# ISE 754: Logistics Engineering

Michael G. Kay

Spring 2017

# Topics

1. Introduction
2. Facility location
3. Freight transport
   – Midterm exam
4. Network models
5. Routing
6. $\boxed{\text{Warehousing}}$
   – Final project
   – Final exam

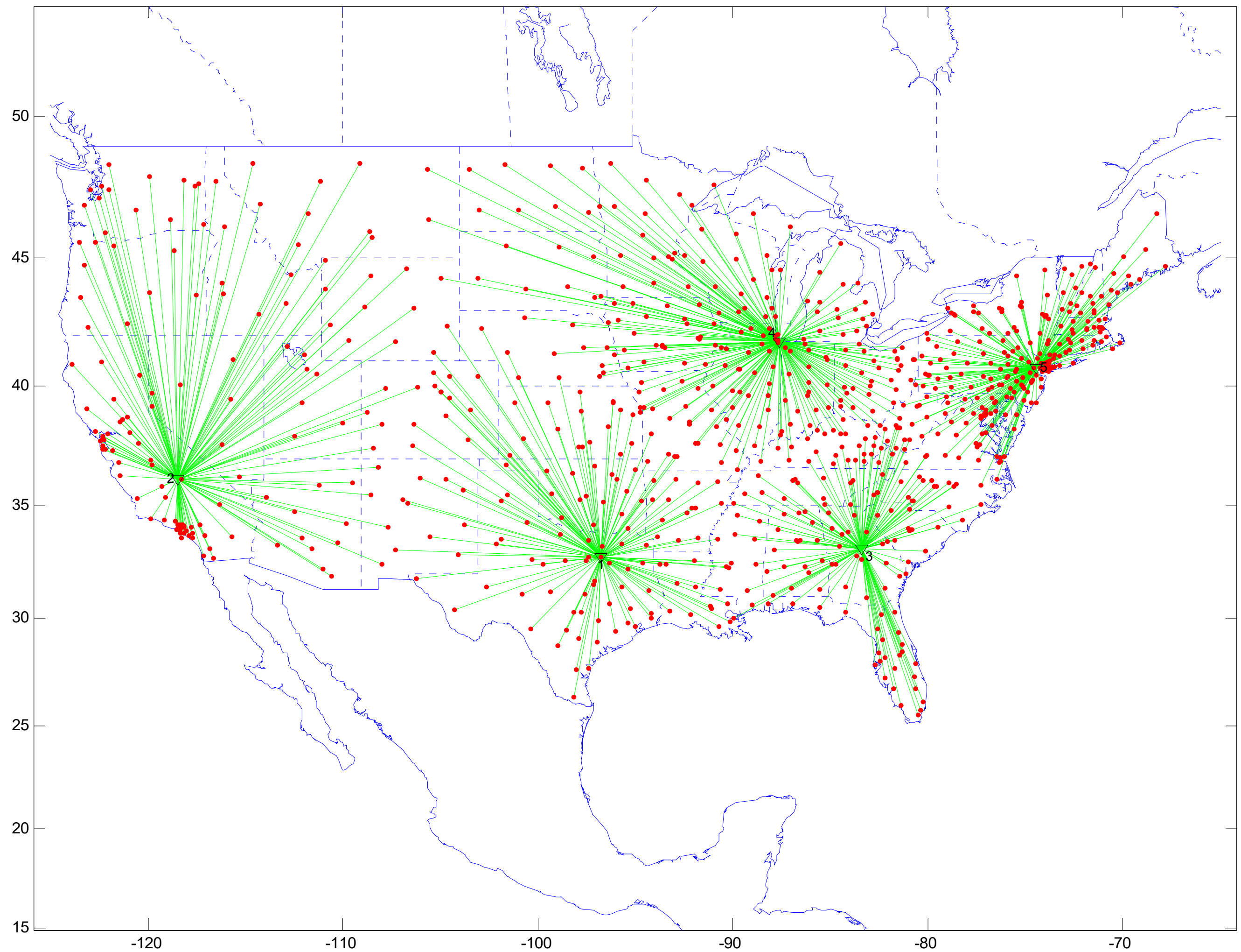
Outside the box

Inside the box

# Scope

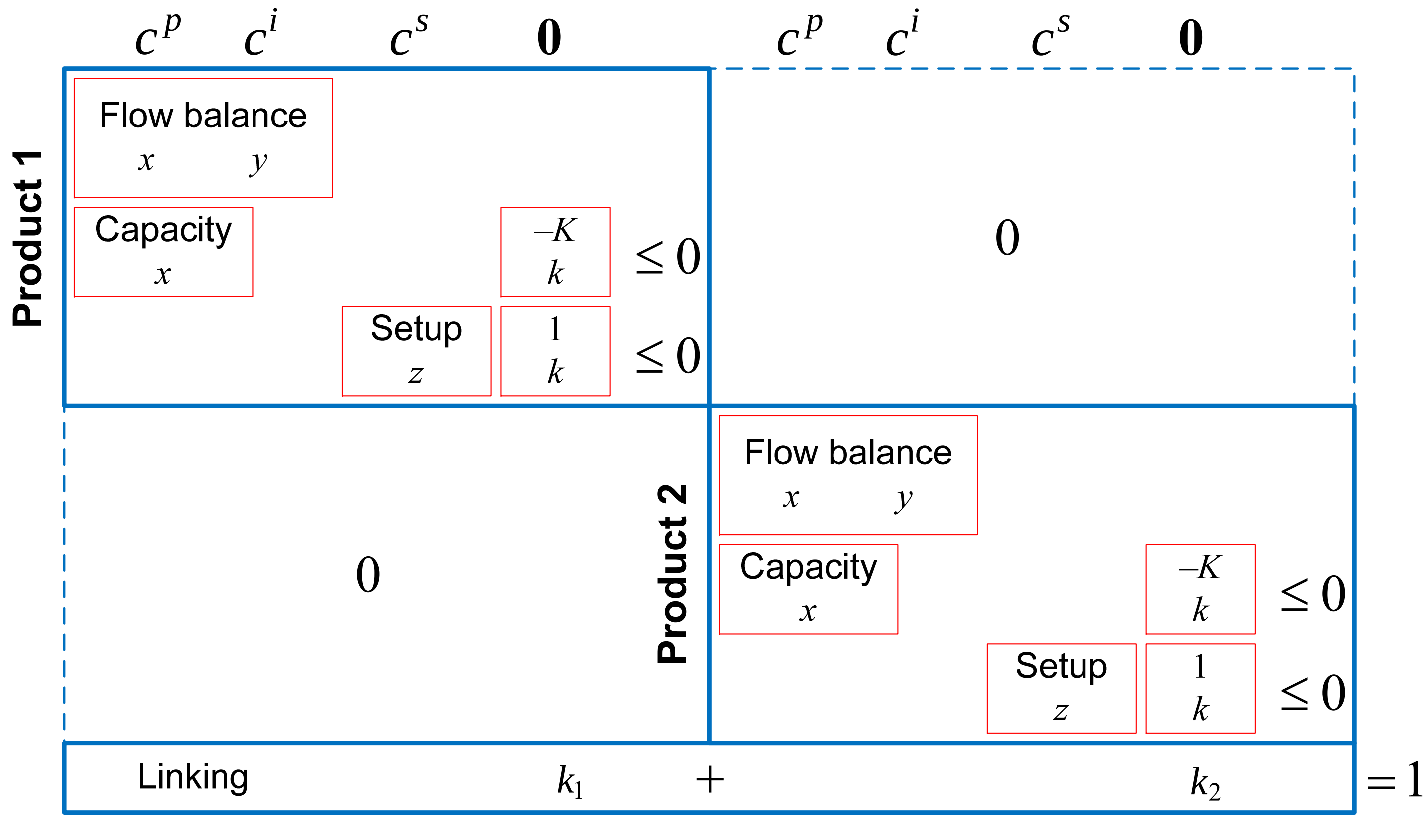
- Strategic (years)
  - Network design
- Tactical (weeks-year)
  - Multi-echelon, multi-period, multi-product production and inventory models
- Operational (minutes-week)
  - Vehicle routing

# Strategic: Network Design




**Optimal locations for five DCs serving 877 customers throughout the U.S.**

# Tactical: Production-Inventory Model



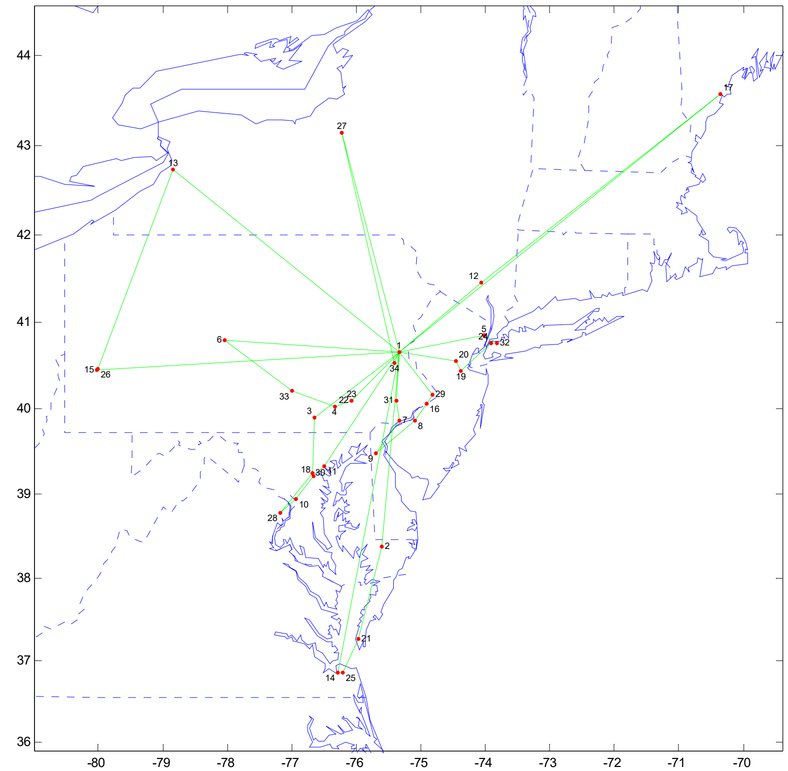

**Constraint matrix for a 2-product, multi-period model with setups**

# Vehicle Routing

## Eight routes served from DC in Harrisburg, PA

### Route Summary Information

| Route | Load Weight | Route Time | Customers in Route | Layover Required |
|---|---|---|---|---|
| 1 | 12,122 | 18.36 | 4 | 1 |
| 2 | 4,833 | 16.05 | 2 | 1 |
| 3 | 9,642 | 17.26 | 3 | 1 |
| 4 | 25,957 | 13.77 | 6 | 0 |
| 5 | 12,512 | 9.90 | 2 | 0 |
| 6 | 15,156 | 13.70 | 5 | 0 |
| 7 | 29,565 | 11.30 | 6 | 0 |
| 8 | 32,496 | 8.84 | 5 | 0 |
| | | 109.18 | | 3 |




### Detailed Route Information

| | | | | | | Start | L/D (hr) | Depart | Total Time (hr) | Zip Code |
|---|---|---|---|---|---|---|---|---|---|---|
| Route | | 1 | | | | | | | | |
| | 1 | | | | | 23:29 | 0 | 23:29 | 0 | 18020 |
| | 14 | 2,328 | 7.51 | 7:00 | 399 | 7:00 | 0.59 | 7:35 | 8.1 | 23510 |
| | 25 | 4,697 | 0.19 | 7:46 | 6 | 7:46 | 0.68 | 8:27 | 0.87 | 23502 |
| | 21 | 3,682 | 1.12 | 9:34 | 37 | 9:34 | 0.64 | 10:13 | 1.76 | 23310 |
| | 2 | 1,415 | 2.51 | 12:43 | 93 | 12:43 | 0.56 | 13:17 | 3.07 | 21801 |
| | 1 | 0 | 4.57 | 17:51 | 196 | 17:51 | 0 | 17:51 | 4.57 | 18020 |
| Total | | 12,122 | 15.89 | | 731 | | 2.47 | | 18.36 | |

# Geometric Mean

- How many people can be crammed into a car?
  - Certainly more than one and less than 100: the average (50) seems to be too high, but the geometric mean (10) is reasonable

$$\text{Geometric Mean:} \quad X = \sqrt{LB \times UB} = \sqrt{1 \times 100} = 10$$

- Often it is difficult to directly estimate input parameter X, but is easy to estimate reasonable lower and upper bounds (LB and UB) for the parameter
  - Since the guessed LB and UB are usually orders of magnitude apart, use of the arithmetic mean would give too much weight to UB
  - Geometric mean gives a more reasonable estimate because it is a logarithmic average of LB and UB

# Fermi Problems

- Involves "reasonable" (i.e., +/– 10%) *guesstimation* of input parameters needed and back-of-the-envelope type approximations
  – Goal is to have an answer that is within an order of magnitude of the correct answer (or what is termed a *zeroth-order approximation*)
  – Works because over- and under-estimations of each parameter tend to cancel each other out as long as there is no consistent bias
- How many McDonald's restaurants in U.S.? (actual 2013: 14,267)

| **Parameter** | **LB** | | **UB** | **Estimate** | |
|---|---|---|---|---|---|
| Annual per capita demand | 1 | 1 order/person-day x 350 day/yr = | 350 | 18.71 | (order/person-yr) |
| U.S. population | | | | 300,000,000 | (person) |
| Operating hours per day | | | | 16 | (hr/day) |
| Orders per store per minute (in-store + drive-thru) | | | | 1 | (order/store-min) |
| **Analysis** | | | | | |
| Annual U.S. demand | | (person) x (order/person-yr) = | | 5,612,486,080 | (order/yr) |
| Daily U.S. demand | | (order/yr)/365 day/yr = | | 15,376,674 | (order/day) |
| Daily demand per store | | (hrs/day) x 60 min/hr x (order/store-min) = | | 960 | (order/store-day) |
| Est. number of U.S. stores | | (order/day) / (order/store-day) = | | 16,017 | (store) |

# System Performance Estimation

- Often easy to estimate performance of a new system if can assume either perfect or no control
- Example: estimate waiting time for a bus
  - 8 min. avg. time (aka “headway”) between buses
  - Customers arrive at random
    - assuming no web-based bus tracking
  - Perfect control (LB): wait time = half of headway
  - No control (*practical* UB): wait time = headway
    - assuming buses arrive at random

$$\text{Estimated wait time} = \sqrt{LB \times UB} = \sqrt{\frac{8}{2} \times 8} = 5.67 \text{ min}$$

  - Bad control can result in higher values than no control

# http://www.nextbuzz.gatech.edu/

SELF-COORDINATING BUSES
REDUCE BUNCHING

HOME THE IDEA PROOF OF CONCEPT HOW IT WORKS CONTRIBUTORS

## A BUS-HEADWAY CONTROLLER

A software system to coördinate buses on a route, based on an idea by John J. Bartholdi III and Donald D. Eisenstein. The current version of the software was designed and largely written by Loren K. Platzman. Implementation has been led by Russ Clark, Jin Lee, and David Williamson.

### THE IDEA

Delaying buses briefly at certain checkpoints equalizes headways

Read more

### PROOF OF CONCEPT

Coordinating trolleys on Georgia Tech's busiest route

Read more

### HOW IT WORKS

Tablets, GPS, cellular networks, and web-based control

Read more

# Crowdsourcing

- Obtain otherwise hard to get information from a large group of online workers
- Amazon's Mechanical Turk is best known
  - Jobs posted as HITs (Human Information Tasks) that typically pay $1-2 per hour
  - Main use has been in machine learning to create tagged data sets for training purposes
  - Has been used in logistics engineering to estimate the percentage homes in U.S. that have sidewalks (sidewalk deliveries by Starship robots)

# Available 2016: Starship Technologies

- Started by Skype co-founders
- 99% autonomous
- Goal: “deliver ‘two grocery bags’ worth of goods (weighing up to 20lbs) in 5-30 minutes for ‘10-15 times less than the cost of current last-mile delivery alternatives.’”

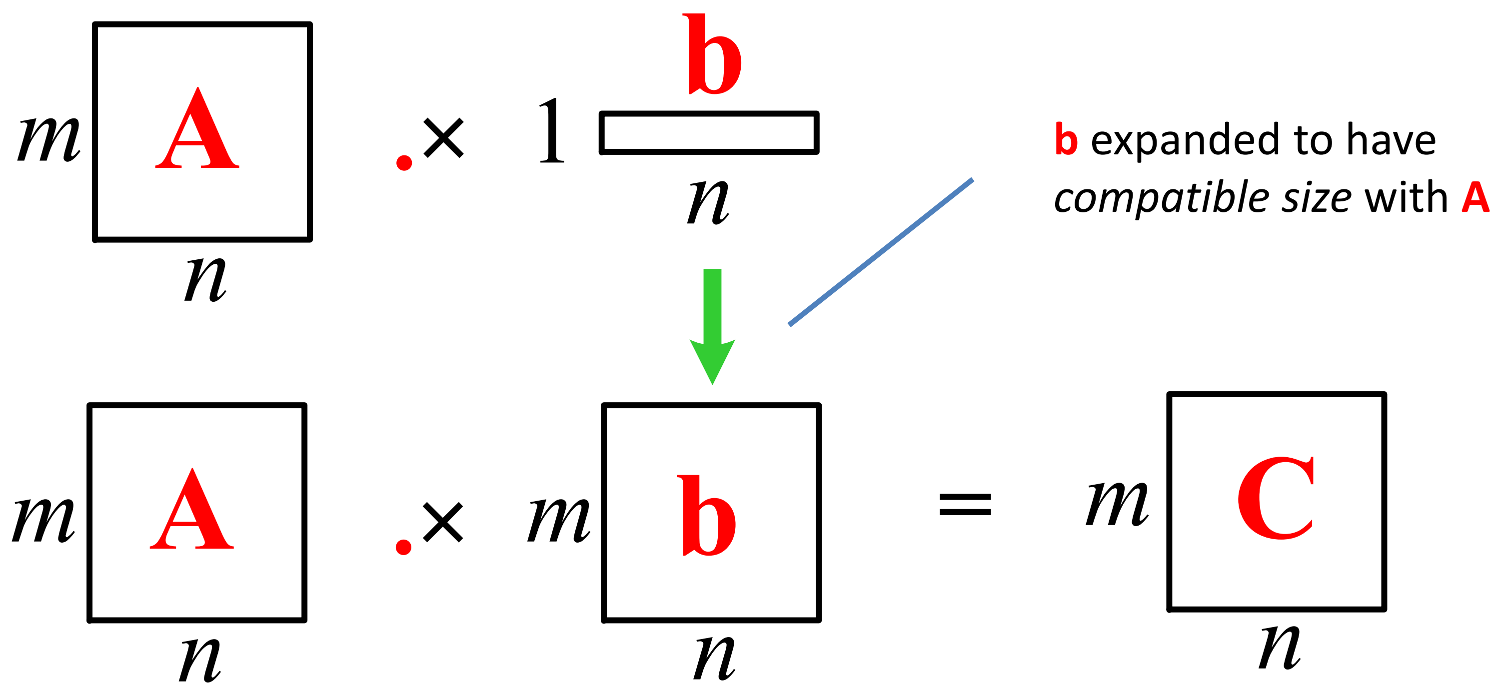
# Matrix Multiplication

$$m\,\underset{n}{\boxed{\mathbf{A}}} \quad \times \quad n\,\underset{p}{\boxed{\mathbf{B}}} \quad = \quad m\,\underset{p}{\boxed{\mathbf{C}}}$$

$$(m \times \cancel{n}) \times (\cancel{n} \times p) \quad = \quad (m \times p)$$

Arrays must have same
*inner dimensions*

# Element-by-Element Multiplication




$$(m \times n) \times (1 \times n) \quad = \quad (m \times n)$$

# Compatible Sizes

- Two arrays have compatible sizes if, for *each respective dimension*, either
  - has the same size, or
  - size of one of arrays is one, in which case it is automatically duplicated so that it matches the size of the other array

$$\mathbf{A}_{m\times n} \;.*\; \mathbf{B}_{m\times n} = \mathbf{C}_{m\times n}$$

$$\mathbf{A}_{m\times n} \;.*\; \mathbf{b}_{1\times n} = \mathbf{C}_{m\times n}$$

$$\mathbf{a}_{m\times 1} \;.*\; \mathbf{B}_{m\times n} = \mathbf{C}_{m\times n}$$

$$\mathbf{a}_{m\times 1} \;.*\; \mathbf{b}_{1\times n} = \mathbf{C}_{m\times n}$$

$$\mathbf{a}_{1\times n} \;.*\; \mathbf{b}_{m\times 1} = \mathbf{C}_{m\times n}$$

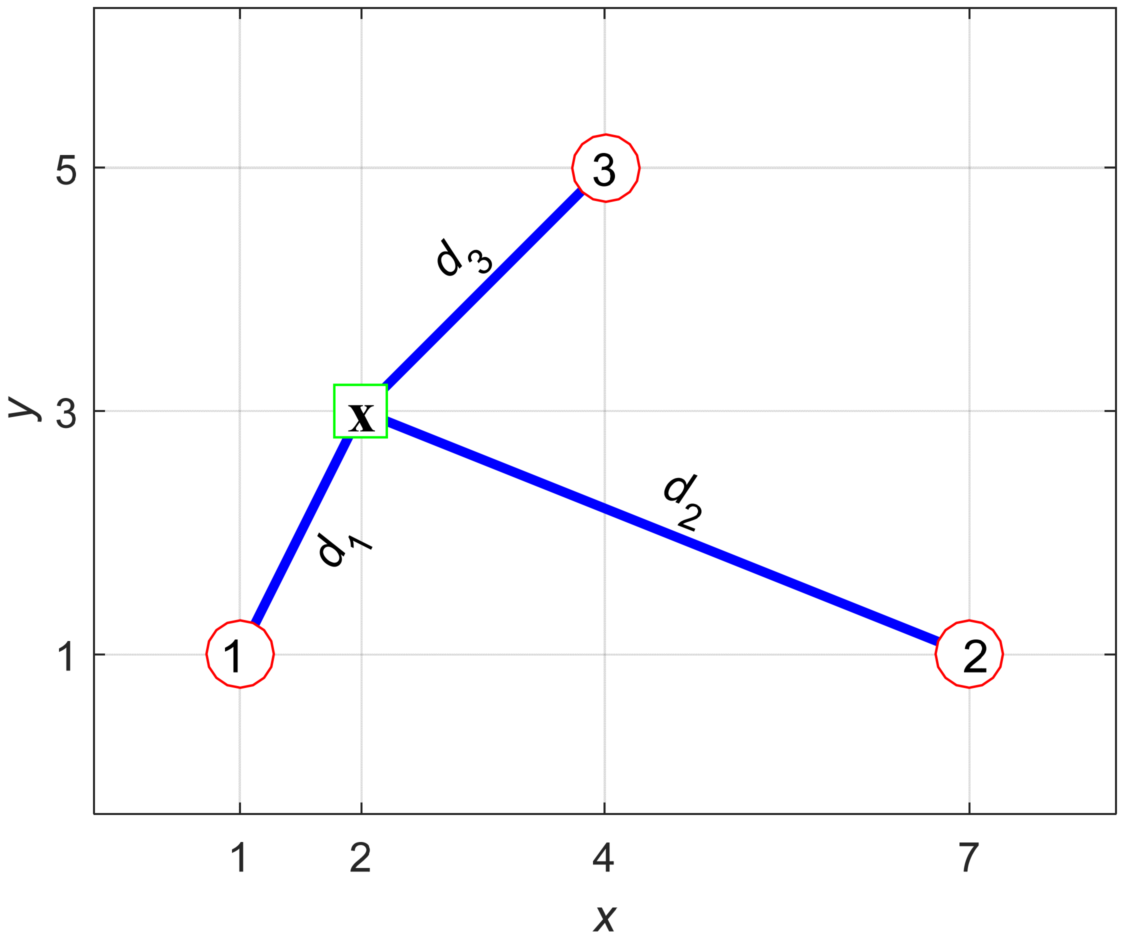
$$\cancel{\mathbf{A}_{m\times n} \;.*\; \mathbf{B}_{m\times p}}$$

$$\cancel{\mathbf{A}_{m\times n} \;.*\; \mathbf{b}_{n\times 1}}$$

$$\cancel{\mathbf{a}_{m\times 1} \;.*\; \mathbf{B}_{p\times n}}$$

$$\cancel{\mathbf{a}_{m\times 1} \;.*\; \mathbf{b}_{n\times 1}}$$

# 2-D Euclidean Distance



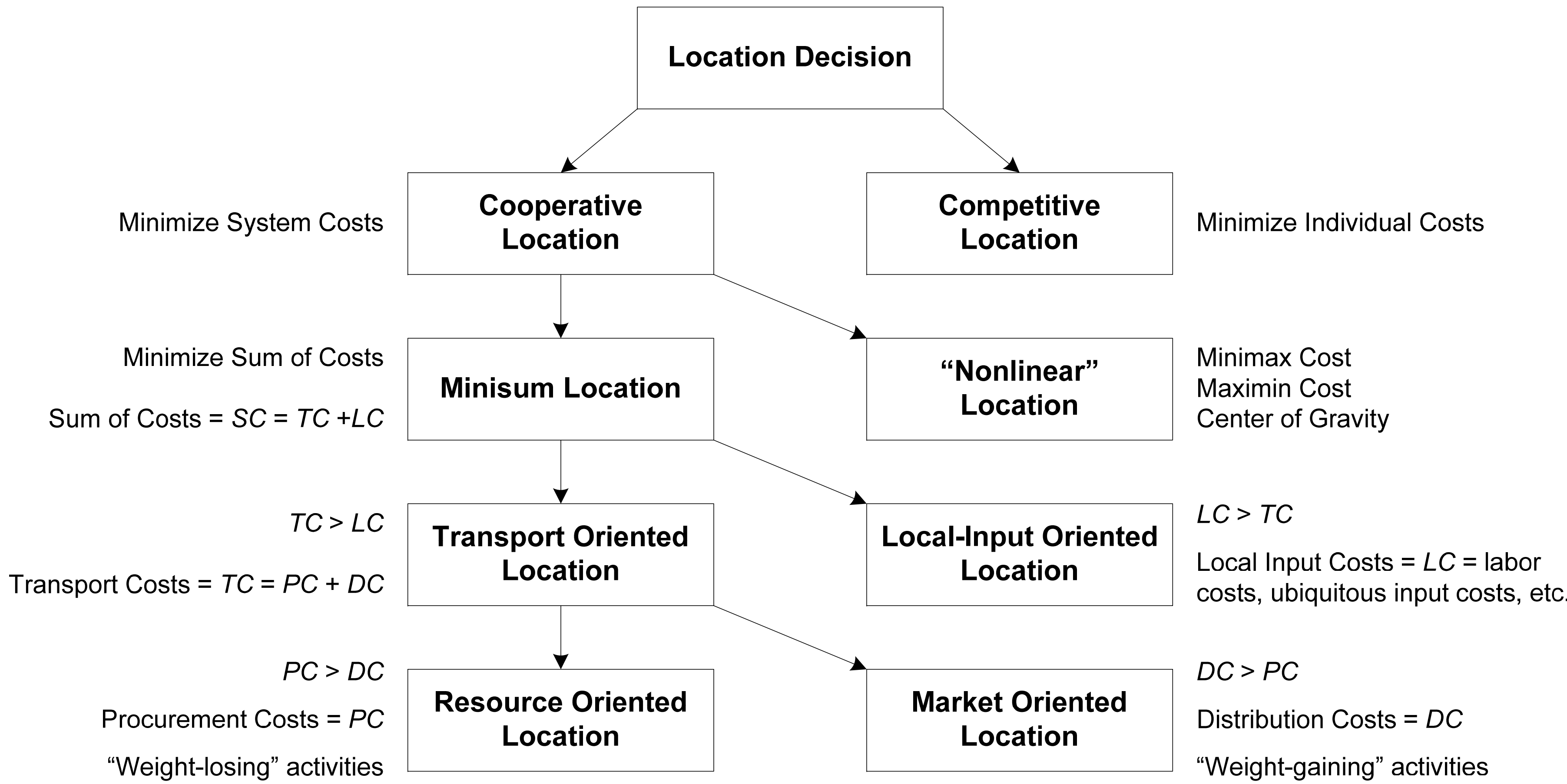

$$\mathbf{x} = \begin{bmatrix} 2 & 3 \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} 1 & 1 \\ 7 & 1 \\ 4 & 5 \end{bmatrix}$$

$$\mathbf{d} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} \sqrt{\left(x_1 - p_{1,1}\right)^2 + \left(x_2 - p_{1,2}\right)^2} \\ \sqrt{\left(x_1 - p_{2,1}\right)^2 + \left(x_2 - p_{2,2}\right)^2} \\ \sqrt{\left(x_1 - p_{3,1}\right)^2 + \left(x_2 - p_{3,2}\right)^2} \end{bmatrix}$$

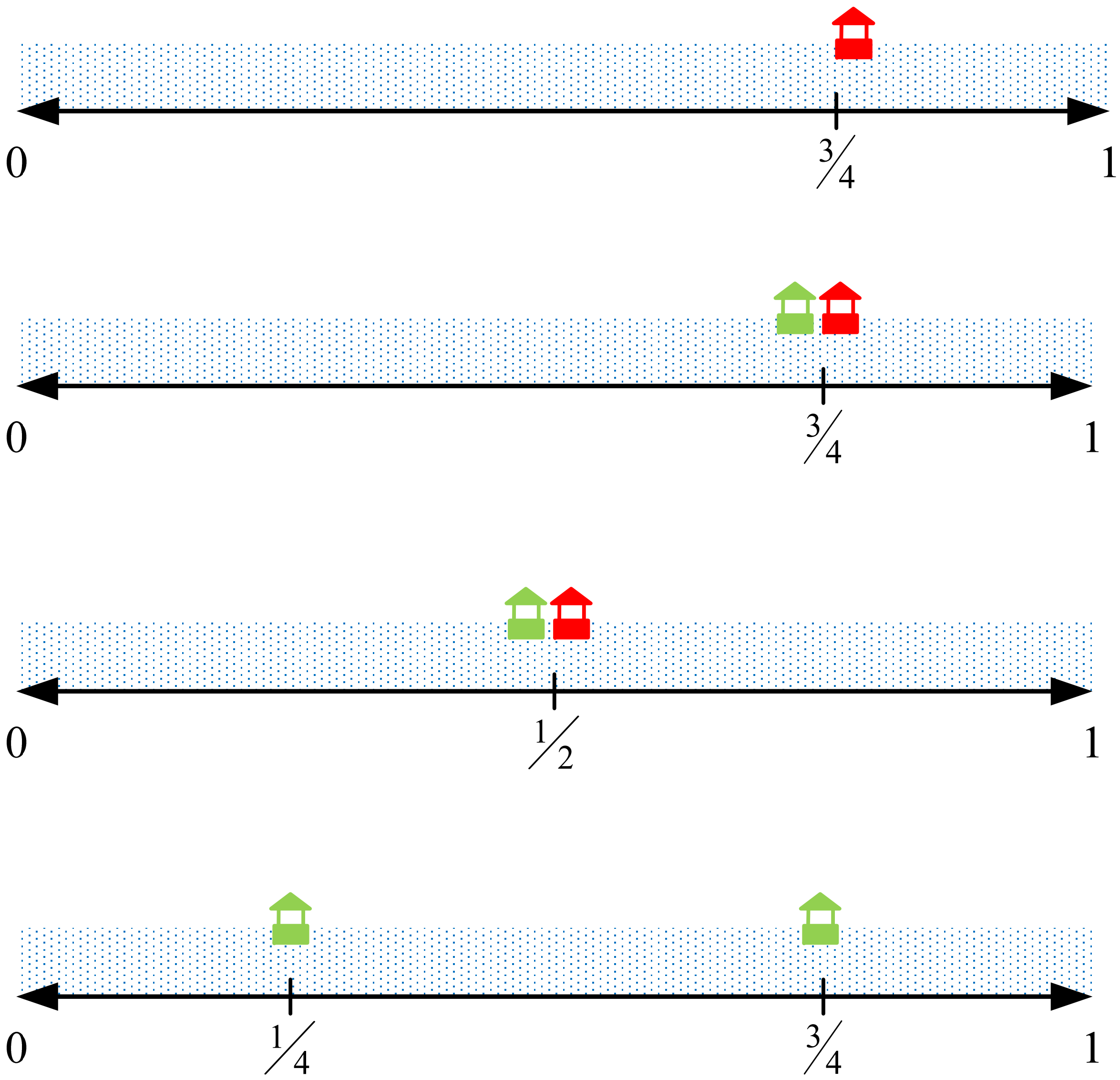
# Basic Matlab Workflow

- Given problem to solve:
  1. Test critical steps at Command Window
  2. Copy working critical steps to a cell (&&) in script file (myscript.m) along with supporting code
  - Repeat using new cells for additional problems
- Once all problems solved, report using:
  - `>> diary hw1soln.txt`
  - Evaluate each cell in script:
    - To see code + results: select text then Evaluate Selection on mouse menu (or F9)
    - To see results: position cursor in cell then Evaluate Current Section (Cntl+Enter)
  - `>> diary off`
- Can also report using Publish (see Matlab menu) as html or Word
- Submit all files created, which may include additional
  - Data files (myscript.mat) or spreadsheet files (myexcel.xlsx)
  - Function files (myfun.m) that can allow use to re-use same code used in multiple problems
    - All code inside function isolated from other code except for inputs/outputs: `[out1,out2] = myfun(input1,input2)`

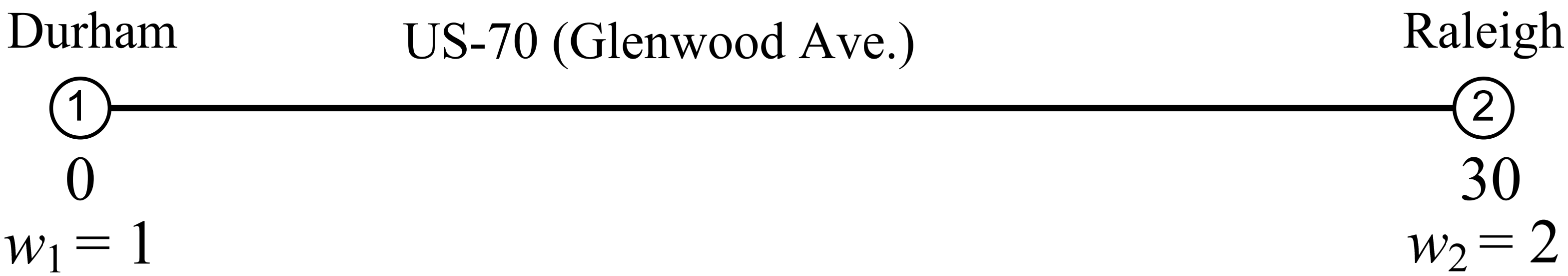
# Taxonomy of Location Problems



# Hotelling's Law



# 1-D Cooperative Location




$$\text{Min } TC = \sum w_i \, d_i$$

$$\text{Min } TC = \sum w_i \, d_i^2$$

$$\text{Min } TC = \sum w_i \, d_i^k$$

$$a_1 = 0, \quad a_2 = 30$$

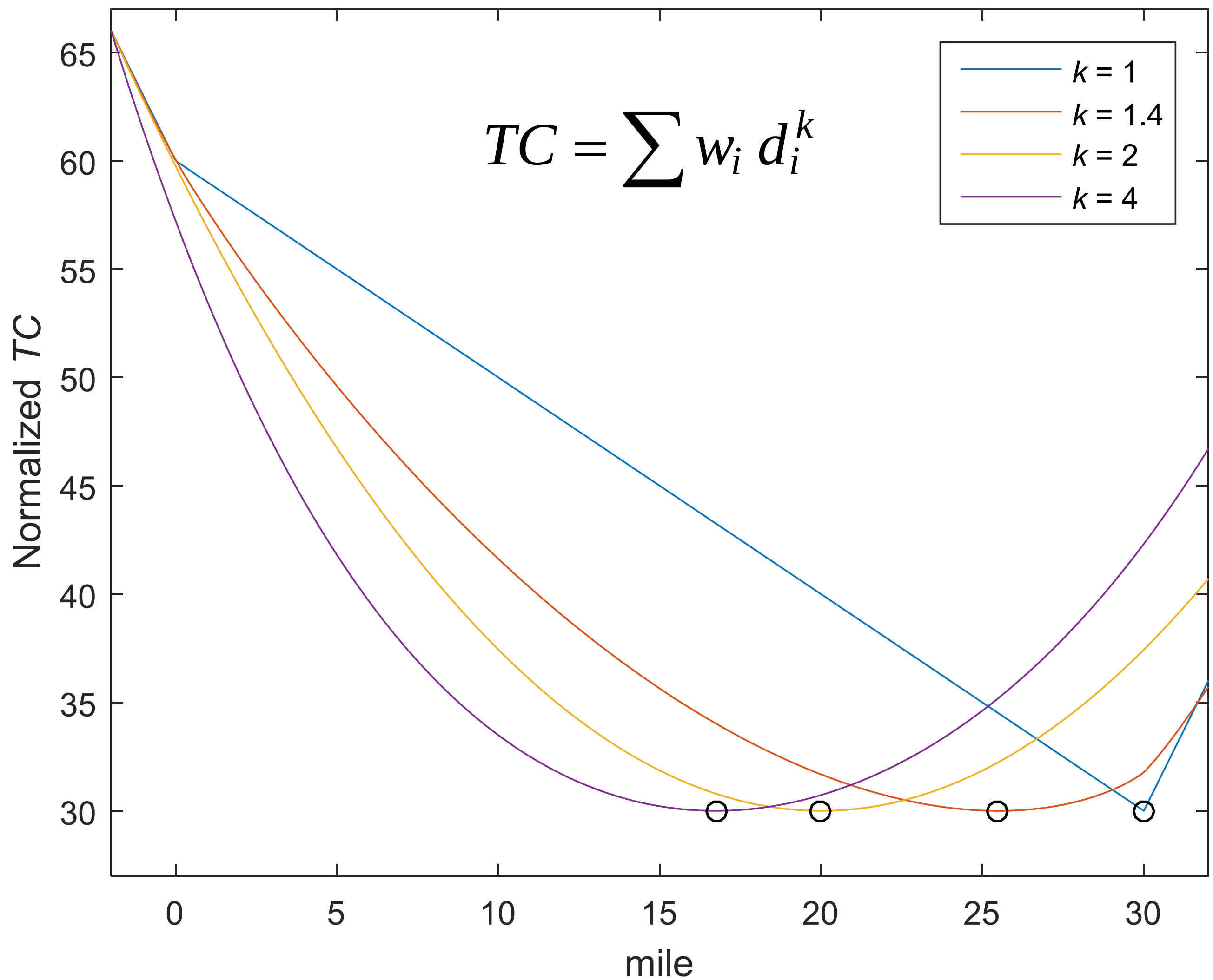
$$TC = \sum w_i \, d_i^2 = \sum w_i \left( x - a_i \right)^2$$

$$\frac{dTC}{dx} = 2 \sum w_i \left( x - a_i \right) = 0 \Rightarrow$$

$$x \sum w_i = \sum w_i \, a_i \Rightarrow$$

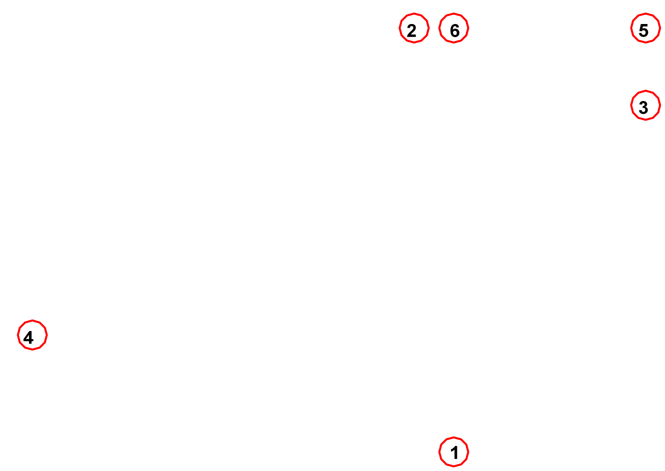
Squared−Euclidean Distance $\Rightarrow$ Center of Gravity: $x^* = \dfrac{\sum w_i \, a_i}{\sum w_i} = \dfrac{1(0) + 2(30)}{1 + 2} = 20$

# “Nonlinear” Location

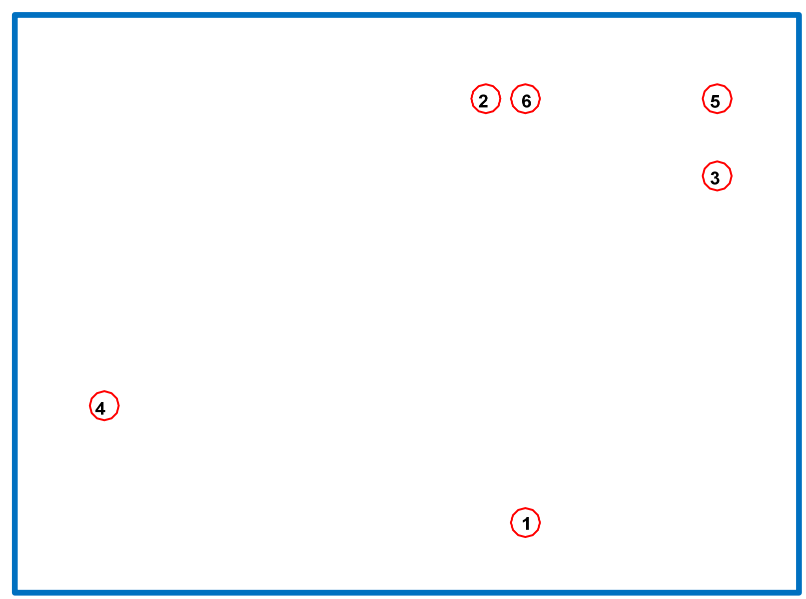


# Minimax and Maximin Location

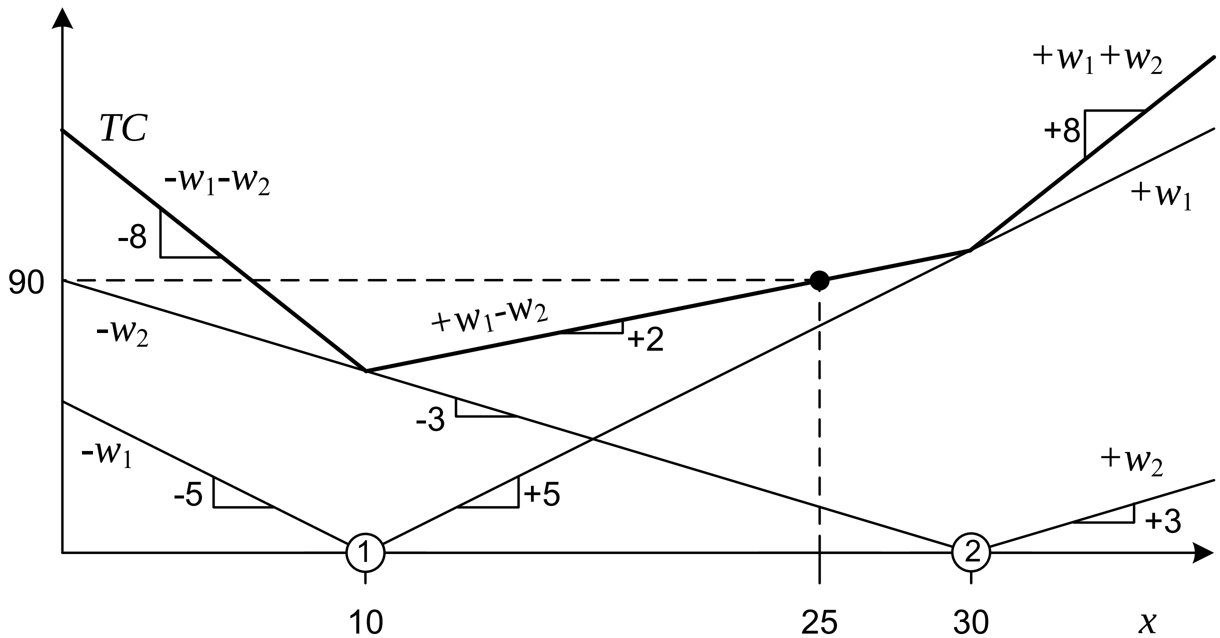
- Minimax
  - Min max distance
  - Set covering problem




- Maximin
  - Max min distance
  - AKA obnoxious facility location



# 2-EF Minisum Location



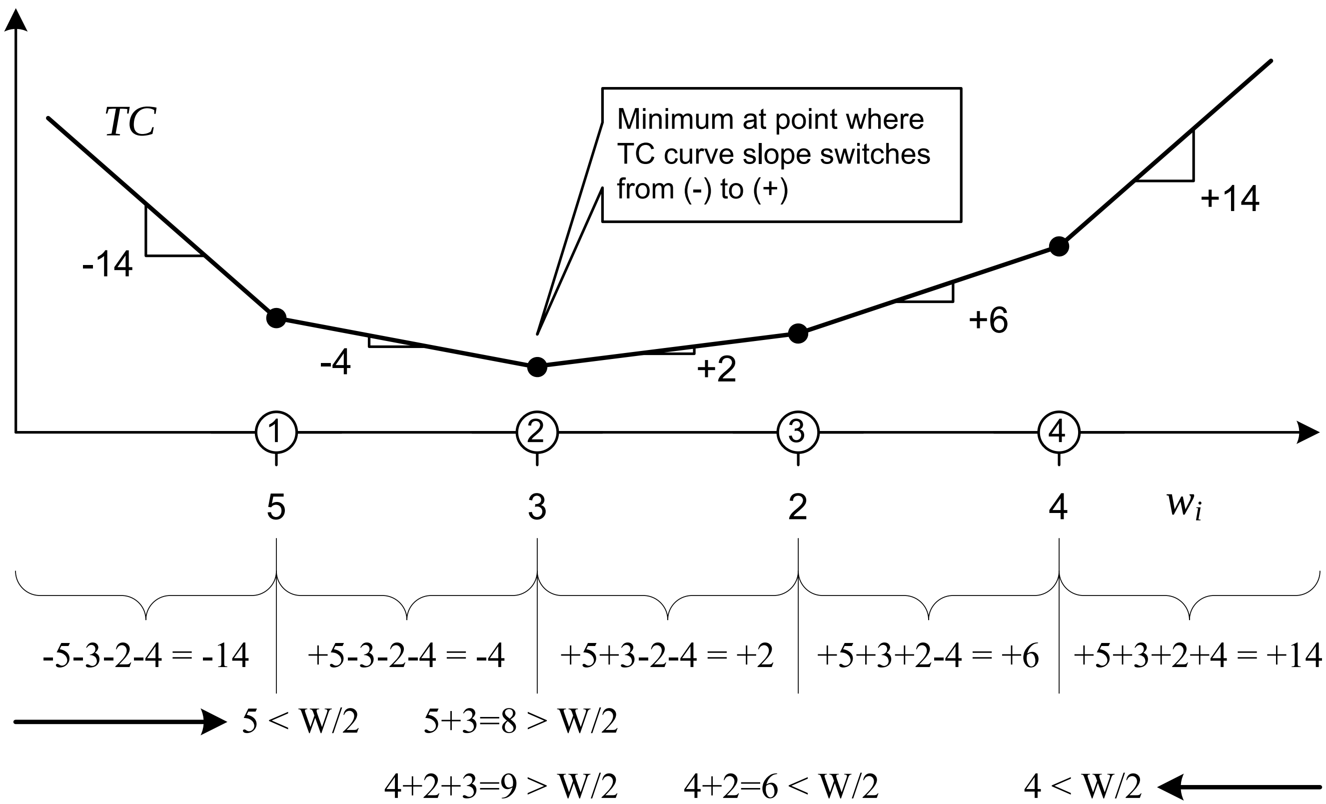

$$TC(x) = \sum w_i\, d_i = \beta_1(x - x_1) + \beta_2(x - x_2), \quad \text{where } \beta_i = \begin{cases} w_i, & \text{if } x \geq x_i \\ -w_i, & \text{if } x < x_i \end{cases}$$

$$\begin{aligned} TC(25) &= w_1(25-10) + (-w_2)(25-30) \\ &= 5(15) + (-3)(-5) = 90 \end{aligned}$$

# Median Location: 1-D 4 EFs

*Median location:* For each dimension $x$ of $X$:

