quad 1 \quad 1]$$

$$p_1 = c_1 - \underline{\lambda}^T \underline{a}_1 = 56 - 10^4 [1 \quad 1 \quad 1] \begin{bmatrix} 0.3 \\ 0.2 \\ 0.3 \end{bmatrix} = -7{,}944; \ p_2 = c_2 - \underline{\lambda}^T \underline{a}_2 = 50 - 10^4 [1 \quad 1 \quad 1] \begin{bmatrix} 0.3 \\ 0.4 \\ 0.2 \end{bmatrix} = -8{,}950$$

$$p_3 = c_3 - \underline{\lambda}^T \underline{a}_3 = 10{,}000; \ p_4 = c_4 - \underline{\lambda}^T \underline{a}_4 = 10{,}000; \ p_5 = c_5 - \underline{\lambda}^T \underline{a}_5 = 10{,}000$$

$$p_6 = p_7 = p_8 = 0$$

$\Rightarrow$ bring $x_2$ into the basis

$$\underline{\alpha} = \begin{bmatrix} 0.3 \\ 0.4 \\ 0.2 \end{bmatrix} \Rightarrow \theta = 10^5 \min\{9/0.3, 8/0.4, 5/0.2\} = 2.10^6 \Rightarrow y_2 \ should\ go\ out$$

# Example: Big M Iterations 2-4

*Iteration* 2:

$$\underline{x}_B = \begin{bmatrix} y_1 \\ x_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 300,000 \\ 2.10^6 \\ 100,000 \end{bmatrix}; B = \begin{bmatrix} 1 & 0.3 & 0 \\ 0 & 0.4 & 0 \\ 0 & 0.2 & 1 \end{bmatrix}; B^{-1} = \begin{bmatrix} 1 & -3/4 & 0 \\ 0 & 5/2 & 0 \\ 0 & -1/2 & 1 \end{bmatrix}; \underline{\lambda}^T = \underline{c}_B^T B^{-1} = \begin{bmatrix} 10000 & -12375 & 10000 \end{bmatrix}$$

$$\underline{p}^T = [-3469 \quad 0 \quad 10000 \quad -12375 \quad 10000 \quad 0 \quad 22375 \quad 0]$$

$$\Rightarrow \text{bring } s_2 \text{ into the basis; } \underline{\alpha} = \begin{bmatrix} 0.75 \\ -2.5 \\ 0.5 \end{bmatrix} \Rightarrow \theta = \min\{300,000/0.75, 100,000/0.5\} = 200,000 \Rightarrow y_3 \textit{ should go out}$$

*Iteration* 3:

$$\underline{x}_B = \begin{bmatrix} y_1 \\ x_2 \\ s_2 \end{bmatrix} = \begin{bmatrix} 150,000 \\ 2.5.10^6 \\ 200,000 \end{bmatrix}; B = \begin{bmatrix} 1 & 0.3 & 0 \\ 0 & 0.4 & -1 \\ 0 & 0.2 & 0 \end{bmatrix}; B^{-1} = \begin{bmatrix} 1 & 0 & -3/2 \\ 0 & 0 & 5 \\ 0 & -1 & 2 \end{bmatrix}; \underline{\lambda}^T = \underline{c}_B^T B^{-1} = \begin{bmatrix} 10000 & 0 & -14750 \end{bmatrix}$$

$$\underline{p}^T = [1,481 \quad 0 \quad 10,000 \quad 0 \quad -14,750 \quad 0 \quad 10,000 \quad 24,750]$$

$$\Rightarrow \text{bring } s_3 \text{ into the basis; } \underline{\alpha} = \begin{bmatrix} 3/2 \\ -5 \\ -2 \end{bmatrix} \Rightarrow \theta = 100,000 \Rightarrow y_1 \textit{ should go out}$$

*Iteration* 4:

$$\underline{x}_B = \begin{bmatrix} s_3 \\ x_2 \\ s_2 \end{bmatrix} = \begin{bmatrix} 100,000 \\ 3.10^6 \\ 400,000 \end{bmatrix}; B = \begin{bmatrix} 0 & 0.3 & 0 \\ 0 & 0.4 & -1 \\ -1 & 0.2 & 0 \end{bmatrix}; B^{-1} = \begin{bmatrix} 2/3 & 0 & -1 \\ 10/3 & 0 & 0 \\ 4/3 & -1 & 0 \end{bmatrix}; \underline{\lambda}^T = \underline{c}_B^T B^{-1} = \begin{bmatrix} 500/3 & 0 & 0 \end{bmatrix}$$

$$\underline{p}^T = [6 \quad 0 \quad 167 \quad 0 \quad 0 \quad 9833 \quad 10,000 \quad 10000]$$

$$\textit{optimal} \Rightarrow x_1 = 0; x_2 = 3.10^6 \textit{barrels}; \cos t = \$150M$$

# Example: Detecting infeasibility (Phase I)

- Consider

$$\min x_1 + x_2$$
$$s.t.\ x_1 + x_2 = -4$$
$$x_1 \geq 0; x_2 \geq 0$$

$$\min x_1 + x_2$$
$$s.t. - x_1 - x_2 = 4$$
$$x_1 \geq 0; x_2 \geq 0$$

For phase I:

$$\min y$$
$$s.t. - x_1 - x_2 + y = 4$$
$$x_1 \geq 0; x_2 \geq 0; y \geq 0$$

$$Phase\ I : \min\ y = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ y \end{bmatrix}$$

$$A = \begin{bmatrix} -1 & -1 & 1 \end{bmatrix}$$

$$Iteration 1 : B = 1 = B^{-1}; x_B = y = 4; \lambda = 1; \cos t = 1$$

$$\text{Re}\,duced\ \cos ts : p_1 = 0 - 1.(-1) = 1; p_2 = 0 - 1.(-1) = 1; p_3 = 1 - 1.1 = 0$$

$$Optimal \Rightarrow y = 1\ and\ \cos t = 1 > 0 \Rightarrow \inf easible$$

# Summary

- LP problem: $\min c^T \underline{x}$ s.t. $A\underline{x} = \underline{b}$, $\underline{x} \geq \underline{0}$
- At least $(n - m)$ components of $\underline{x}$ are zero
- Such solutions are called basic feasible solutions (bfs)
- They are also extreme points of $K = \{\underline{x} : A\underline{x} = \underline{b}, \underline{x} \geq \underline{0}\}$
- An LP may have no solution: detected in Phase 1 of Simplex
- Unbounded solution: $\underline{\alpha}_j \leq \underline{0}$ detected in Phase 2
- Unique solution: $\underline{p} \geq \underline{0}$ and detected in Phase 2 of Simplex
- Monitor residuals and be aware of finite precision arithmetic
- Must use factorization schemes for efficiency of updating the basis matrix… Lecture 3