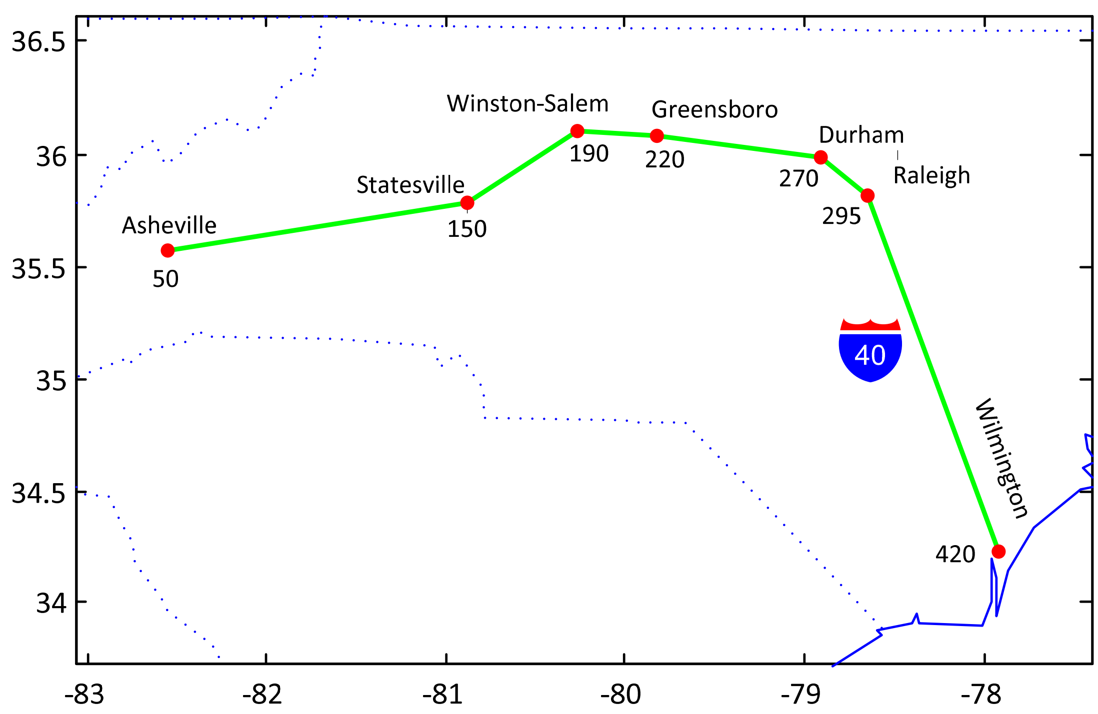
1. Order EFs so that $|x_1| \le |x_2| \le \cdots \le |x_m|$
2. Locate $x$-dimension of NF at the first EF$j$ where $\sum_{i=1}^{j} w_i \ge \frac{W}{2}$, where $W = \sum_{i=1}^{m} w_i$

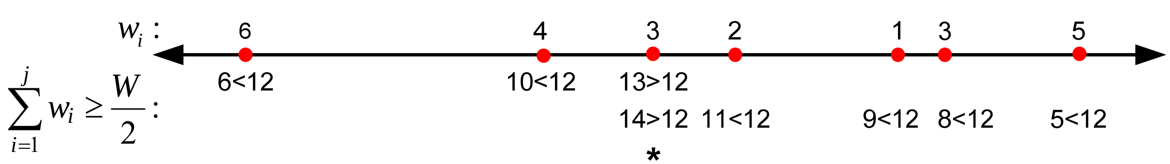


# Median Location: 1-D 7 EFs

*Median location:* For each dimension $x$ of $X$:

1. Order EFs so that $|x_1| \le |x_2| \le \cdots \le |x_m|$
2. Locate $x$-dimension of NF at the first EF $j$ where $\sum_{i=1}^{j} w_i \ge \frac{W}{2}$, where $W = \sum_{i=1}^{m} w_i$

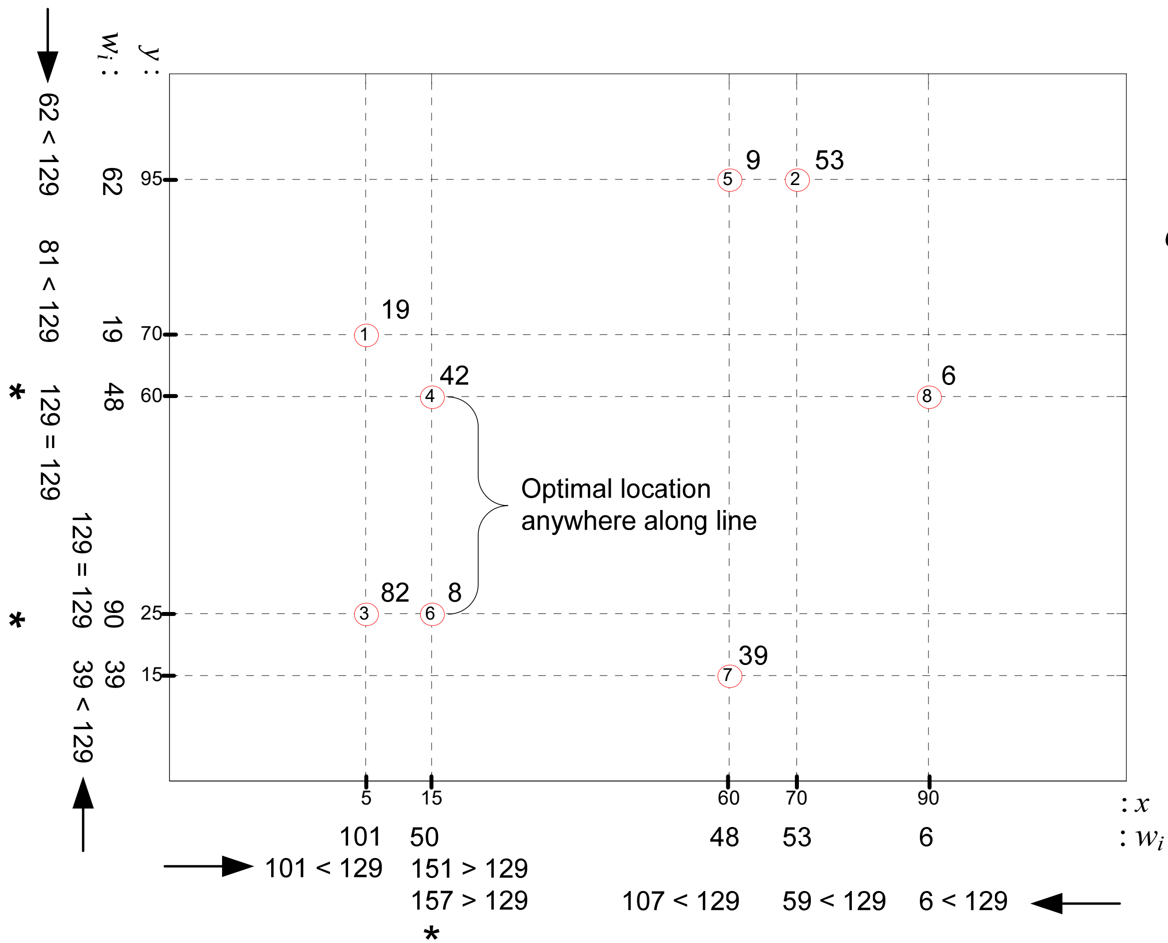


# Median Location: 2-D Rectilinear Distance 8 EFs

*Median location:* For each dimension $x$ of $X$:

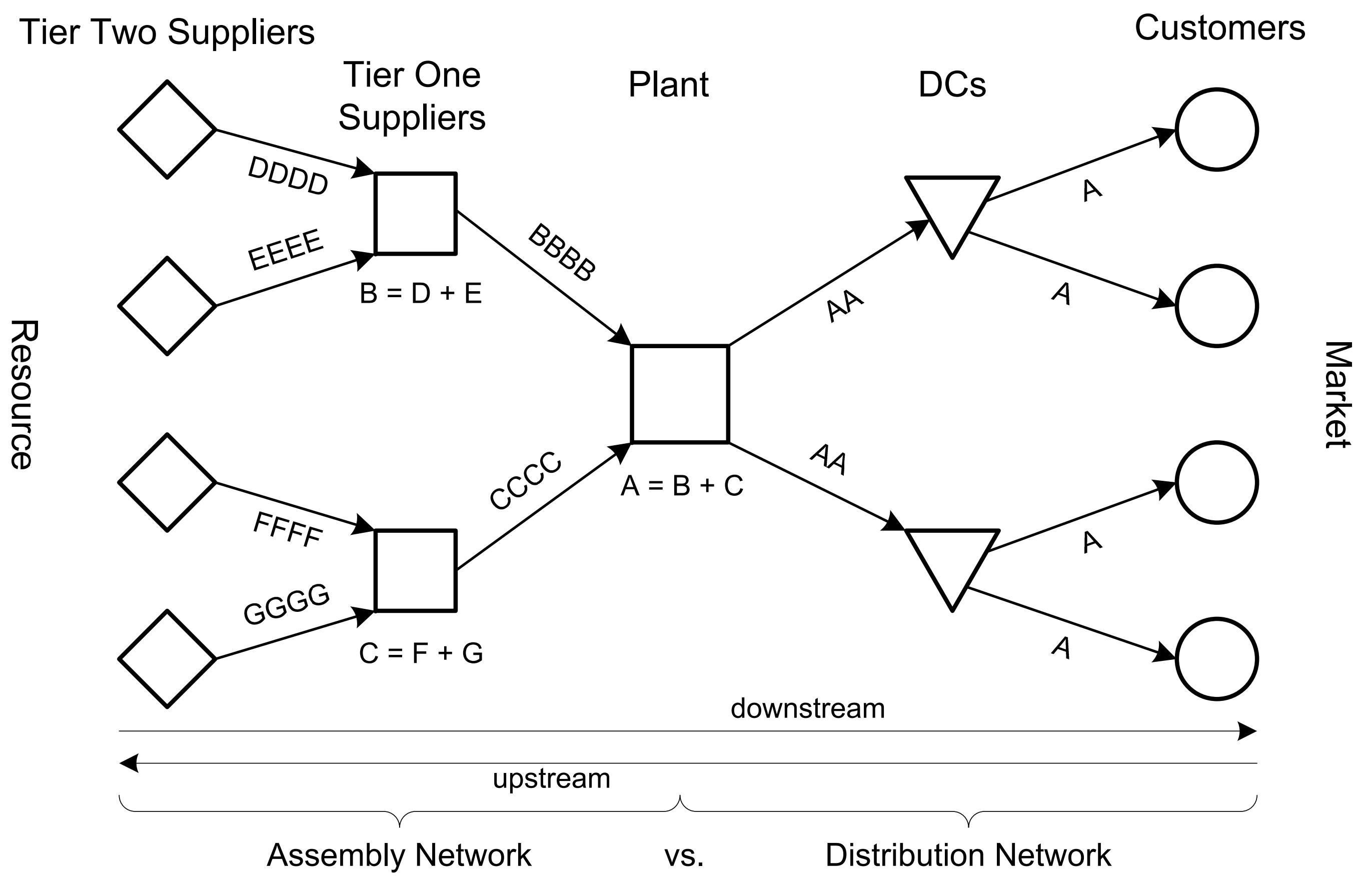
1. Order EFs so that $|x_1| \le |x_2| \le \cdots \le |x_m|$
2. Locate $x$-dimension of NF at the first EF$j$ where $\sum_{i=1}^{j} w_i \ge \frac{W}{2}$, where $W = \sum_{i=1}^{m} w_i$




$$d_1(P_1, P_2) = |x_1 - x_2| + |y_1 - y_2|$$

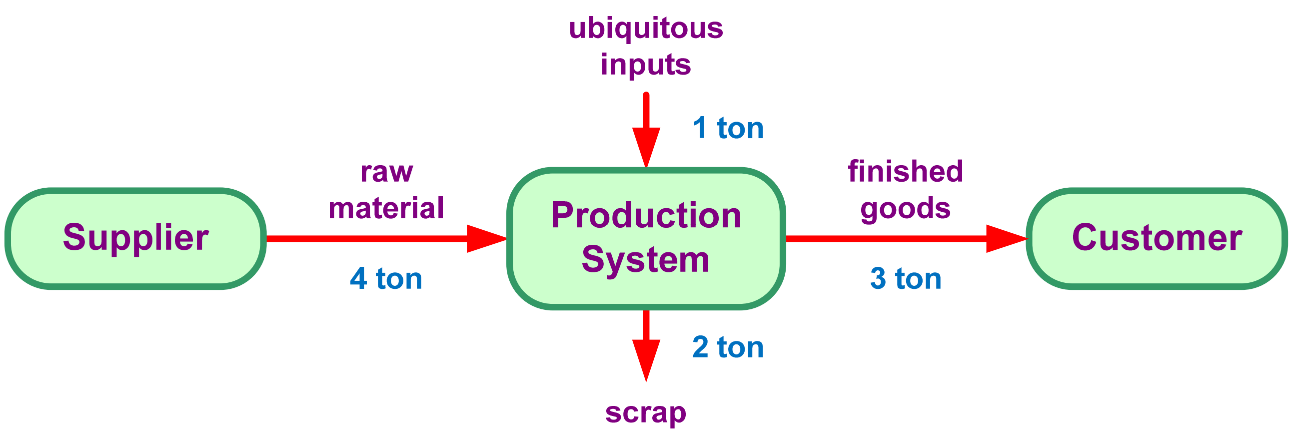
$$d_2(P_1, P_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

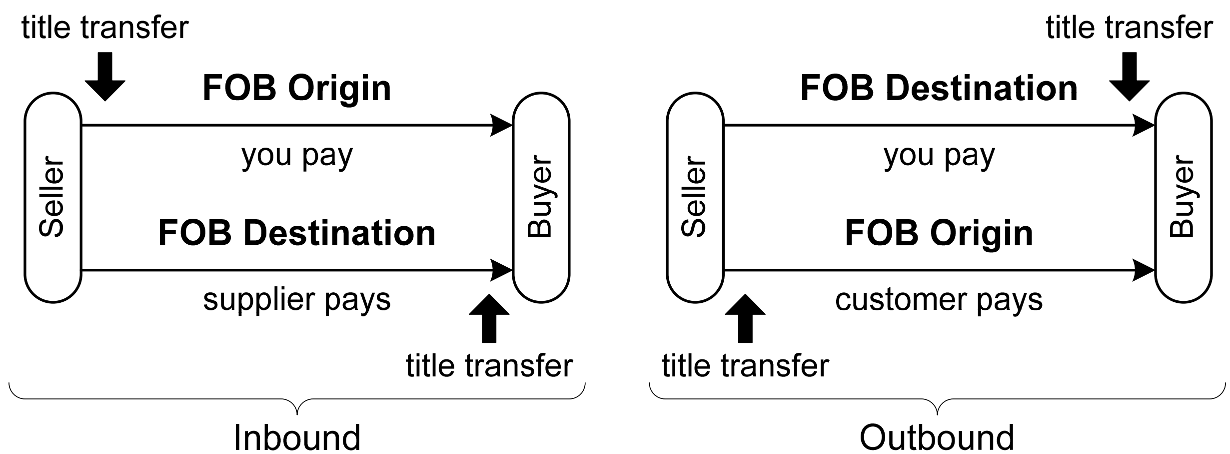
# Logistics Network for a Plant




| | | |
|---|---|---|
| Procurement | vs. | Distribution |
| Inbound Logistics | vs. | Outbound Logistics |
| Raw Materials | vs. | Finished Goods |

# Basic Production System




**FOB (free on board)**

# Monetary vs. Physical Weight

$$\min TC(X) = \sum_{i=1}^{m} w_i \, d(X, P_i) = \sum_{i=1}^{m} \underbrace{f_i \, r_i}_{w_i} d(X, P_i)$$

where

$TC$ = total transport cost (\$/yr)

$w_i$ = monetary weight (\$/mi-yr)

$f_i$ = physical weight rate (ton/yr)

$r_i$ = transport rate (\$/ton-mi)

$d(X, P_i)$ = distance between NF at $X$ and EF$_i$ at $P_i$ (mi)

NF = new facility to be located

EF = existing facility

$m$ = number of EFs

(Montetary) Weight Gaining: $\Sigma w_{\text{in}} < \Sigma w_{\text{out}}$

Physically Weight Losing: $\Sigma f_{\text{in}} > \Sigma f_{\text{out}}$

# Minisum Location: TC vs. TD

- Assuming local input costs are
  - same at every location or
  - insignificant as compared to transport costs,

  the minisum transport-oriented single-facility location problem is to locate NF to minimize TC
- Can minimize total distance (TD) if transport rate is same:

$$\min TD(X) = \sum_{i=1}^{m} w_i \, d(X, P_i) = \sum_{i=1}^{m} \underbrace{f_i \, r_i}_{w_i} d(X, P_i)$$

where $TD$ = total transport distance (mi/yr)

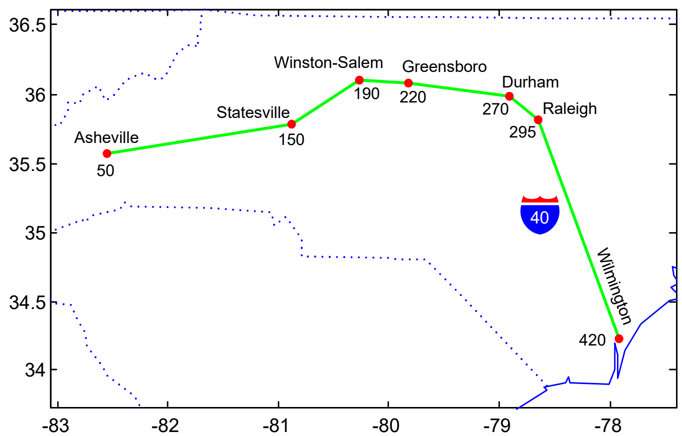
$w_i$ = monetary weight (trip/yr)

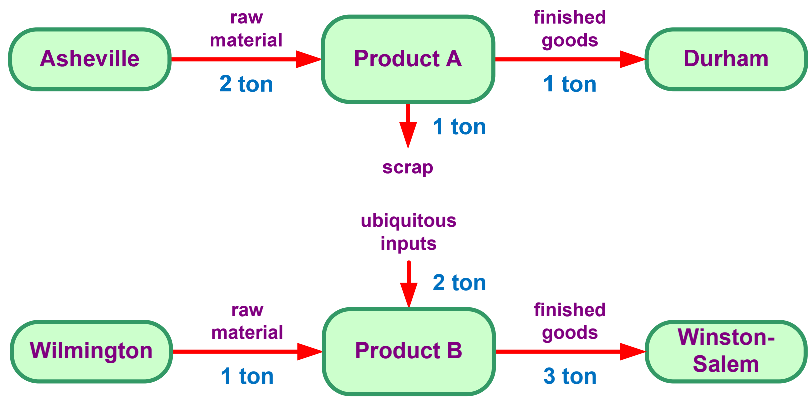
$f_i$ = trips per year (trip/yr)

$r_i$ = transport rate = 1

$d(X, P_i)$ = per-trip distance between NF and $EF_i$ (mi/trip)

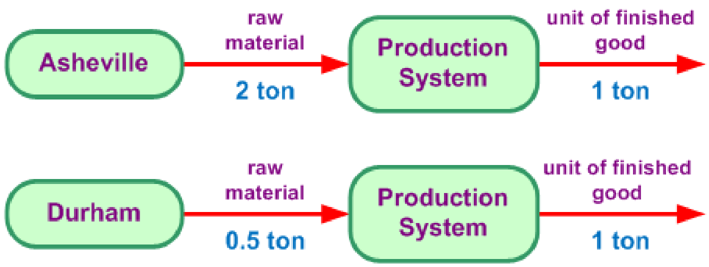
# Example: Single Supplier/Customer

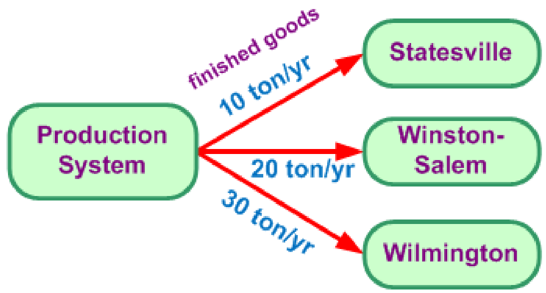



- The cost per ton-mile (i.e., the cost to ship one ton, one mile) for both raw materials and finished goods is $0.10.
  1. Where should the plant for each product be located so that procurement and distribution costs (i.e., transportation costs to and from the plant) are minimized?
  2. How would the location decision change if the customers paid for distribution costs (FOB Origin) instead of the producer (FOB Destination)?
     - In particular, what would be the impact if there were competitors located along I-40 producing the same product?

# 1-D Location with Procurement and Distribution Costs




$$\underset{(\$/\text{yr})}{TC} = \sum \underset{(\$/\text{mi-yr})}{w_i} \times \underset{(\text{mi})}{d_i}$$

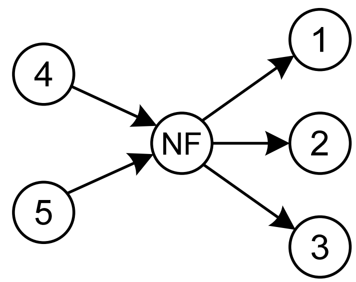
$$\underbrace{\underset{(\$/\text{mi-yr})}{w_i}}^{\text{monetary weight}} = \underbrace{\underset{(\text{ton/yr})}{f_i}}^{\text{physical weight}} \times \underset{(\$/\text{ton-mi})}{r_i}$$

Assume: all scrap is disposed of locally

$$r_{\text{in}} = \$0.33/\text{ton-mi} \qquad r_{\text{out}} = \$1.00/\text{ton-mi}$$

$$f_4 = BOM_4 \sum_{i=1}^{3} f_i = 2(60) = 120, \quad w_4 = f_4 r_{\text{in}} = 40$$

$$f_5 = BOM_5 \sum_{i=1}^{3} f_i = 0.5(60) = 30, \quad w_5 = f_5 r_{\text{in}} = 10$$



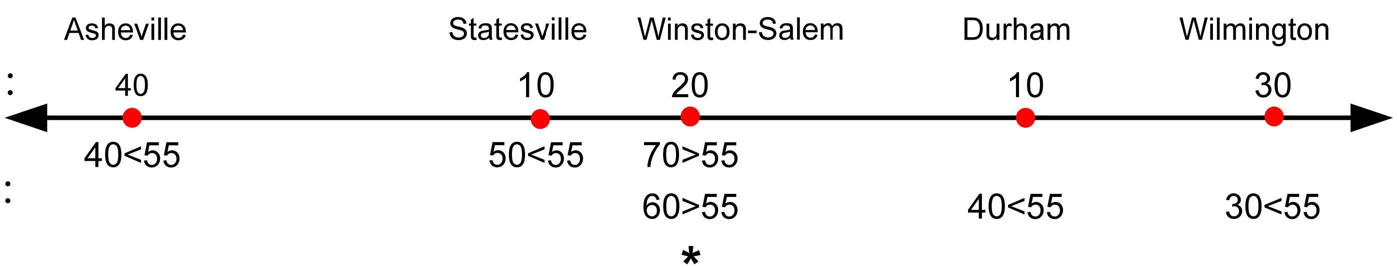

$$f_1 = 10, \quad w_1 = f_1 r_{\text{out}} = 10$$

$$f_2 = 20, \quad w_2 = f_2 r_{\text{out}} = 20$$

$$f_3 = 30, \quad w_3 = f_3 r_{\text{out}} = 30$$

$w_i$ :

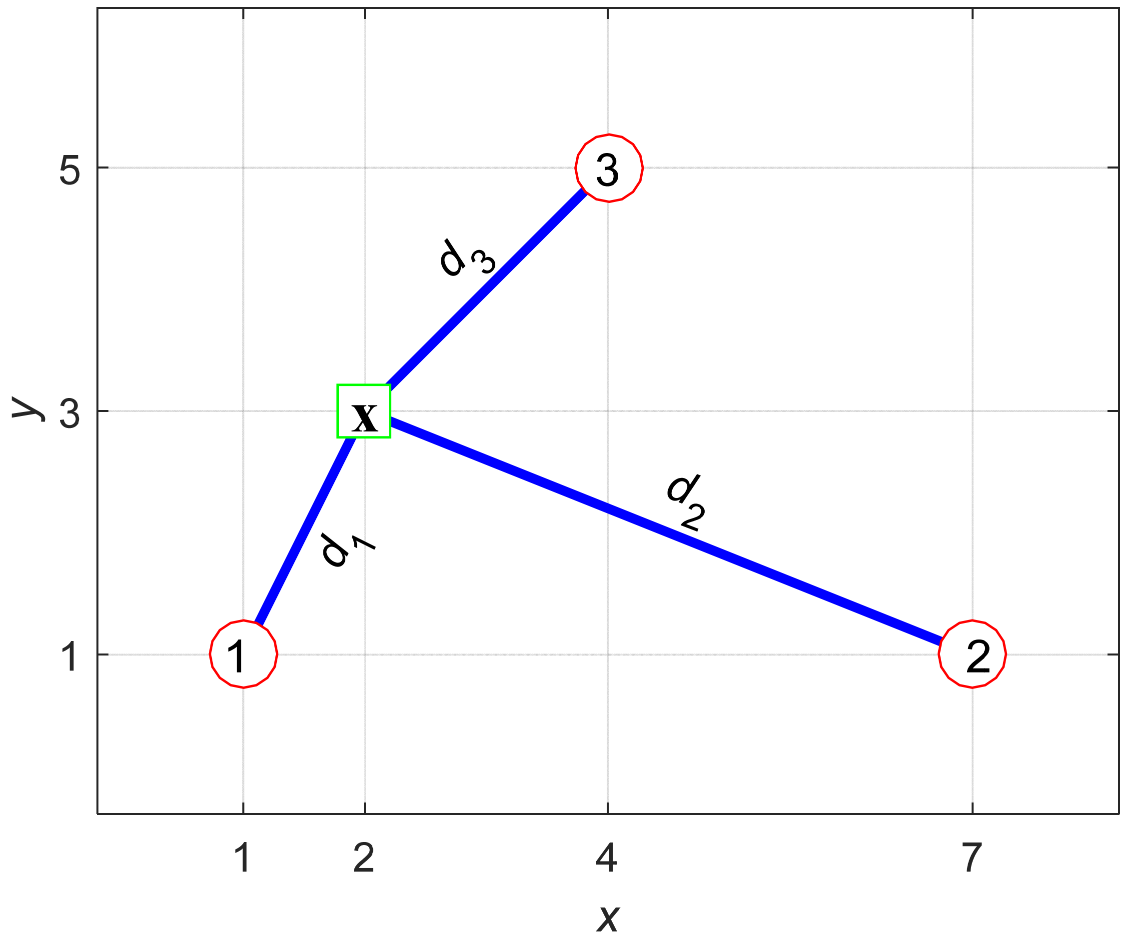
$\sum_{i=1}^{j} w_i \le \frac{W}{2}$ :




$$\text{(Montetary) Weight Gaining:} \quad \Sigma w_{\text{in}} = 50 < \Sigma w_{\text{out}} = 60$$

$$\text{Physically Weight Losing:} \quad \Sigma f_{\text{in}} = 150 > \Sigma f_{\text{out}} = 60$$

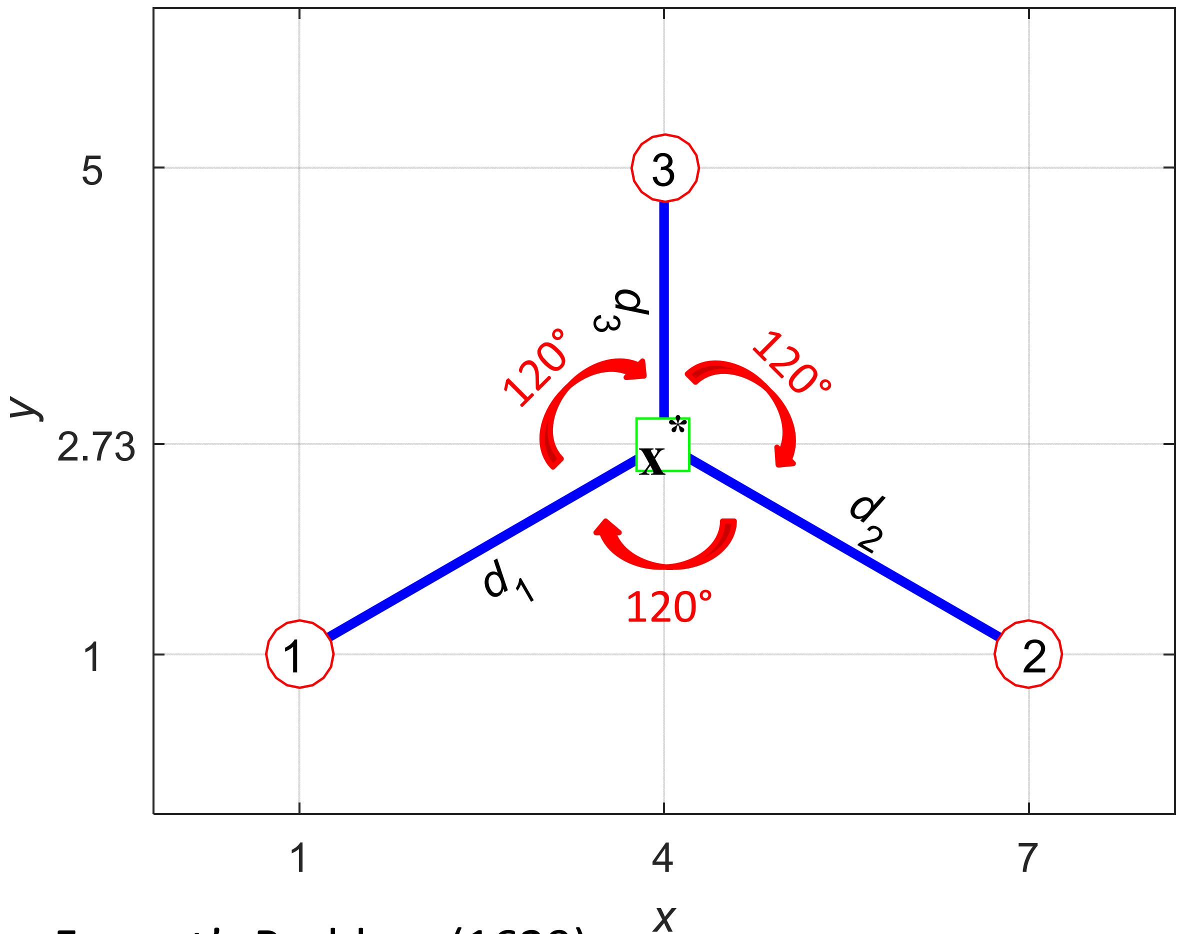
# 2-D Euclidean Distance




$$\mathbf{x} = \begin{bmatrix} 2 & 3 \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} 1 & 1 \\ 7 & 1 \\ 4 & 5 \end{bmatrix}$$

$$\mathbf{d} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} \sqrt{\left(x_1 - p_{1,1}\right)^2 + \left(x_2 - p_{1,2}\right)^2} \\ \sqrt{\left(x_1 - p_{2,1}\right)^2 + \left(x_2 - p_{2,2}\right)^2} \\ \sqrt{\left(x_1 - p_{3,1}\right)^2 + \left(x_2 - p_{3,2}\right)^2} \end{bmatrix}$$

# Minisum Distance Location




$$\mathbf{P} = \begin{bmatrix} 1 & 1 \\ 7 & 1 \\ 4 & 5 \end{bmatrix}$$

$$d_i(\mathbf{x}) = \sqrt{\left(x_1 - p_{i,1}\right)^2 + \left(x_2 - p_{i,2}\right)^2}$$

$$TD(\mathbf{x}) = \sum_{i=1}^{3} d_i(\mathbf{x})$$

$$\mathrm{x}^* = \arg\min_{\mathbf{x}} TD(\mathbf{x})$$

$$TD^* = TD(\mathbf{x}^*)$$

Fermat’s Problem (1629):

Given three points, find fourth (Steiner point) such that sum to others is minimized

(Solution: Optimal location corresponds to all angles = 120°)

# Minisum Weighted-Distance Location

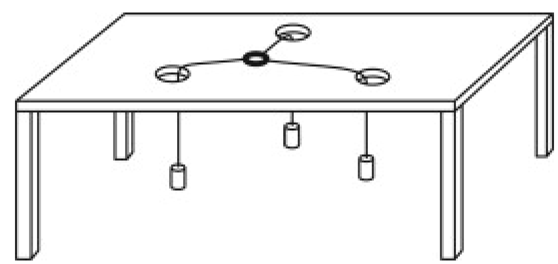
- Solution for 2-D+ and non-rectangular distances:
  - *Majority Theorem:* Locate NF at EF*j* if $w_j \geq \frac{W}{2}, \quad \text{where } W = \sum_{i=1}^{m} w_i$
  - Mechanical (Varigon frame)
  - 2-D rectangular approximation
  - Numerical: nonlinear unconstrained optimization
    - Analytical/estimated derivative (quasi-Newton, `fminunc`)
    - Direct, derivative-free (Nelder-Mead, `fminsearch`)

$$m = \text{number of EFs}$$

$$TC(\mathbf{x}) = \sum_{i=1}^{m} w_i d_i(\mathbf{x})$$

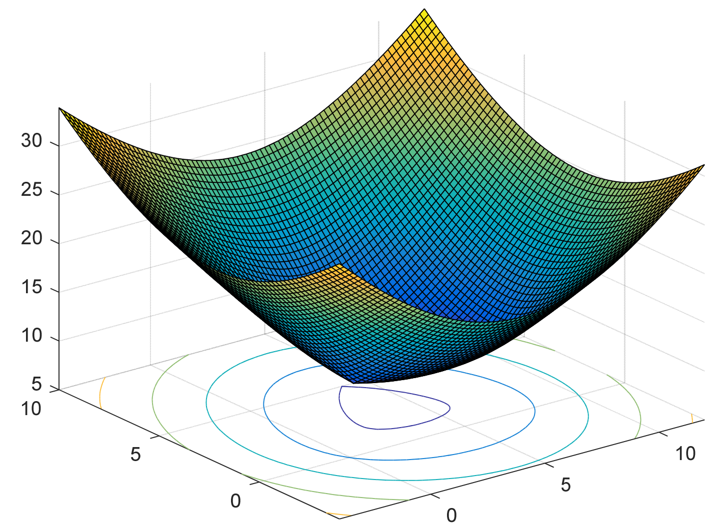
$$\mathbf{x}^* = \arg\min_{\mathbf{x}} TC(\mathbf{x})$$

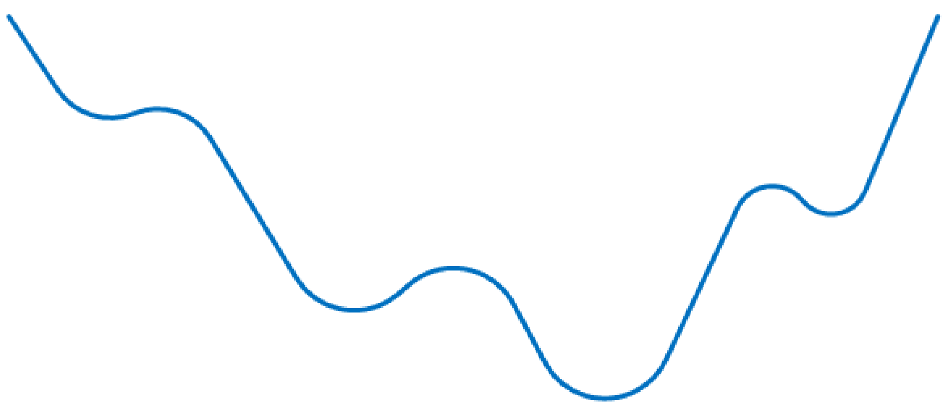
$$TC^* = TC(\mathbf{x}^*)$$



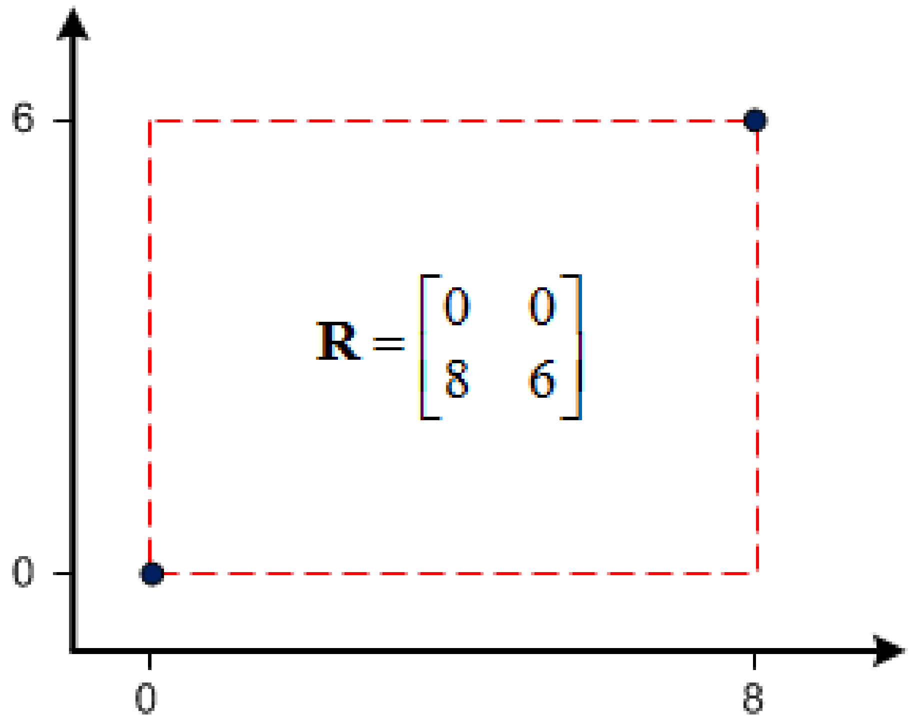
Varignon Frame

# Convex vs Nonconvex Optimization

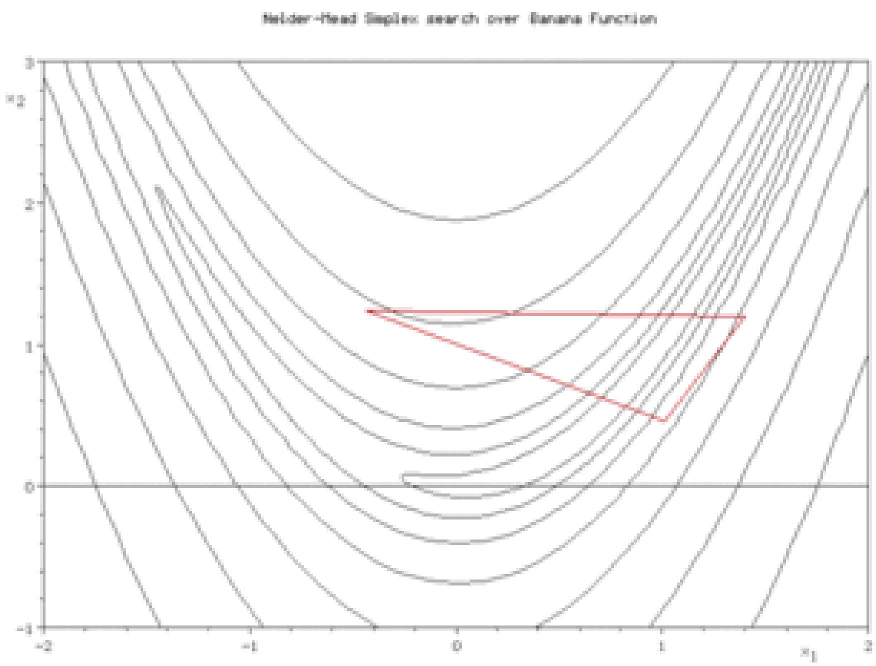


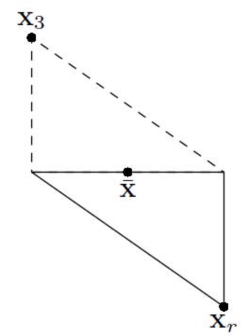



# Feasible Region

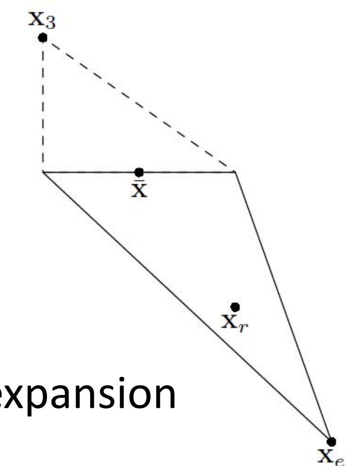


# Nelder-Mead Simplex Method



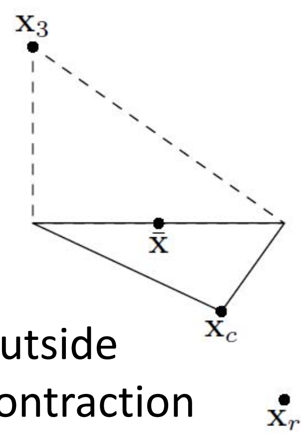




reflection

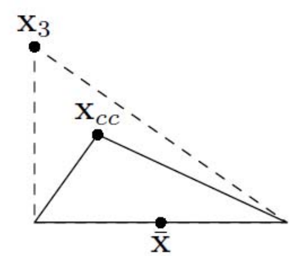



expansion

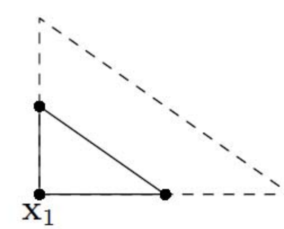
- AKA amoeba method
- Simplex is triangle in 2-D (dashed line in figures)




outside contraction



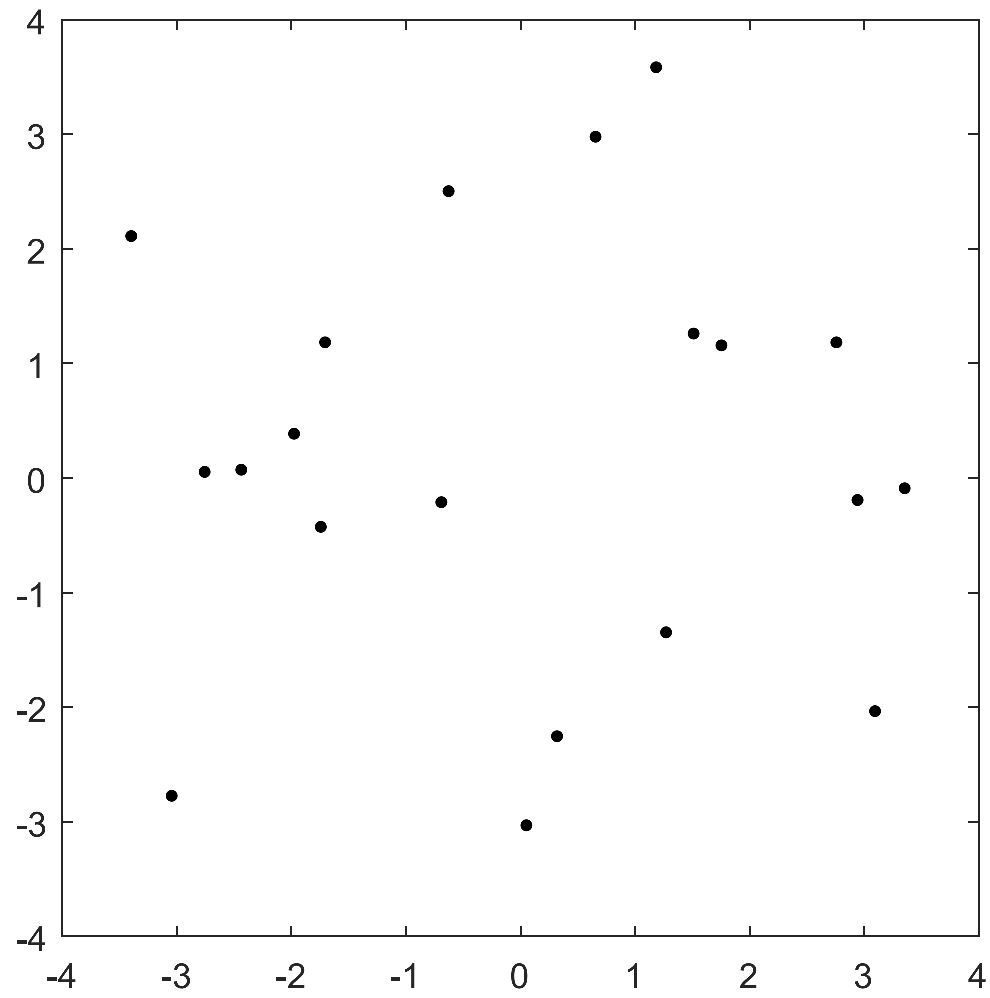

inside contraction




a shrink

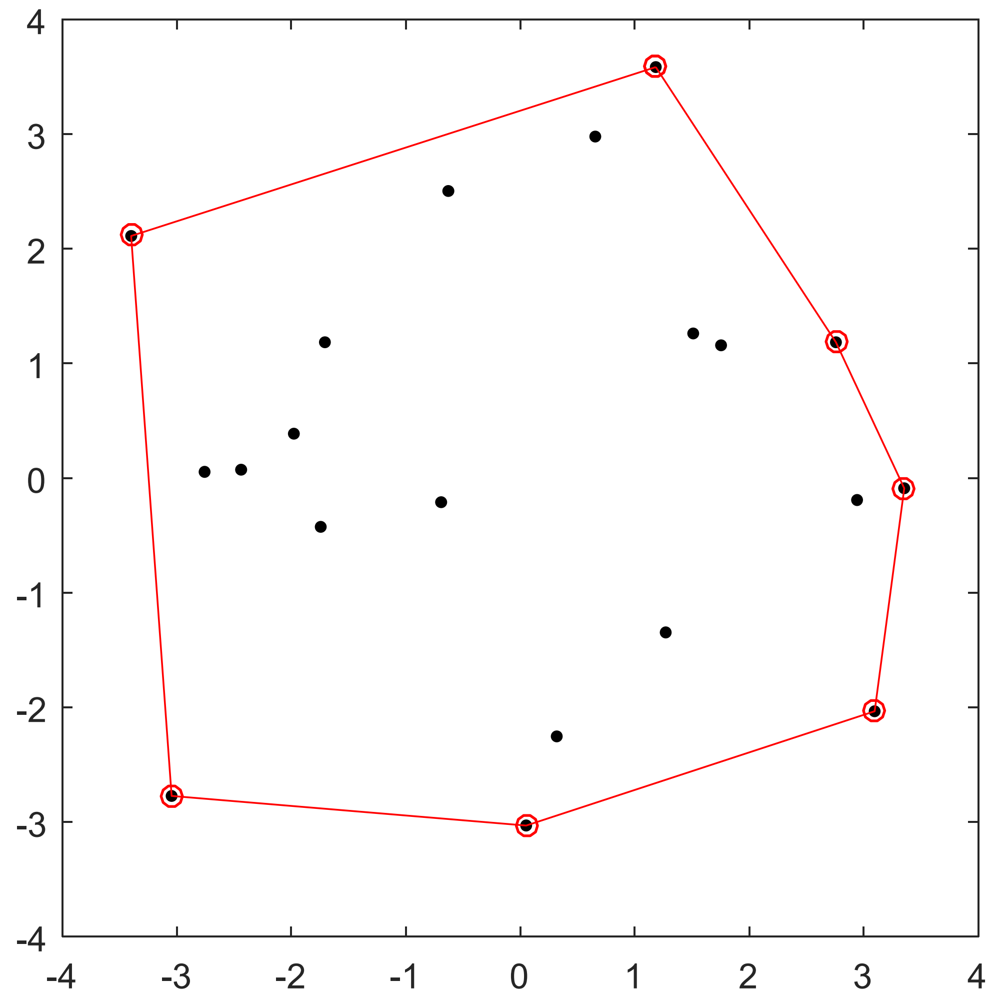
# Computational Geometry

- Design and analysis of algorithms for solving geometric problems
  - Modern study started with Michael Shamos in 1975
- Facility location:
  - geometric data structures used to “simplify” solution procedures



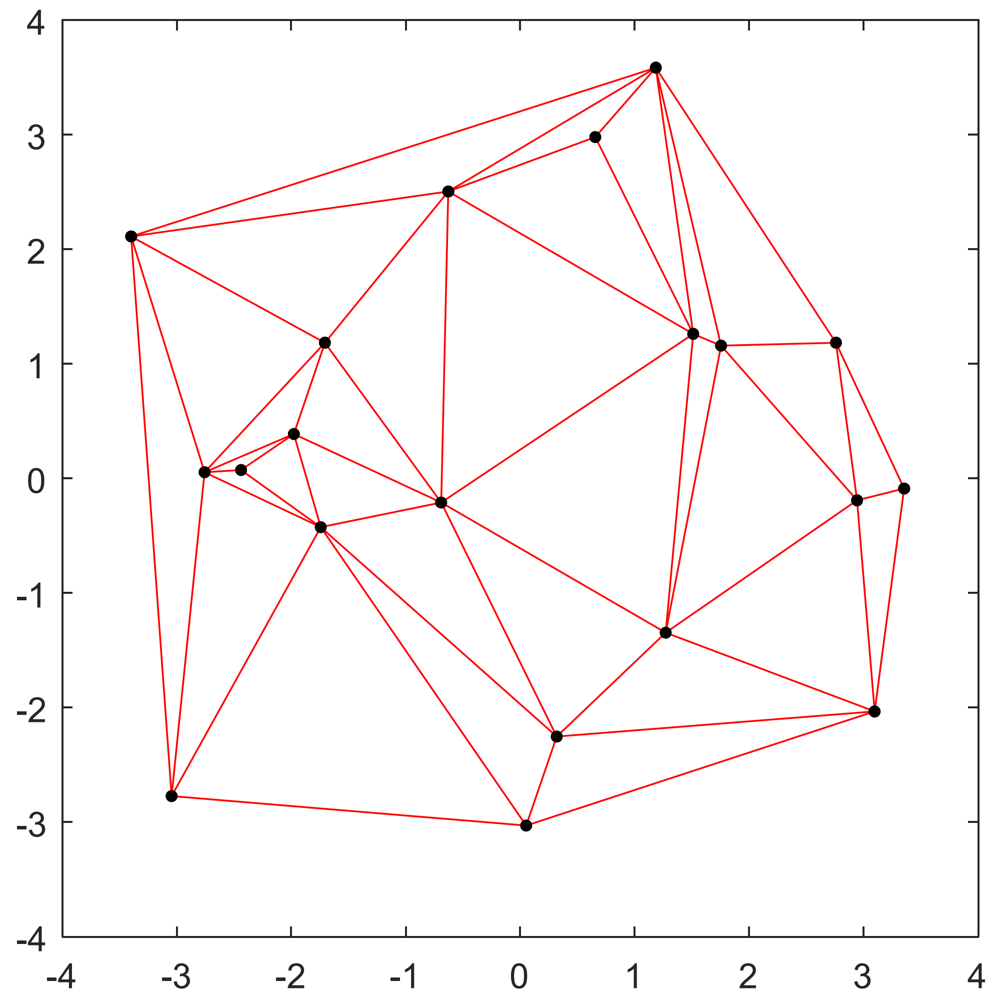
# Convex Hull

- Find the points that enclose all points
  - Most important data structure
  - Calculated, via Graham's scan in $O(n \log n), \quad n$ points



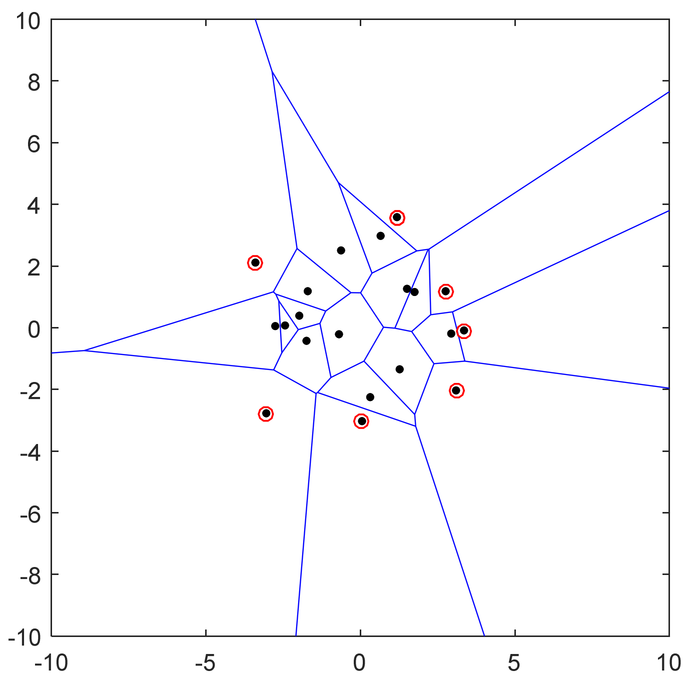
# Delaunay Triangulation

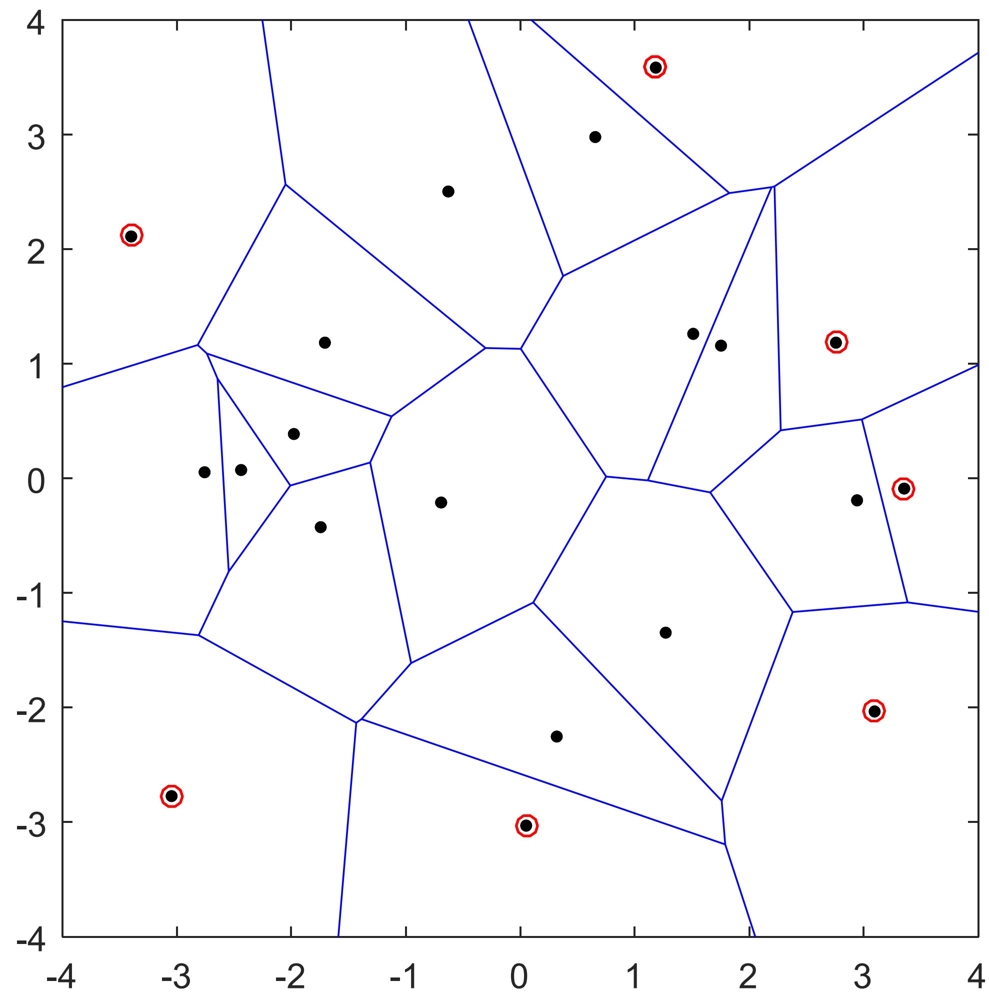
- Find the triangulation of points that maximizes the minimum angle of any triangle
  - Captures proximity relationships
  - Used in 3-D animation
  - Calculated, via divide and conquer, in $O(n \log n)$, $n$ points



# Voronoi Diagram

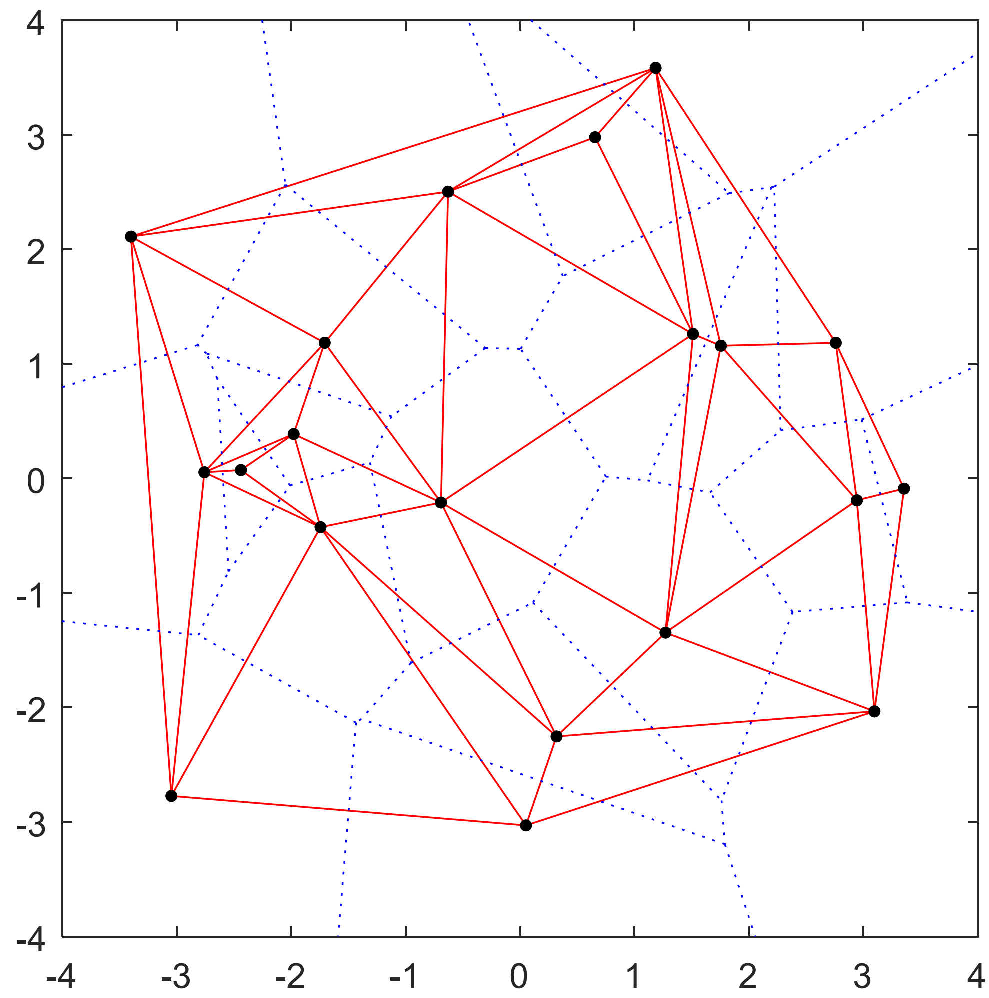
- Each region defines area closest to a point
  - Open face regions indicate points in convex hull



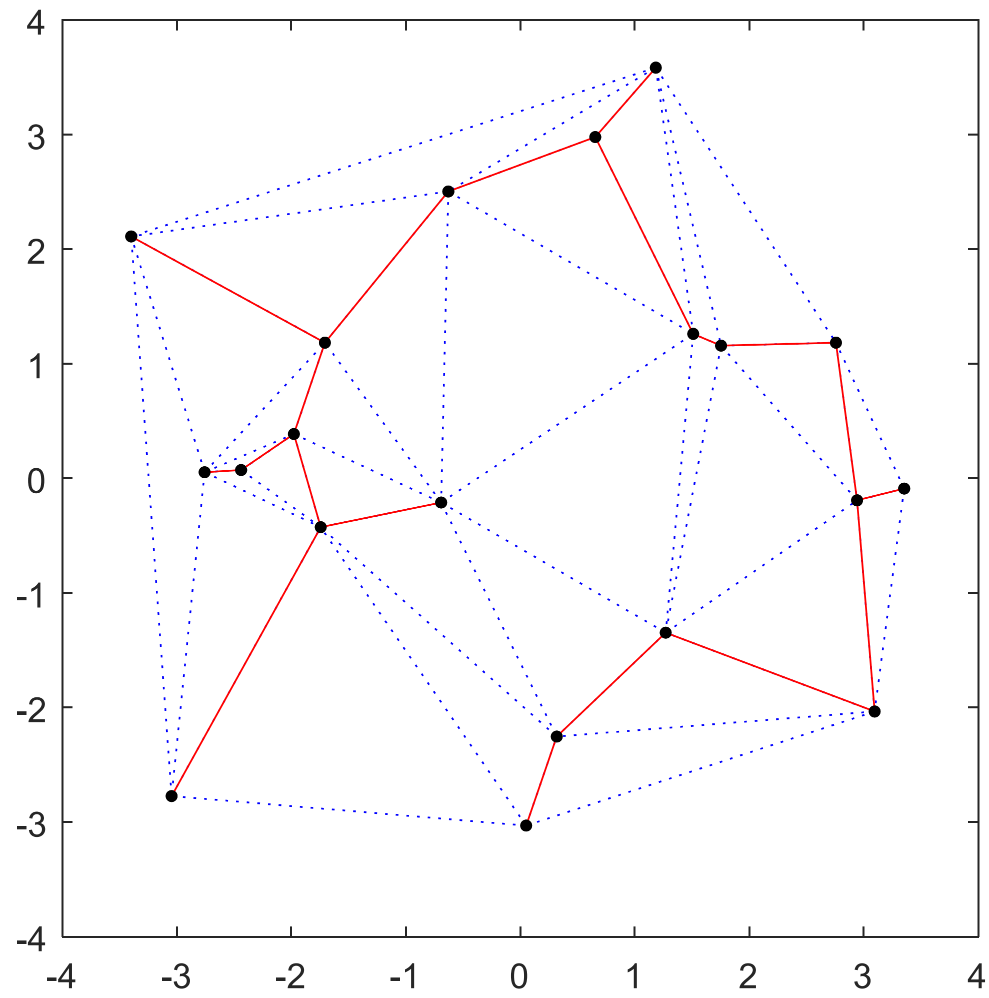
# Delaunay-Voronoi

- Delaunay triangulation is straight-line dual of Voronoi diagram
  - Can easily convert from one to another



# Minimum Spanning Tree

- Find the minimum weight set of arcs that connect all nodes in a graph
  - *Undirected* arcs: calculated, via Kruskal's algorithm, $O(m \log n)$, $m$ arcs, $n$ nodes
  - *Directed* arcs: calculated, via Edmond's branching algorithm, in $O(mn)$, $m$ arcs, $n$ nodes



# Steiner Network

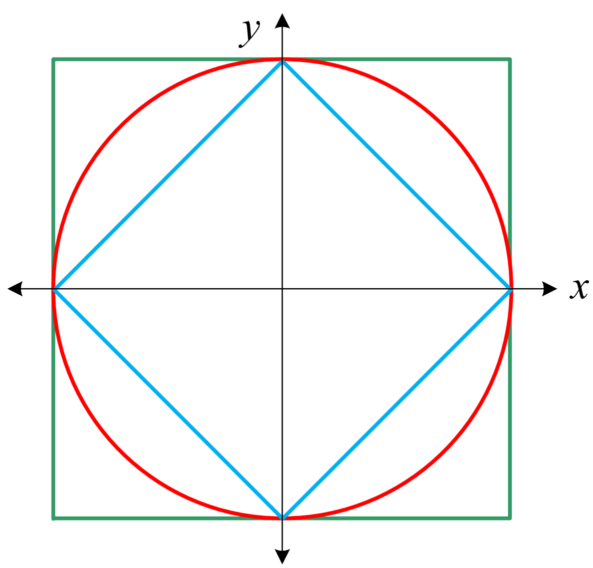
# Metric Distances

General $l_p$: $\quad d_p(P_1, P_2) = \left[ |x_1 - x_2|^p + |y_1 - y_2|^p \right]^{\frac{1}{p}}, \quad p \geq 1$

Rectilinear $(p=1)$: $\quad d_1(P_1, P_2) = |x_1 - x_2| + |y_1 - y_2|$

Euclidean $(p=2)$: $\quad d_2(P_1, P_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

Chebychev $(p \to \infty)$: $\quad d_\infty(P_1, P_2) = \max\left\{ |x_1 - x_2|, |y_1 - y_2| \right\}$



# Chebychev Distances

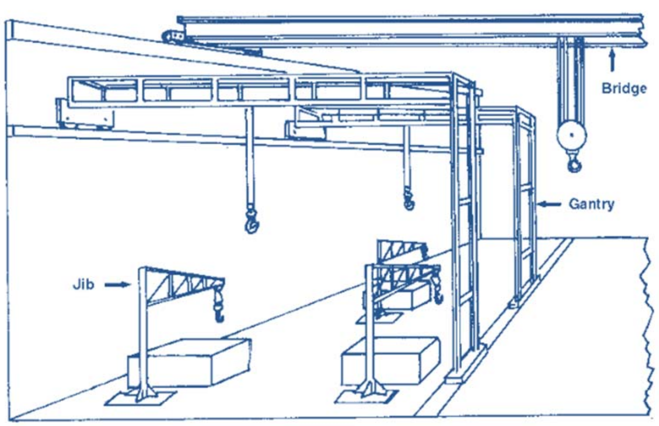
## Proof

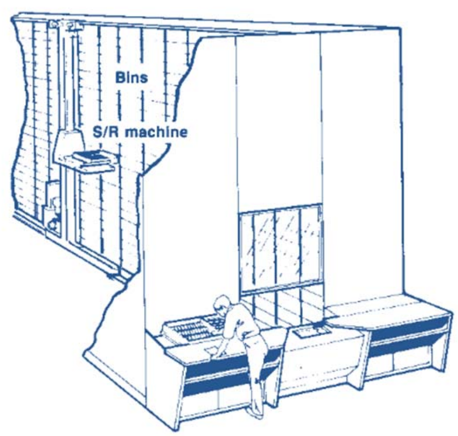
Without loss of generality, let $P_1 = (x, y)$, for $x, y \geq 0$, and $P_2 = (0, 0)$. Then $d_\infty(P_1, P_2) = \max\{x, y\}$ and $d_p(P_1, P_2) = \left[x^p + y^p\right]^{1/p}$.

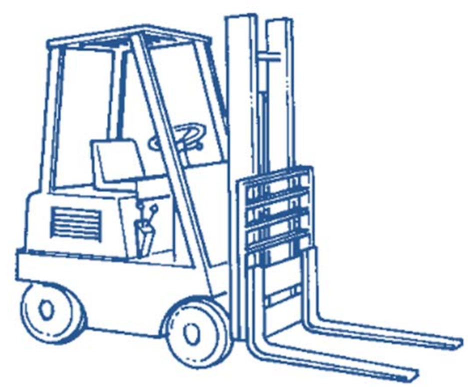
If $x = y$, then $\lim_{p\to\infty}\left[x^p + y^p\right]^{1/p} = \lim_{p\to\infty}\left[2x^p\right]^{1/p} = \lim_{p\to\infty}\left[2^{1/p}x\right] = x$.

If $x < y$, then $\lim_{p\to\infty}\left[x^p + y^p\right]^{1/p} = \lim_{p\to\infty}\left[\left((x/y)^p + 1\right)y^p\right]^{1/p} = \lim_{p\to\infty}\left((x/y)^p + 1\right)^{1/p} y = 1 \cdot y = y$.

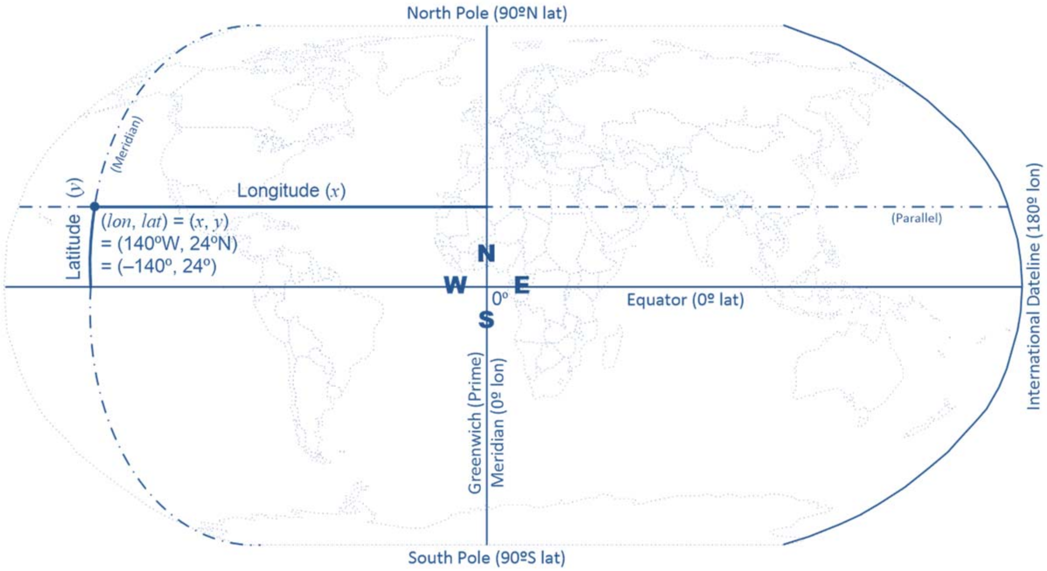
A similar argument can be made if $x > y$. ■

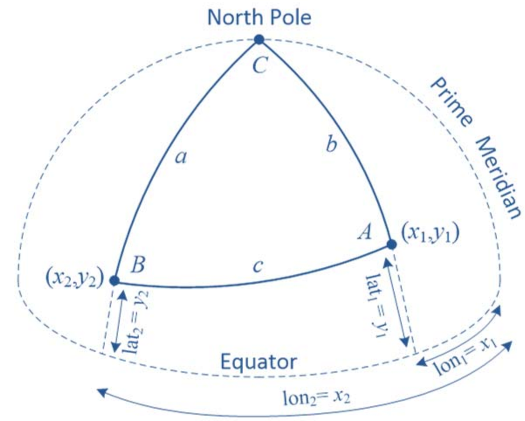


# Great Circle Distances

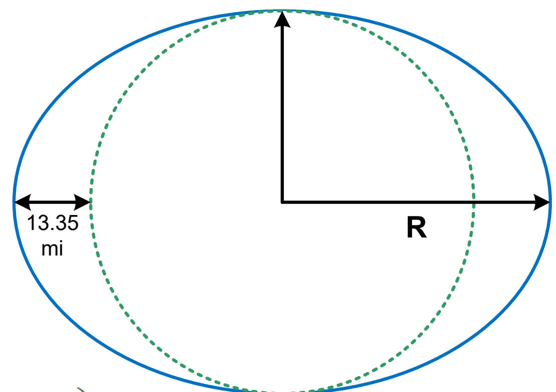



$$(lon_1, lat_1) = (x_1, y_1), \quad (lon_2, lat_2) = (x_2, y_2)$$

$$d_{rad} = (\text{great circle distance in radians of a sphere})$$

$$= \cos^{-1}\left[\sin y_2 \sin y_1 + \cos y_2 \cos y_1 \cos(x_1 - x_2)\right]$$




$$R = (\text{radius of earth at equator}) - (\text{bulge from north pole to equator})$$

$$= 3{,}963.34 - 13.35 \sin\left(\frac{y_1 + y_2}{2}\right) \text{ mi}, \quad = 6{,}378.388 - 21.476 \sin\left(\frac{y_1 + y_2}{2}\right) \text{ km}$$

$$d_{GC} = \text{distance } (x_1, y_1) \text{ to } (x_2, y_2) = \boxed{d_{rad} \cdot R}$$

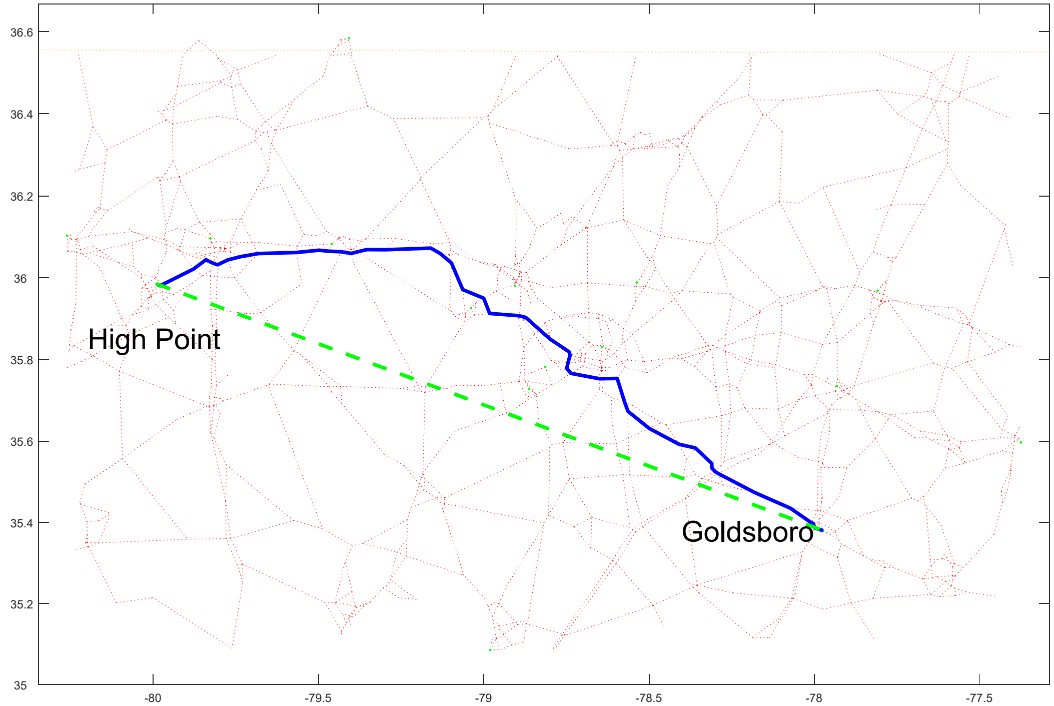
$$x_{\deg} = \begin{cases} DD + \dfrac{MM}{60} + \dfrac{SS}{3{,}600}, & \text{if } E \text{ or } N \\ -DD - \dfrac{MM}{60} - \dfrac{SS}{3{,}600}, & \text{if } W \text{ or } S \end{cases}$$

# Circuity Factor

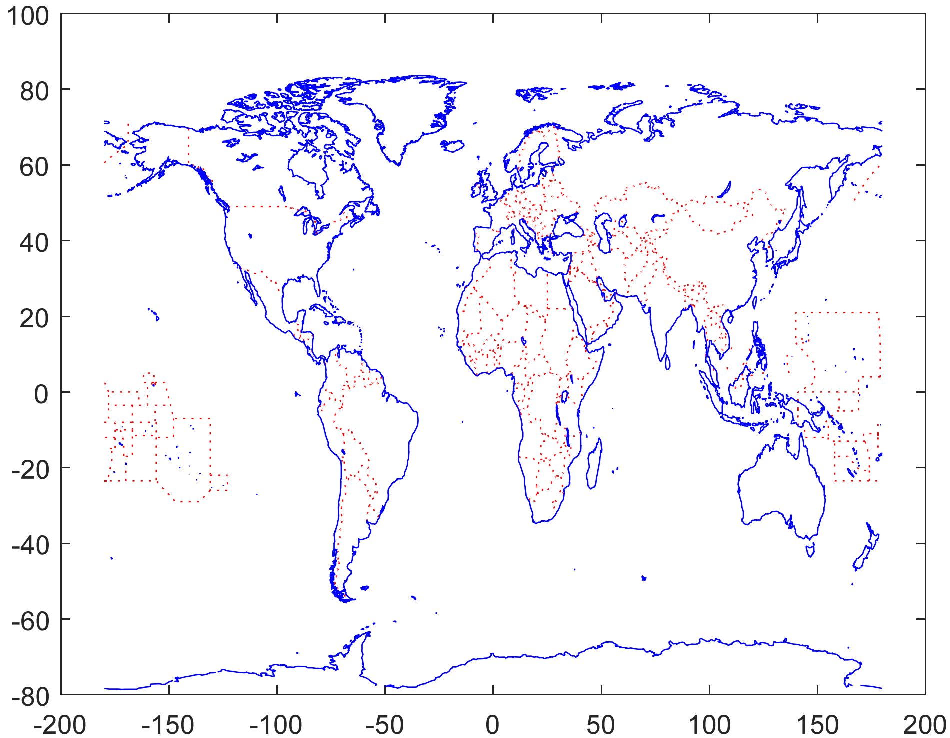
$$\textit{Circuity Factor}: \quad g = \sum \frac{d_{\mathrm{road}_i}}{d_{GC_i}}, \quad \text{where usually } 1.15 \le g \le 1.5$$

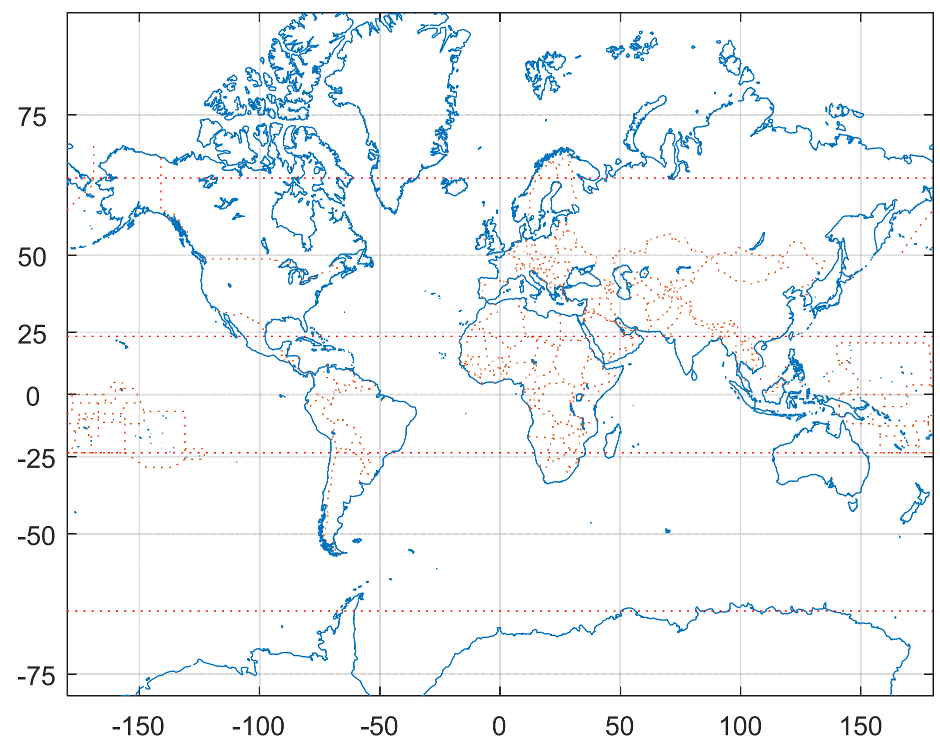
$$d_{\mathrm{road}} \approx g \cdot d_{GC}(P_1, P_2), \quad \text{estimated road distance from } P_1 \text{ to } P_2$$

**From High Point to Goldsboro: Road = 143 mi, Great Circle = 121 mi, Circuity = 1.19**



# Mercator Projection



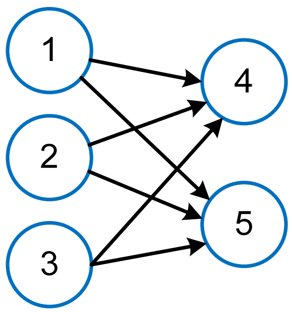

$$x_{\text{rad}} = \frac{x_{\text{deg}}}{180}\pi \quad \text{and} \quad x_{\text{deg}} = \frac{x_{\text{rad}} \cdot 180}{\pi}$$

$$x_{\text{proj}} = x$$

$$y_{\text{proj}} = \sinh^{-1}(\tan y)$$

$$y = \tan^{-1}(\sinh y_{\text{proj}})$$

# Metric Distances using `dists`




$$\mathbf{D} = \begin{array}{c|cc} & 4 & 5 \\ \hline 1 & \bullet & \bullet \\ 2 & \bullet & \bullet \\ 3 & \bullet & \bullet \end{array}_{\substack{3\times 2 \\ n\times m}} = \text{dists}(\underset{\substack{3\times 2 \\ n\times d}}{\mathbf{X1}}, \underset{\substack{2\times 2 \\ m\times d}}{\mathbf{X2}}, p), \quad p = \begin{cases} \text{'mi'} & \text{'km'} \\ 1 & 2 & \text{Inf} \end{cases}$$

***d* = 2**

$$\mathbf{X1} = \begin{bmatrix} \bullet & \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet & \bullet \end{bmatrix} \Rightarrow \mathbf{d} = \begin{bmatrix} \bullet \end{bmatrix}$$

$$\mathbf{X1} = \begin{bmatrix} \bullet & \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \\ \bullet & \bullet \end{bmatrix} \Rightarrow \mathbf{d} = \begin{bmatrix} \bullet & \bullet & \bullet \end{bmatrix}$$

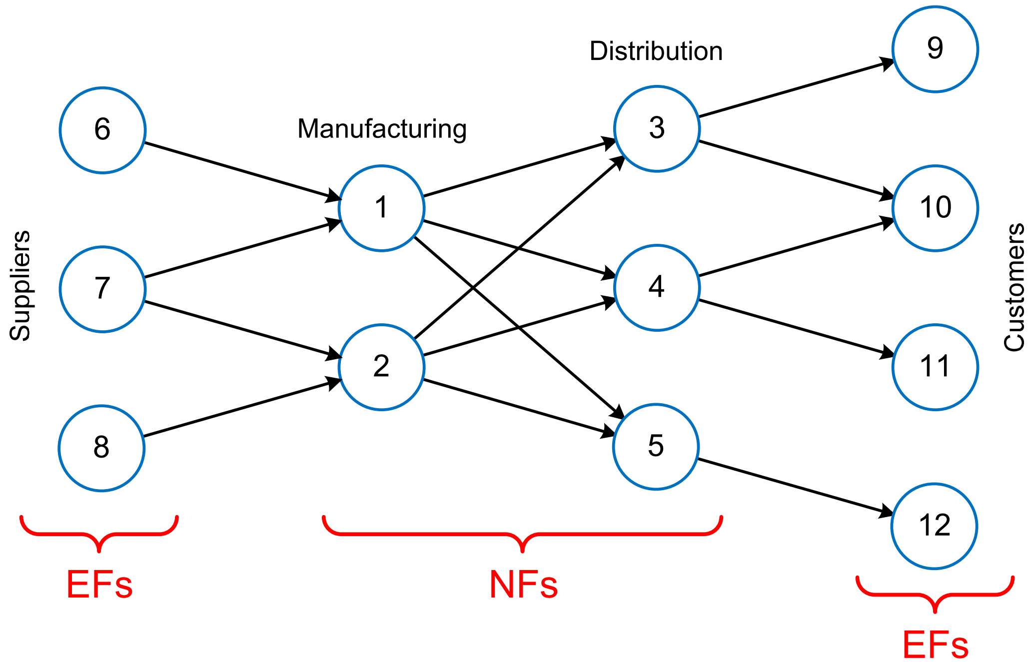
$$\mathbf{X1} = \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \\ \bullet & \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \\ \bullet & \bullet \end{bmatrix} \Rightarrow \mathbf{D} = \begin{bmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{bmatrix}$$

***d* = 1**

$$\mathbf{X1} = \begin{bmatrix} \bullet \\ \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet \\ \bullet \end{bmatrix} \Rightarrow \mathbf{D} = \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \end{bmatrix}$$

$$\mathbf{X1} = \begin{bmatrix} \bullet \\ \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet & \bullet \end{bmatrix} \Rightarrow \textit{Error}$$

# Minisum Multifacility Location




$n =$ no. of NFs, $\quad m =$ no. of EFs

$\mathbf{X}_{n\times d} =$ NF locations, $\quad \mathbf{P}_{m\times d} =$ EF locations

$$\mathbf{V}_{n\times n} = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 0 & + & + & + \\ 2 & 0 & 0 & + & + & + \\ 3 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 & 0 & 0 \end{array}$$

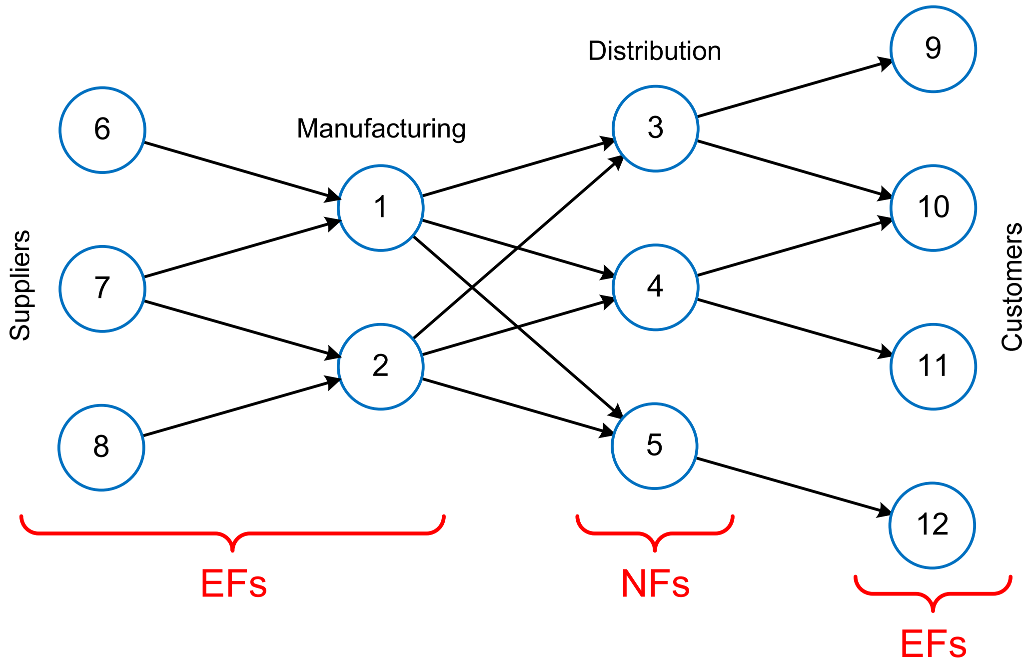
$$\mathbf{W}_{n\times m} = \begin{array}{c|ccccccc} & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ \hline 1 & + & + & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & + & + & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & + & + & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & + & + & 0 \\ 5 & 0 & 0 & 0 & 0 & 0 & 0 & + \end{array}$$

$$TC(\mathbf{X}) = \sum_{j=1}^{n}\sum_{k=1}^{n} v_{jk} d(\mathbf{X}_j, \mathbf{X}_k) + \sum_{j=1}^{n}\sum_{i=1}^{m} w_{ji} d(\mathbf{X}_j, \mathbf{P}_i)$$

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} TC(\mathbf{X})$$

$$TC^* = TC(\mathbf{X}^*)$$

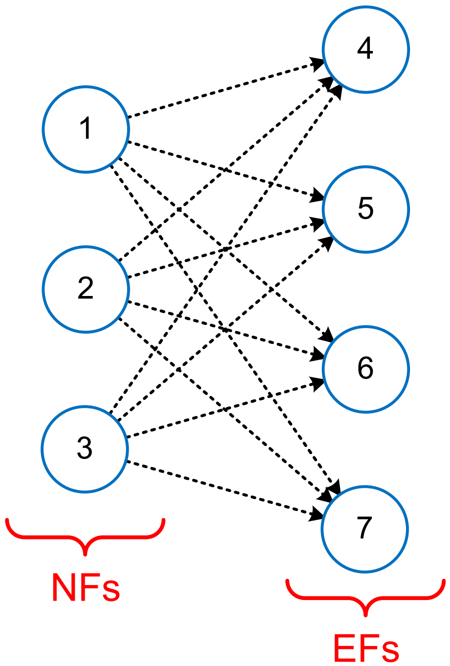
# Multiple Single-Facility Location




$$TC(\mathbf{X}) = \sum_{j=1}^{n}\sum_{k=1}^{n} v_{jk} d(\mathbf{X}_j, \mathbf{X}_k) + \sum_{j=1}^{n}\sum_{i=1}^{m} w_{ji} d(\mathbf{X}_j, \mathbf{P}_i)$$

$$= \sum_{j=1}^{n} TC(\mathbf{X}_j)$$

# Facility Location–Allocation Problem




- Determine both the location of *n* NFs and the allocation of flow requirements of *m* EFs that minimize TC

$$w_{ji} = r_{ji} f_{ji} = (1) f_{ji} = \text{flow between NF}j \text{ and EF}i$$

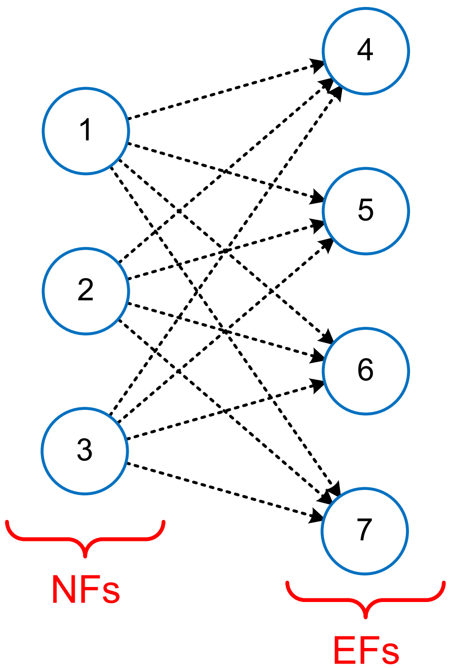
$$w_i = \text{total flow requirememts of EF}i$$

$$TC(\mathbf{X},\mathbf{W}) = \sum_{j=1}^{n}\sum_{i=1}^{m} w_{ji} d(\mathbf{X}_j, \mathbf{P}_i)$$

$$\mathbf{X}^*, \mathbf{W}^* = \arg\min_{\mathbf{X},\mathbf{W}} \left\{ TC(\mathbf{X},\mathbf{W}) : \sum_{j=1}^{n} w_{ji} = w_i,\ w_{ji} \geq 0 \right\}$$

$$TC^* = TC(\mathbf{X}^*, \mathbf{W}^*)$$

# Integrated Formulation




- If there are no capacity constraints on NFs, it is optimal to always satisfy all the flow requirements of an EF from its closest NF
- Requires search of ($n$ x $d$)-dimensional TC that combines location with allocation

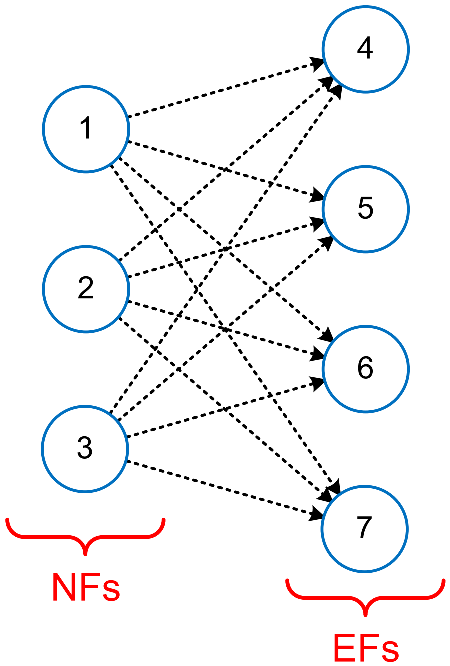
$$\alpha_i(\mathbf{X}) = \arg\min_j d(\mathbf{X}_j, \mathbf{P}_i)$$

$$TC(\mathbf{X}) = \sum_{i=1}^{m} w_i d(\mathbf{X}_{\alpha_i(\mathbf{X})}, \mathbf{P}_i)$$

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} TC(\mathbf{X})$$

$$TC^* = TC(\mathbf{X}^*)$$

# Alternating Formulation



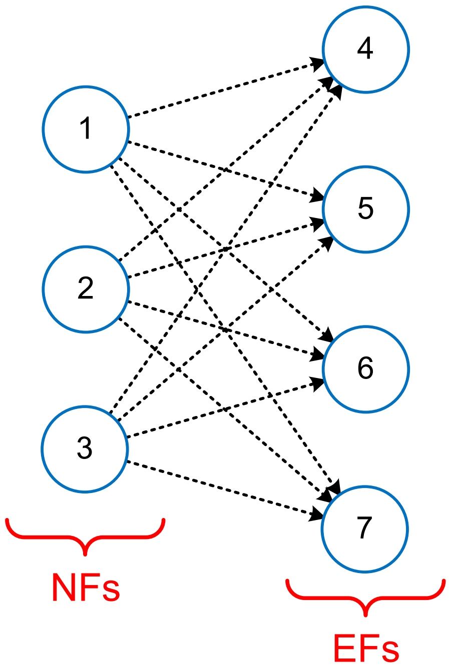

- Alternate between finding locations and finding allocations until no further TC improvement
- Requires *n* *d*-dimensional location searches together with separate allocation procedure
- Separating location from allocation allows other types of location and/or allocation procedures to be used:
  - Allocation with NF with capacity constraints (solved as minimum cost network flow problem)
  - Location with some NFs at fixed locations

$$allocate(\mathbf{X}) = \left[ w_{ji} \right] = \begin{cases} w_i, & \text{if } \arg\min_k d(\mathbf{X}_k, \mathbf{P}_i) = j \\ 0, & \text{otherwise} \end{cases}$$

$$TC(\mathbf{X}, \mathbf{W}) = \sum_{j=1}^{n} \sum_{i=1}^{m} w_{ji} d(\mathbf{X}_j, \mathbf{P}_i)$$

$$locate(\mathbf{W}, \mathbf{X}) = \arg\min_{\mathbf{X}} TC(\mathbf{X}, \mathbf{W})$$

# ALA: Alternate Location–Allocation



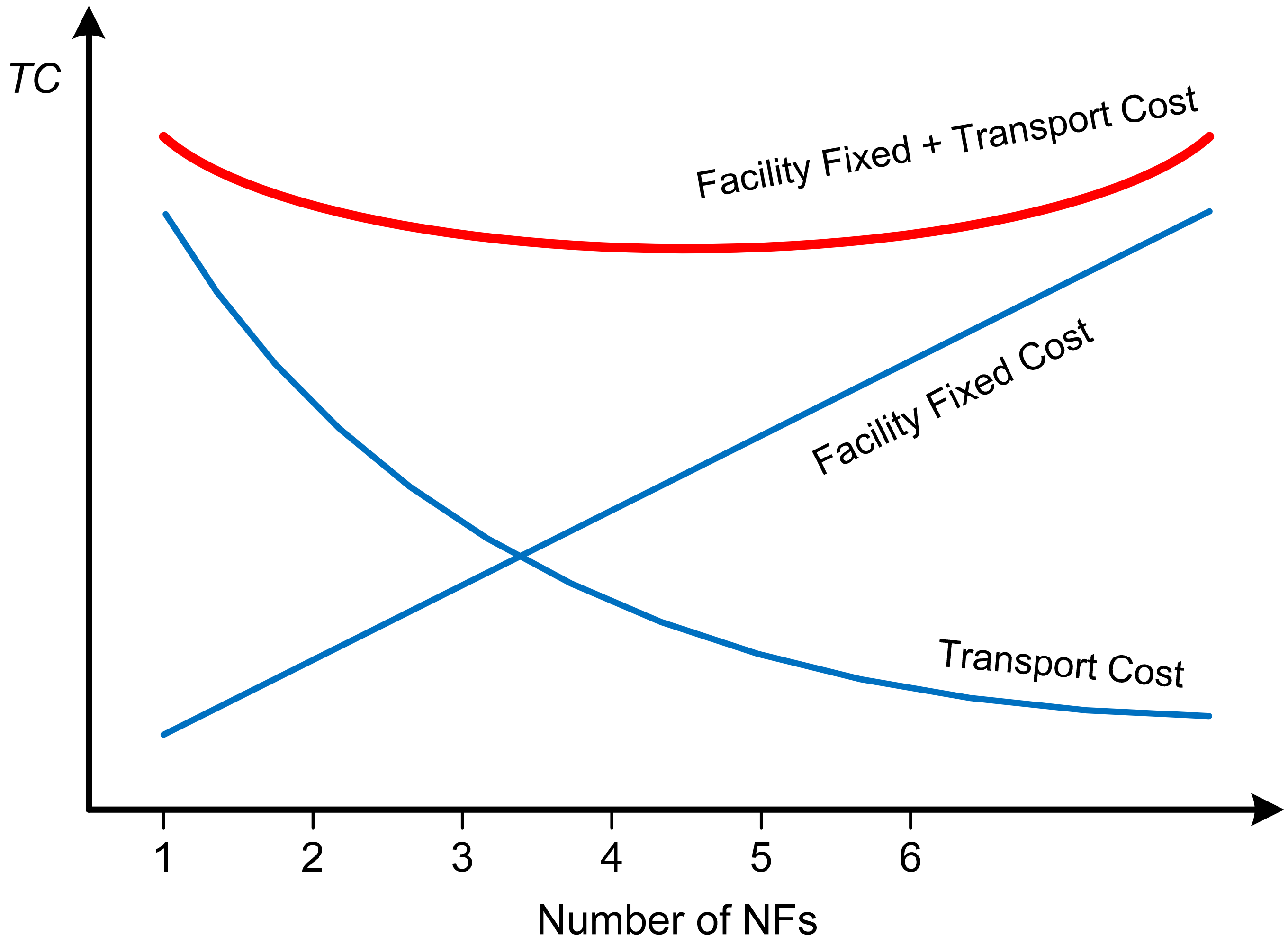

**procedure** $ala(\mathbf{X})$
$TC \leftarrow \infty$, $done \leftarrow$ false
**repeat**
  $\mathbf{W}' \leftarrow allocate(\mathbf{X})$
  $\mathbf{X}' \leftarrow locate(\mathbf{W}', \mathbf{X})$
  $TC' \leftarrow TC(\mathbf{X}', \mathbf{W}')$
  **if** $TC' < TC$
    $TC \leftarrow TC'$, $\mathbf{X} \leftarrow \mathbf{X}'$, $\mathbf{W} \leftarrow \mathbf{W}'$
  **else**
    $done \leftarrow$ true
  **endif**
**until** $done =$ true
**return** $\mathbf{X}$, $\mathbf{W}$

```
%% ALA Matlab Code
X = randX(P,n);
TC = Inf; done = false;
while ~done
   Wi = alloc_h(X);
   Xi = loc_h(Wi,X);
   TCi = TCh(Wi,Xi);
   if TCi < TC
      TC = TCi; X = Xi; W = Wi;
   else
      done = true;
   end
end
X, W
```

# Best Retail Warehouse Locations

| Number of Locations | Average Transit Time (days) | Warehouse Location | | |
|---|---|---|---|---|
| 1 | 2.20 | Bloomington, IN | | |
| 2 | 1.48 | Ashland, KY | Palmdale, CA | |
| 3 | 1.29 | Allentown, PA | Palmdale, CA | McKenzie, TN |
| 4 | 1.20 | Edison, NJ<br>Meridian, MS | Palmdale; CA | Chicago, IL |
| 5 | 1.13 | Madison, NJ<br>Dallas, TX | Palmdale, CA<br>Macon, GA | Chicago, IL |
| 6 | 1.08 | Madison, NJ<br>Dallas, TX | Pasadena, CA<br>Macon, GA | Chicago, IL<br>Tacoma, WA |
| 7 | 1.07 | Madison, NJ<br>Dallas, TX<br>Lakeland, FL | Pasadena, CA<br>Gainesville, GA | Chicago, IL<br>Tacoma, WA |
| 8 | 1.05 | Madison, NJ<br>Dallas, TX<br>Lakeland, FL | Pasadena, CA<br>Gainesville, GA<br>Denver, CO | Chicago, IL<br>Tacoma, WA |
| 9 | 1.04 | Madison, NJ<br>Dallas, TX<br>Lakeland. FL | Alhambra, CA<br>Gainesville, GA<br>Denver, CO | Chicago, IL<br>Tacoma, WA<br>Oakland, CA |
| 10 | 1.04 | Newark, NJ<br>Palistine, TX<br>Lakeland, FL<br>Mansfield, OH | Alhambra, CA<br>Gainesville, GA<br>Denver, CO | Rockford, IL<br>Tacoma, WA<br>Oakland. CA |

# Optimal Number of NFs



# Uncapacitated Facility Location Problem

$$\text{Min } TC = \sum_{i=1}^{n} k_i y_i + \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij}$$

subject to

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, \ldots, m$$

$$y_i \geq x_{ij}, \quad i = 1, \ldots, n; \ j = 1, \ldots, m$$

$$0 \leq x_{ij} \leq 1, \quad i = 1, \ldots, n; \ j = 1, \ldots, m$$

$$y_i \in \{0, 1\}, \quad i = 1, \ldots, n,$$

where

$k_i$ = fixed cost of establishing a NF at site $i$

$c_{ij}$ = variable cost to serve all of EF $j$'s demand from site $i$

$y_i$ = 1, if NF established at site $i$; 0, otherwise

$x_{ij}$ = fraction of EF $j$'s demand served from NF at site $i$.

# UFLADD: Add Construction Procedure

procedure $uf\!ladd(\mathbf{k},\mathbf{C})$

$Y \leftarrow \{\}$

$TC \leftarrow \infty,\ done \leftarrow \text{false}$

repeat

  $TC' \leftarrow \infty$

  for $i' \in \{1,\ldots,n\} \setminus Y$

    $$TC'' \leftarrow \sum_{h \in Y \cup i'} k_h + \sum_{j=1}^{m} \min_{h \in Y \cup i'} c_{hj}$$

    if $TC'' < TC'$

      $TC' \leftarrow TC'',\ i \leftarrow i'$

    endif

  endfor

  if $TC' < TC$

    $TC \leftarrow TC',\ Y \leftarrow Y \cup i$

  else

    $done \leftarrow \text{true}$

  endif

until $done = \text{true}$

return $Y, TC$

```
%% UFLADD Matlab code, given k and C as inputs
y = [];
TC = Inf; done = false;
while ~done
   TC1 = Inf;                        % Stops if y = all NF,
   for i1 = setdiff(1:size(C,1),y)  % since i1 = []
      TC2 = sum(k([y i1])) + sum(min(C([y i1],:),[],1));
      if TC2 < TC1
         TC1 = TC2; i = i1;
      end
   end
   if TC1 < TC % not true if y = all NF, since TC1 = Inf
      TC = TC1; y = [y i];
   else
      done = true;
   end
end
y,TC
```

# UFLXCHG: Exchange Improvement Procedure

**procedure** $uflxchg(\mathbf{k}, \mathbf{C}, Y)$

$TC \leftarrow \sum_{i \in Y} k_i + \sum_{j=1}^{m} \min_{i \in Y} c_{ij}$

$TC' \leftarrow \infty$, $done \leftarrow$ false

**while** $|y| > 1$ and $done =$ false

    **for** $i' \in y$

        **for** $j' \in \{1, \ldots, n\} \setminus Y$

            $Y' \leftarrow Y \setminus i' \cup j'$

            $TC'' \leftarrow \sum_{i \in Y'} k_i + \sum_{j=1}^{m} \min_{i \in Y'} c_{ij}$

            **if** $TC'' < TC'$

                $TC' \leftarrow TC''$, $i \leftarrow i'$, $j \leftarrow j'$

            **endif**

        **endfor**

    **endfor**

    **if** $TC' < TC$

        $TC \leftarrow TC', Y \leftarrow Y \setminus i \cup j$

    **else**

        $done \leftarrow$ true

    **endif**

**endwhile**

**return** $Y, TC$

```
%% UFLXCHG Matlab code, given k, C, and y as inputs
TC = sum(k(y)) + sum(min(C(y,:),[],1));
TC1 = Inf; done = false;
while length(y) > 1 && ~done
   for i1 = y
      for j1 = setdiff(1:size(C,1),y)
         y1 = [setdiff(y,i1) j1];
         TC2 = sum(k(y1)) + sum(min(C(y1,:),[],1));
         if TC2 < TC1
            TC1 = TC2; i = i1; j = j1;
         end
      end
   end
   if TC1 < TC
      TC = TC1; y = [setdiff(y,i) j];
   else
      done = true;
   end
end
y, TC
```

# Modified UFLADD

**procedure** $\mathit{ufladd}(\mathbf{k}, \mathbf{C}, \boxed{Y, p})$

$\cancel{Y \leftarrow \{\}}$

$TC \leftarrow \infty$, $\mathit{done} \leftarrow \text{false}$

**repeat**

  $TC' \leftarrow \infty$

  **for** $i' \in \{1, \ldots, n\} \setminus Y$

    $$TC'' \leftarrow \sum_{h \in Y \cup i'} k_h + \sum_{j=1}^{m} \min_{h \in Y \cup i'} c_{hj}$$

    **if** $TC'' < TC'$

      $TC' \leftarrow TC''$, $i \leftarrow i'$

    **endif**

  **endfor**

  **if** $\boxed{(p = \{\} \text{ and } TC' < TC) \text{ or } (p \neq \{\} \text{ and } |Y| < p)}$

    $TC \leftarrow TC'$, $Y \leftarrow Y \cup i$

  **else**

    $\mathit{done} \leftarrow \text{true}$

  **endif**

**until** $\mathit{done} = \text{true}$

**return** $Y, TC$

# UFL: Hybrid Algorithm

$$
\begin{array}{l}
\textbf{procedure } \mathit{ufl}(\mathbf{k},\mathbf{C}) \\
Y', TC' \leftarrow \mathit{ufladd}(\mathbf{k},\mathbf{C}) \\
\mathit{done} \leftarrow \text{false} \\
\textbf{repeat} \\
\quad Y, TC \leftarrow \mathit{uflxchg}(\mathbf{k},\mathbf{C},Y') \\
\quad \textbf{if } Y \neq Y' \\
\qquad Y', TC' \leftarrow \mathit{ufladd}(\mathbf{k},\mathbf{C},Y) \\
\qquad Y'', TC'' \leftarrow \mathit{ufldrop}(\mathbf{k},\mathbf{C},Y) \\
\qquad \textbf{if } TC'' < TC' \\
\qquad\quad TC' \leftarrow TC'', Y' \leftarrow Y'' \\
\qquad \textbf{endif} \\
\qquad \textbf{if } TC' \geq TC \\
\qquad\quad \mathit{done} \leftarrow \text{true} \\
\qquad \textbf{endif} \\
\quad \textbf{else} \\
\qquad \mathit{done} \leftarrow \text{true} \\
\quad \textbf{endif} \\
\textbf{until } \mathit{done} = \text{true} \\
\textbf{return } Y, TC
\end{array}
$$

```
%% UFL Matlab code, given k and C
[y1,TC1] = ufladd(k,C);
done = false;
while ~done
   [y,TC] = uflxchg(k,C,y1);
   if ~isequal(y,y1)
      [y1,TC1] = ufladd(k,C,y);
      [y2,TC2] = ufldrop(k,C,y);
      if TC2 < TC1
         TC1 = TC2; y1 = y2;
      end
      if TC1 >= TC
         done = true;
      end
   else
      done = true;
   end
end
y, TC
```

# PMEDIAN: Discrete Location Algorithm

**procedure** $pmedian(p, \mathbf{C})$

$Y \leftarrow ufladd(0, \mathbf{C}, \{\}, p)$

$Y, TC \leftarrow uflxchg(0, \mathbf{C}, Y)$

$Y' \leftarrow ufldrop(0, \mathbf{C}, \{\}, p)$

$Y', TC' \leftarrow uflxchg(0, \mathbf{C}, Y)$

**if** $TC' < TC$

$\quad TC \leftarrow TC', Y \leftarrow Y'$

**endif**

**return** $Y, TC$

# Aggregate Demand Point Data Sources

- Aggregate demand point: centroid of population
- Good rule of thumb: use 100x number of NFs (≈ 1000 pts provides good coverage for locating ≈ 10 NFs)

1. City data: ONLY USE FOR LABELING!, not as demand points
2. 3-digit ZIP codes: ≈ 1000 pts covering U.S., = 20 pts NC
3. County data: ≈ 3000 pts covering U.S., = 100 pts NC
   - Grouped by state or MSA (Metropolitan Statistical Area)
   - MSA = defined by set of counties (382 MSAs in U.S.)
   - FIPS code = 5-digit state-county FIPS code<br>= 2-digit state code + 3-digit county code<br>= 37183 = 37 NC FIPS + 183 Wake FIPS
   - MSA List: www.census.gov/population/metro/files/lists/2009/List1.txt
4. 5-digit ZIP codes: > 35K pts U.S., ≈ 1000 pts NC
5. Census Block Group: > 220K pts U.S., ≈ 1000 pts Raleigh-Durham MSA
   - Grouped by state, county, or MSA

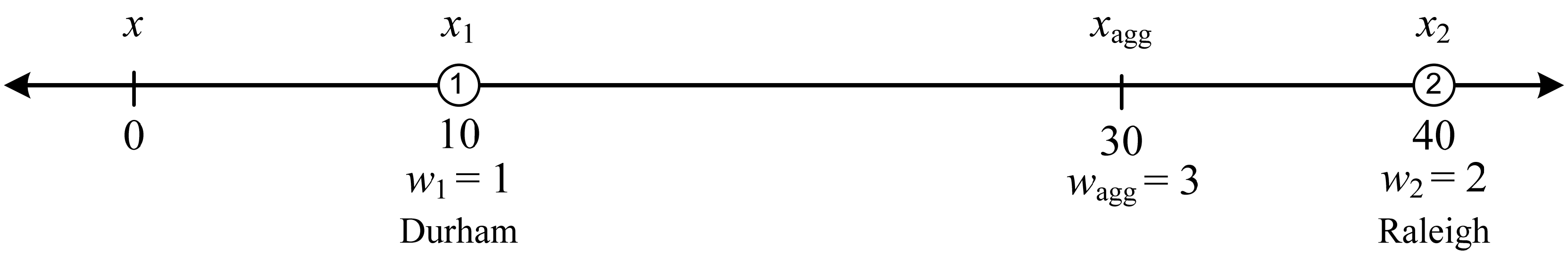
# City vs MSA Population Data

| Rank | City; State | 2010 population | 2012 population |
|---|---|---|---|
| 1 | New York City; New York | 8,175,133 | **8,336,697** |
| 2 | Los Angeles; California | 3,792,621 | **3,857,799** |
| 3 | Chicago; Illinois | 2,695,598 | **2,714,856** |
| 4 | Houston; Texas | 2,099,451 | **2,160,821** |
| 5 | Philadelphia; Pennsylvania | 1,526,006 | **1,547,607** |
| 6 | Phoenix; Arizona | 1,445,632 | **1,488,750** |
| 7 | San Antonio; Texas | 1,327,407 | **1,382,951** |
| 8 | San Diego; California | 1,307,402 | **1,338,348** |
| 9 | Dallas; Texas | 1,197,816 | **1,241,162** |
| 10 | San Jose; California | 945,942 | **982,765** |
| 11 | Austin; Texas | 790,390 | **842,592** |
| 12 | Jacksonville; Florida | 821,784 | **836,507** |
| 13 | Indianapolis; Indiana | 820,445 | **834,852** |
| 14 | San Francisco; California | 805,235 | **825,863** |
| 15 | Columbus; Ohio | 787,033 | **809,798** |
| 16 | Fort Worth; Texas | 741,206 | **777,992** |
| 17 | Charlotte; North Carolina | 731,424 | **775,202** |
| 18 | Detroit; Michigan | 713,777 | **701,475** |
| 19 | El Paso; Texas | 649,121 | **672,538** |
| 20 | Memphis; Tennessee | 646,889 | **655,155** |

| Metropolitan Area | 2010 Population | City |
|---|---|---|
| New York-Northern NJ-Long Island, NY-NJ-PA | 18,897,109 | New York |
| Los Angeles-Long Beach-Santa Ana, CA | 12,828,837 | Los Angeles |
| Chicago-Joliet-Naperville, IL-IN-WI | 9,461,105 | Chicago |
| Dallas-Fort Worth-Arlington, TX | 6,371,773 | Dallas |
| Philadelphia-Camden-Wilmington, PA-NJ-DE-MD | 5,965,343 | Philadelphia |
| Houston-Sugar Land-Baytown, TX | 5,946,800 | Houston |
| Washington-Arlington-Alexandria, DC-VA-MD-WV | 5,582,170 | Washington |
| Miami-Fort Lauderdale-Pompano Beach, FL | 5,564,635 | Miami |
| Atlanta-Sandy Springs-Marietta, GA | 5,268,860 | Atlanta |
| Boston-Cambridge-Quincy, MA-NH | 4,552,402 | Boston |
| San Francisco-Oakland-Fremont, CA | 4,335,391 | San Francisco |
| Detroit-Warren-Livonia, MI | 4,296,250 | Detroit |
| Riverside-San Bernardino-Ontario, CA | 4,224,851 | Riverside |
| Phoenix-Mesa-Glendale, AZ | 4,192,887 | Phoenix |
| Seattle-Tacoma-Bellevue, WA | 3,439,809 | Seattle |
| Minneapolis-St. Paul-Bloomington, MN-WI | 3,279,833 | Minneapolis |
| San Diego-Carlsbad-San Marcos, CA | 3,095,313 | San Diego |
| St. Louis, MO-IL | 2,812,896 | St. Louis |
| Tampa-St. Petersburg-Clearwater, FL | 2,783,243 | Tampa |
| Baltimore-Towson, MD | 2,710,489 | Baltimore |

# Demand Point Aggregation

- *Existing facility* (EF): actual physical location of demand source
- *Aggregate demand point*: single location representing multiple demand sources




- For minisum location, would like for any location $x$:

$$\left(w_1 + w_2\right) d(x, x_{agg}) = w_1 d(x, x_1) + w_2 d(x, x_2), \quad \text{w.l.o.g., let } x = 0,\ x_1, x_2 > 0$$

$$\left(w_1 + w_2\right) x_{agg} = w_1 x_1 + w_2 x_2$$

$$x_{agg} = \frac{w_1 x_1 + w_2 x_2}{w_1 + w_2} \Rightarrow \quad \text{centroid}$$

# Bottom-Up vs Top-Down Analysis

- Bottom-Up: HW 3 Q 3

$$\mathbf{P}_{3\times 2} = \text{lon-lat of EFs}$$

$$\mathbf{f} = [45, 25, 35] \quad \text{(TL/yr)}$$

$$r = 2 \quad \text{(\$/TL-mi)}$$

$$g = \frac{1}{3}\left[\frac{d_{RD}(\mathbf{P}_1,\mathbf{P}_2)}{d_{GC}(\mathbf{P}_1,\mathbf{P}_2)} + \frac{d_{RD}(\mathbf{P}_2,\mathbf{P}_3)}{d_{GC}(\mathbf{P}_2,\mathbf{P}_3)} + \frac{d_{RD}(\mathbf{P}_3,\mathbf{P}_1)}{d_{GC}(\mathbf{P}_3,\mathbf{P}_1)}\right]$$

$$TC(\mathbf{x}) = \sum_{i=1}^{3} f_i\, r\, g\, d_{GC}(\mathbf{x},\mathbf{P}_i)$$

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} TC(\mathbf{x})$$

$$TC^* = TC(\mathbf{x}^*)$$

$$\mathbf{x}^{\text{cary}} = \text{lon-lat of Cary}$$

$$TC^{\text{cary}} = TC(\mathbf{x}^{\text{cary}})$$

$$\Delta TC = TC^{\text{cary}} - TC^*$$

- Top-Down: estimate *r* (circuity factor cancels, so not needed)

$$TC^{\text{cary}} = \text{current known } TC$$

$$\mathbf{f} = [450, 250, 350] \quad \text{(ton/yr)}$$

$$r_{\text{nom}} = \frac{TC^{\text{cary}}}{\sum_{i=1}^{3} f_i\, d_{GC}(\mathbf{x}^{\text{cary}},\mathbf{P}_i)} \quad \text{(\$/ton-mi)}$$

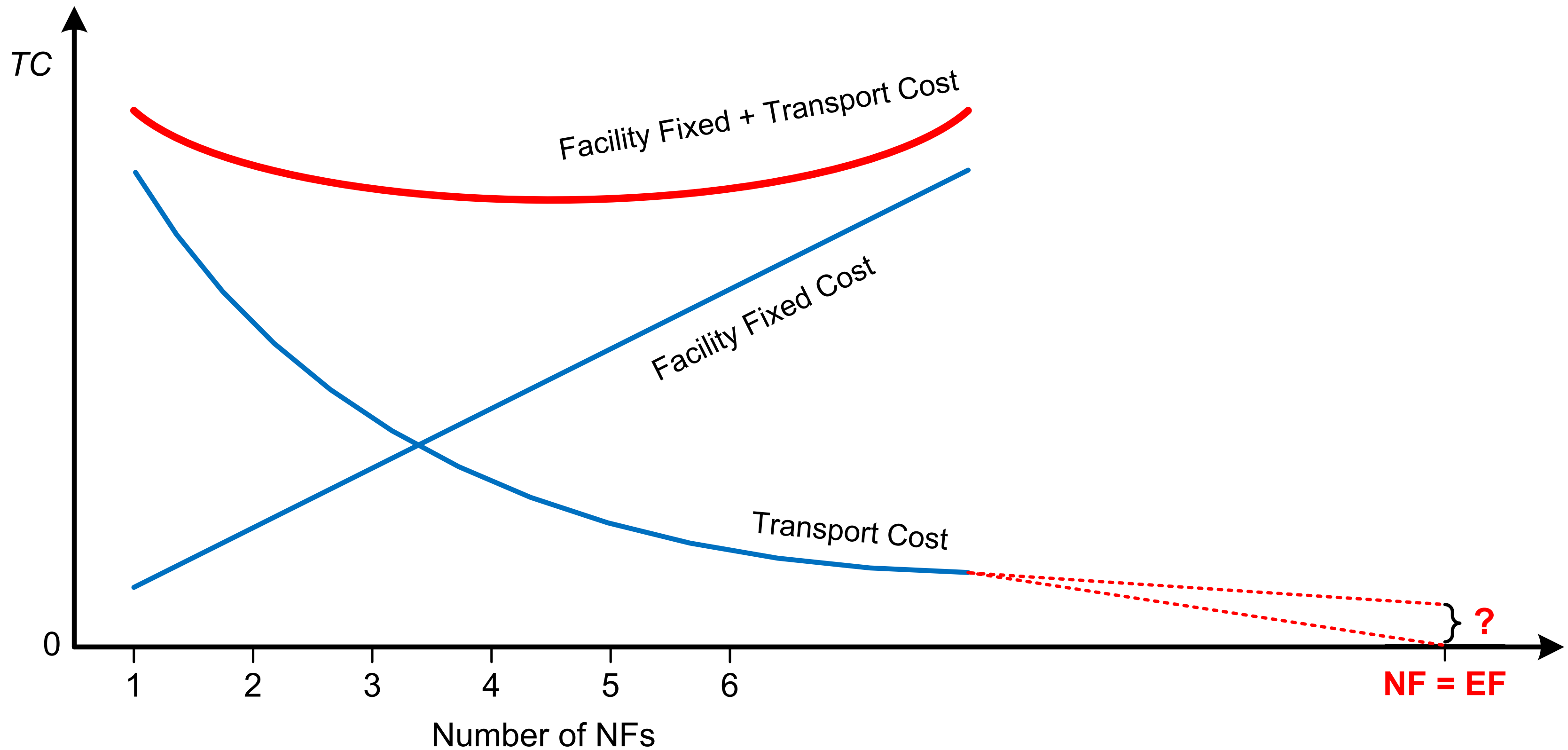
$$TC(\mathbf{x}) = \sum_{i=1}^{3} f_i\, r_{\text{nom}} d_{GC}(\mathbf{x},\mathbf{P}_i)$$

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} TC(\mathbf{x})$$

$$TC^* = TC(\mathbf{x}^*)$$

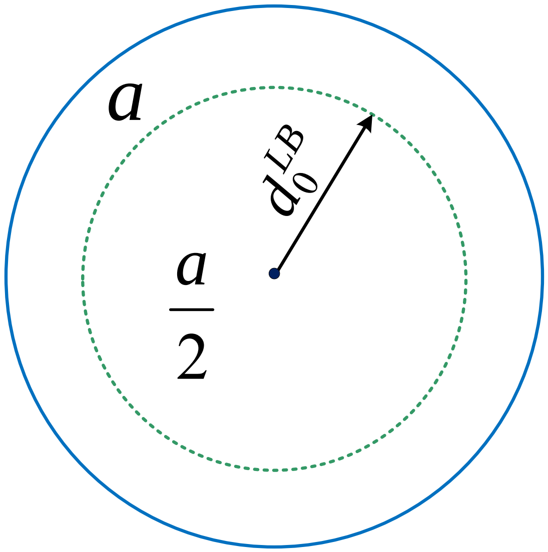
$$\Delta TC = TC^{\text{cary}} - TC^*$$

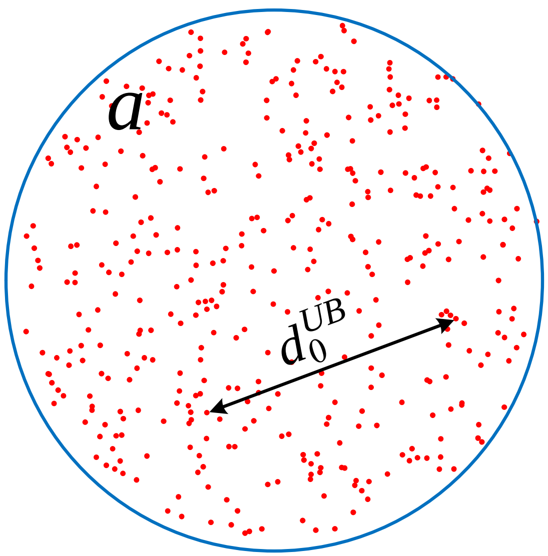
# Transport Cost if NF at every EF




$$TC = \overbrace{\sum_{i=1}^{n} k_i y_i}^{\text{fixed cost}} + \overbrace{\sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij}}^{\text{transport cost}}$$

# Area Adjustment for Aggregate Data Distances




- *LB:* average distance from center to all points in area
- *UB:* average distance between all random pairs of points

$$\frac{a}{2} = \pi \left( d_0^{LB} \right)^2$$

$$d_0^{LB} = \sqrt{\frac{a}{2\pi}} \approx 0.40\sqrt{a}$$

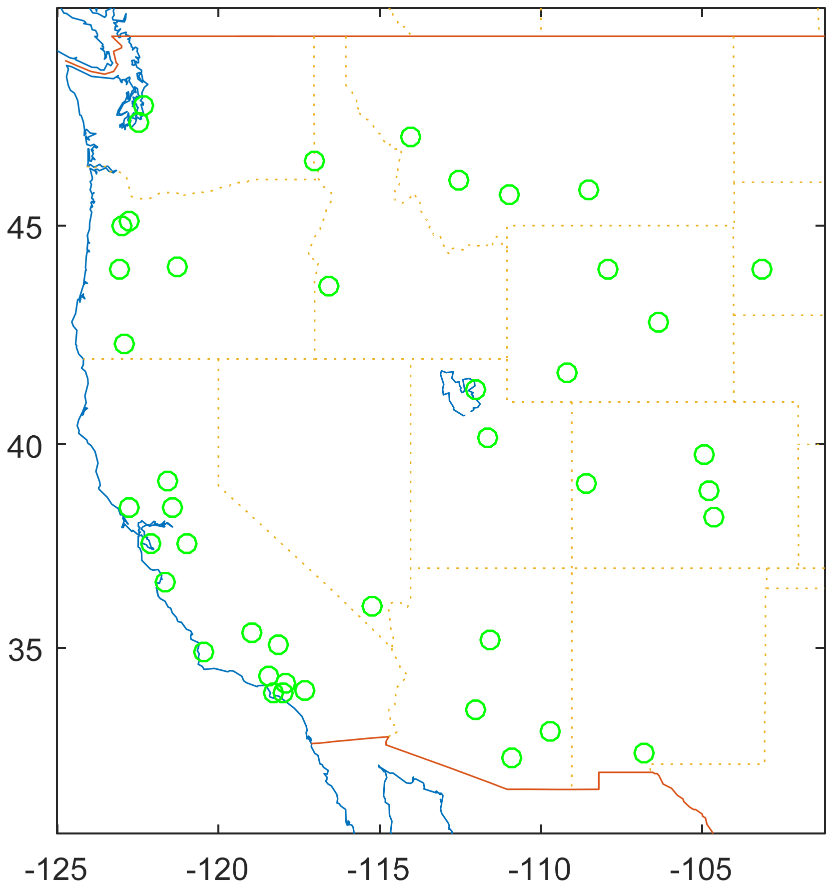
$$d_0^{UB} = \frac{32}{15} \frac{\sqrt{a}}{\pi \Gamma\left( 5/2 \right)} \approx 0.51\sqrt{a}$$

Mathai, A.M., *An Intro to Geo Prob*, p. 207 (2.3.68)

$$d_0 = \sqrt{d_0^{LB} d_0^{UB}} \approx 0.45\sqrt{a}$$

$$d_a(\mathbf{X}_1, \mathbf{X}_2) = \max\left\{ d_{GC}(\mathbf{X}_1, \mathbf{X}_2), 0.45 \max\left\{ \sqrt{a_1}, \sqrt{a_2} \right\} \right\}$$

# Popco Bottling Company Example



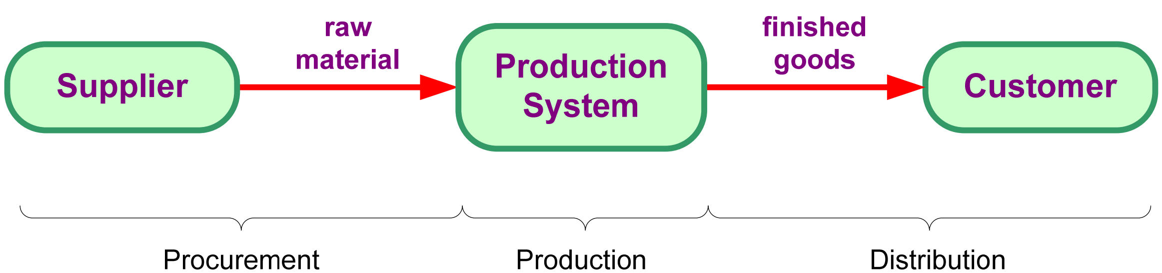

- **Problem:** Popco currently has 43 bottling plants across the western U.S. and wants to know if they should consider reducing or adding plants to improve their profitability.
- **Solution:** Formulate as an UFL to determine the number of plants that minimize Popco's production, procurement, and distribution costs.

# Popco Bottling Company Example

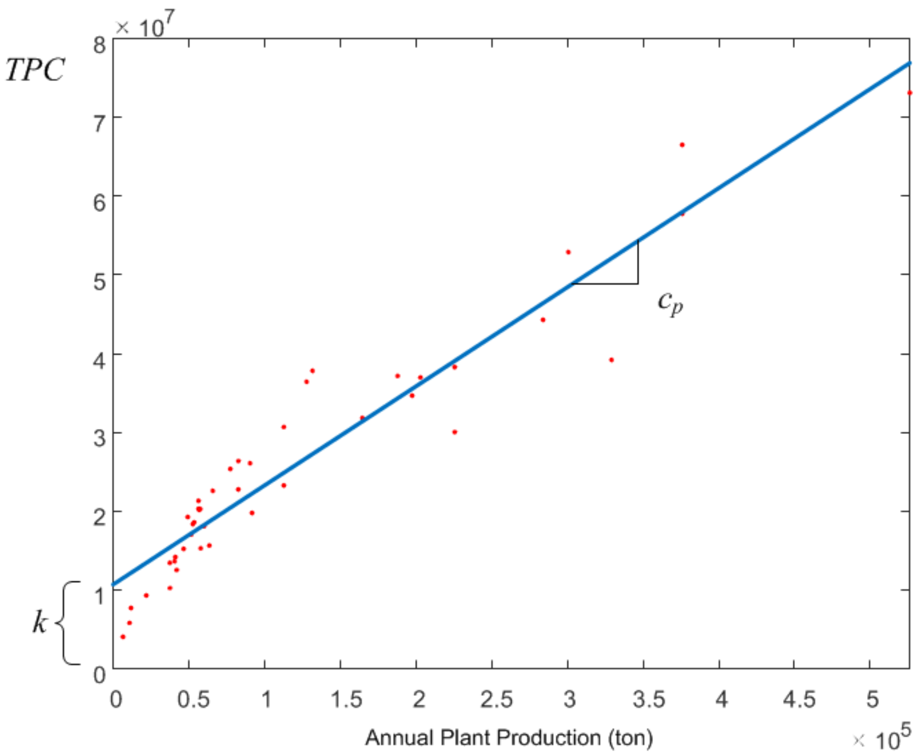
- Following representative information is available for each of *N* current plants (DC) *i*:

$$
\begin{aligned}
xy_i &= \text{location} \\
f_i^{DC} &= \text{aggregate production (tons)} \\
TPC_i &= \text{total production and procurement cost} \\
TDC_i &= \text{total distribution cost}
\end{aligned}
$$

- Assuming plants are (monetarily) weight gaining, so UFL can ignore inbound procurement costs related to location



# Popco Bottling Company Example



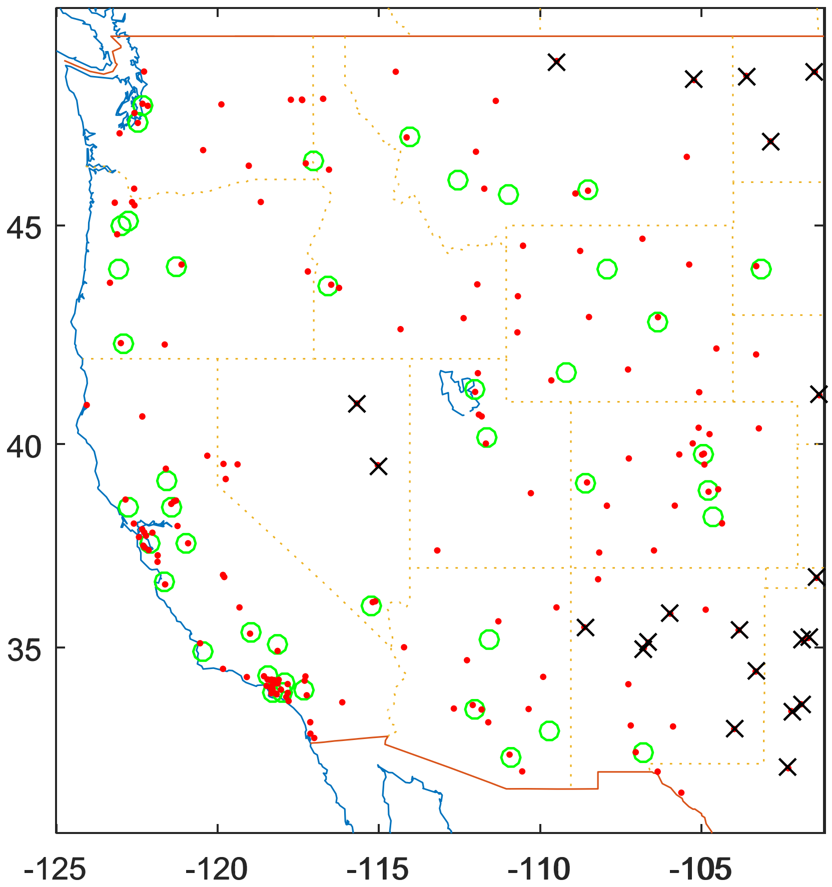

- Difficult to estimate fixed cost of each new facility because this cost must not include any cost related to quantity of product produced at facility.

1. Use plant (DC) production costs to find UFL fixed costs via linear regression
   – variable production costs $c_p$ do not change and can be cut

$$TPC = \sum_{i \in N} TPC_i = \sum_{i \in N} \left( k + c_p f_i^{DC} \right)$$

$$\left(\text{only keep } k \text{ for UFL}\right)$$

# Popco Bottling Company Example




2. Allocate all 3-digit ZIP codes to closest plant (up to 200 mi max) to serve as aggregate customer demand points.

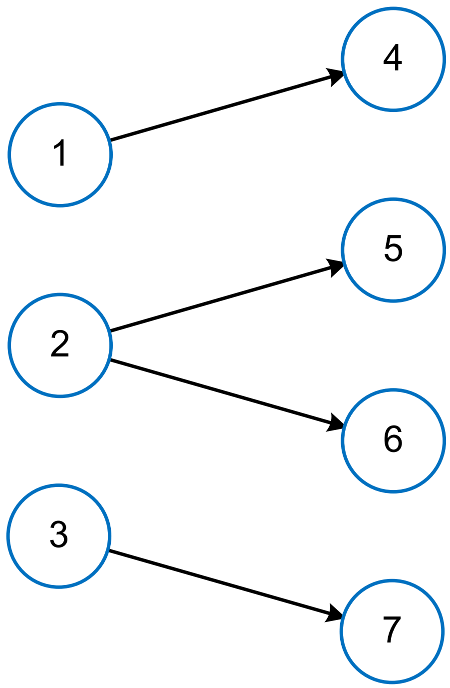
$$M_i = \left\{ j : \arg\min_h d_{hj}^a = i \text{ and } d_{ij}^a \le d_{\max} \right\}$$

$$d_{\max} = 200 \text{ mi}$$

$$M = \bigcup_{i \in N} M_i$$

# Popco Bottling Company Example

3. Allocate each plant's demand (tons of product) to each of its customers based on its population.




$$f_{j \in M_i} = f_i^{DC} \frac{q_j}{\sum_{h \in M_i} q_h}$$

$$q_j = \text{population of EF}j$$

$$f_5 = f_2^{DC} \frac{q_5}{q_5 + q_6}$$

# Popco Bottling Company Example

4. Estimate a nominal transport rate (\$/ton-mi) using the ratio of total distribution cost (\$) to the sum of the product of the demand (ton) at each customer and its distance to its plant (mi).

$$r_{\text{nom}} = \frac{\sum_{i \in N} TDC_i}{\sum_{i \in N} \sum_{j \in M_j} f_j \, d_{ij}^a}$$

# Popco Bottling Company Example

5. Calculate UFL variable transportation cost $c_{ij}$ ($) for each possible NF site *i* (all customer and plant locations) and EF site *j* (all customer locations) as the product of customer *j* demand (ton), distance from site *i* to *j* (mi), and the nominal transport rate ($/ton-mi).

$$\mathbf{C} = \left[ c_{ij} \right]_{\substack{i \in M \cup N \\ j \in M}} = \left[ r_{\text{nom}} \, f_j \, d_{ij}^{a} \right]_{\substack{i \in M \cup N \\ j \in M}}$$

6. Solve as UFL, where *TC* returned includes all new distribution costs and the fixed portion of production costs.

$$TC = \overbrace{\sum_{i=1}^{n} k \, y_i}^{\text{fixed cost}} + \overbrace{\sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij}}^{\text{transport cost}}$$

$$n = \left| M \cup N \right|, \quad \text{number of potential NF sites}$$

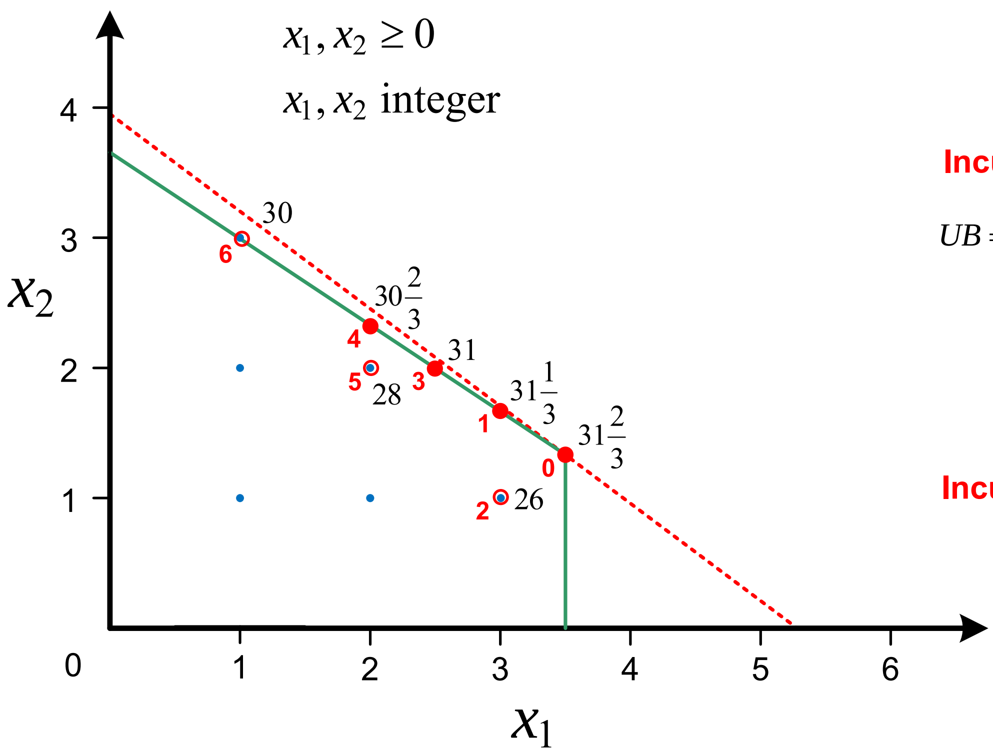
$$m = \left| M \right|, \quad \text{number of EF sites}$$

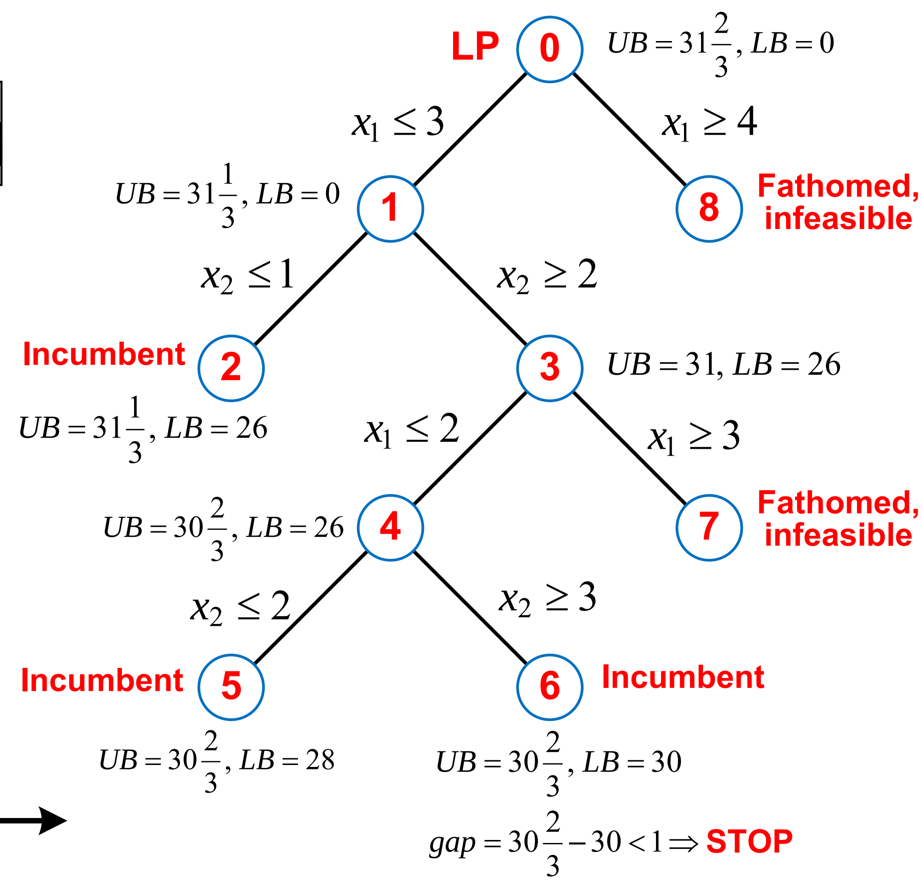
# MILP

$$\begin{array}{rll} \text{LP:} & \max & \mathbf{c}'\mathbf{x} \\ & \text{s.t.} & \mathbf{A}\mathbf{x} \le \mathbf{b} \\ & & \mathbf{x} \ge 0 \\ \text{MILP:} & \multicolumn{2}{l}{\text{some } x_i \text{ integer}} \\ \text{ILP:} & \multicolumn{2}{l}{\mathbf{x} \text{ integer}} \\ \text{BLP:} & \multicolumn{2}{l}{\mathbf{x} \in \{0,1\}} \end{array}$$

# Branch and Bound

$$
\begin{aligned}
\max\quad & 6x_1 + 8x_2 \\
\text{s.t.}\quad & 2x_1 + 3x_2 \le 11 \\
& 2x_1 \qquad\;\; \le 7 \\
& x_1, x_2 \ge 0 \\
& x_1, x_2 \text{ integer}
\end{aligned}
\qquad
\mathbf{c} = \begin{bmatrix} 6 & 8 \end{bmatrix}
\qquad
\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 2 & 0 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 11 \\ 7 \end{bmatrix}
$$



# MILP Solvers

LP: $\max\ \mathbf{c}'\mathbf{x}$

$\text{s.t.}\quad \mathbf{A}\mathbf{x} \le \mathbf{b}$

$\mathbf{x} \ge 0$

MILP: some $x_i$ integer

ILP: $\mathbf{x}$ integer

BLP: $\mathbf{x} \in \{0,1\}$

intlinprog: $\min\ \mathbf{c} \quad (\max\ -\mathbf{c})$

$$\begin{aligned} \text{s.t.}\quad & \mathbf{A}_{\text{lt}} \le \mathbf{b}_{\text{lt}} \\ & \mathbf{A}_{\text{eq}} = \mathbf{b}_{\text{eq}} \\ & LB \le \mathbf{x} \le UB \\ & \text{integer variable indices} \end{aligned}$$

cplex: $\mathbf{c} \quad (\text{sense } \mathit{min} \text{ or } \mathit{max})$

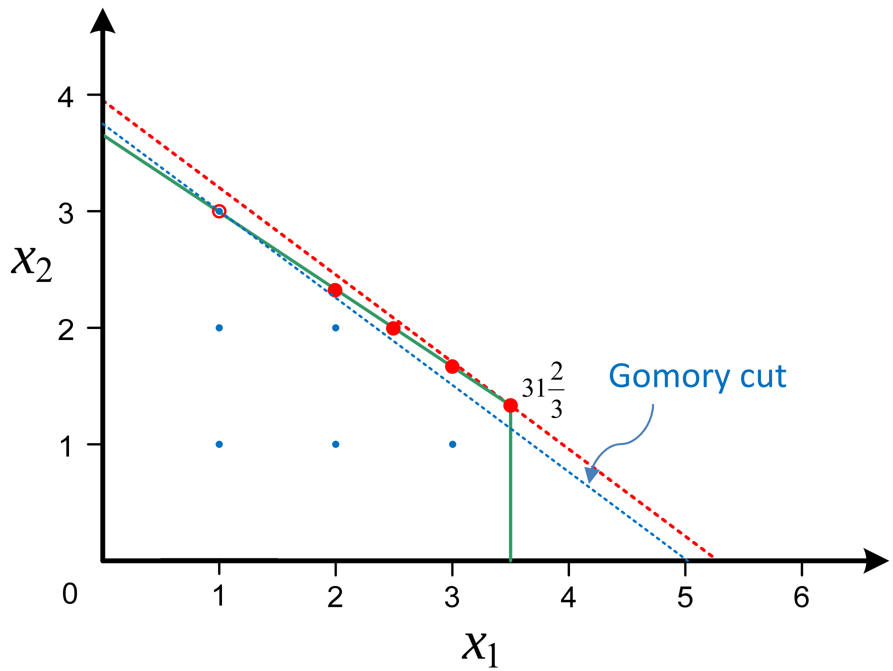
$$\begin{aligned} \text{s.t.}\quad & lhs \le \mathbf{A} \le rhs \\ & LB \le \mathbf{x} \le UB \\ & \text{variable:} \begin{cases} C & \text{continuous} \\ B & \text{binary} \\ I & \text{general integer} \end{cases} \end{aligned}$$

| $lhs$ | $rhs$ | | |
|---|---|---|---|
| $-\infty$ | $\mathbf{b}$ | $\Rightarrow$ | $\le$ |
| $\mathbf{b}$ | $\infty$ | $\Rightarrow$ | $\ge$ |
| $\mathbf{b}$ | $\mathbf{b}$ | $\Rightarrow$ | $=$ |

gurobi: $\mathbf{c} \quad (\text{modelsense } \mathit{min} \text{ or } \mathit{max})$

$$\begin{aligned} \text{s.t.}\quad & \mathbf{A} \begin{Bmatrix} < \\ = \\ > \end{Bmatrix} \mathbf{b} \\ & LB \le \mathbf{x} \le UB \\ & \text{variable:} \begin{cases} C & \text{continuous} \\ B & \text{binary} \\ I & \text{general integer} \end{cases} \end{aligned}$$

# MILP Solvers




- **Presolve**: eliminate variables

$$2x_1 + 2x_2 \le 1,\ x_1, x_2 \ge 0 \text{ and integer}$$

$$\Rightarrow x_1 = x_2 = 0$$

- **Cutting planes**: keeps all integer solutions and cuts off LP solutions (Gomory cut)
- **Heuristics**: find good initial incumbent solution
- **Parallel**: use separate cores to solve nodes in B&B tree
- **Speedup** from 1990-2014:
  - 320,000× computer speed
  - 580,000× algorithm improvements

# MILP Formulation of UFL

$$
\begin{aligned}
\min \quad & \sum_{i\in N} k_i y_i + \sum_{i\in N}\sum_{j\in M} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{i\in N} x_{ij} = 1, && j\in M \\
& \cancel{m y_i \ge \sum_{j\in M} x_{ij},} && \cancel{i\in N} \\
& 0 \le x_{ij} \le 1, && i\in N,\, j\in M \\
& y_i \in \{0,1\}, && i\in N
\end{aligned}
$$

```
%% UFL MILP Matlab code, given k and C
mp.addobj('min',k,C)
for j = M
   mp.addcstr(0,{':',j},'=',1)
end
for i = N
   mp.addcstr({m,{i}},'>=',{i,':'})
end
mp.addub(1,1)
mp.addctype('B','C')
```

$$\boxed{y_i \ge x_{ij}, \qquad i\in N,\, j\in M}$$

where

$$k_i = \text{fixed cost of NF at site } i\in N = \{1,\ldots,n\}$$

$$c_{ij} = \text{variable cost from } i \text{ to serve EF } j\in M = \{1,\ldots,m\}$$

$$y_i = \begin{cases} 1, & \text{if NF established at site } i \\ 0, & \text{otherwise} \end{cases}$$

$$x_{ij} = \text{fraction of EF } j \text{ demand served from NF at site } i.$$

# Set Covering

$$M = \{1,\ldots,m\}, \quad \text{objects to be covered}$$

$$M_i \subseteq M,\, i \in N = \{1,\ldots,n\}, \quad \text{subsets of } M$$

$$c_i = \text{cost of using } M_i \text{ in cover}$$

$$I^* = \arg\min_I \left\{ \sum_{i\in I} c_i : \bigcup_{i\in I} M_i = M \right\}, \quad \text{min cost covering of } M$$

$$\begin{aligned} \min \quad & \sum_{i\in N} c_i x_i \\ \text{s.t.} \quad & \sum_{i\in N} a_{ji} x_i \ge 1, \qquad j \in M \\ & x_i \in \{0,1\}, \qquad i \in N \end{aligned}$$

where

$$x_i = \begin{cases} 1, & \text{if } M_i \text{ is in cover} \\ 0, & \text{otherwise} \end{cases}$$

$$a_{ji} = \begin{cases} 1, & \text{if } j \in M_i \\ 0, & \text{otherwise.} \end{cases}$$

```
%% Set Covering BLP Matlab code,
%  given c and A
mp = Milp('Set Cover')
mp.addobj('min',c)
mp.addcstr(A,'>=',1)
mp.addctype('B')
```

# Bin Packing

$$M = \{1,\ldots,m\}, \quad \text{objects to be packed}$$
$$v_j = \text{volume of object } j$$
$$V = \text{volume of each bin } B_i \left(\max v_j \le V\right)$$
$$B^* = \arg\min_B \left\{ |B| : \sum_{j \in B_i} v_j \le V, \bigcup_{B_i \in B} B_i = M \right\}, \quad \text{min packing of } M$$

$$\begin{aligned}
\min \quad & \sum_{i \in M} y_i \\
\text{s.t.} \quad & V y_i \ge \sum_{j \in M} v_j x_{ij}, && i \in M \\
& \sum_{i \in M} x_{ij} = 1, && j \in M \\
& y_i \in \{0,1\}, && i \in M \\
& x_{ij} \in \{0,1\}, && i \in M, j \in M
\end{aligned}$$

where

$$y_i = \begin{cases} 1, & \text{if bin } B_i \text{ is used in packing} \\ 0, & \text{otherwise} \end{cases}$$

$$x_{ij} = \begin{cases} 1, & \text{if object } j \text{ packed into bin } B_i \\ 0, & \text{otherwise.} \end{cases}$$

```matlab
%% Bin Packing BLP Matlab code,
%  given v and V
mp = Milp('Bin Packing')
mp.addobj('min',ones(1,m),zeros(m))
for i = M
   mp.addcstr({V,{i}},'>=',{v,{i,':'}})
end
for j = M
   mp.addcstr(0,{':',j},'=',1)
end
mp.addctype('B','B')
```

# Topics

1. Introduction
2. Facility location
3. **Freight transport**
   – Midterm exam
4. Network models
5. Routing
6. Warehousing
   – Final project
   – Final exam

# Logistics Engineering Design Constants

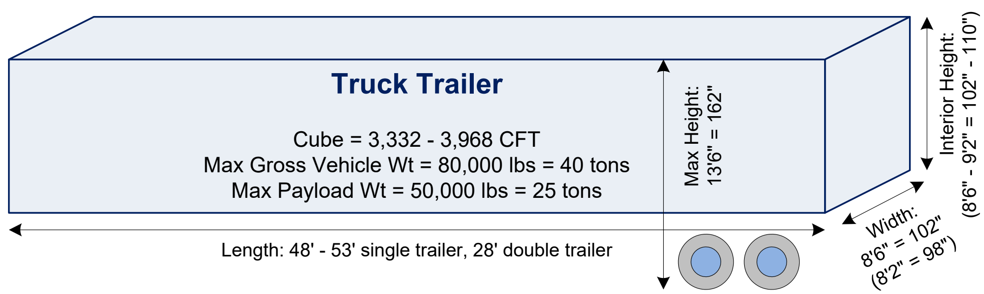
1. Circuity Factor: 1.2 ( $g$ )
   - 1.2 × GC distance ≈ actual road distance
2. Local vs. Intercity Transport:
   - Local: < 50 mi ⇒ use actual road distances
   - Intercity: > 50 mi ⇒ can estimate road distances
     - 50-250 mi ⇒ return possible (11 HOS)
     - > 250 mi ⇒ always one-way transport
     - > 500-750 mi ⇒ intermodal rail possible
3. Inventory Carrying Cost ( $h$ ) = funds + storage + obsolescence
   - 20% average (no product information)
   - 10% low-value product (construction)
   - 25-30% general durable manufactured goods
   - 50% computer equipment
   - >> 100% perishable goods (produce)

# Logistics Engineering Design Constants

4. $\dfrac{\text{Value}}{\text{Transport Cost}} \gg 1: \quad \$1\ \text{ft}^3 \approx \dfrac{\$2{,}620\ \text{Shanghai-LA/LB shipping cost}}{2{,}400\ \text{ft}^3\ 40'\ \text{ISO container capacity}}$

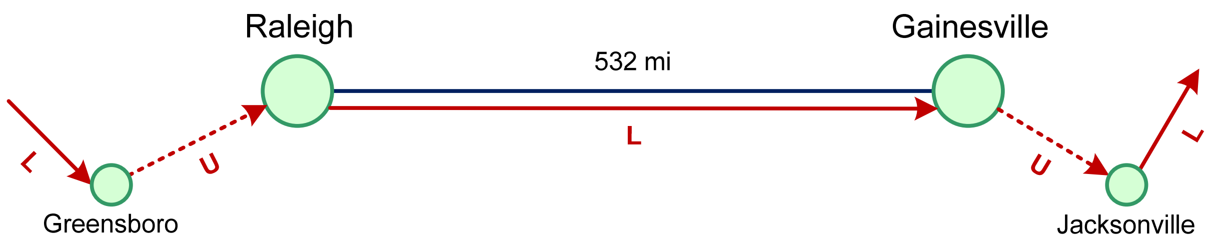


5. TL Weight Capacity: 25 tons ( $K_{wt}$ )
   - (40 ton max per regulation) – (15 ton tare for tractor-trailer) = 25 ton max payload
   - Weight capacity = 100% of physical capacity
6. TL Cube Capacity: 2,750 $\text{ft}^3$ ( $K_{cu}$ )
   - Trailer physical capacity = 3,332 $\text{ft}^3$
   - Effective capacity = 3,332 × 0.80 ≈ 2,750 $\text{ft}^3$
   - Cube capacity = 80% of physical capacity



# Logistics Engineering Design Constants

7. TL Revenue per Loaded Truck-Mile: $2/mi in 2004 ( $r$ )
   – TL revenue for the carrier is your TL cost as a shipper




$$15\%, \quad \text{average deadhead travel}$$

$$\$1.60, \quad \text{cost per mile in 2004}$$

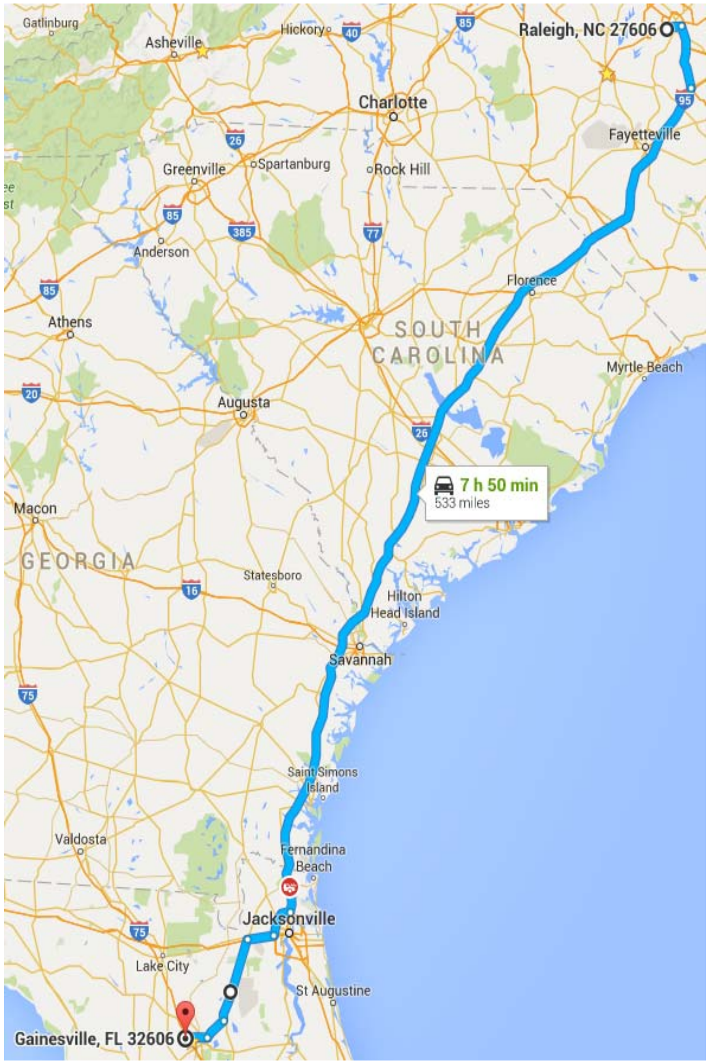
$$\frac{\$1.60}{1-0.15} = \$1.88, \quad \text{cost per loaded-mile}$$

$$6.35\%, \quad \text{average operating margin for trucking}$$

$$\frac{\$1.88}{1-0.0635} \approx \$2.00, \quad \text{revenue per loaded-mile}$$

# Truck Shipment Example

- Product is to be shipped in cartons from Raleigh, NC (27606) to Gainesville, FL (32606). Each unit weighs 40 lb and occupies 9 ft$^3$, and units can be stacked on top of each other in a trailer.
- **One-Time Shipments** (operational decision):
  - Know when and how much to ship, need to determine if TL and/or LTL to be used
  - Need to know much it will cost
- **Periodic Shipments** (tactical decision):
  - Need to determine how often and how much to ship



# Truck Shipment Example: One-Time

1. Assuming that the product is to be shipped P2P TL, what is the maximum payload for each trailer used for the shipment?

$$q_{\max}^{wt} = K_{wt} = 25 \text{ ton}$$

$$K_{cu} = 2750 \text{ ft}^3$$

$$s = \frac{40 \text{ lb/unit}}{9 \text{ ft}^3\text{/unit}} = 4.4444 \text{ lb/ft}^3$$

$$K_{cu} = \frac{q_{\max}^{cu}}{\left(\dfrac{s}{2000}\right)} \Rightarrow q_{\max}^{cu} = \frac{sK_{cu}}{2000}$$

$$q_{\max} = \min\left\{q_{\max}^{wt}, q_{\max}^{cu}\right\} = \min\left\{K_{wt}, \frac{sK_{cu}}{2000}\right\}$$

$$= \min\left\{25, \frac{4.4444(2750)}{2000}\right\} = 6.1111 \text{ ton}$$

# Truck Shipment Example: One-Time

2. Next Monday, 350 units of the product are to be shipped. How many truckloads are required for this shipment?

$$q = 350\frac{40}{2000} = 7 \text{ ton}, \quad \left\lceil \frac{q}{q_{\max}} \right\rceil = \left\lceil \frac{7}{6.1111} \right\rceil = 2 \text{ truckloads}$$

3. Using the most recent rate estimate available, what is the TL transport charge for this shipment?
   - Ratio of most recent Producer Price Index (PPI) for TL (see course website for link) to PPI in 2004 used to inflate known \$2/mi 2004 value

$$d = 532 \text{ mi}$$

$$r = \frac{PPI_{TL}^{\text{Jan 2017}}}{PPI_{TL}^{2004}} \times r_{2004} = \frac{124.3}{102.7} \times \$2.00/\text{mi} = \$2.4206/\text{mi}$$

$$c_{TL} = \left\lceil \frac{q}{q_{\max}} \right\rceil r\, d = \left\lceil \frac{7}{6.1111} \right\rceil (2.4206)(532) = \$2,575.56$$

# Truck Shipment Example: One-Time

4. Using the most recent LTL rate estimate, what is transport charge to ship the fractional portion of the shipment LTL (i.e., the last partially full truckload portion)?

$$q_{\text{frac}} = q - q_{\max} = 7 - 6.1111 = 0.8889 \text{ ton}$$

$$r_{LTL} = PPI_{LTL}^{\text{Jan 2017}} \left[ \frac{\dfrac{s^2}{8} + 14}{\left( q_{\text{frac}}^{\frac{1}{7}} d^{\frac{15}{29}} - \dfrac{7}{2} \right) \left( s^2 + 2s + 14 \right)} \right]$$

$$= 168.2 \left[ \frac{\dfrac{4.49^2}{8} + 14}{\left( 7^{\frac{1}{7}} 532^{\frac{15}{29}} - \dfrac{7}{2} \right) \left( 4.49^2 + 2(4.49) + 14 \right)} \right] = \$2.9837 \,/\, \text{ton-mi}$$

$$c_{LTL} = r_{LTL}\, q_{\text{frac}}\, d = 2.9837(0.8889)(532) = \$1{,}410.95$$

# Truck Shipment Example: One-Time

5. What is the change in total charge associated with the combining TL and LTL as compared to just using TL?

$$\Delta c = c_{TL} - \left( \frac{c_{TL}}{2} + c_{LTL} \right)$$

$$= \left\lceil \frac{q}{q_{\max}} \right\rceil r\, d - \left( \left\lfloor \frac{q}{q_{\max}} \right\rfloor r\, d + r_{LTL}\, q_{\text{frac}}\, d \right)$$

$$= 2{,}575.56 - (1{,}287.78 + 1{,}410.95)$$

$$= 2{,}575.56 - 2{,}698.73 = -\$123.17$$

# Truck Shipment Example: One-Time

6. What would the fractional portion have to be so that the TL and LTL chargers are equal?

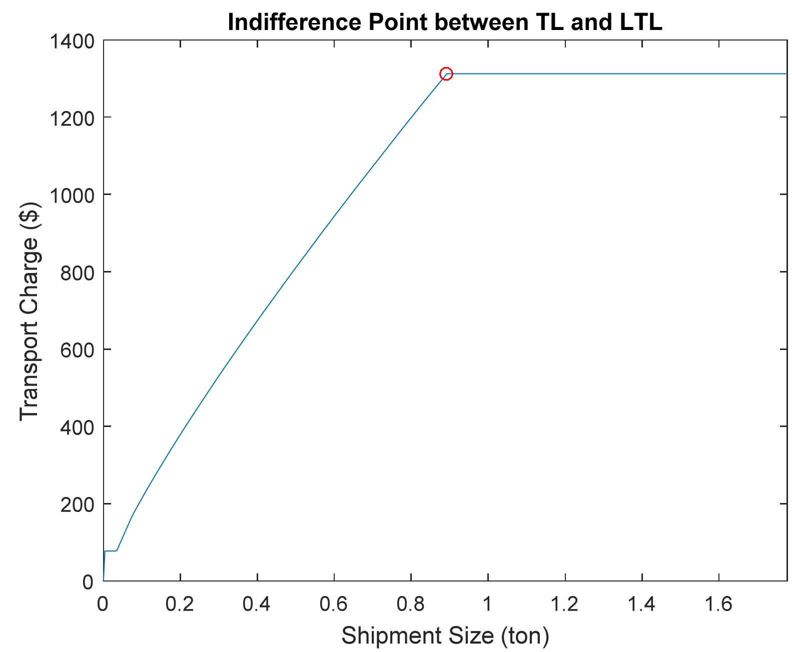
$$c_{TL}(q) = \left\lceil \frac{q}{q_{\max}} \right\rceil r\, d$$

$$r_{LTL}(q) = PPI_{LTL} \left[ \frac{\frac{s^2}{8} + 14}{\left( q^{\frac{1}{7}} d^{\frac{15}{29}} - \frac{7}{2} \right)\left( s^2 + 2s + 14 \right)} \right]$$

$$c_{LTL}(q) = r_{LTL}(q)\, q d$$

$$q_I = \arg\min_q \left( \left\| c_{TL}(q) - c_{LTL}(q) \right\| \right)$$

$$= 0.7967 \text{ ton}$$



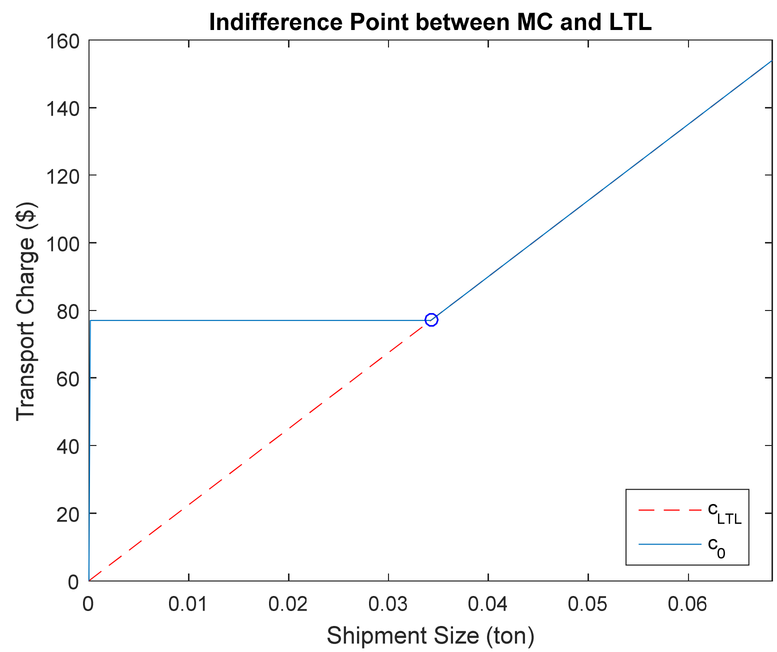
# Truck Shipment Example: One-Time

7. What are the TL and LTL minimum charges?

$$MC_{TL} = \left(\frac{r}{2}\right)45 = \left(\frac{2.4206}{2}\right)45 = \$54.46$$

$$MC_{LTL} = \left(\frac{PPI_{LTL}}{104.2}\right)\left(45 + \frac{d^{\frac{28}{19}}}{1625}\right)$$

$$= \left(\frac{168.2}{104.2}\right)\left(45 + \frac{532^{\frac{28}{19}}}{1625}\right) = \$82.97$$

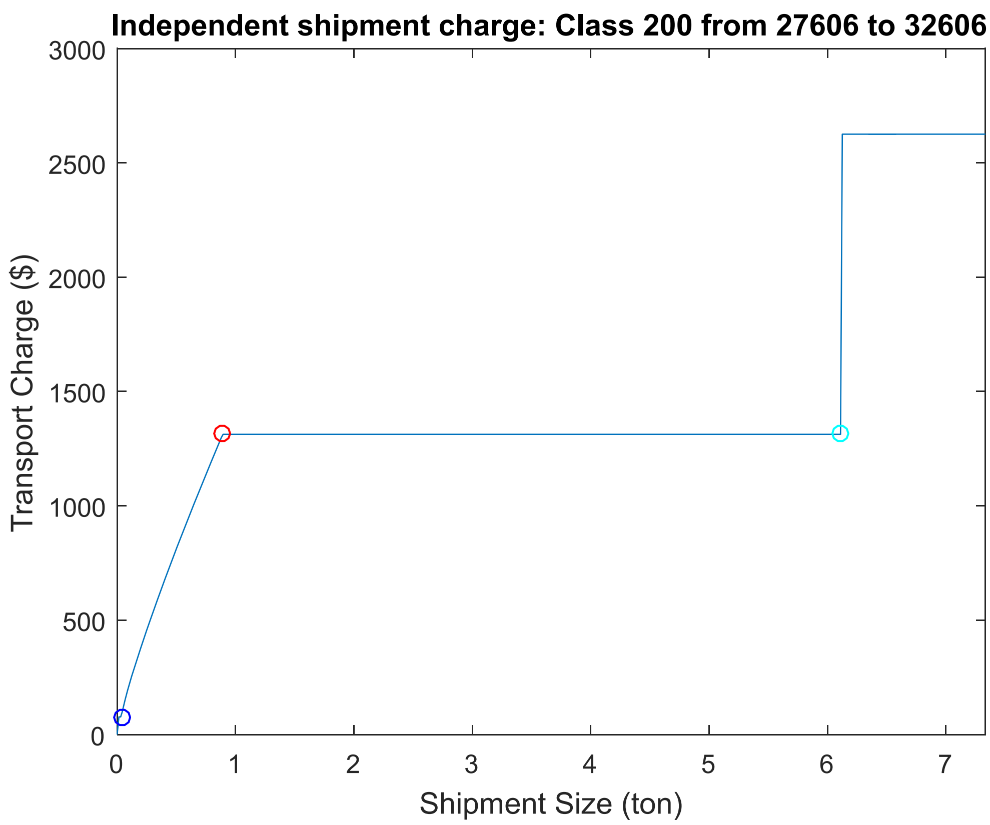



- Why do these charges not depend on the size of the shipment?
- Why does only the LTL minimum charge depend of the distance of the shipment?

# Truck Shipment Example: One-Time

- Independent Transport Charge ($):

$$c_0(q) = \min\left\{\max\left\{c_{TL}(q), MC_{TL}\right\}, \max\left\{c_{LTL}(q), MC_{LTL}\right\}\right\}$$



# Truck Shipment Example: One-Time

8. Using the same LTL shipment, find online one-time (spot) LTL rate quotes using the FedEx LTL website

$$q_{\text{frac}} = 0.8889 \text{ ton} \Rightarrow 0.8889(2000) = 1778 \text{ lb}$$

$$\Rightarrow \frac{1778}{40} \approx 44 \text{ cartons}$$

- Most likely freight class:

$$s = \frac{40 \text{ lb/unit}}{9 \text{ ft}^3\text{/unit}}$$

$$= 4.4444 \text{ lb/ft}^3$$

$$\Rightarrow \text{Class 200}$$

- What is the rate quote for the reverse trip from Gainesville (32606) to Raleigh (27606)?

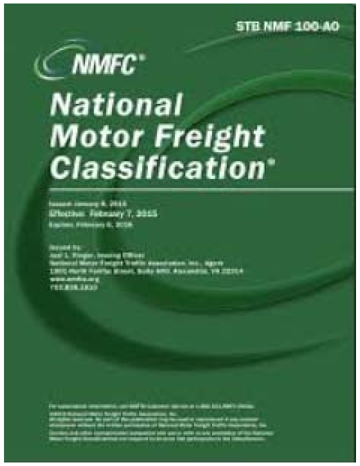
**Class-Density Relationship**

| **Class** | **Load Density** ($\text{lb/ft}^3$) Minimum | Average | **Max Physical Weight** (tons) | **Max Effective Cube** ($\text{ft}^3$) |
|---|---|---|---|---|
| 500 | – | 0.52 | 0.72 | *2,750* |
| 400 | 1 | 1.49 | 2.06 | *2,750* |
| 300 | 2 | 2.49 | 3.43 | *2,750* |
| 250 | 3 | 3.49 | 4.80 | *2,750* |
| 200 | 4 | 4.49 | 6.17 | *2,750* |
| 175 | 5 | 5.49 | 7.55 | *2,750* |
| 150 | 6 | 6.49 | 8.92 | *2,750* |
| 125 | 7 | 7.49 | 10.30 | *2,750* |
| 110 | 8 | 8.49 | 11.67 | *2,750* |
| 100 | 9 | 9.72 | 13.37 | *2,750* |
| 92.5 | 10.5 | 11.22 | 15.43 | *2,750* |
| 85 | 12 | 12.72 | 17.49 | *2,750* |
| 77.5 | 13.5 | 14.22 | 19.55 | *2,750* |
| 70 | 15 | 18.01 | 24.76 | *2,750* |
| 65 | 22.5 | 25.50 | *25* | 1,961 |
| 60 | 30 | 32.16 | *25* | 1,555 |
| 55 | 35 | 39.68 | *25* | 1,260 |
| 50 | 50 | 56.18 | *25* | 890 |

# Truck Shipment Example: One-Time

- The *National Motor Freight Classification* (NMFC) can be used to determine the product class
- Based on:
  1. Load density
  2. Special handling
  3. Stowability
  4. Liability

| Item | Description | Class | NMFC | Sub |
|---|---|---|---|---|
| **Abietic Acid** | Abietic Acid, in drums | 55 | 42605 | - |
| **Accordions** | Accordions, in boxes | 125 | 138820 | - |
| **Acetonitrile** | Acetonitrile, in boxes or drums. See item 60000 for class dependent upon released value | 85 | 42645 | - |
| **Acetylene** | in steel cylinders | 70 | 85520 | - |
| **Acid Fish Scrap** | Fish Scrap, NOI, dry, not ground, pulverized nor screened, or Acid Fish Scrap, in bags | 77.5 | 69980 | - |
| **Aircraft Parts** | metal, struts, skins, panels | 200 | 11790 | 01 |
| **Aluminum Channel** | U channel | 60 | 13340 | - |
| **Aluminum Table Set** | aluminum table SU | 200 | 82105 | 01 |
| **Ambulance Stretcher** | stretcher | 200 | 56920 | 06 |
| **Arches Support** | Iron Steel | 60 | 52460 | - |
| **Architectural Details** | 6 - 8 lbs per cubic foot | 125 | 56290 | 05 |
| **Architectural Details** | 2 - 4 lbs per cubic ft | 250 | 56290 | 03 |
| **Assembled Furniture** | Bathroom cabinet set up | 300 | 39220 | 01 |
| **Assembled Furniture** | Highboys, dressers, wooden set up | 125 | 80120 | 01 |
| **Assembled Furniture** | Wood furniture 4-6 Lbs per cu ft | 150 | 82270 | 04 |
| **Assembled Furniture** | Chairs wooden setup w/out upholstery | 300 | 80770 | 01 |
| **Assembled Furniture** | Chairs wooden setup w/out upholstery KD | 125 | 80770 | 03 |
| **Assembled Furniture** | Couch w/ back & arms put together | 175 | 80865 | 03 |
| **Assembled Furniture** | Chairs put together w/ upholstery | 200 | 79255 | 01 |
| **Assembled Furniture** | Metal cabinets in boxes | 110 | 39270 | 06 |
| **Assembled Furniture** | 18 gauge steel cabinet | 70 | 39340 | - |
| **Assembled Furniture** | Benches, cabinets, tables for workstations | 125 | 23410 | - |
| **Assembled Furniture** | Buffets, china cabinets put together | 125 | 80080 | - |
| **Assembled Furniture** | Cabinets of metal or plastic for storage | 92.5 | 39235 | - |
| **Assembled Furniture** | Tanning bed | 150 | 109050 | - |
| **Assembled Furniture** | Mattresses, in packages or boxes | 200 | 79550 | - |
| **Athletic / Sporting Goods** | Gym equipment, playground, sports items. Density Item | | | |
| **Attachments: Backhoe** | NOI: Attachments, backhoe (Backhoes), tractor or truck, on lift truck skids or pallets: | 175 | 114217 | 01 |
| **Attachments: Backhoe** | Attachments, backhoe (Backhoes), tractor or truck, on lift truck skids or pallets: Each shipped with all components secured to a single pallet, platform or skid, weighing 1100 pounds or more and having a density of 8 pounds or greater per cubic foot | 100 | 114217 | 02 |



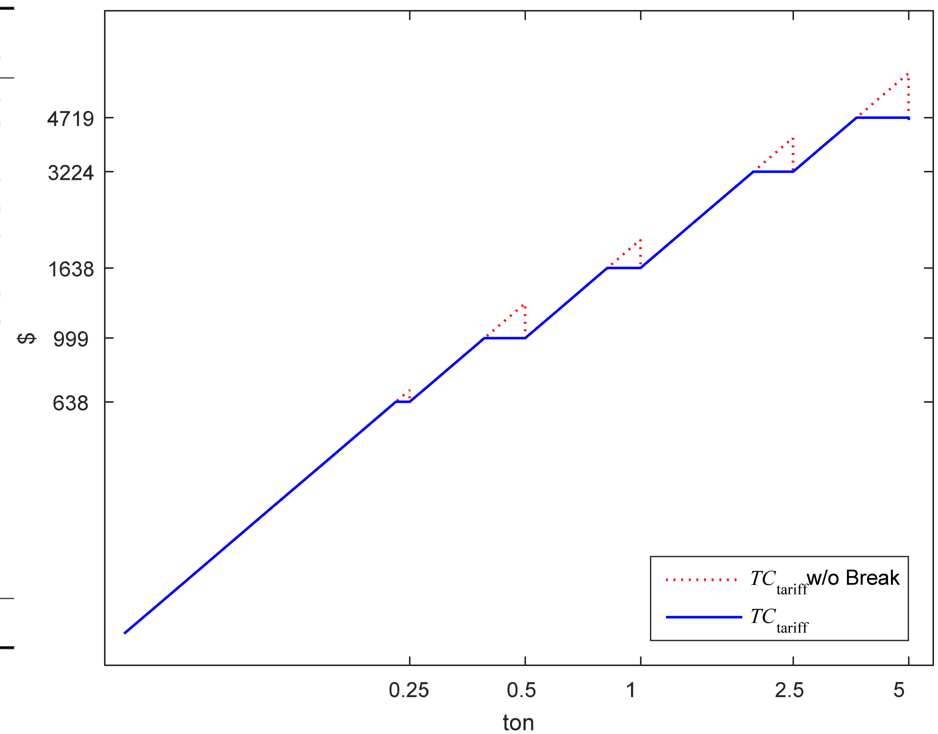
# Truck Shipment Example: One-Time

- CzarLite tariff table for O-D pair 27606-32606

$$cwt = \text{hundredweight} = 100 \text{ lb} = \frac{100}{2000} = \frac{1}{20} \text{ ton}$$

**Tariff (in \$/cwt) from Raleigh, NC (27606) to Gainesville, FL (32606)**
**(532 mi, CzarLite DEMOCZ02 04-01-2000, minimum charge = \$95.23)**

| Freight Class | Rate Breaks ($i$) 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9&10 |
|---|---|---|---|---|---|---|---|---|---|
| 500 | 341.42 | 314.14 | 245.80 | 201.48 | 158.60 | 112.37 | 55.66 | 55.66 | 55.66 |
| 400 | 273.88 | 251.99 | 197.19 | 161.61 | 127.22 | 91.12 | 45.10 | 45.10 | 45.10 |
| 300 | 206.34 | 189.85 | 148.56 | 121.76 | 95.85 | 69.47 | 34.43 | 34.43 | 34.43 |
| 250 | 172.56 | 158.77 | 124.23 | 101.83 | 80.15 | 58.03 | 28.79 | 28.79 | 28.79 |
| 200 | 138.78 | 127.69 | 99.92 | 81.89 | 64.47 | 47.19 | 23.40 | 23.40 | 23.40 |
| 175 | 121.37 | 111.68 | 87.39 | 71.62 | 56.38 | 41.27 | 20.39 | 20.39 | 20.39 |
| 150 | 104.49 | 96.13 | 75.22 | 61.66 | 48.53 | 35.96 | 17.75 | 17.75 | 17.75 |
| 125 | 87.59 | 80.60 | 63.07 | 51.69 | 40.69 | 30.24 | 15.00 | 15.00 | 15.00 |
| 110 | 77.57 | 71.37 | 55.85 | 45.77 | 36.04 | 28.61 | 14.40 | 14.40 | 14.40 |
| 100 | 71.23 | 65.55 | 51.29 | 42.04 | 33.09 | 27.58 | 14.03 | 10.80 | 9.90 |
| 92 | 66.48 | 61.18 | 47.88 | 39.24 | 30.89 | 25.75 | 13.68 | 10.52 | 9.66 |
| 85 | 61.74 | 56.80 | 44.45 | 36.43 | 28.68 | 23.91 | 13.20 | 10.15 | 9.32 |
| 77 | 56.99 | 52.44 | 41.04 | 33.63 | 26.48 | 22.07 | 12.60 | 9.68 | 8.89 |
| 70 | 52.77 | 48.55 | 37.99 | 31.14 | 24.51 | 20.43 | 12.00 | 9.23 | 8.47 |
| 65 | 50.07 | 46.08 | 36.05 | 29.56 | 23.04 | 19.39 | 11.87 | 9.14 | 8.39 |
| 60 | 47.44 | 43.64 | 34.15 | 28.00 | 21.82 | 18.37 | 11.76 | 9.04 | 8.30 |
| 55 | 44.75 | 41.17 | 32.22 | 26.40 | 20.59 | 17.32 | 11.64 | 8.96 | 8.22 |
| 50 | 41.57 | 38.26 | 29.94 | 24.54 | 19.12 | 16.10 | 11.52 | 8.85 | 8.14 |
| Tons ($q_i^B$) | 0.25 | 0.5 | 1 | 2.5 | 5 | 10 | 15 | 20 | $\infty$ |



# Truck Shipment Example: One-Time

9. Using the same LTL shipment, what is the transport cost found using the undiscounted CzarLite tariff?

$$q = 0.8889, \quad class = 200$$

$$disc = 0, \quad MC = 95.23$$

| Freight Class | Rate Breaks ($i$) 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9&10 |
|---|---|---|---|---|---|---|---|---|---|
| 500 | 341.42 | 314.14 | 245.80 | 201.48 | 158.60 | 112.37 | 55.66 | 55.66 | 55.66 |
| 400 | 273.88 | 251.99 | 197.19 | 161.61 | 127.22 | 91.12 | 45.10 | 45.10 | 45.10 |
| 300 | 206.34 | 189.85 | 148.56 | 121.76 | 95.85 | 69.47 | 34.43 | 34.43 | 34.43 |
| 250 | 172.56 | 158.77 | 124.23 | 101.83 | 80.15 | 58.03 | 28.79 | 28.79 | 28.79 |
| 200 | 138.78 | 127.69 | 99.92 | 81.89 | 64.47 | 47.19 | 23.40 | 23.40 | 23.40 |
| 175 | 121.37 | 111.68 | 87.39 | 71.62 | 56.38 | 41.27 | 20.39 | 20.39 | 20.39 |
| 50 | 41.57 | 38.26 | 29.94 | 24.54 | 19.12 | 16.10 | 11.52 | 8.85 | 8.14 |
| **Tons** ($q_i^B$) | 0.25 | 0.5 | 1 | 2.5 | 5 | 10 | 15 | 20 | $\infty$ |

$$\begin{aligned} i &= \arg\left\{q_i^B \,\middle|\, q_{i-1}^B \le q < q_i^B\right\} \\ &= \arg\left\{q_3^B \,\middle|\, q_2^B \le q < q_3^B\right\} \\ &= \arg\left\{q_3^B \,\middle|\, 0.5 \le 0.8889 < 1\right\} = 3 \end{aligned}$$

$$\begin{aligned} c_{\text{tariff}} &= \left(1 - disc\right) \max\left\{MC, \min\left\{OD(class, i)\, 20q,\, OD(class, i+1)\, 20q_i^B\right\}\right\} \\ &= \left(1 - 0\right) \max\left\{95.23, \min\left\{OD(200,3)\, 20(0.8889),\, OD(200,4)\, 20(1)\right\}\right\} \\ &= \max\left\{95.23, \min\left\{(99.92)\, 20(0.8889),\, (81.89)20(1)\right\}\right\} \\ &= \max\left\{95.23, \min\left\{1{,}776.23, 1{,}637.80\right\}\right\} = \$1{,}637.80 \end{aligned}$$

# Truck Shipment Example: One-Time

10. What is the implied discount of the estimated charge from the CzarLite tariff cost?

$$disc = \frac{c_{\text{tariff}} - c_{LTL}}{c_{\text{tariff}}}$$

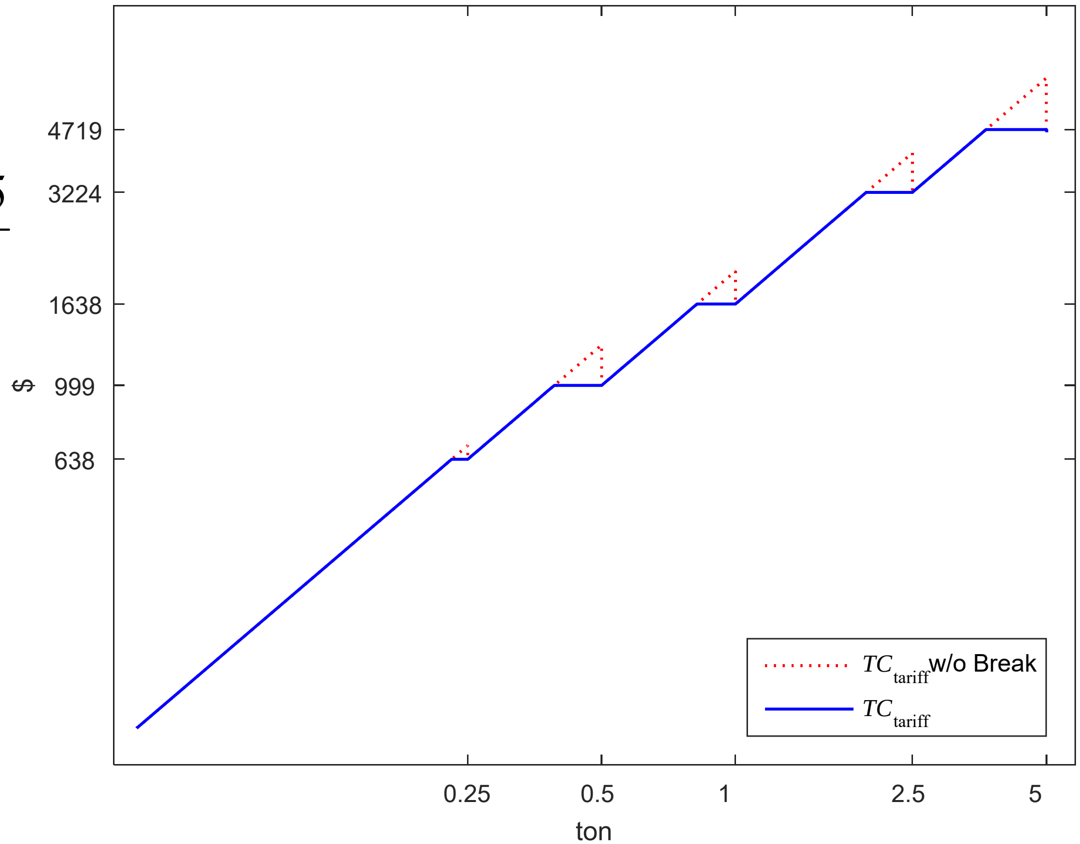
$$= \frac{1,637.80 - 1,410.95}{1,637.80}$$

$$= 14\%$$

- What is the weight break between the rate breaks?

$$q_i^W = \frac{OD(class, i+1)}{OD(class, i)} q_i^B$$

$$= \frac{81.89}{99.92}(1) = 0.8196 \text{ ton}$$



# Truck Shipment Example: Periodic

11. Continuing with the example: assuming a constant annual demand for the product of 20 tons, what is the number of full truckloads per year?

$$f = 20 \text{ ton/yr}$$

$$q = q_{\text{max}} = 6.1111 \text{ ton/ TL} \quad (\text{full truckload} \Rightarrow q \equiv q_{\text{max}})$$

$$n = \frac{f}{q} = \frac{20}{6.1111} = 3.2727 \text{ TL/yr}, \quad \text{average shipment frequency}$$

- Why should this number not be rounded to an integer value?

# Truck Shipment Example: Periodic

12. What is the shipment interval?

$$t = \frac{1}{n} = \frac{q}{f} = \frac{6.1111}{20} = 0.3056 \text{ yr/TL}, \quad \text{average shipment interval}$$

- How many days are there between shipments?

$$365.25 \text{ day/yr}$$

$$t \times 365.25 = \frac{365.25}{n} = 111.6042 \text{ day/TL}$$

# Truck Shipment Example: Periodic

13. What is the annual full-truckload transport cost?

$$d = 532 \text{ mi}, \quad r = \$2.4206 \,/\, \text{mi}$$

$$r_{FTL} = \frac{r}{q_{\max}} = \frac{2.4206}{6.1111} = \$0.3961 \,/\, \text{ton-mi}$$

$$TC_{FTL} = f\, r_{FTL}\, d = n\, r\, d \quad \left(= w\, d,\; w = \text{monetary weight in \$/mi}\right)$$

$$= 3.2727\,(2.4206)\,532 = \$4,214.56\text{/yr}$$

- What would be the cost if the shipments were to be made at least every three months?

$$t_{\max} = \frac{3}{12} \text{ yr/TL} \quad \Rightarrow \quad n_{\min} = \frac{1}{t_{\max}} = 4 \text{ TL/yr}$$

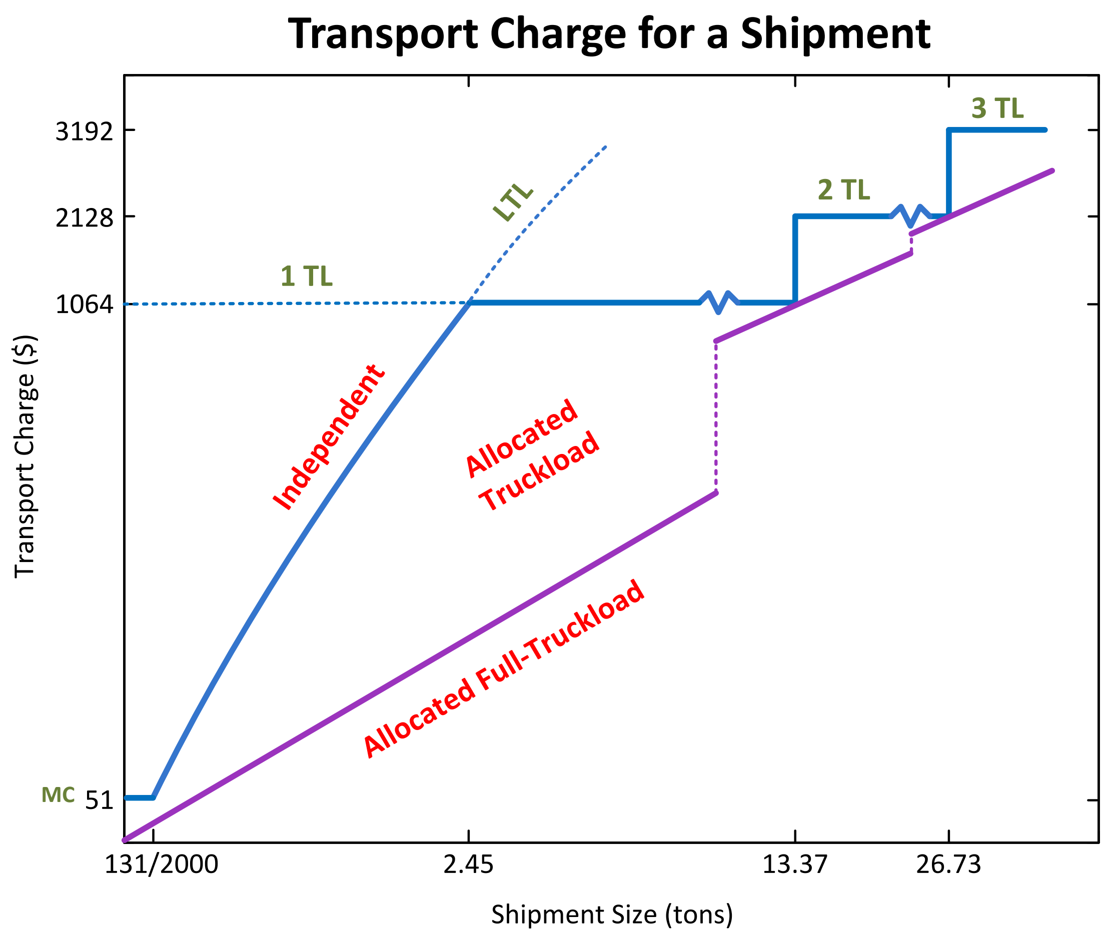
$$TC'_{FTL} = \max\left\{n, n_{\min}\right\} r\, d$$

$$= \max\left\{3.2727, 4\right\} 2.4206\,(532) = \$5,151.13\text{/yr}$$

# Truck Shipment Example: Periodic

- Independent and allocated full-truckload charges:

$$q \le q_{\max} \Rightarrow \left[c_{UB}, c_{LB}\right] = \left[\mathrm{c}_0(q),\, q r_{FTL}\, d\right]$$



# Truck Shipment Example: Periodic

- *Total Logistics Cost* (TLC) includes all costs that could change as a result of a logistics-related decision

$$
\begin{aligned}
TLC &= TC + IC + PC \\
TC &= \text{transport cost} \\
IC &= \text{inventory cost} \\
&= IC_{\text{cycle}} + IC_{\text{pipeline}} + IC_{\text{safety}} \\
PC &= \text{purchase cost}
\end{aligned}
$$

- *Cycle inventory*: held to allow cheaper large shipments
- *Pipeline inventory*: goods in transit or awaiting transshipment
- *Safety stock*: held due to transport uncertainty
- *Purchase cost*: can be different for different suppliers

# Truck Shipment Example: Periodic

14. Since demand is constant throughout the year, one half of a shipment is stored at the customer, on average. Assuming that the production rate is also constant, one half of a shipment will also be stored at the supplier, on average. Assuming each ton of the product is valued at $25,000, what is a “reasonable estimate” for the total annual cost for this cycle inventory?

$$IC_{\text{cycle}} = (\text{annual cost of holding one ton})(\text{average annual inventory level})$$

$$= (vh)(\alpha q)$$

$$v = \text{unit value of shipment (\$/ton)}$$

$$h = \text{inventory carrying rate, the cost per dollar of inventory per year (1/yr)}$$

$$\alpha = \text{average inter-shipment inventory fraction at Origin and Destination}$$

$$q = \text{shipment size (ton)}$$

# Truck Shipment Example: Periodic

Rate ($h$) = interest + warehousing + obsolescence rate

- Interest: 4% real interest rate (prime + 2% – inflation rate)
- Warehousing: 6% per Total U.S. Logistics Costs
- Obsolescence: default rate $h_{annual}$ = 0.3 $\Rightarrow$ $h_{obsol}$ = 0.2
  - Estimate $h_{obsol}$ using "fifth-of-value interval" method:
    given $t$ when product worth 20% of its original value $v$, find $h_{obsol}$

$$th_{\text{obsol}}v = \frac{4}{5}v \Rightarrow h_{\text{obsol}} = \frac{4v}{5tv} = \frac{4}{5t}, \quad t = \frac{4}{5h_{\text{obsol}}}$$

  - Example: If a product loses 80% of its value after 5 years:

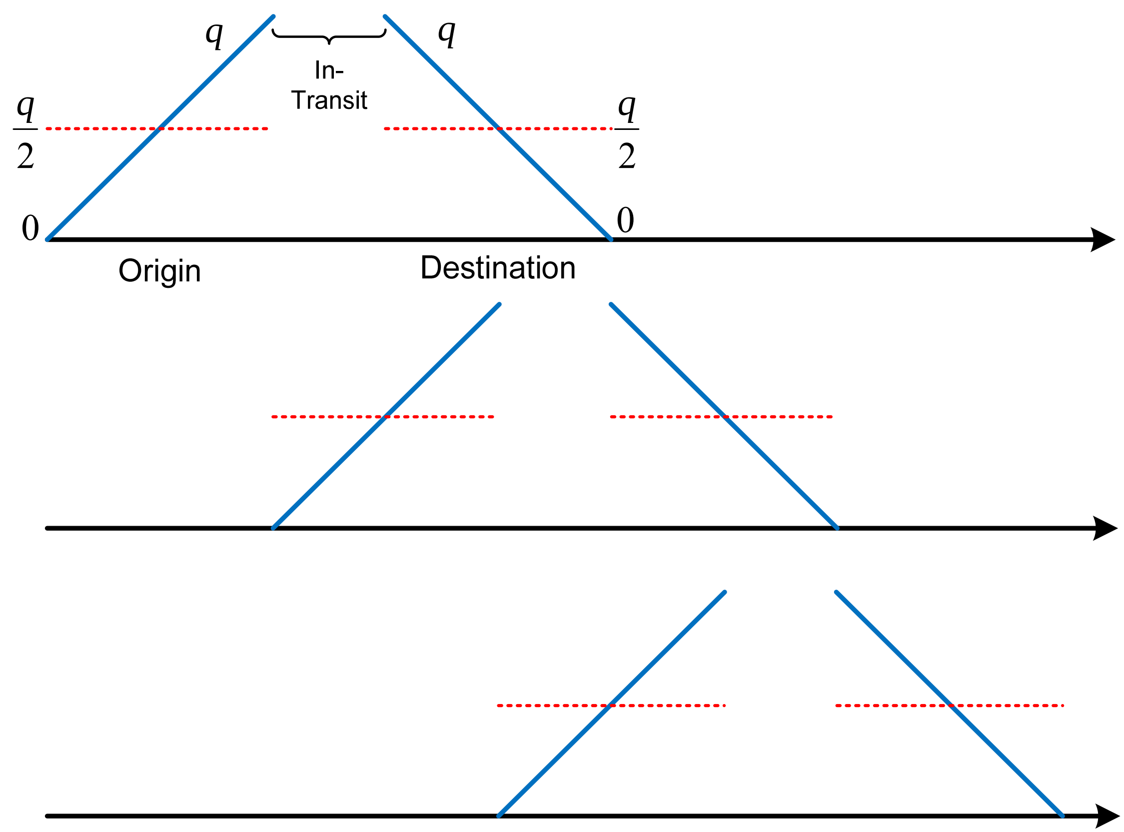
$$t = 5 \Rightarrow h_{\text{obsol}} = \frac{4}{5t} = \frac{4}{5(5)} = 0.16 \quad (16\%)$$

  - Example: If a product loses 80% of its value after two weeks:

$$t = \frac{14}{365.25} \Rightarrow h_{\text{obsol}} = \frac{4}{5t} = \frac{4}{5(0.0383)} = 20.87 \quad (2{,}087\%)$$

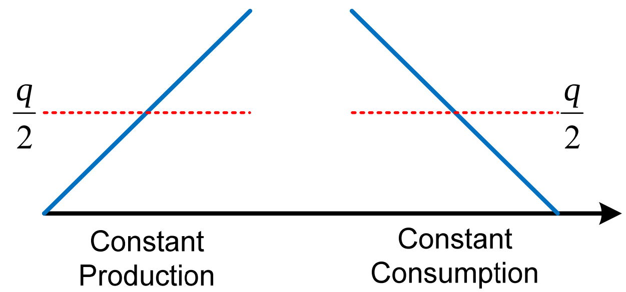
# Truck Shipment Example: Periodic

- Average annual inventory level $= \dfrac{q}{2} + \dfrac{q}{2} = \left(\dfrac{1}{2} + \dfrac{1}{2}\right) q = (1)q \Rightarrow \alpha = 1$

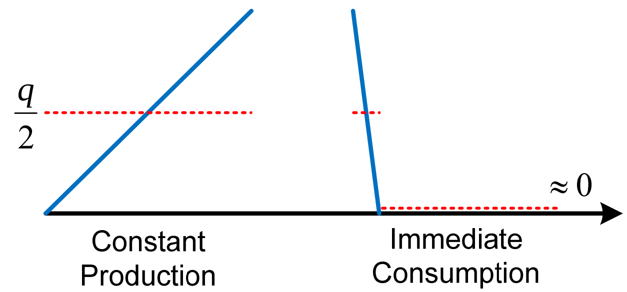


# Truck Shipment Example: Periodic

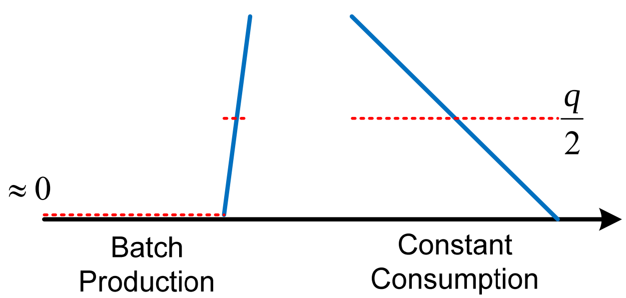
- Inter-shipment inventory fraction alternatives: $\alpha = \alpha_O + \alpha_D$



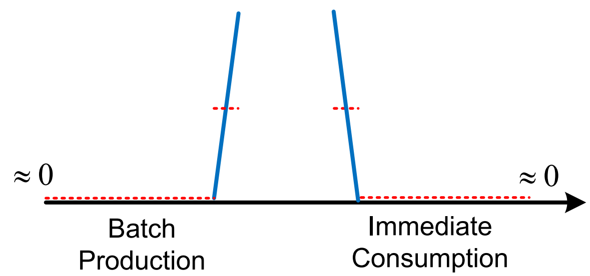

$$\alpha = \frac{1}{2} + \frac{1}{2} = 1$$




$$\alpha = \frac{1}{2} + 0 = \frac{1}{2}$$




$$\alpha = 0 + \frac{1}{2} = \frac{1}{2}$$




$$\alpha = 0 + 0 = 0$$

# Truck Shipment Example: Periodic

- "Reasonable estimate" for the total annual cost for the cycle inventory:

$$IC_{\text{cycle}} = \alpha vhq$$
$$= (1)(25,000)(0.3)6.1111$$
$$= \$45,833.33 / \text{yr}$$

where

$$\alpha = \frac{1}{2}\text{at Origin} + \frac{1}{2}\text{at Destination} = 1$$
$$v = \$25,000 = \text{unit value of shipment (\$/ton)}$$
$$h = 0.3 = \text{estimated carrying rate for manufactured products (1/yr)}$$
$$q = q_{\text{max}} = 6.111 = \text{FTL shipment size (ton)}$$

# Truck Shipment Example: Periodic

15. What is the annual total logistics cost (TLC) for these full-truckload TL shipments?

$$
\begin{aligned}
TLC_{FTL} &= TC_{FTL} + IC_{\text{cycle}} \\
&= n\,r\,d + \alpha vhq \\
&= 3.2727\,(2.4206)\,532 + (1)(25,000)(0.3)6.1111 \\
&= 4,214.56 + 45,833.33 \\
&= \$50,047.89\,/\,yr
\end{aligned}
$$

# Truck Shipment Example: Periodic

16. What is minimum possible annual total logistics cost for TL shipments, where the shipment size can now be less than a full truckload?

$$TLC_{TL}(q) = TC_{TL}(q) + IC(q) = \frac{f}{q} c_{TL}(q) + \alpha vhq = \frac{f}{q} rd + \alpha vhq$$

$$\frac{dTLC_{TL}(q)}{dq} = 0 \Rightarrow q_{TL}^* = \sqrt{\frac{frd}{\alpha vh}} = \sqrt{\frac{20(2.4206)532}{(1)25000(0.3)}} = 1.8531 \text{ ton}$$

$$\begin{aligned} TLC_{TL}(q_{TL}^*) &= \frac{f}{q_{TL}^*} rd + \alpha vhq_{TL}^* \\ &= \frac{20}{1.8553}(2.4206)532 + (1)25000(0.3)1.8553 \\ &= 13{,}898.46 + 13{,}898.46 \\ &= \$27{,}796.93 / \text{yr} \end{aligned}$$

# Truck Shipment Example: Periodic

- Including the minimum charge and maximum payload restrictions:

$$q_{TL}^{*} = \min\left\{\sqrt{\frac{f\ \max\left\{rd, MC_{TL}\right\}}{\alpha vh}}, q_{\max}\right\} \approx \sqrt{\frac{frd}{\alpha vh}}$$

# Truck Shipment Example: Periodic

17. What is the optimal LTL shipment size?

$$TLC_{LTL}(q) = TC_{LTL}(q) + IC(q) = \frac{f}{q} c_{LTL}(q) + \alpha v h q$$

$$q^*_{LTL} = \arg\min_q TLC_{LTL}(q) = 0.7288 \text{ ton}$$

- Must be careful in picking starting point for optimization since random choice likely to be infeasible:

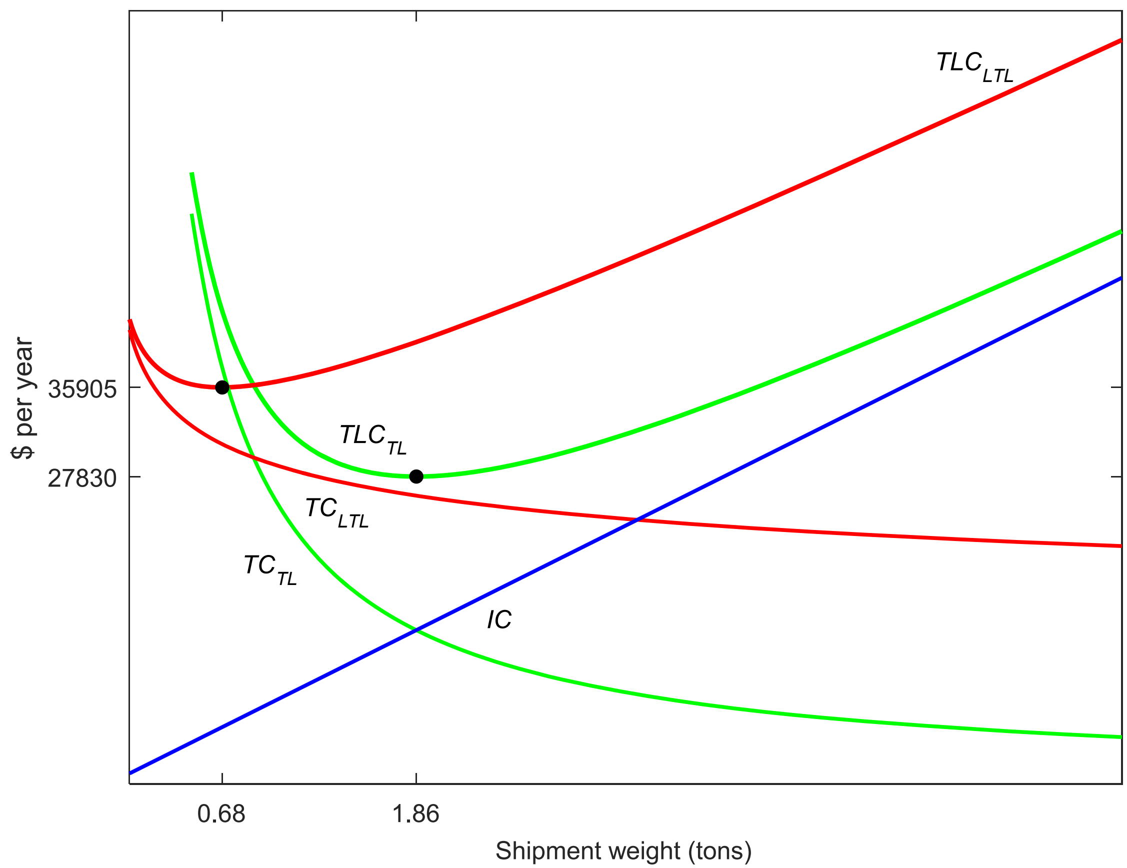
$$r_{LTL} = PPI_{LTL} \left[ \frac{\frac{s^2}{8} + 14}{\left( q^{\frac{1}{7}} d^{\frac{15}{29}} - \frac{7}{2} \right) \left( s^2 + 2s + 14 \right)} \right]$$

$$\frac{150}{2{,}000} \le q \le \frac{10{,}000}{2{,}000}, \quad 37 \le d \le 3354, \quad 2000 \frac{q}{s} \le 650 \text{ ft}^3$$

# Truck Shipment Example: Periodic

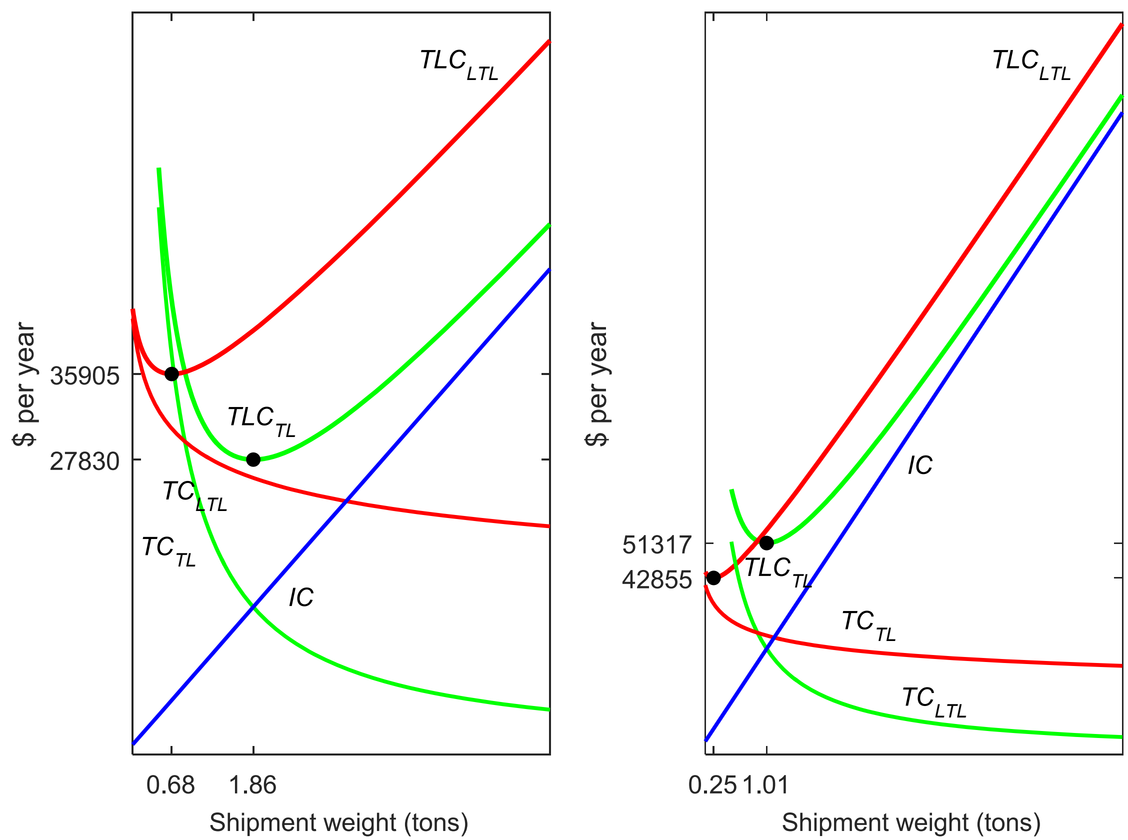
18. Should the product be shipped TL or LTL?

$$TLC_{LTL}(q^*_{LTL}) = TC_{LTL}(q^*_{LTL}) + IC(q^*_{LTL}) = 32{,}811.57 + 5{,}466.06 = \$38{,}277.63\,/\,\text{yr}$$



# Truck Shipment Example: Periodic

19. If the value of the product increased to \$85,000 per ton, should the product be shipped TL or LTL?



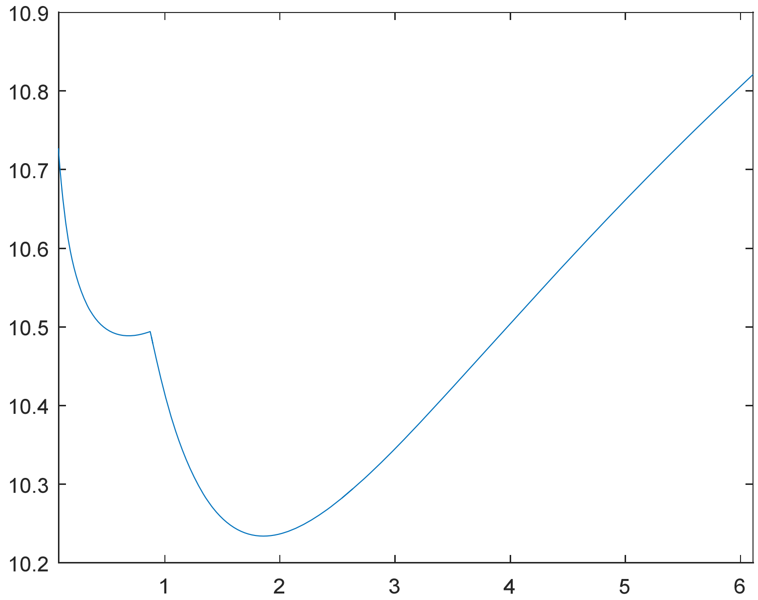

(a) \$25000 value per ton

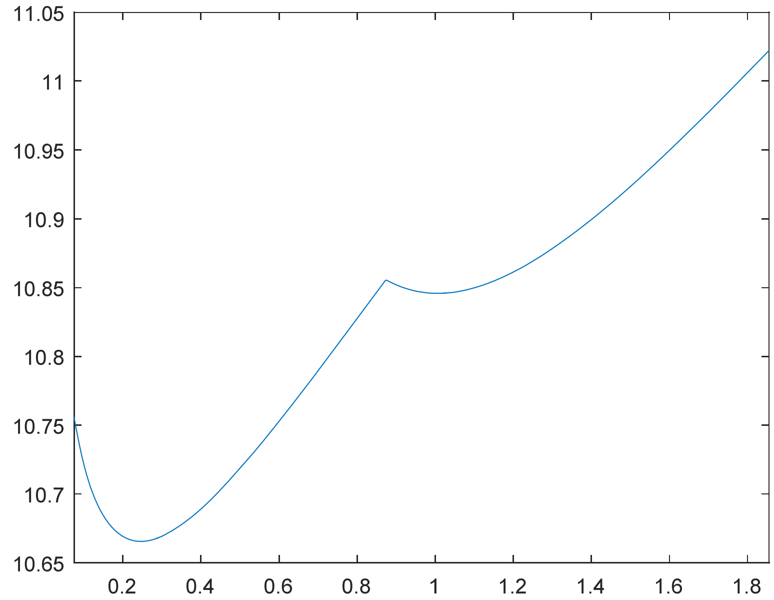
(b) \$85000 value per ton

# Truck Shipment Example: Periodic

- Better to pick from separate optimal TL and LTL because independent charge has two local minima:

$$q_0^* = \arg\min_q \left\{ TLC_{TL}(q), TLC_{LTL}(q) \right\} \qquad q_0^* \overset{!}{=} \arg\min_q \left\{ \frac{f}{q} c_0(q) + \alpha v h q \right\}$$



# Truck Shipment Example: Periodic

20. What is optimal independent shipment size to ship 80 tons per year of a Class 60 product valued at \$5000 per ton between Raleigh and Gainesville?

$$s = 32.16 \text{ lb/ft}^3$$

$$q_0^* = \arg\min_q \left\{ TLC_{TL}(q), TLC_{LTL}(q) \right\} = 8.29 \text{ ton}$$

$$TLC_{TL}(q_0^*) = \$24,862.33 \,/\, \text{yr} < TLC_{LTL}(q_0^*)$$

# Truck Shipment Example: Periodic

21. What is the optimal shipment size if both shipments will always be shipped together on the same truck (with same shipment interval)?

$$d_1 = d_2, \quad h_1 = h_2, \quad \alpha_1 = \alpha_2$$

$$f_{\text{agg}} = f_1 + f_2 = 20 + 80 = 100 \text{ ton}$$

$$s_{\text{agg}} = \frac{(\text{aggregate weight, in lb})}{(\text{aggregate cube, in ft}^3)} = \frac{f_{\text{agg}}}{\frac{f_1}{s_1} + \frac{f_2}{s_2}} = \frac{100}{\frac{20}{4.44} + \frac{80}{32.16}} = 14.31 \text{ lb/ft}^3$$

$$v_{\text{agg}} = \frac{f_1}{f_{\text{agg}}} v_1 + \frac{f_2}{f_{\text{agg}}} v_2 = \frac{20}{100} 85{,}000 + \frac{80}{100} 5000 = \$21{,}000 / \text{ton}$$

$$q_{TL}^* = \sqrt{\frac{f_{\text{agg}} r d}{\alpha v_{\text{agg}} h}} = \sqrt{\frac{100(2.4206)532}{(1)21000(0.3)}} = 4.5212 \text{ ton}$$

# Truck Shipment Example: Periodic

- Summary of results:

| | f | s | v | qmax | TLC | q | t |
|---|---|---|---|---|---|---|---|
| 1: | 20 | 4.44 | 85,000 | 6.11 | 45,675.80 | 0.26 | 4.78 |
| 2: | 80 | 32.16 | 5,000 | 25.00 | 24,862.33 | 8.29 | 37.84 |
| 1+2: | | | | | 70,538.13 | | |
| Aggregate: | 100 | 14.31 | 21,000 | 19.68 | 56,966.75 | 4.52 | 16.51 |

# Location and Transport Costs

- Monetary weights *w* used for location are, in general, a function of the location of a NF
  - Distance *d* appears in optimal TL size formula
  - TC & IC functions of location $\Rightarrow$ Need to minimize TLC instead of TC
  - FTL (since size is fixed at max payload) results in only constant weights for location $\Rightarrow$ Need to only minimize TC since IC is constant in TLC

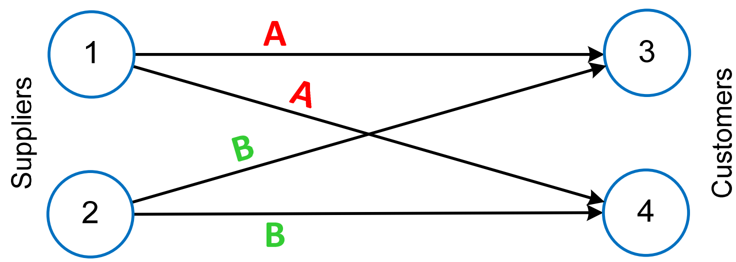
$$TLC_{TL}(\mathbf{x}) = \sum_{i=1}^{m} w_i(\mathbf{x}) d_i(\mathbf{x}) + \alpha v h q_i(\mathbf{x}) = \sum_{i=1}^{m} \frac{f_i}{q_i(\mathbf{x})} r d_i(\mathbf{x}) + \alpha v h\, q_i(\mathbf{x})$$

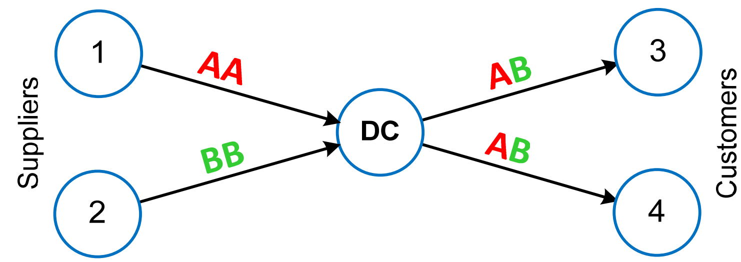
$$= \sum_{i=1}^{m} \frac{f_i}{\sqrt{\frac{f_i r d_i(\mathbf{x})}{\alpha v h}}} r d_i(\mathbf{x}) + \alpha v h \sqrt{\frac{f_i r d_i(\mathbf{x})}{\alpha v h}} = \sum_{i=1}^{m} \sqrt{\frac{f_i r d_i(\mathbf{x})}{\alpha v h}} + \sqrt{f_i r d_i(\mathbf{x})}\sqrt{\alpha v h}$$

$$TLC_{FTL}(\mathbf{x}) = \sum_{i=1}^{m} \frac{f_i}{q_{\max}} r d_i(\mathbf{x}) + \alpha v h\, q_{\max} = \sum_{i=1}^{m} w_i\, d_i(\mathbf{x}) + \alpha v h\, q_{\max} = TC_{FTL}(\mathbf{x}) + \text{constant}$$

# Transshipment

- *Direct*: P2P shipments from Suppliers to Customers



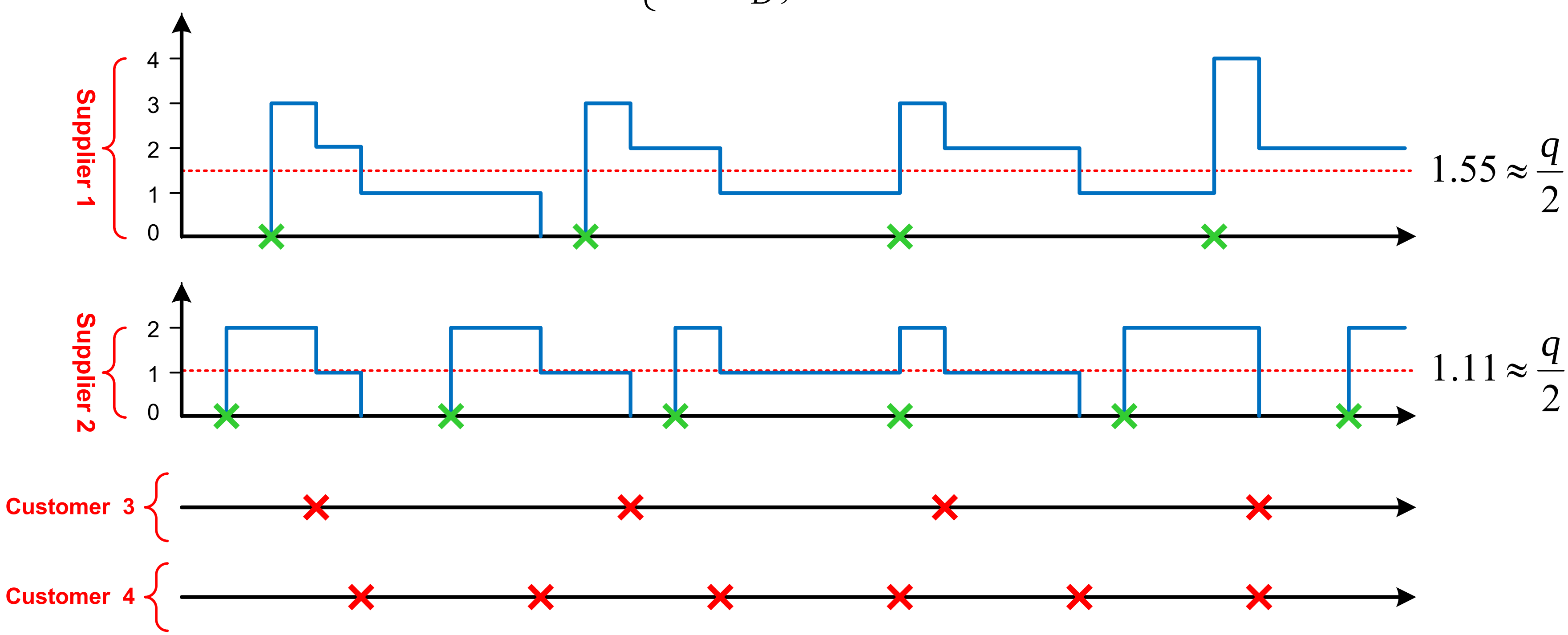

- *Transshipment*: use DC to consolidate outbound shipments
  - *Uncoordinated*: determine separately each optimal inbound and outbound shipment ⇒ hold inventory at DC
  - *Cross-dock*: use single shipment interval for all inbound and outbound shipments ⇒ no inventory at DC

# Uncoordinated Inventory

- Average pipeline inventory level at DC:

$$\alpha = \alpha_O + \alpha_D$$

$$= \begin{cases} \alpha_O + \dfrac{1}{2}, & \text{inbound} \\ 0 + \alpha_D, & \text{outbound} \end{cases}$$



# TLC with Transshipment

- Uncoordinated: $TLC_i = TLC$ of supplier/customer $i$

$$q_i^* = \arg\min_q TLC_i(q)$$

$$TLC^* = \sum TLC_i\left(q_i^*\right)$$

- Cross-docking: $t = \dfrac{q}{f}$, shipment interval

$$TLC_i(t) = \frac{c_0(t)}{t} + \alpha v h f t$$

$$c_0(t) = \text{independent transport charge as function of } t$$

$$\alpha = \begin{cases} \alpha_O + 0, & \text{inbound} \\ 0 + \alpha_D, & \text{outbound} \end{cases}$$

$$t^* = \arg\min_t \sum TLC_i(t)$$

$$TLC^* = \sum TLC_i\left(t^*\right)$$

# TLC and Location

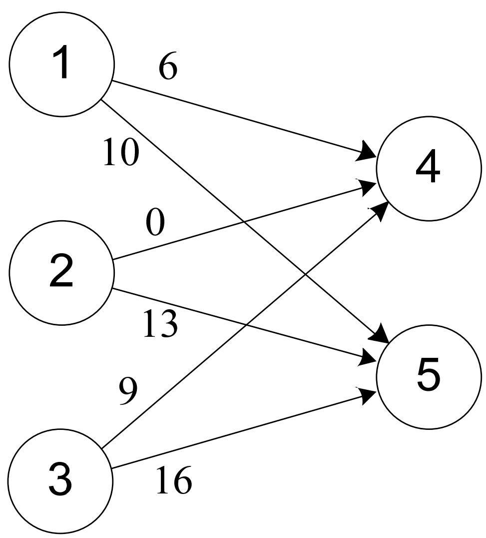
- TLC should include all logistics-related costs
  - $\Rightarrow$ TLC can be used as sole objective for network design (incl. location)
- Assume public warehouses to be used for all DCs
  - $\Rightarrow$ Pay only for time each unit spends in WH $\Rightarrow$ No fixed cost at DC
- Transport fixed costs:
  - Costs that are independent of shipment size (e.g., $/mi), that make it worthwhile to incur the inventory cost associated with larger shipment sizes in order to spread out the fixed cost
- Example: Optimal Number DCs for Lowe's
  - Mix of top-down (COGS) and bottom-up (typical load/TL parameters)
  - FTL for all inbound and outbound shipments
  - ALA used to determine TC for given number of DCs
  - IC = $\alpha v h q_{max}$ x (number of suppliers x number of DCs + number stores)
  - Ignoring max DC-to-store distance constraints, consolidation, etc.

# Topics

1. Introduction
2. Facility location
3. Freight transport
   – Midterm exam
4. **Network models**
5. Routing
6. Warehousing
   – Final project
   – Final exam

# Graph Representations

- Complete bipartite directed (or digraph):
  - Suppliers to multiple DCs, single mode of transport




```
C:   1   2
-:-------
1:   6  10
2:   0  13
3:   9  16
```

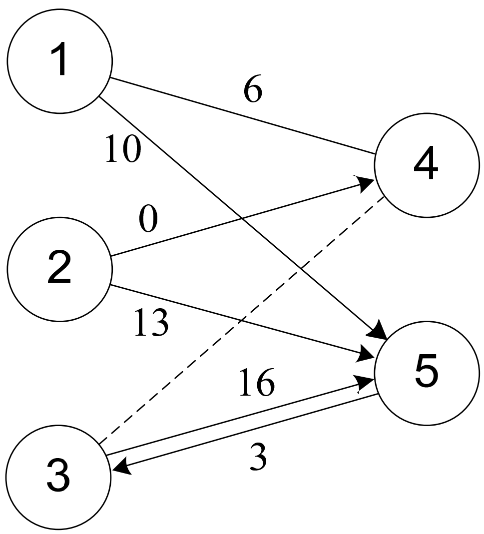
Interlevel matrix

```
W =    0     0     0     6    10
       0     0     0   NaN    13
       0     0     0     9    16
       0     0     0     0     0
       0     0     0     0     0
```

Weighted adjacency matrix

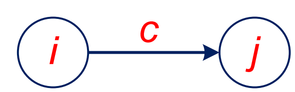
# Graph Representations

- Bipartite:
  - One- or two-way connections between nodes in two groups



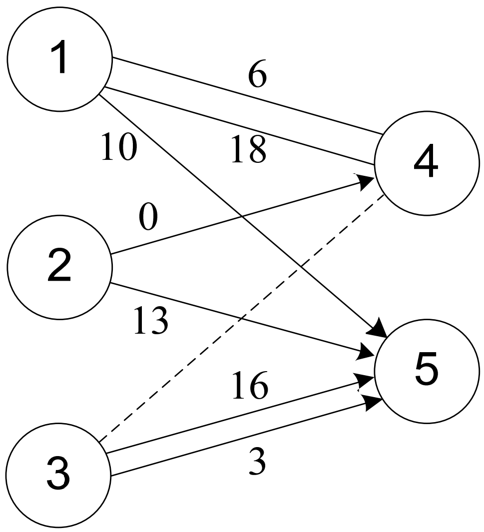

```
W =    0    0    0    6    10
       0    0    0  NaN   13
       0    0    0    0   16
       6    0    0    0    0
       0    0    3    0    0
IJC =
       4    1    6
       5    3    3
       1    4    6
       2    4    0
       1    5   10
       2    5   13
       3    5   16
```

Arc list matrix



# Graph Representations

- Multigraph:
  - Multiple connections, multiple modes of transport

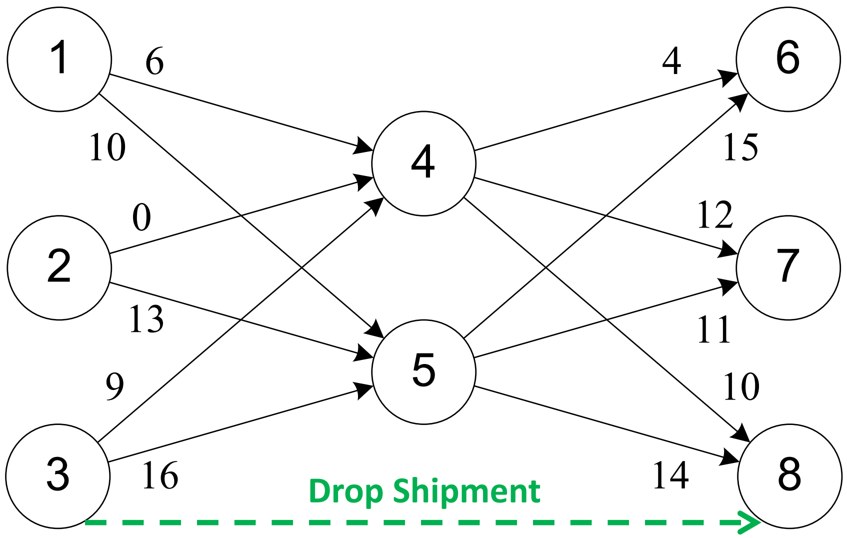



```
IJC = 1     -4     6
      1     -4    18
      1      5    10
      2      4     0
      2      5    13
      3      5    16
      3      5     3
no_W =
      0      0     0    24    10
      0      0     0   NaN    13
      0      0     0     0    19
     24      0     0     0     0
      0      0     0     0     0
```

Can’t represent using adjacency matrix

# Graph Representations

- Complete multipartite directed:
  - Typical supply chain (no drop shipments)




Level 1 (suppliers)

Level 2 (DCs)

Level 3 (customers)

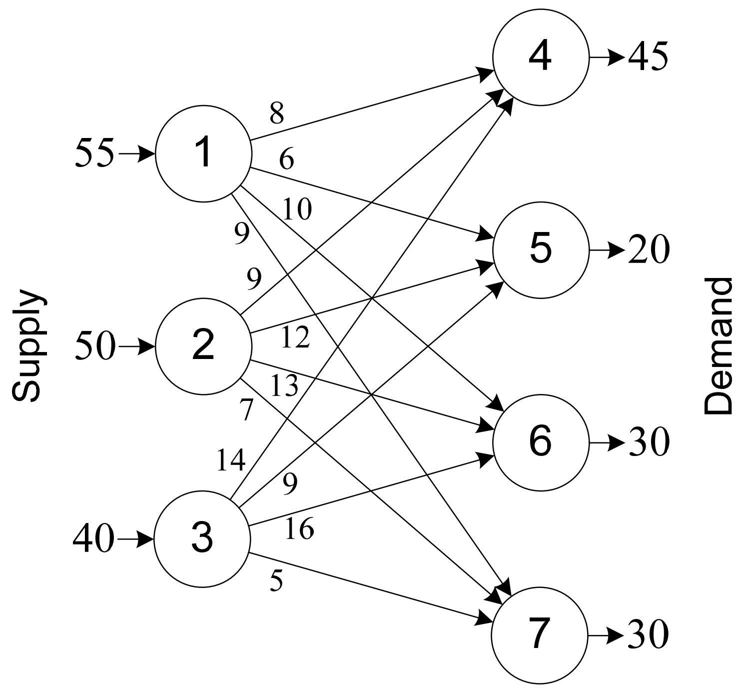
| C12: | 1 | 2 |
|---|---|---|
| 1: | 6 | 10 |
| 2: | 0 | 13 |
| 3: | 9 | 16 |

| C23: | 1 | 2 | 3 |
|---|---|---|---|
| 1: | 4 | 12 | 10 |
| 2: | 15 | 11 | 14 |

| W: | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1: | 0 | 0 | 0 | 6 | 10 | 0 | 0 | 0 |
| 2: | 0 | 0 | 0 | | 13 | 0 | 0 | 0 |
| 3: | 0 | 0 | 0 | 9 | 16 | 0 | 0 | 0 |
| 4: | 0 | 0 | 0 | 0 | 0 | 4 | 12 | 10 |
| 5: | 0 | 0 | 0 | 0 | 0 | 15 | 11 | 14 |
| 6: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# Transportation Problem

- Satisfy node demand from supply nodes
  - Can be used for allocation in ALA when NFs have capacity constraints
  - Min cost/distance allocation = infinite supply at each node




| Trans | 4 | 5 | 6 | 7 | Supply |
|---|---|---|---|---|---|
| 1 | 8 | 6 | 10 | 9 | 55 |
| 2 | 9 | 12 | 13 | 7 | 50 |
| 3 | 14 | 9 | 16 | 5 | 40 |
| Demand | 45 | 20 | 30 | 30 | |

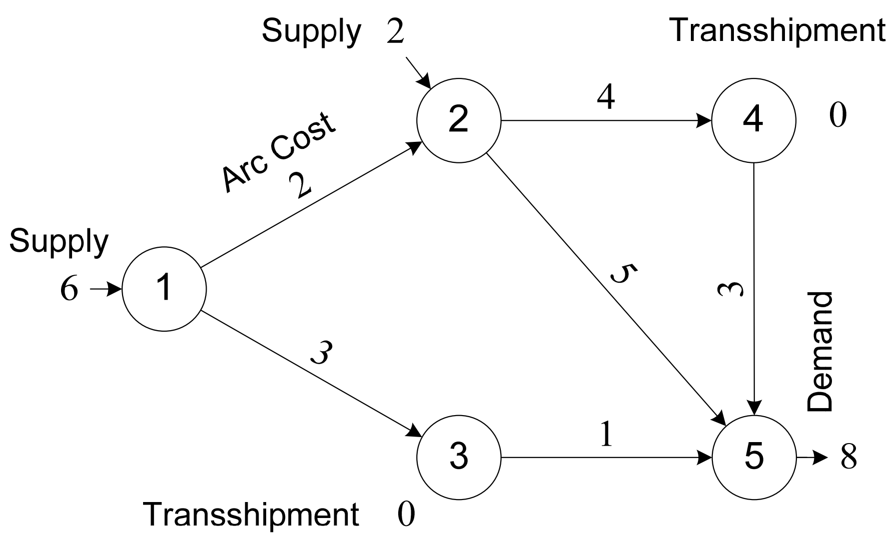
# Greedy Solution Procedure

- Procedure for transportation problem: *Continue to select lowest cost supply until all demand is satisfied*
  - Fast, but not always optimal for transportation problem
  - Dijkstra's shortest path and simplex method for LP are optimal greedy procedures

| Trans | 4 | 5 | 6 | 7 | Supply |
|---|---|---|---|---|---|
| 1 | 8 | 6 | 10 | 9 | 55 |
| 2 | 9 | 12 | 13 | 7 | 50 |
| 3 | 14 | 9 | 16 | 5 | 40 |
| Demand | 45 | 20 | 30 | 30 | |

# Min Cost Network Flow (MCNF) Problem

- Most general network problem, can solve using any type of graph representation




$$s_i = \text{net supply of node } i$$

$$= \begin{cases} > 0, & \text{supply node} \\ < 0, & \text{demand node} \\ = 0, & \text{transshipment node} \end{cases}$$

$$\text{Arc cost:} \quad \mathbf{c} = \begin{bmatrix} 2 & 3 & 4 & 5 & 1 & 3 \end{bmatrix}'$$

$$\text{Net node supply:} \quad \mathbf{s} = \begin{bmatrix} 6 & 2 & 0 & 0 & -8 \end{bmatrix}'$$

$$\text{Incidence Matrix:} \ \mathbf{A} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 & -1 & -1 \end{bmatrix}$$

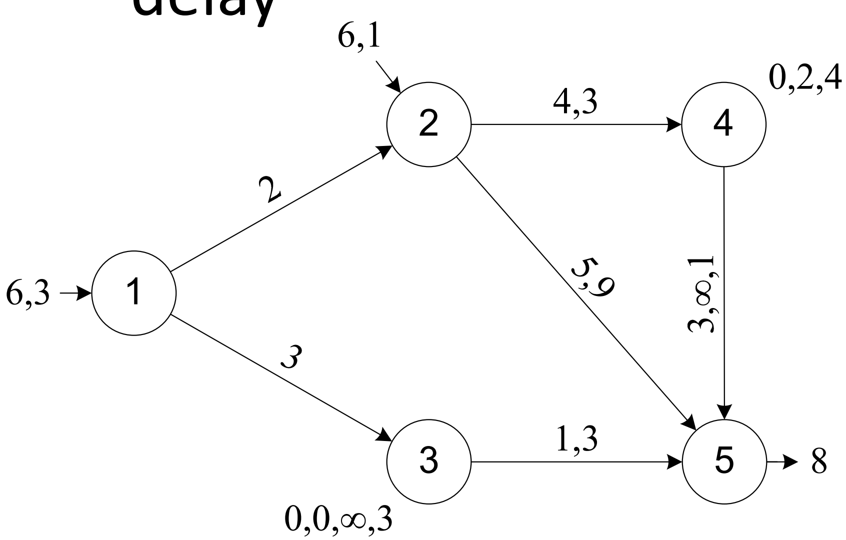
```
MCNF: lhs   C    C    C    C    C    C   rhs
----:--------------------------------------
 Min:        2    3    4    5    1    3
   1:   6    1    1    0    0    0    0   6
   2:   2   -1    0    1    1    0    0   2
   3:   0    0   -1    0    0    1    0   0
   4:   0    0    0   -1    0    0    1   0
  lb:        0    0    0    0    0    0
  ub:      Inf  Inf  Inf  Inf  Inf  Inf
```

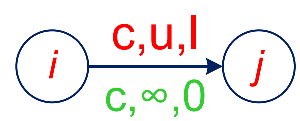
Row for node 5 is redundant

$$\begin{aligned} \text{MCNF:} \quad & \max \ \mathbf{c}'\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{s} \\ & \mathbf{x} \ge 0 \end{aligned}$$

# MCNF with Arc/Node Bounds and Node Costs

- Bounds on arcs/nodes can represent capacity constraints in a logistic network
- Node cost can represent production cost or intersection delay

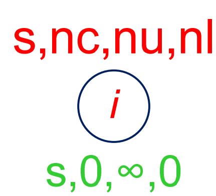



| IJCUL: | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1: | 1 | 2 | 2 | Inf | 0 |
| 2: | 1 | 3 | 3 | Inf | 0 |
| 3: | 2 | 4 | 4 | 3 | 0 |
| 4: | 2 | 5 | 5 | 9 | 0 |
| 5: | 3 | 5 | 1 | 3 | 0 |
| 6: | 4 | 5 | 3 | Inf | 1 |

$$s_i = \text{net supply of node } i$$

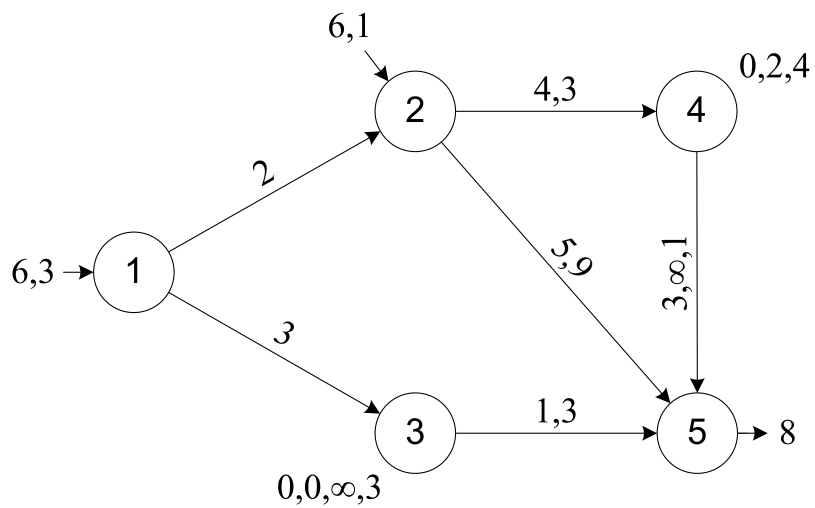
$$= \begin{cases} > 0, & \text{supply node} \\ < 0, & \text{demand node} \\ = 0, & \text{transshipment node} \end{cases}$$

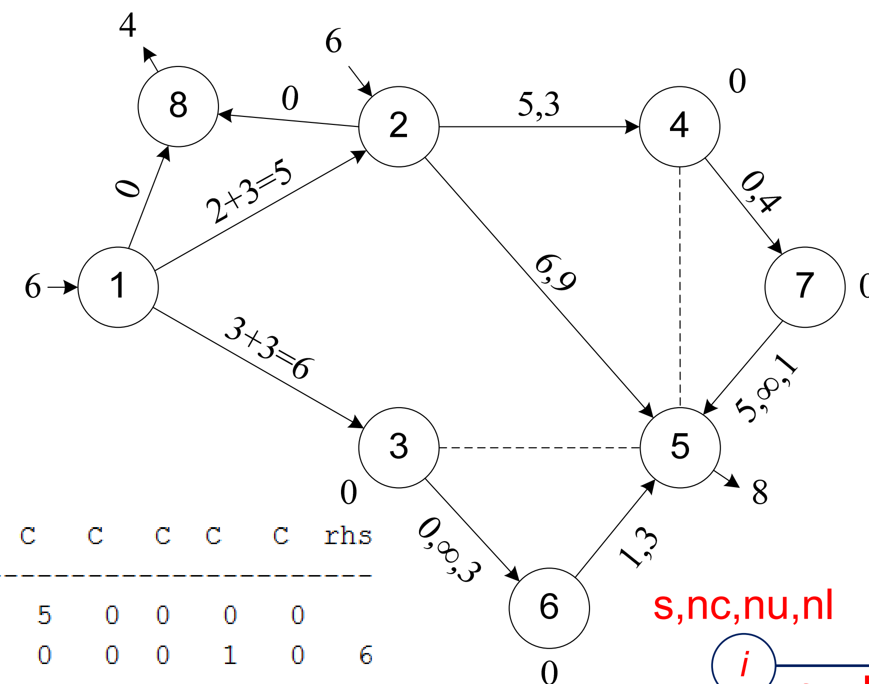



| SCUL: | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1: | 6 | 3 | Inf | 0 |
| 2: | 6 | 1 | Inf | 0 |
| 3: | 0 | 0 | Inf | 3 |
| 4: | 0 | 2 | 4 | 0 |
| 5: | -8 | 0 | Inf | 0 |

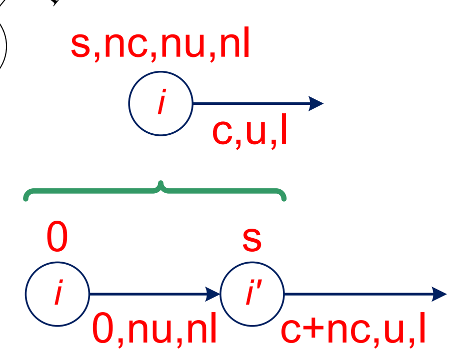
# Expanded-Node Formulation of MCNF

- Node cost/constraints converted to arc cost/constraints
  - Dummy node (8) added so that supply = demand



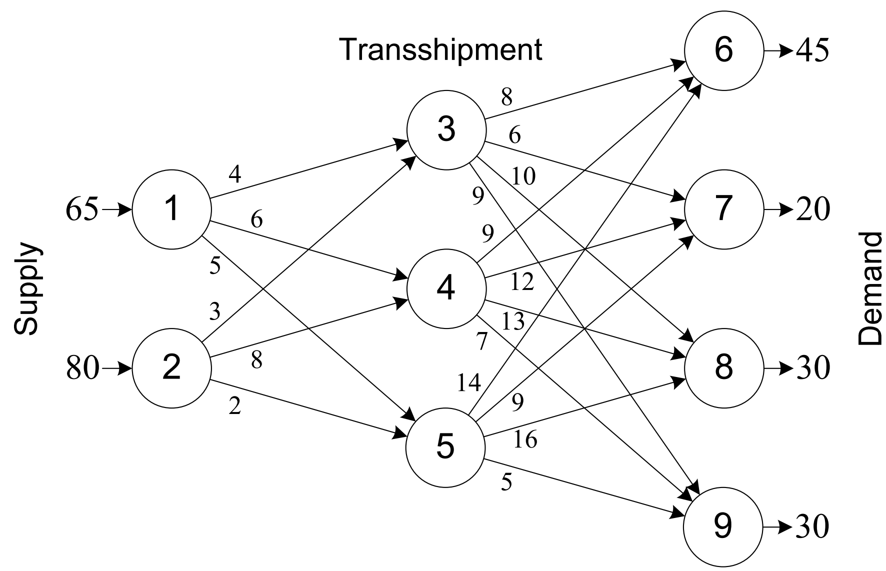




| MCNF: | lhs | C | C | C | C | C | C | C | C | C | C | rhs |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Min: | | 5 | 6 | 5 | 6 | 1 | 5 | 0 | 0 | 0 | 0 | |
| 1: | 6 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 6 |
| 2: | 6 | -1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 6 |
| 3: | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 4: | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 5: | -8 | 0 | 0 | 0 | -1 | -1 | -1 | 0 | 0 | 0 | 0 | -8 |
| 6: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | -1 | 0 | 0 | 0 | 0 |
| 7: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | -1 | 0 | 0 | 0 |
| lb: | | 0 | 0 | 0 | 0 | 0 | 1 | 3 | 0 | 0 | 0 | |
| ub: | | Inf | Inf | 3 | 9 | 3 | Inf | Inf | 4 | Inf | Inf | |

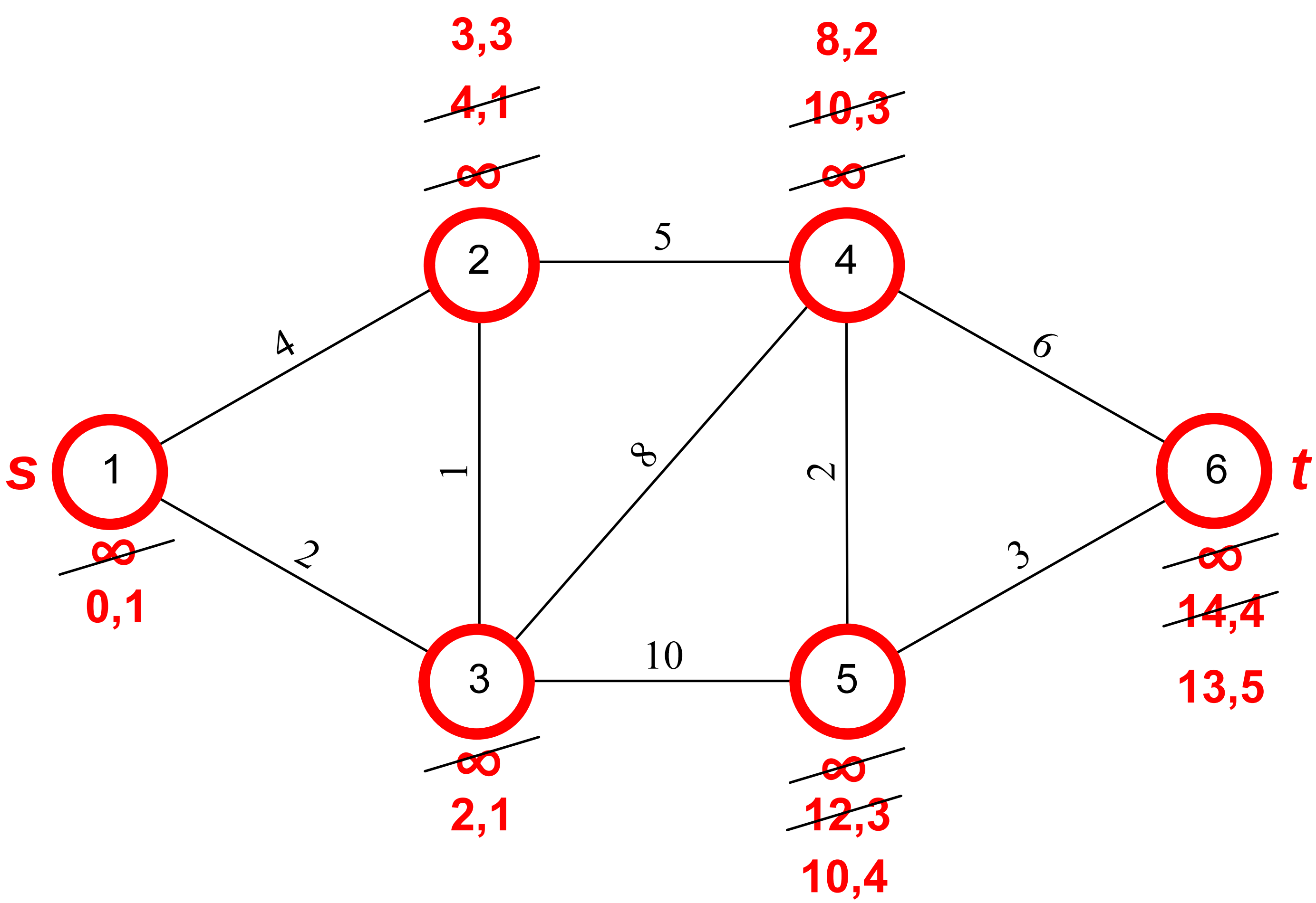


# Solving an MCNF as an LP

- Special procedures more efficient than LP were developed to solve MCNF and Transportation problems
  - e.g., Network simplex algorithm (MCNF)
  - e.g., Hungarian method (Transportation and *Transshipment*)
- Now usually easier to transform into LP since solvers are so good, with MCNF just aiding in formulation of problem:
  - Trans $\Rightarrow$ MCNF $\Rightarrow$ LP
  - Special, very efficient procedures only used for shortest path problem (Dijkstra)



# Dijkstra Shortest Path Procedure




Path: $1 \leftarrow 3 \leftarrow 2 \leftarrow 4 \leftarrow 5 \leftarrow 6:\quad 13$

# Dijkstra Shortest Path Procedure

**procedure** $dijkstra(\mathbf{W}, n, s)$
$S \leftarrow \{\ \}, \quad \bar{S} \leftarrow \{1, \ldots, n\}$
**for** $i \in \bar{S}$, $d(i) \leftarrow \infty$, **endfor**
$d(s) \leftarrow 0, \quad pred(s) \leftarrow 0$
**while** $|S| < n$
$\quad i \leftarrow \arg\min_j \{d(j) : j \in \bar{S}\}$
$\quad S \leftarrow S \cup i, \quad \bar{S} \leftarrow \bar{S} \setminus i$
$\quad$ **for** $j \in \arg\{W_{i(j)} : W_{ij} \neq 0\}$
$\quad\quad$ **if** $d(j) > d(i) + W_{ij}$
$\quad\quad\quad d(j) \leftarrow d(i) + W_{ij}$
$\quad\quad\quad pred(j) \leftarrow i$
$\quad\quad$ **endif**
$\quad$ **endfor**
**endwhile**
**return d, pred**

```
%% DIJKSTRA Matlab code,
%  given W, n, and s
S = []; nS = 1:n;
d = inf(1,n);
d(s) = 0; pred(s) = 0;
while length(S) < n
    [di,idx] = min(d(nS));
    i = nS(idx);
    S = [S i];
    nS(idx) = [];
    pred(d > di + W(i,:)) = i;
    d = min(d,di + W(i,:));
end
d, pred
```

Index to index vector nS

Order important

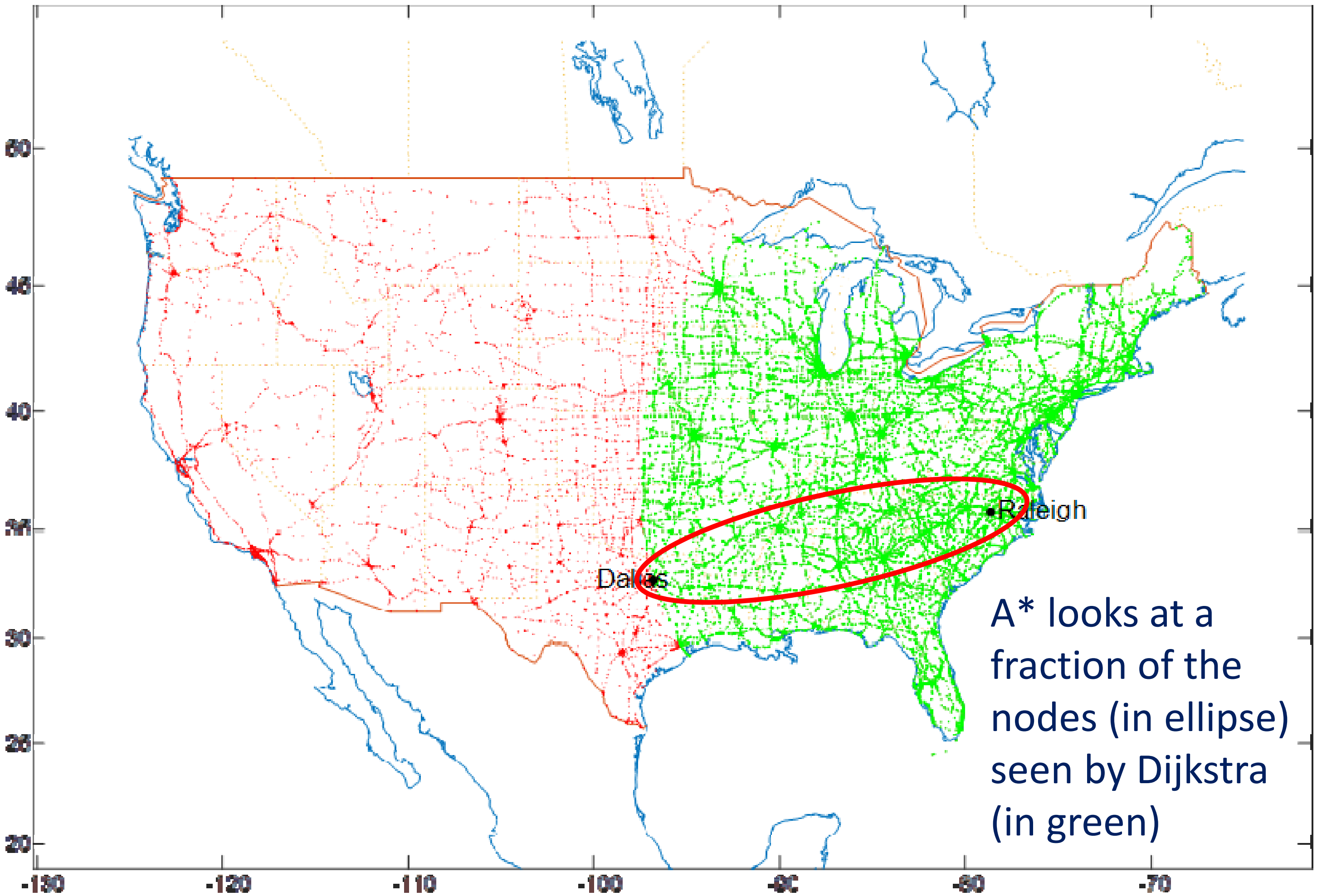
| | |
|---|---|
| $O(2^n)$ | Simplex (LP) |
| $O(n^4)$ | Ellipsoid (LP) |
| $O(n^3)$ | Hungarian (transportation) |
| $O(n^2)$ | Dijkstra (linear min) |
| $\boxed{O(m \log n)}$ | Dijkstra (Fibonocci heap) |
| $m$ | no. arcs |

# Other Shortest Path Procedures

- Dijkstra requires that all arcs have nonnegative lengths
  - Is a “label setting” algorithm since step to final solution made as each node labeled
  - Can find longest path (used in CPM) by making by negating arc lengths
- Networks with some negative arcs require slower “label correcting” procedures that repeatedly check for optimality at all nodes or detect a negative cycle
  - Negative arcs used in project scheduling to represent maximum lags between activities
- A* algorithm adds to Dijkstra an heuristic estimate of each node’s remaining distance to destination
  - Used in AI search for all types of applications (tic-tac-toe, chess)
  - In path planning applications, great circle distance from each node to destination could be used as estimate of remaining distance

# A* Path Planning Example 1

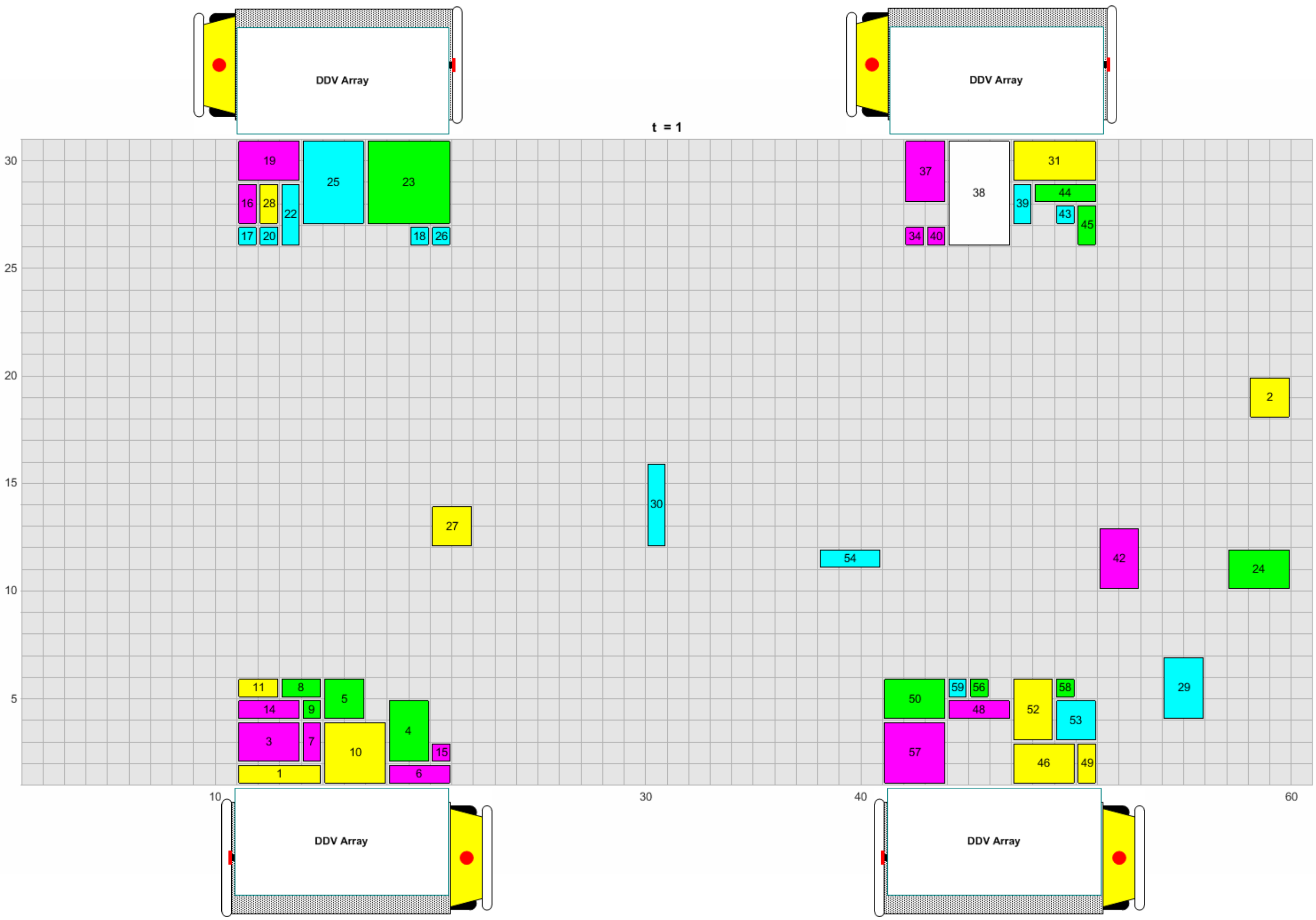
$$d_{A^*}(\text{Raleigh, Dallas}) = d_{dijk}(\text{Raleigh}, i) + d_{GC}(i, \text{Dallas}), \quad \text{for each node } i$$



# A* Path Planning Example 2

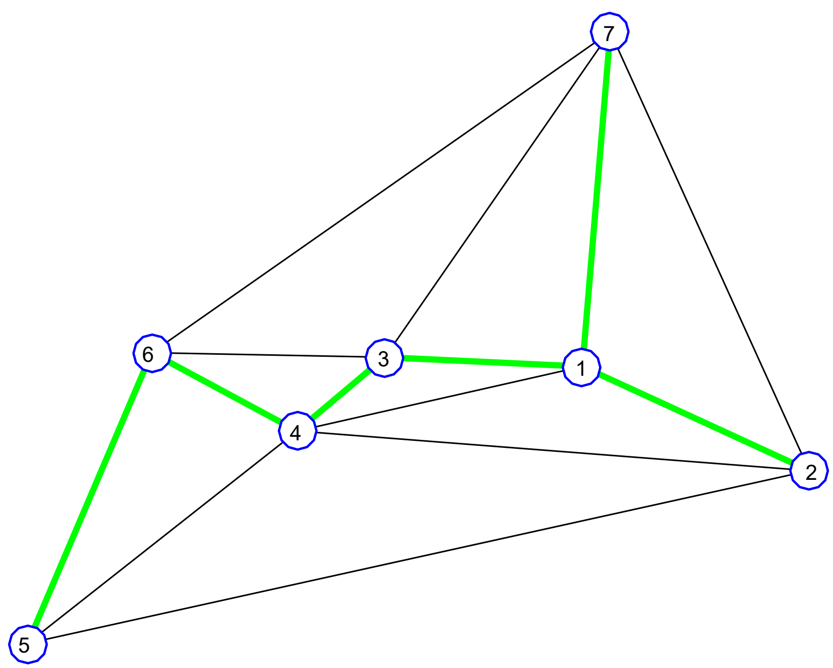
- 3-D ($x,y,t$) A* used for planning path of each container in a DC
- Each container assigned unique priority that determines planning sequence
  - Paths of higher-priority containers become obstacles for subsequent containers

# A* Path Planning Example 2



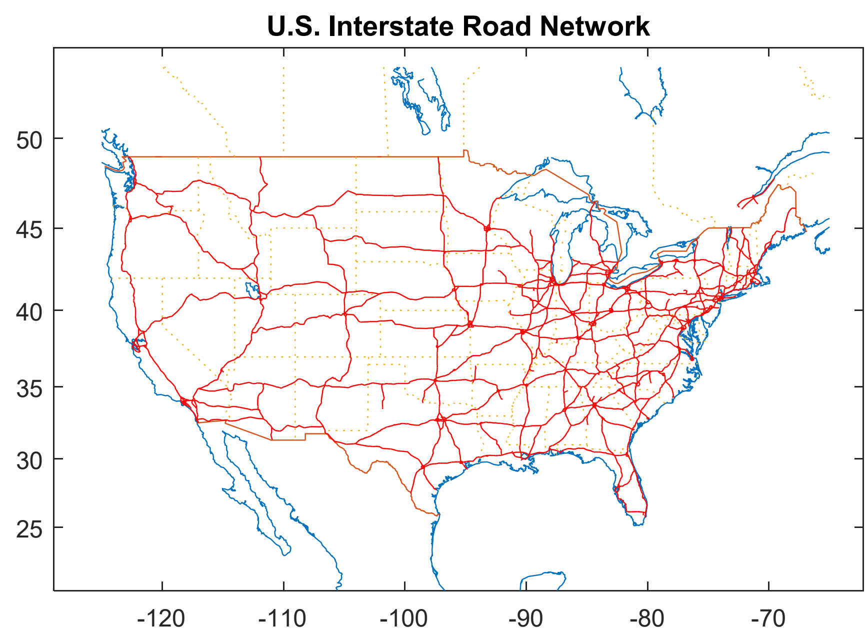
# Minimum Spanning Tree

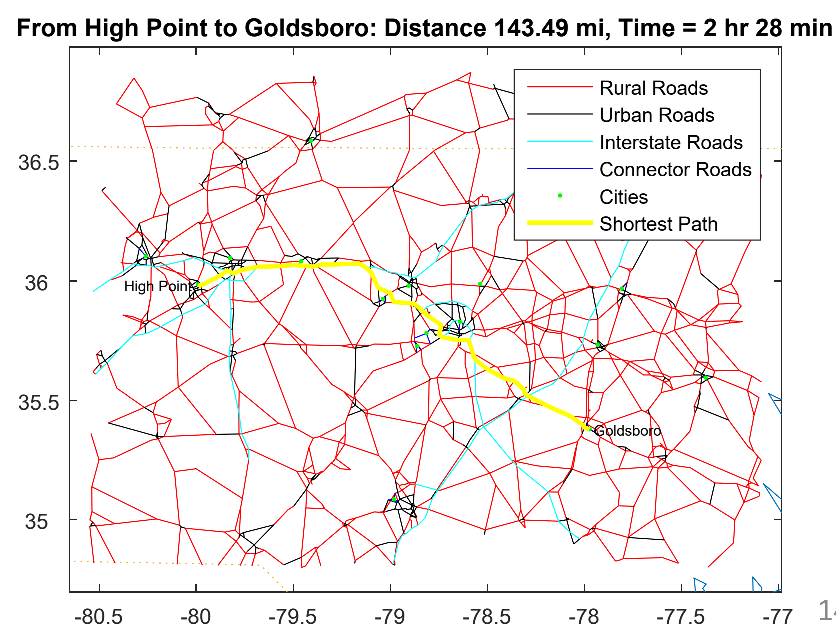
- Find the minimum cost set of arcs that connect all nodes
  - Undirected arcs: Kruskal's algorithm (easy to code)
  - Directed arcs: Edmond's branching algorithm (hard to code)



# U.S. Highway Network

- Oak Ridge National Highway Network
  - Approximately 500,000 miles of roadway in US, Canada, and Mexico
  - Created for truck routing, does not include residential
  - Nodes attributes: XY, FIPS code
  - Arc attributes: IJD, Type (Interstate, US route), Urban



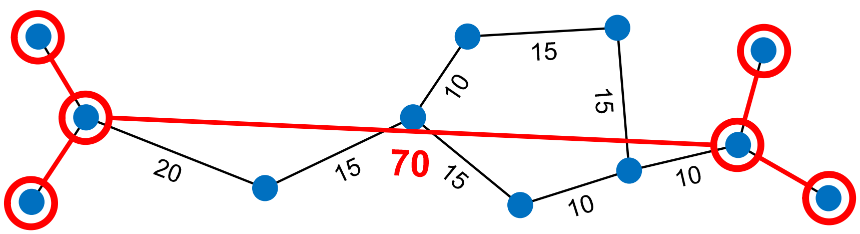
# FIPS Codes

- Federal Information Processing Standard (FIPS) codes used to uniquely identify states (2-digit) and counties (3-digit)
  - 5-digit Wake county code = 2-digit state + 3-digit county = 37183 = 37 NC FIPS + 183 Wake FIPS

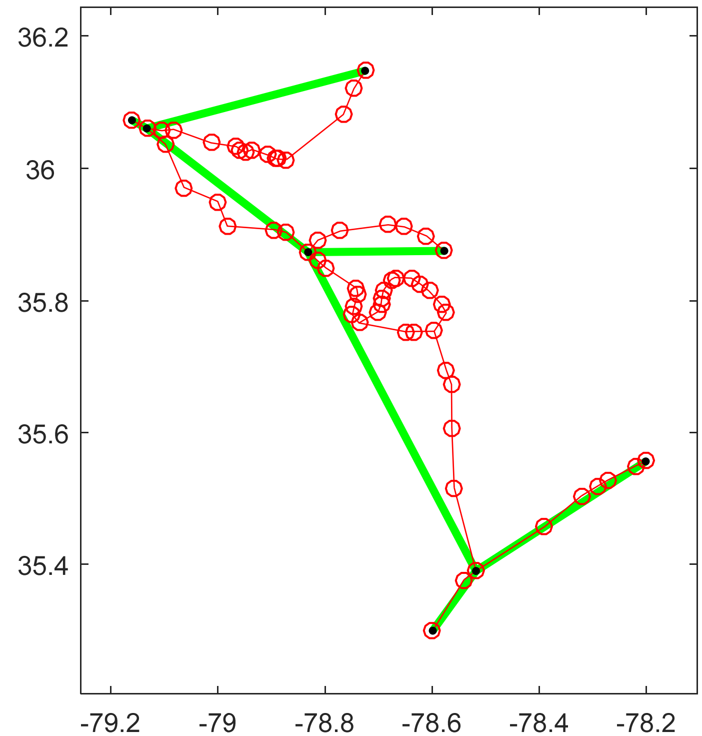
```
 1 AL Alabama         22 LA Louisiana       40 OK Oklahoma
 2 AK Alaska          23 ME Maine           41 OR Oregon
 4 AZ Arizona         24 MD Maryland        42 PA Pennsylvania
 5 AR Arkansas        25 MA Massachusetts   44 RI Rhode Island
 6 CA California      26 MI Michigan        45 SC South Carolina
 8 CO Colorado        27 MN Minnesota       46 SD South Dakota
 9 CT Connecticut     28 MS Mississippi     47 TN Tennessee
10 DE Delaware        29 MO Missouri        48 TX Texas
11 DC Dist Columbia   30 MT Montana         49 UT Utah
12 FL Florida         31 NE Nebraska        50 VT Vermont
13 GA Georgia         32 NV Nevada          51 VA Virginia
15 HI Hawaii          33 NH New Hampshire   53 WA Washington
16 ID Idaho           34 NJ New Jersey      54 WV West Virginia
17 IL Illinois        35 NM New Mexico      55 WI Wisconsin
18 IN Indiana         36 NY New York        56 WY Wyoming
19 IA Iowa            37 NC North Carolina  72 PR Puerto Rico
20 KS Kansas          38 ND North Dakota    88    Canada
21 KY Kentucky        39 OH Ohio            91    Mexico
```

# Road Network Modifications

1. **Thin**
   - Remove all degree-2 nodes from network
   - Add cost of both arcs incident to each degree-2 node
   - If results in multiple arcs between pair of nodes, keep minimum cost




**Thinned I-40 Around Raleigh**

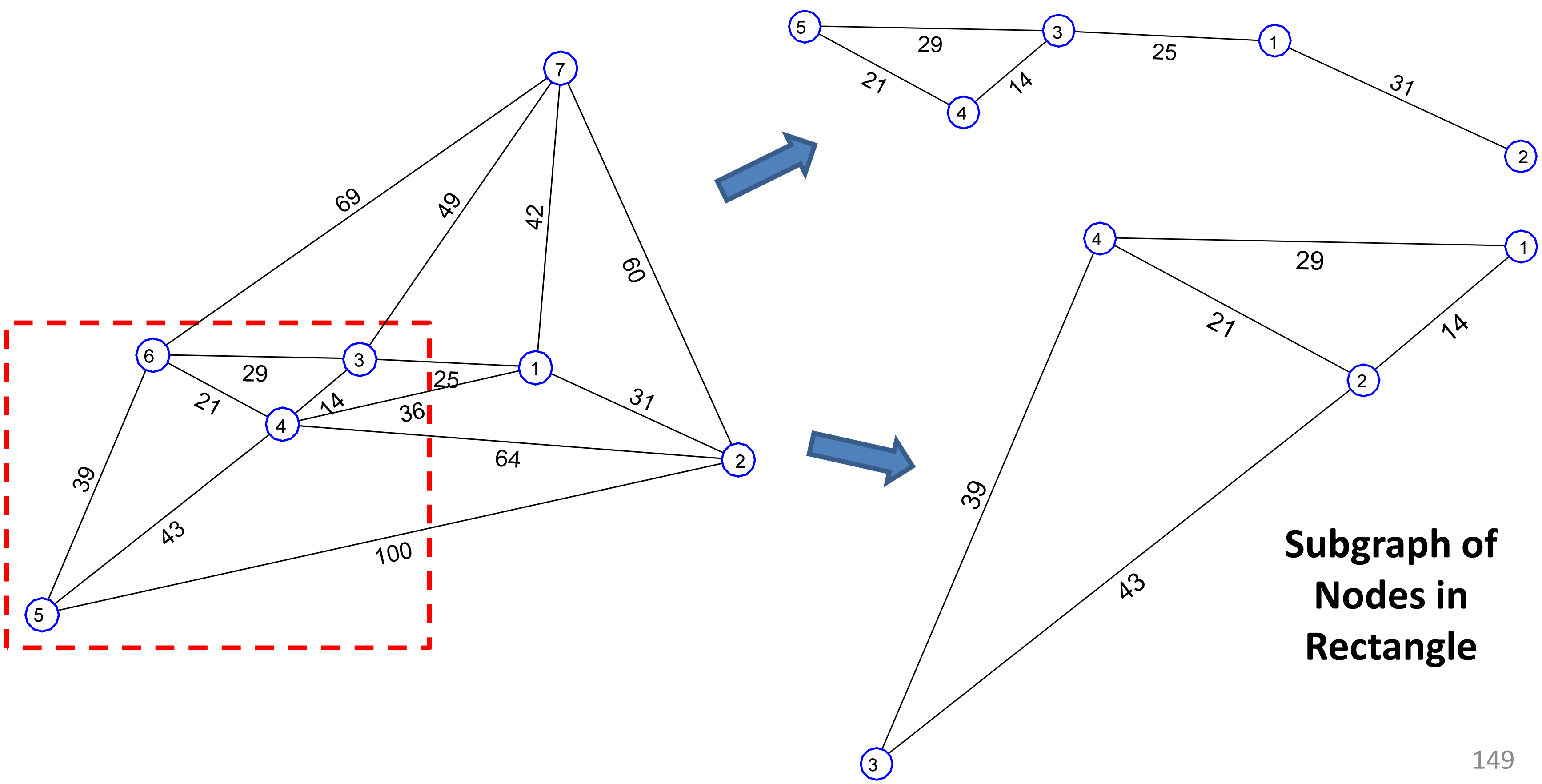


# Road Network Modifications

## 2. Subgraph

– Extract portion of graph with only those nodes and/or arcs that satisfy some condition

**Subgraph of Arcs < 35**

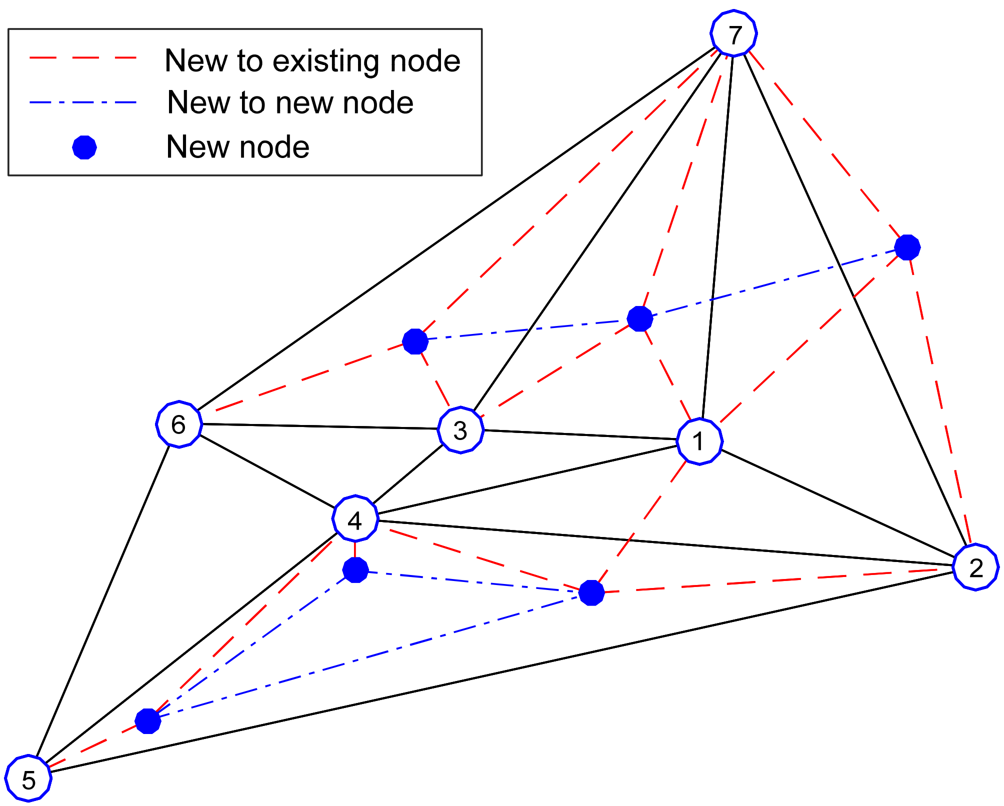



**Subgraph of Nodes in Rectangle**

# Road Network Modifications

## 3. Add connector

– Given new nodes, add arcs that connect the new nodes to the existing nodes in a graph and to each other



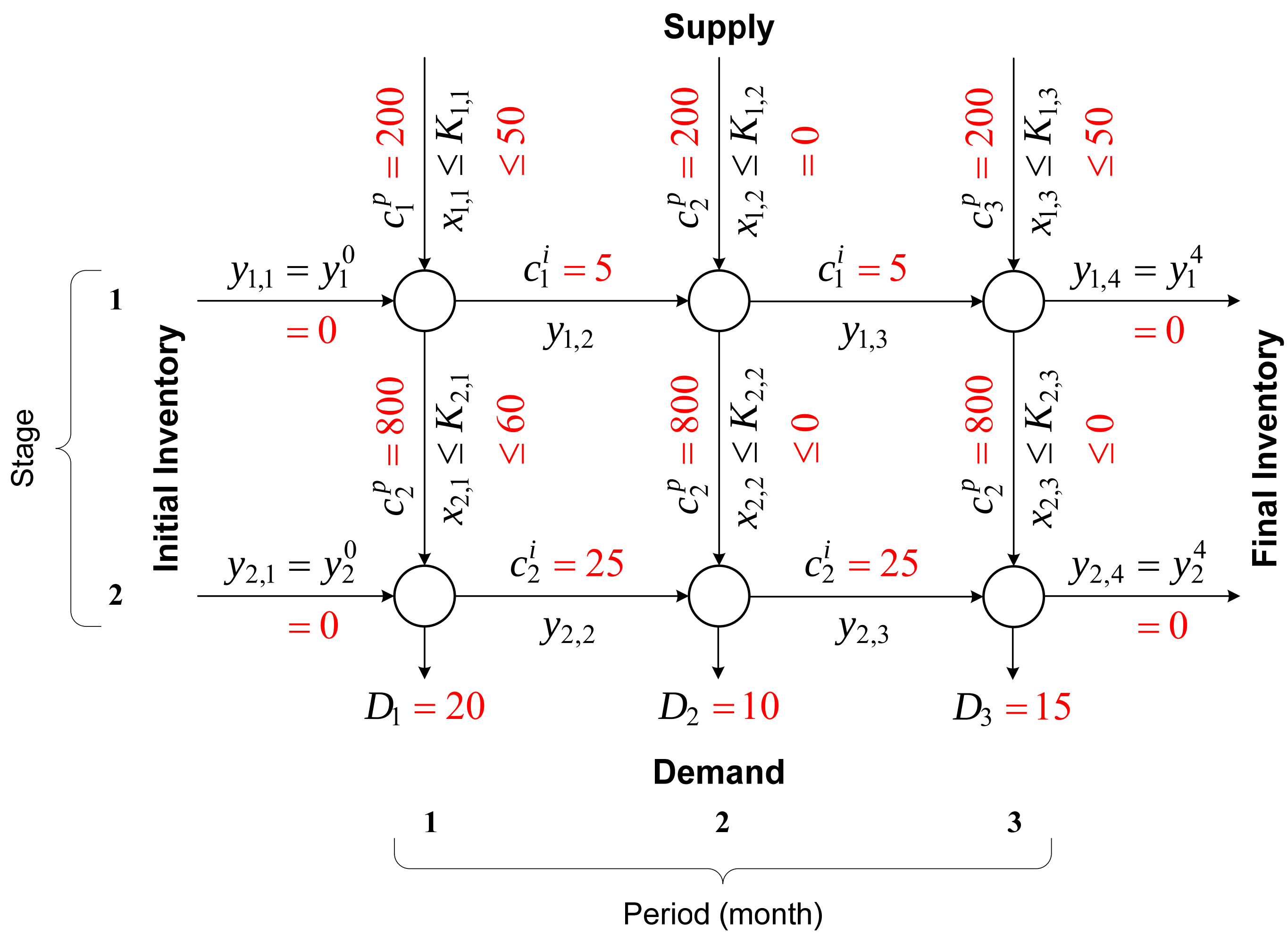

– Distance of connector arcs = GC distance x circuity factor (1.5)

– New node connected to 3 closest existing nodes, except if

  – Ratio of closest to 2nd and 3rd closest < threshold (0.1)

  – Distance shorter using other connector and graph

# Production and Inventory: One Product




$$h = 0.3 \frac{\$}{\$\text{-yr}} = \frac{0.3}{\text{yr}}$$

$$\frac{h}{T} = \frac{0.3}{12} = \frac{0.025}{\text{month}}$$

$$c_m^i = \frac{h}{T} \sum_{j=1}^{m} c_j^p$$

$$c_1^i = \frac{0.3}{12} 200 = 5$$

$$c_2^i = \frac{0.3}{12}(200 + 800)$$

$$= 25$$

# Production and Inventory: One Product

$$\min \sum_{m=1}^{M}\sum_{t=1}^{T} c_m^p\, x_{mt} + \sum_{m=1}^{M}\sum_{t=1}^{T+1} c_m^i\, y_{mt}$$

subject to

Flow balance

$$x_{mt} - x_{(m+1)t} + y_{mt} - y_{m(t+1)} = 0, \qquad m = 1,\ldots,M-1;\ t = 1,\ldots,T$$

$$x_{Mt} + y_{Mt} - y_{M(t+1)} = D_t, \qquad t = 1,\ldots,T$$

Capacity

$$x_{mt} \le K_{mt}, \qquad m = 1,\ldots,M;\ t = 1,\ldots,T$$

Initial/Final inventory

$$y_{m1} = y_m^0, \qquad m = 1,\ldots,M$$

$$y_{m(T+1)} = y_m^{T+1}, \qquad m = 1,\ldots,M$$

$$x, y \ge 0 \text{ and continuous}$$

Use var. LB & UB instead of constraints

where

$M$ = number of production stages

$c_m^p$ = production cost in stage $m$ (\$/ton)

$x_{mt}$ = production at stage $m$ in period $t$ (ton)

$c_m^i$ = inventory cost for stage $m$ (\$/ton)

$y_{mt}$ = stage-$m$ inventory period $t-1$ to $t$ (ton)

$T$ = number of periods of production

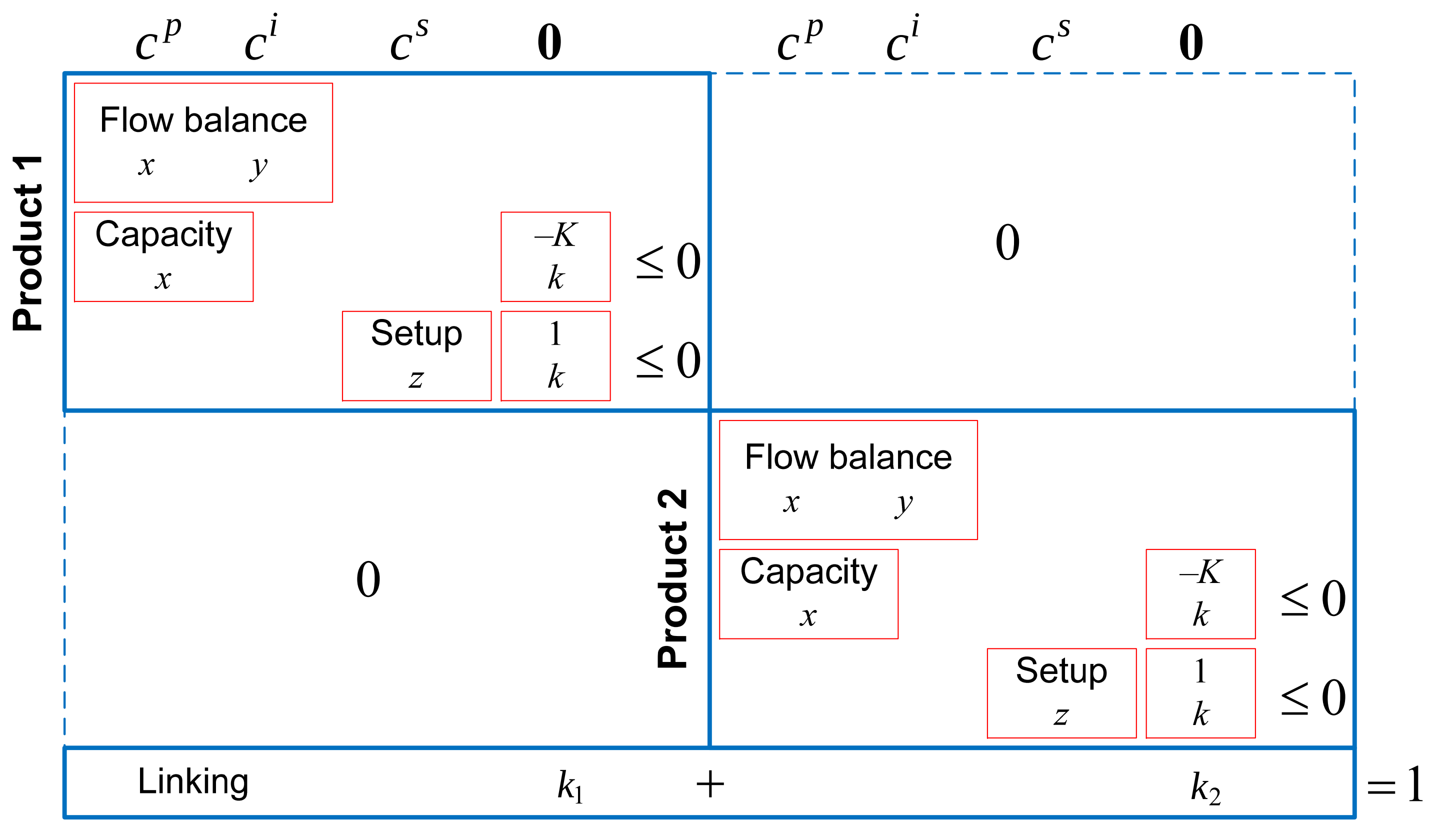
$D_t$ = demand in period $t$ (ton)

$K_{mt}$ = capacity of stage $m$ in period $t$ (ton)

$y_m^0$ = initial inventory of stage $m$ (ton)

$y_m^{T+1}$ = final inventory of stage $m$ (ton)

# Production and Inventory: Multiple Products



# Production and Inventory: Multiple Products

$k_{mtg} \in \{0,1\},\quad$ production indicator

| $k_{mtg}$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| 2 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

$z_{mtg} \in \{0,1\},\quad$ setup indicator

| $z_{mtg}$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

| | $-z_t$ | $+\ k_t$ | $-\ k_{t-1}$ | $\leq\ 0$ |
|---|---|---|---|---|
| | 0 | 0 | 0 | 0 |
| | 0 | 0 | 1 | $-1$ |
| Don't want (not feasible) | 0 | 1 | 0 | 1 |
| | 0 | 1 | 1 | 0 |
| | 1 | 0 | 0 | $-1$ |
| | 1 | 0 | 1 | $-2$ |
| Want (feasible) | 1 | 1 | 0 | 0 |
| | 1 | 1 | 1 | $-1$ |

Feasible, but not min cost

# Production and Inventory: Multiple Products

$$\min \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{g=1}^{G} c_{mg}^{p}\, x_{mtg} + \sum_{m=1}^{M}\sum_{t=1}^{T+1}\sum_{g=1}^{G} c_{mg}^{i}\, y_{mtg} + \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{g=1}^{G} c_{mg}^{s}\, z_{mtg} \left(\underbrace{+\mathbf{0}_{MTG}\, k_{mtg}}_{\text{dummy}}\right)$$

subject to

Flow balance

$$x_{mtg} - x_{(m+1)tg} + y_{mtg} - y_{m(t+1)g} = 0, \qquad m = 1,\dots,M-1; \quad t = 1,\dots,T;\ g = 1,\dots,G$$

$$x_{Mtg} + y_{Mtg} - y_{M(t+1)g} = D_{tg}, \qquad t = 1,\dots,T;\ g = 1,\dots,G$$

Capacity

$$x_{mtg} \le K_{mg} k_{mtg}, \qquad m = 1,\dots,M;\ t = 1,\dots,T;\ g = 1,\dots,G$$

Setup

$$-z_{m1g} + k_{m1g} \le k_{mg}^{0}, \qquad m = 1,\dots,M;\ g = 1,\dots,G$$

$$-z_{mtg} + k_{mtg} - k_{m(t-1)g} \le 0, \qquad m = 1,\dots,M;\ t = 2,\dots,T;\ g = 1,\dots,G$$

Linking

$$\sum_{g=1}^{G} k_{mtg} = 1, \qquad m = 1,\dots,M;\ t = 1,\dots,T$$

$$y_{m1g} = y_{mg}^{0}, \qquad m = 1,\dots,M;\ g = 1,\dots,G$$

$$y_{m(T+1)g} = y_{mg}^{T+1}, \qquad m = 1,\dots,M;\ g = 1,\dots,G$$

$$x, y \ge 0 \text{ and continuous; } k, z \text{ binary}$$

MILP

# Production and Inventory: Multiple Products

where $\mathbf{0}_{MTG}$ is a matrix of zeroes and

$M$ = number of production stages

$T$ = number of periods of production

$G$ = number of products produced

$c^p_{mg}$ = production cost of product $g$ at stage $m$ (\$/ton)

$x_{mtg}$ = production at stage $m$ in period $t$ of product $g$ (ton)

$c^i_{mg}$ = inventory cost of product $g$ for stage $m$ (\$/ton)

$y_{mtg}$ = inventory at stage $m$ between periods $t-1$ and $t$ of product $g$ (ton)

$c^s_{mg}$ = stage-$m$ product-$g$ setup cost (\$)

$z_{mtg}$ = setup indicator at stage $m$ in period $t$ for product $g$

$k_{mtg}$ = production indicator at stage $m$ in period $t$ for product $g$

$D_{tg}$ = demand for product $g$ in period $t$ (ton)

$K_{mg}$ = capacity for product $g$ in stage $m$ (ton)

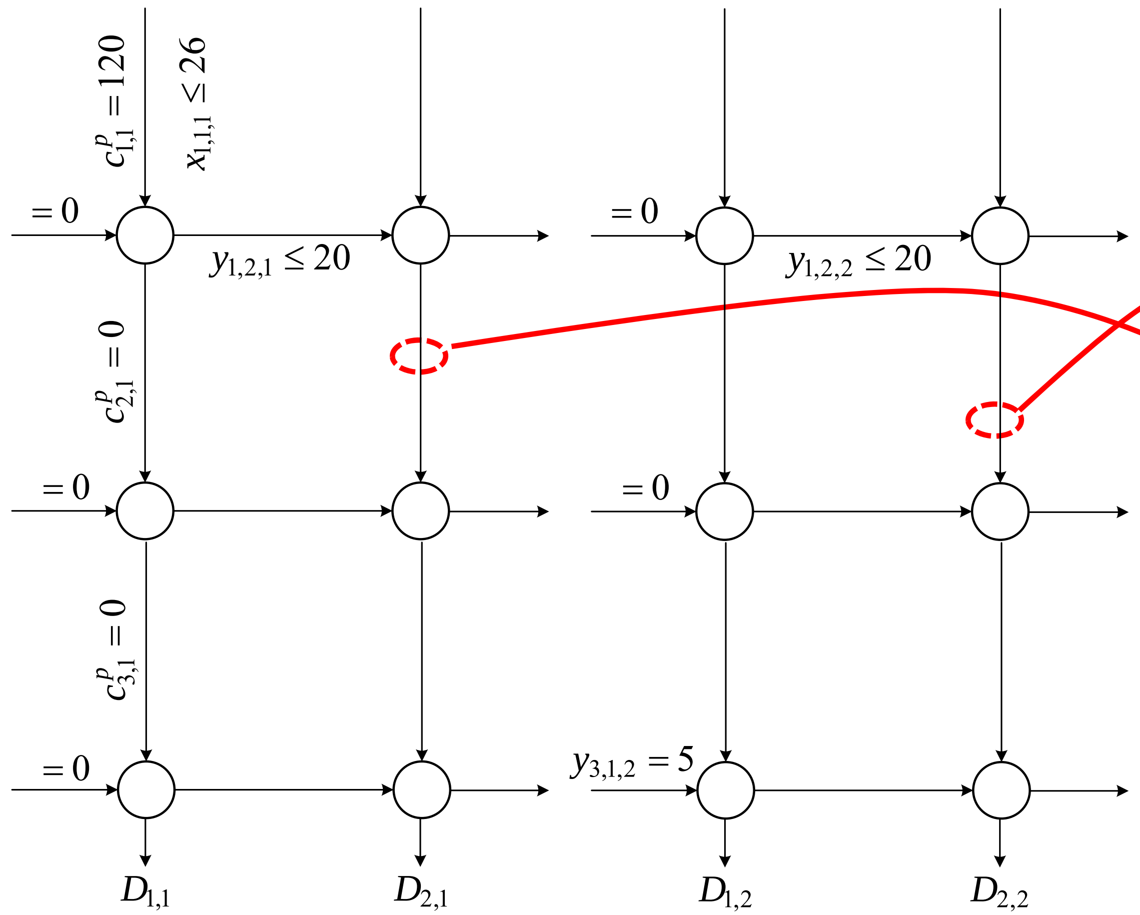
$k^0_{mg}$ = initial setup at stage $m$ for product $g$

$y^0_{mg}$ = initial product $g$ inventory at stage $m$ (ton)

$y^{T+1}_{mg}$ = final product $g$ inventory at stage $m$ (ton)

# Example: HW 6 Problem 4

- Separate networks for each raw material are coupled via sharing the same trucks (added as constraint to model)




Couples each network

$$x_{2,2,1} + x_{2,2,2} \le Q_2^{\max} z_{2,2}$$
$$\le 10 z_{2,2}$$

Number of trucks

# Example: HW 6 Problem 4

- Math programming model:

$$\min \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{g=1}^{G} c_{mg}^{p}\, x_{mtg} + \sum_{m=1}^{M}\sum_{t=1}^{T+1}\sum_{g=1}^{G} c_{mg}^{i}\, y_{mtg} + \sum_{m=1}^{M}\sum_{t=1}^{T} c_{m}^{t} z_{mt}$$

subject to

$$x_{mtg} - x_{(m+1)tg} + y_{mtg} - y_{m(t+1)g} = 0, \qquad m = 1,\ldots,M-1; \quad t = 1,\ldots,T;\ g = 1,\ldots,G$$

$$x_{Mtg} + y_{Mtg} - y_{M(t+1)g} = D_{tg}, \qquad t = 1,\ldots,T;\ g = 1,\ldots,G$$

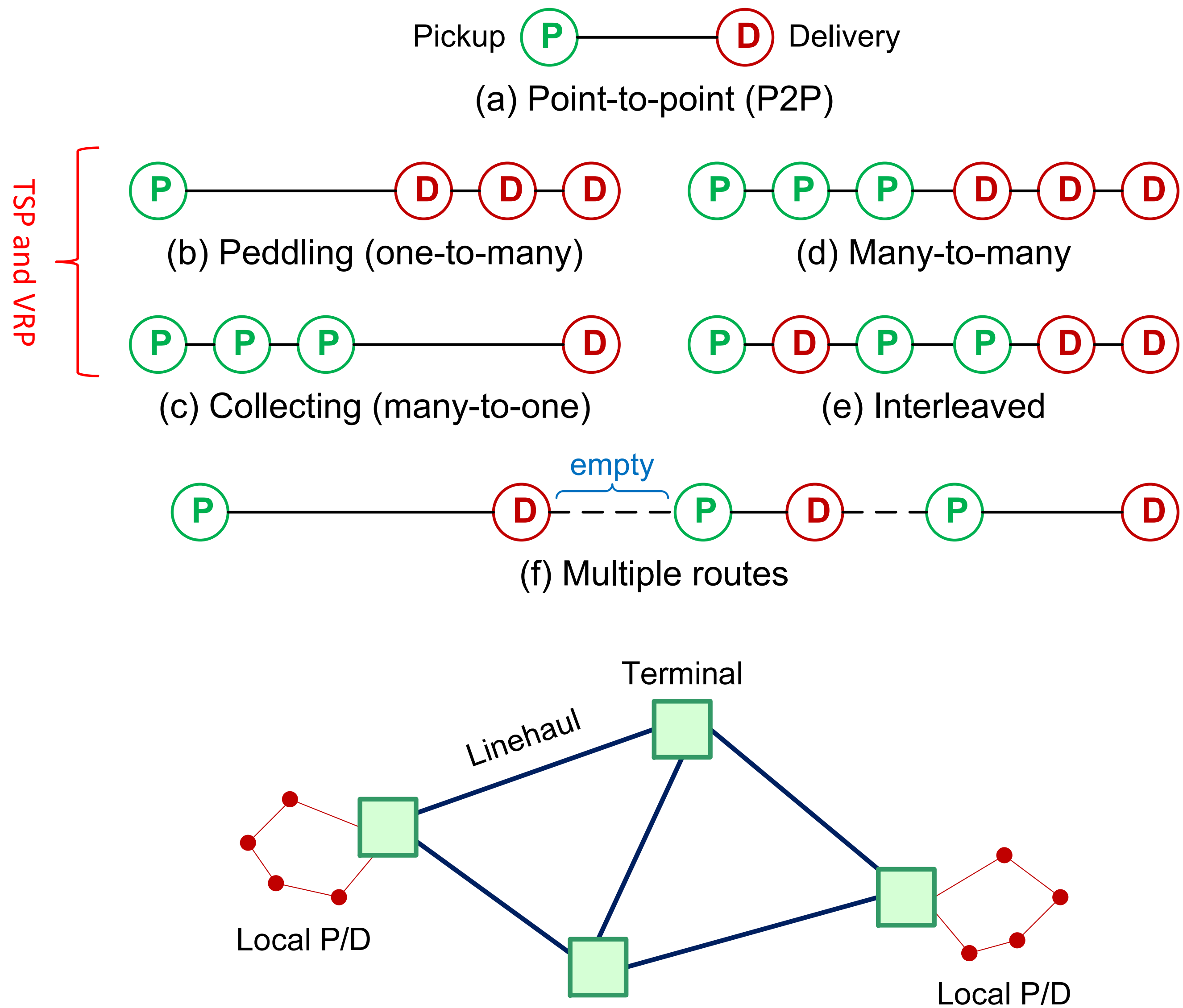
$$\sum_{g=1}^{G} x_{mtg} \le Q_{m}^{\max} z_{mt}, \quad m = 1,\ldots,M;\ t = 1,\ldots,T$$

Use LB,UB for capacity constraints

# Topics

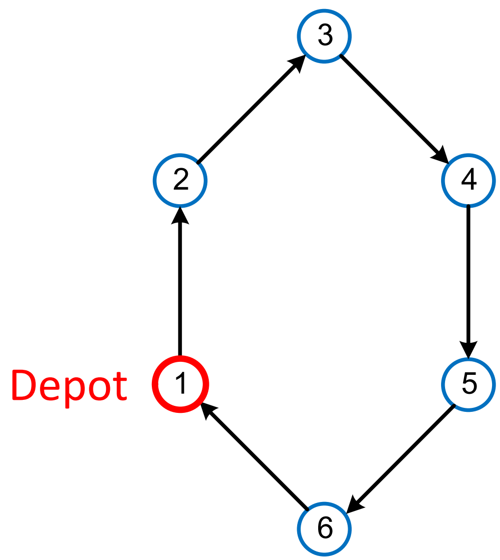
1. Introduction
2. Facility location
3. Freight transport
   – Midterm exam
4. Network models
5. **Routing**
6. Warehousing
   – Final project
   – Final exam

# Routing Alternatives



# TSP

- Problem: find connected sequence through all nodes of a graph that minimizes total arc cost
  - Subroutine in most vehicle routing problems
  - Node sequence can represent a route only if all pickups and/or deliveries occur at a single node (depot)




Node sequence = permutation + start node

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 1 |

$$n = 6 \Rightarrow (n-1)! = 120 \quad \text{possible solutions}$$

# TSP

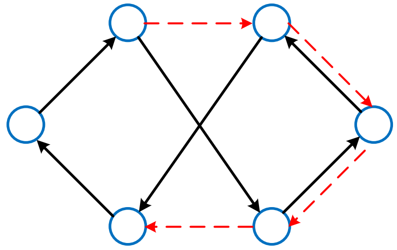
- TSP can be solved by a mix of *construction* and *improvement* procedures
  - BIP formulation has an exponential number of constraints to eliminate subtours ($\Rightarrow$ column generation techniques)
- Asymmetric: only best known solutions for large *n*

$$(n-1)! \quad n = 13 \Rightarrow \approx \frac{1}{2} \text{billion solutions}$$

- Symmetric: solved to optimal using BIP

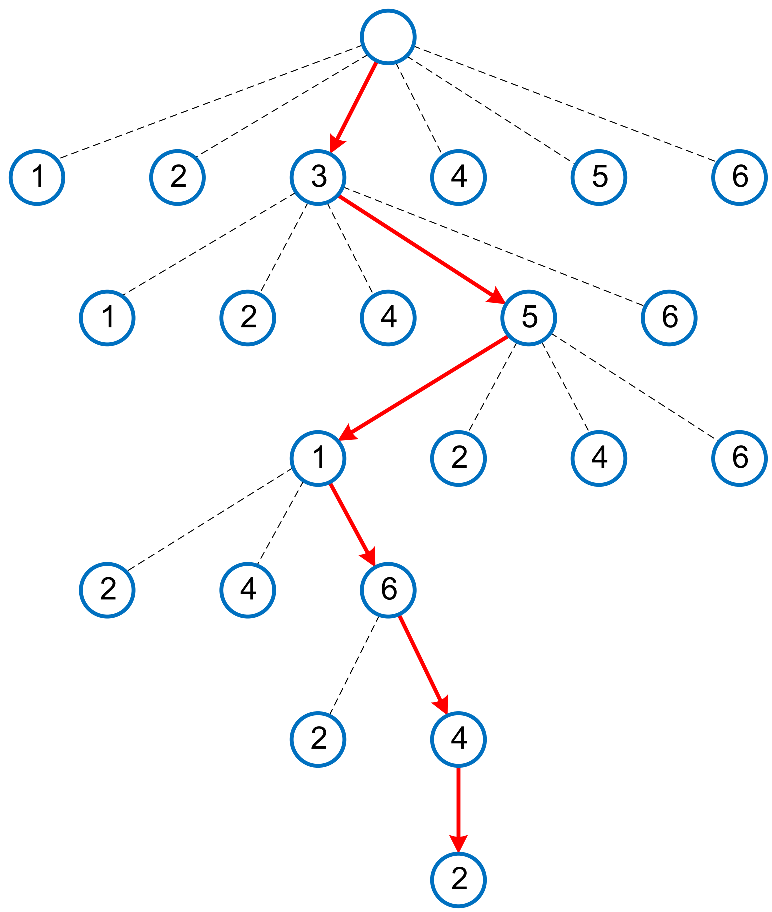
$$c_{ij} = c_{ji} \Rightarrow \frac{(n-1)!}{2} \quad \text{solutions}$$

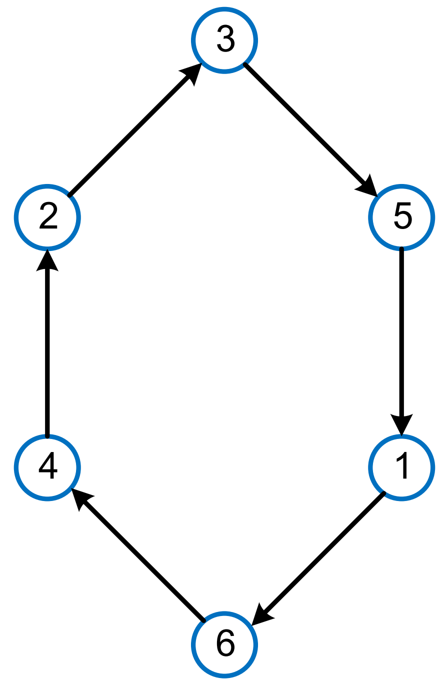
- Euclidian: arcs costs = distance between nodes



# TSP Construction

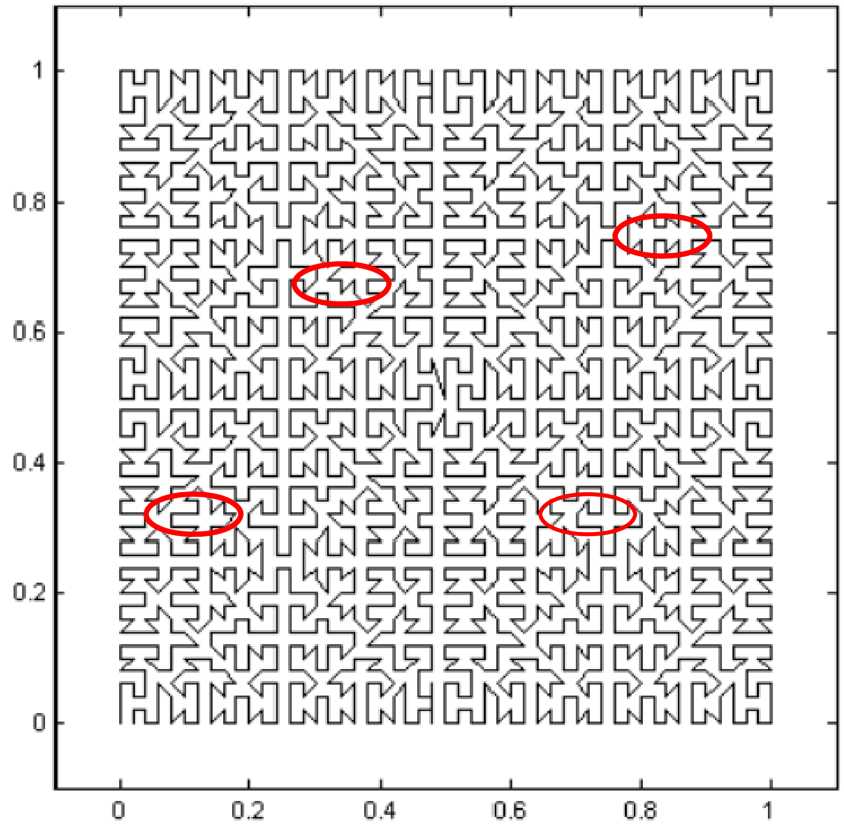
- Construction easy since any permutation is feasible and can then be improved



# Spacefilling Curve

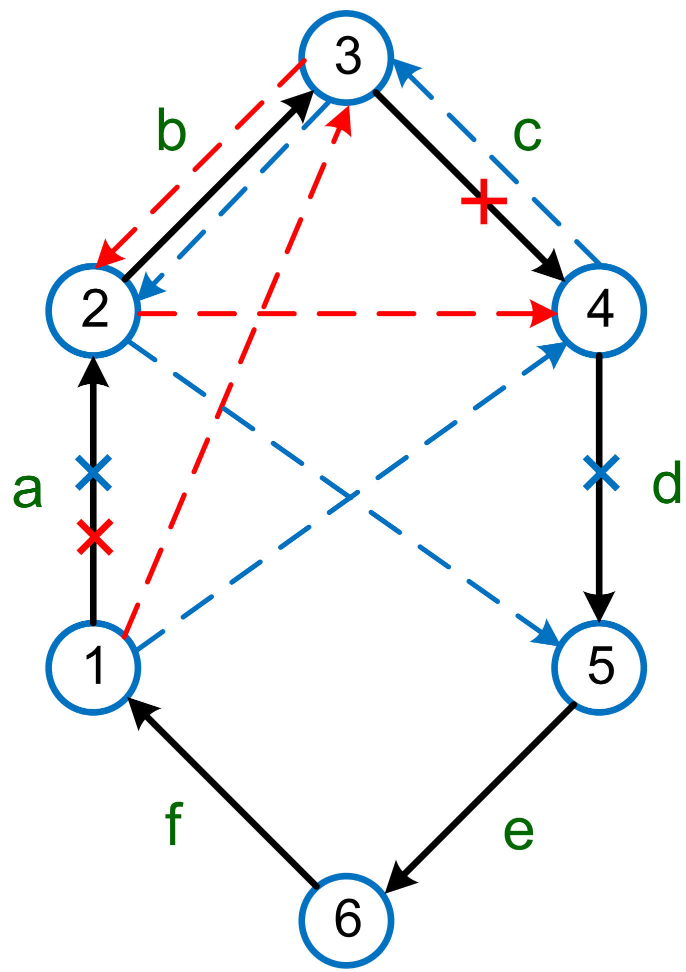
| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **1.0** | 0.250 | 0.254 | 0.265 | 0.298 | 0.309 | 0.438 | 0.441 | 0.452 | 0.485 | 0.496 | 0.500 |
| **0.9** | 0.246 | 0.257 | 0.271 | 0.292 | 0.305 | 0.434 | 0.445 | 0.458 | 0.479 | 0.493 | 0.504 |
| **0.8** | 0.235 | 0.229 | 0.279 | 0.283 | 0.333 | 0.423 | 0.417 | 0.467 | 0.471 | 0.521 | 0.515 |
| **0.7** | 0.202 | 0.208 | 0.158 | 0.154 | 0.354 | 0.390 | 0.396 | 0.596 | 0.592 | 0.542 | 0.548 |
| **0.6** | 0.191 | 0.180 | 0.167 | 0.146 | 0.132 | 0.379 | 0.618 | 0.604 | 0.583 | 0.570 | 0.559 |
| **0.5** | 0.188 | 0.184 | 0.173 | 0.140 | 0.129 | 0.375 | 0.621 | 0.610 | 0.577 | 0.566 | 0.563 |
| **0.4** | 0.059 | 0.070 | 0.083 | 0.104 | 0.118 | 0.871 | 0.632 | 0.646 | 0.667 | 0.680 | 0.691 |
| **0.3** | 0.048 | 0.042 | 0.092 | 0.096 | 0.896 | 0.860 | 0.854 | 0.654 | 0.658 | 0.708 | 0.702 |
| **0.2** | 0.015 | 0.021 | 0.971 | 0.967 | 0.917 | 0.827 | 0.833 | 0.783 | 0.779 | 0.729 | 0.735 |
| **0.1** | 0.004 | 0.993 | 0.979 | 0.958 | 0.945 | 0.816 | 0.805 | 0.792 | 0.771 | 0.757 | 0.746 |
| **0.0** | 0.000 | 0.996 | 0.985 | 0.952 | 0.941 | 0.813 | 0.809 | 0.798 | 0.765 | 0.754 | 0.750 |
| $\theta$ | **0.0** | **0.1** | **0.2** | **0.3** | **0.4** | **0.5** | **0.6** | **0.7** | **0.8** | **0.9** | **1.0** |




Sequence determined by sorting position along 1-D line covering 2-D space

2: 0.021
3: 0.154
1: 0.471
4: 0.783

# Two-Opt Improvement




| | a | b | c | d | e | f | |
|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 1 |
| **a-c** | 1 | 3 | 2 | 4 | 5 | 6 | 1 |
| **a-d** | 1 | 4 | 3 | 2 | 5 | 6 | 1 |
| **a-e** | 1 | 5 | 4 | 3 | 2 | 6 | 1 |
| **b-d** | | | | | | | |
| **b-e** | | | | | | | |
| **b-f** | | | | | | | |
| **c-e** | | | | | | | |
| **c-f** | | | | | | | |
| **d-f** | | | | | | | |

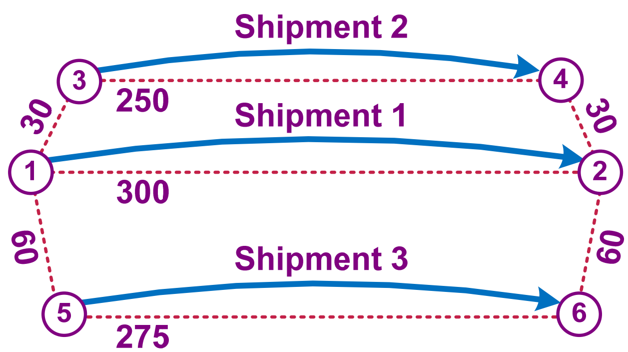
Sequences considered at end to verify local optimum: $n$ nodes $\Rightarrow \sum_{i=1}^{n-2} \sum_{j=i+2}^{n-1} (1) = \frac{n^2 - 5n}{2} + 3 = 9$ for $n = 6$

# TSP Comparison

| | TSP Procedure | Total Cost |
|---|---|---|
| 1 | Spacefilling curve | 482.7110 |
| 2 | 1 + 2-opt | 456 |
| 3 | Convex hull insert + 2-opt | 452 |
| 4 | Nearest neighbor + 2-opt | 439.6 |
| 5 | Random construction + 2-opt | 450, 456 |
| 6 | Eil51 in TSPLIP | 426* optimal |

# Multi-Stop Routing

- Each shipment might have a different origin and/or destination $\Rightarrow$ node/location sequence not adequate




$$L = (y_1, \ldots, y_n) = (1,2,3) \qquad n \text{ shipments}$$

$$R = (z_1, \ldots, z_{2n}) = (3,1,2,2,1,3) \qquad 2n\text{-element route sequence}$$

$$X = (x_1, \ldots, x_{2n}) = (5,1,3,4,2,6) \qquad 2n\text{-element location (node) sequence}$$

$$c_{ij} = \text{cost between locations } i \text{ and } j$$

$$c(R) = \sum_{i=1}^{2n-1} c_{x_i, x_{i+1}} = 60 + 30 + 250 + 30 + 60 = 430, \quad \text{total cost of route } R$$

# Min Cost Insert

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | 1 | | | 1 | | | |
| 1 | | • | | | • | 2 | 2 | × |
| 2 | 2 | • | | | • | 2 | | $c_2$ |
| 3 | | • | 2 | 2 | • | | | $c_3^*$ |
| 4 | | • | | 2 | • | 2 | | $c_4$ |
| 5 | 2 | • | 2 | | • | | | $c_5$ |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | | | 2 | | 2 | | 1 | |
| 1 | 3 | • | 3 | | • | | • | | • | |
| 2 | 3 | • | | | • | 3 | • | | • | |
| 3 | 3 | • | | | • | | • | 3 | • | |
| 4 | 3 | • | | | • | | • | | • | 3 |
| 5 | | • | 3 | 3 | • | | • | | • | |
| 6 | | • | | 3 | • | 3 | • | | | |
| 7 | | • | | 3 | • | | • | 3 | | |
| ⋮ | | ⋮ | | | ⋮ | | ⋮ | | ⋮ | |

# Route Sequencing Procedures

- Online procedure: add a shipment to an existing route as it becomes available
  - Insert and Improve: for each shipment, mincostinsert + twoopt
- Offline procedure: consider all shipments to decide order in which each added to route
  - Savings and Improve: determine insert ordering based on "savings," then improve final route

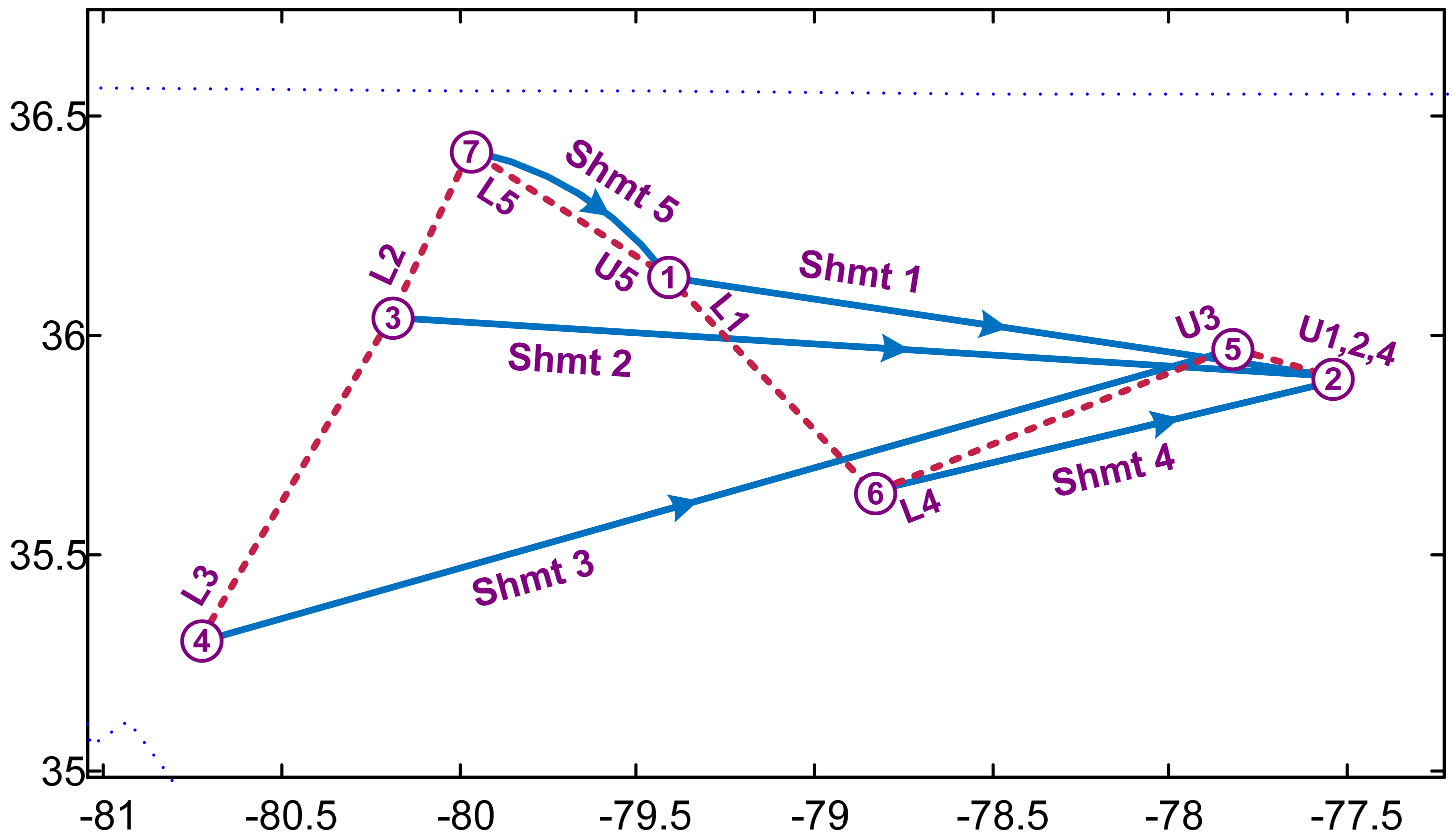
# Insert and Improve Online Procedure

```
procedure insertImprove(y_i ∈ L)
R = (y_1, y_1)
for i = 2, ..., |L|
    R = minCostInsert(y_i, R)
    R = twoOpt(R)
endfor
return R
```

```
subprocedure minCostInsert(y, z_i ∈ R)
c_R = c(R)
for i = 1, ..., |R|+1, for j = 1, ..., |R|+1
    R' = (z_1, ..., z_{i-1}, y, z_i, ..., z_{j-1}, y, z_j, ..., z_{|R|})
    if c(R') < c_R, c_R = c(R'), R = R', endif
endfor, endfor
return R
```

```
subprocedure twoOpt(z_i ∈ R)
c_R = c(R)
repeat
    done = true, i = 1, j = 2
    while done and i < |R|
        while done and j < |R|+1
            R' = (z_1, ..., z_{i-1}, reverseSequence(z_i, ..., z_j), z_{j+1}, ..., z_{|R|})
            if c(R') < c_R
                c_R = c(R'), R = R', done = false
            endif
            j = j+1
        endwhile
        i = i+1, j = i+1
    endwhile
until done = true
return R
```

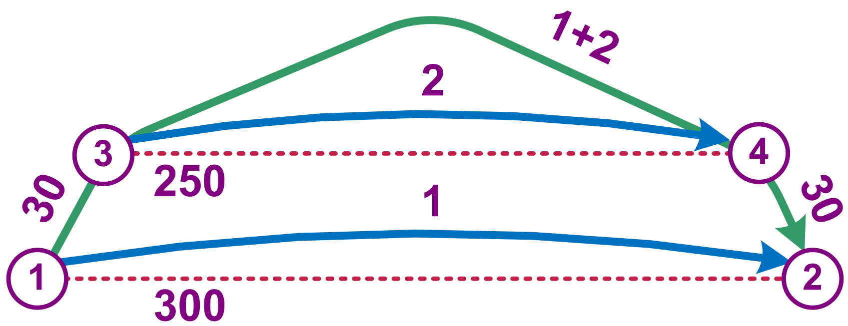
# 5-Shipment Example




$$R = (3,2,5,5,1,4,3,1,2,4)$$

$$X = (4,3,7,1,1,6,5,2,2,2)$$

# Pairwise Savings




$$s_{ij} = \text{pairwise savings between shipments } i \text{ and } j$$

$$= c_i + c_j - c_{ij}$$

$$s_{1,2} = 300 + 250 - 310$$

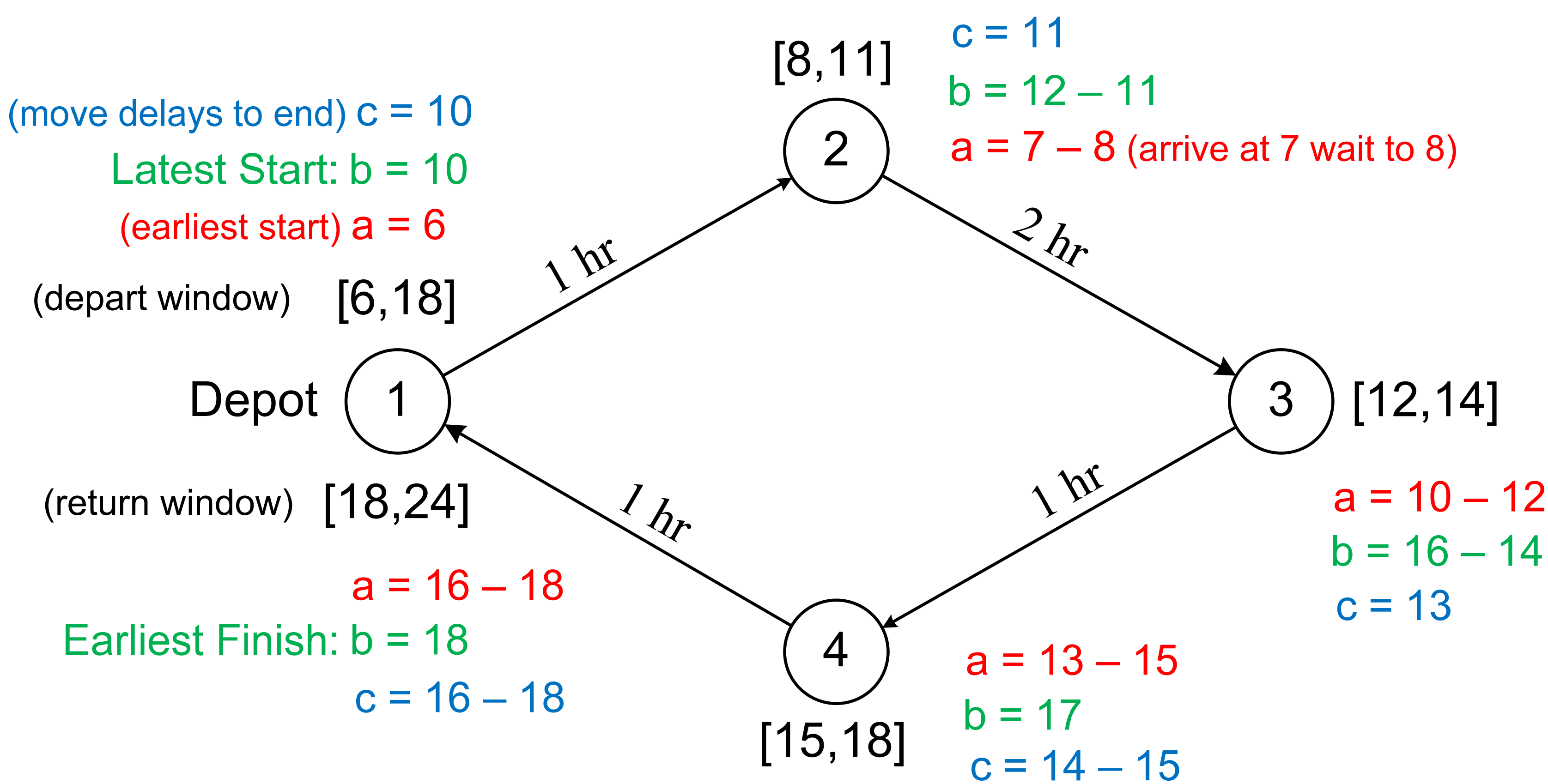
$$= 240$$

# Vehicle Routing Problem

- VRP = TSP + vehicle constraints
- Constraints:
  - Capacity (weight, cube, etc.)
  - Maximum TC (HOS: 11 hr max)
  - Time windows (with/without delay at customer)
    - VRP uses absolute windows that can be checked by simple scanning
    - Project scheduling uses relative windows solved by shortest path with negative arcs
  - Maximum number of routes/vehicles (hard)
- Criteria:
  1. Number of routes/vehicles
  2. TC

# Time Window Example

[0,24] hr; Loading/unloading time = 0; Capacity = ∞; LB = 5 hr




Earliest Finish – Latest Start = 18 – 10 = 8 hr = 5 travel + 3 delay

# Savings Route Construction Procedure

- Pairs of shipments are ordered in terms of their decreasing pairwise savings to create $\mathbf{i}$ and $\mathbf{j}$
- Creates set of multi-shipment routes

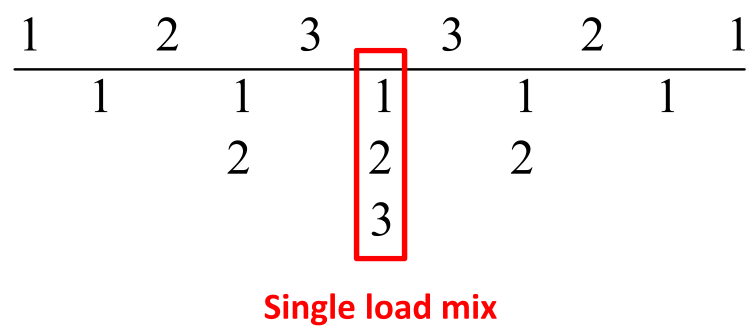
$$R = \left\{ R_1, \ldots, R_m \right\}$$

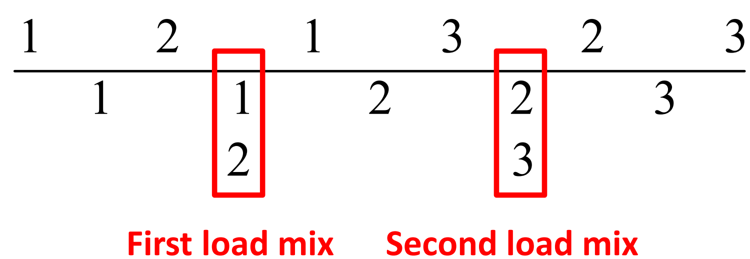
  – Shipments with no pairwise savings are not included (use `sh2rte` to add)
- Given savings pair *i,j*, the original Clark-Wright savings procedure (`vrpsavings`):
  – adds *i* to route only if *j* at beginning or end of route
  – routes combined only if *i* and *j* are endpoints of each route

**procedure** $savings(\mathbf{i},\mathbf{j})$
$R \leftarrow \{\,\}$
**for** $k = \{1,\ldots,|\mathbf{i}|\}$
  **if** $i_k \notin R$ and $j_k \notin R$  **1. Form new route**
    $R \leftarrow R \cup minCostInsert(i_k, j_k)$
  **elseif** $(i_k \notin R \text{ and } j_k \in R) \text{ or } (i_k \in R \text{ and } j_k \notin R)$
    **if** $j_k \notin R$  **2. Add shipment to route**
      $temp \leftarrow i_k,\quad i_k \leftarrow j_k,\quad j_k \leftarrow temp$
    **endif**
    $h \leftarrow \arg\{R_l : j_k \in R_l\}$
    $R' \leftarrow minCostInsert(i_k, R_h)$
    **if** $c(R') < c(i_k) + c(R_h)$
      $R_h \leftarrow R'$
    **endif**
  **else**  **3. Combine two routes**
    $g \leftarrow \arg\{R_l : i_k \in R_l\},\quad h \leftarrow \arg\{R_l : j_k \in R_l\}$
    **if** $g \neq h$
      $R' \leftarrow minCostInsert(R_g, R_h)$
      **if** $c(R') < c(R_g) + c(R_h)$
        $R_g \leftarrow \{\,\},\quad R_h \leftarrow R'$
      **endif**
    **endif**
  **endif**
**endfor**
**return** $R$

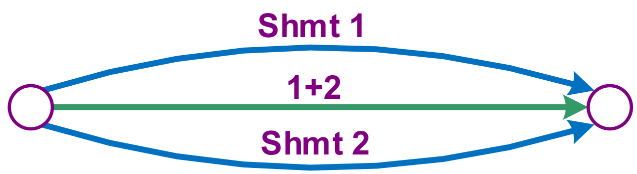
# Min TLC Multi-Stop Routing

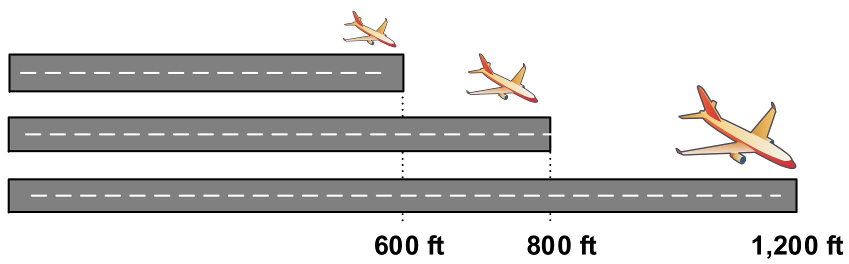
- Periodic consolidated shipments that have the same frequency/interval
  - Similar to a "milk run"
- Min TLC of aggregate shipment may not be feasible
  - Different combinations of shipments (*load mix*) may be on board during each segment of route
  - Each load mix may have different max payload
  - If needed, reduce all shipment sizes in proportion to load mix with the minimum max payload



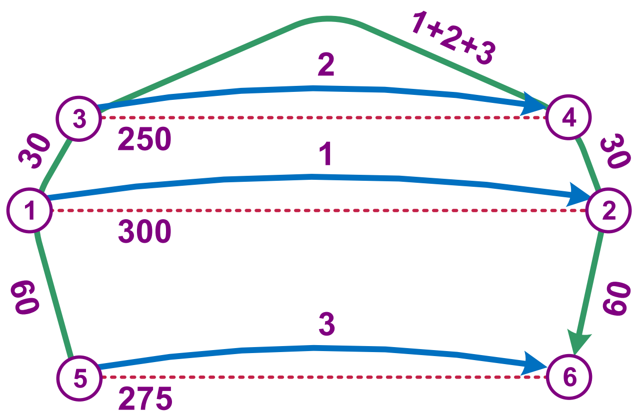



# Cost Allocation for Routing



# Cost Allocation for Routing



# Shapley Value

$$c_L^{\text{sav}} = \sum_{i=1}^{n} c_i^0 - c_L$$

$$c_{ij}^{\text{sav}} = c_i^0 + c_j^0 - c_{(i,j)}$$

$$\alpha_i = \sum_{0 \le m \le n-1} \frac{m!(n-m-1)!}{n!} \sum_{\substack{M \subset N \setminus i \\ |M|=m}} \left( \sigma_{M \cup \{i\}} - \sigma_M \right)$$

$$c_i^{\text{sav}} = \frac{c_L^{\text{sav}}}{n} + \frac{1}{n-1} \sum_{j=1}^{n} \frac{c_{ij}^{\text{sav}} + c_{ji}^{\text{sav}}}{2} - \frac{1}{n(n-1)} \sum_{j=1}^{n} \sum_{k=1}^{n} c_{jk}^{\text{sav}}$$

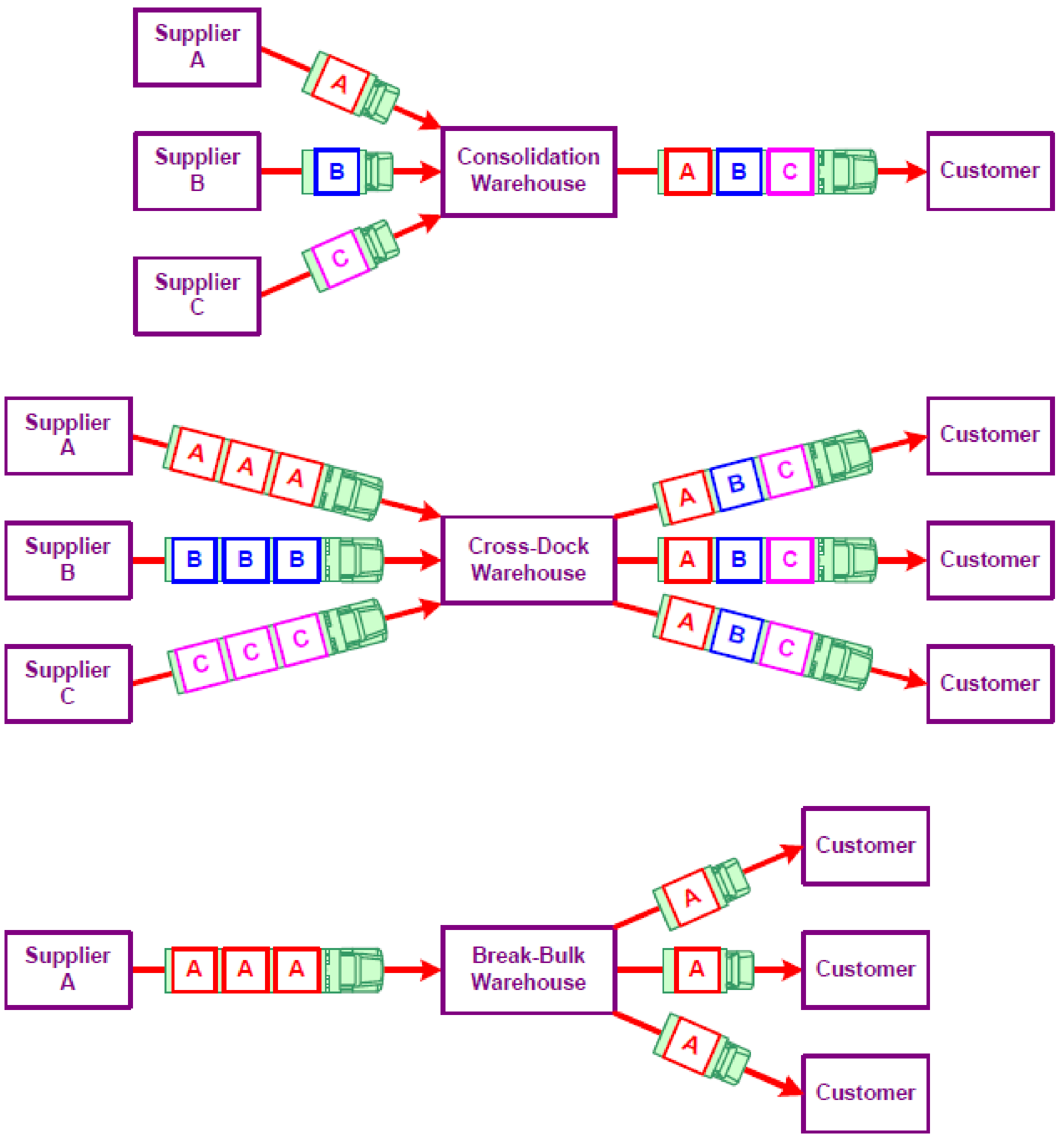
# Topics

1. Introduction
2. Facility location
3. Freight transport
   – Midterm exam
4. Network models
5. Routing
6. **Warehousing**
   – Final project
   – Final exam

# Warehousing

- *Warehousing* are the activities involved in the design and operation of warehouses
- A *warehouse* is the point in the supply chain where raw materials, work-in-process (WIP), or finished goods are stored for varying lengths of time.
- Warehouses can be used to add value to a supply chain in two basic ways:
  1. Storage. Allows product to be available where and when its needed.
  2. Transport Economies. Allows product to be collected, sorted, and distributed efficiently.
- A *public warehouse* is a business that rents storage space to other firms on a month-to-month basis. They are often used by firms to supplement their own *private warehouses*.

# Types of Warehouses



# Warehouse Design Process

- The objectives for warehouse design can include:
  - maximizing cube utilization
  - minimizing total storage costs (including building, equipment, and labor costs)
  - achieving the required storage throughput
  - enabling efficient order picking
- In planning a storage layout: either a storage layout is required to fit into an existing facility, or the facility will be designed to accommodate the storage layout.

# Warehouse Design Elements

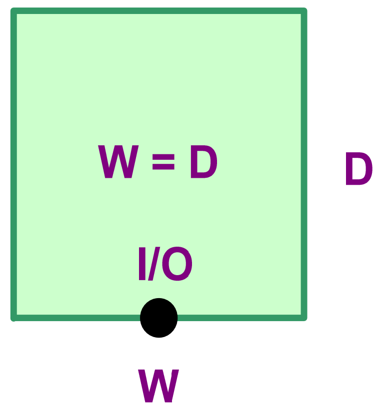
- The design of a new warehouse includes the following elements:
  1. Determining the layout of the storage locations (i.e., the warehouse layout).
  2. Determining the number and location of the input/output (I/O) ports (e.g., the shipping/receiving docks).
  3. Assigning items (stock-keeping units or *SKUs*) to storage locations (*slots*).
- A typical objective in warehouse design is to minimize the overall storage cost while providing the required levels of service.

# Design Trade-Off

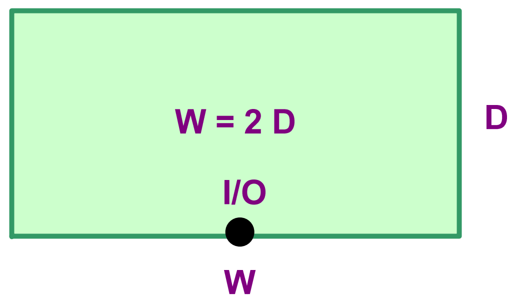
- Warehouse design involves the trade-off between building and handling costs:

| | | |
|---|---|---|
| min Building Costs | vs. | min Handling Costs |
| ⇕ | | ⇕ |
| max Cube Utilization | vs. | max Material Accessibility |

# Shape Trade-Off



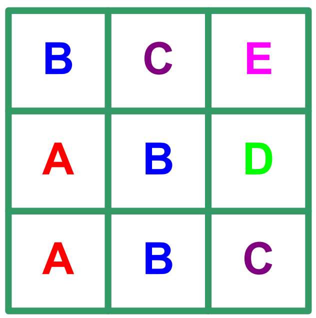

vs.



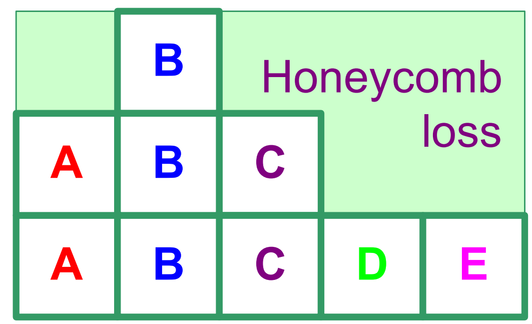

Square shape minimizes perimeter length for a given area, thus minimizing building costs

Aspect ratio of 2 (W = 2D) min. expected distance from I/O port to slots, thus minimizing handling costs

# Storage Trade-Off




vs.



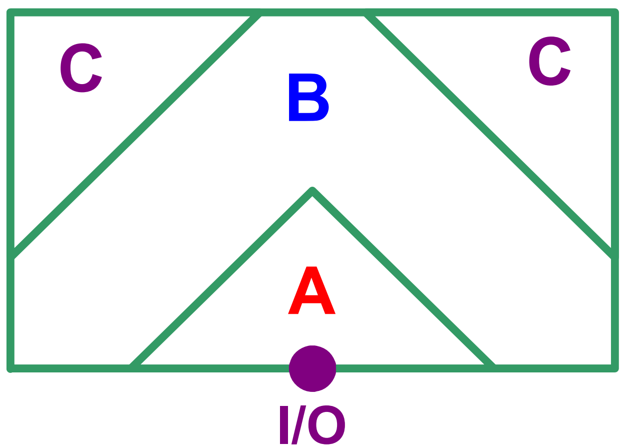

Maximizes cube utilization, but minimizes material accessibility

Making at least one unit of each item accessible decreases cube utilization

# Storage Policies

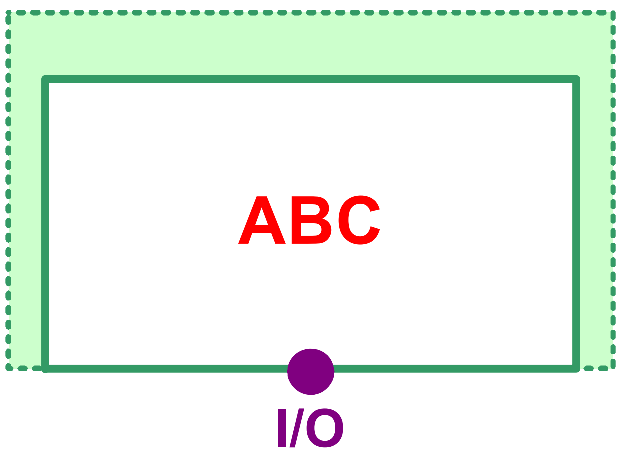
- A storage policy determines how the slots in a storage region are assigned to the different SKUs to the stored in the region.
- The differences between storage polices illustrate the trade-off between minimizing building cost and minimizing handling cost.
- Type of policies:
  - Dedicated
  - Randomized
  - Class-based

# Dedicated Storage



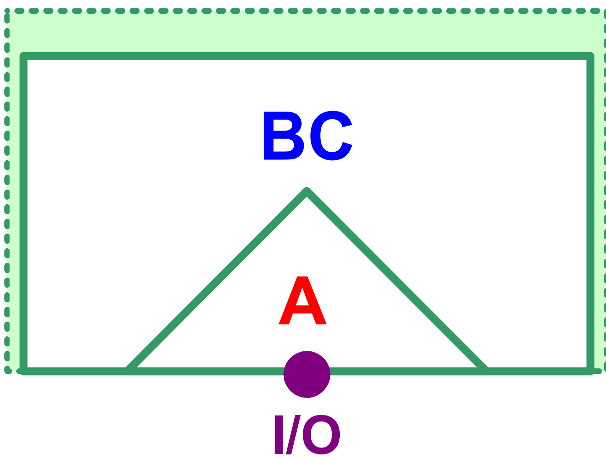

- Each SKU has a predetermined number of slots assigned to it.
- Total capacity of the slots assigned to each SKU must equal the storage space corresponding to the maximum inventory level of each *individual* SKU.
- Minimizes handling cost.
- Maximizes building cost.

# Randomized Storage



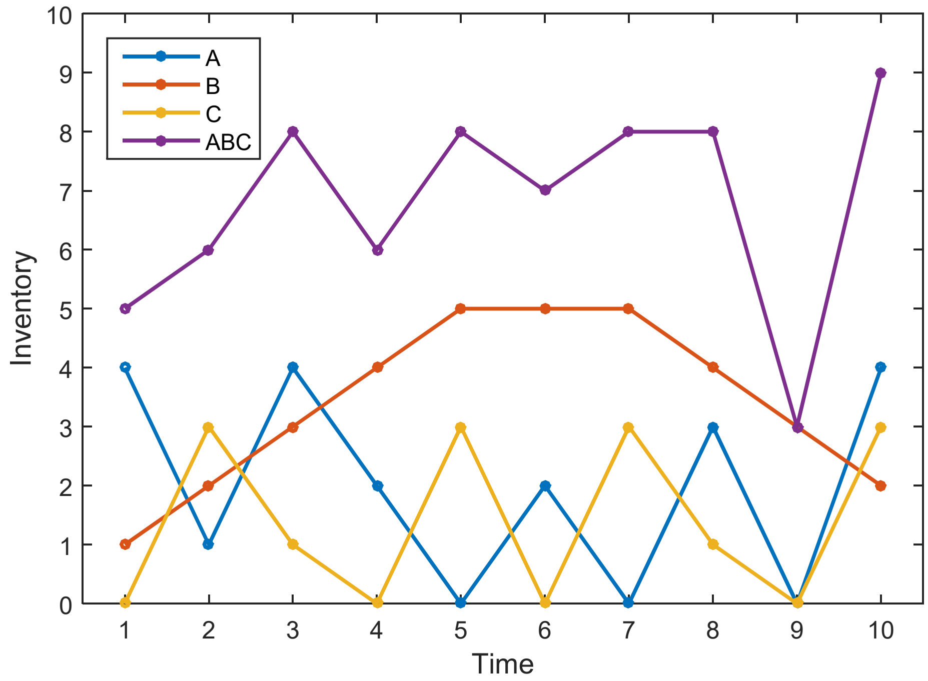

- Each SKU can be stored in any available slot.
- Total capacity of all the slots must equal the storage space corresponding to the maximum *aggregate* inventory level of all of the SKUs.
- Maximizes handling cost.
- Minimizes building cost.

# Class-based Storage




- Combination of dedicated and randomized storage, where each SKU is assigned to one of several different storage classes.
- Randomized storage is used for each SKU within a class, and dedicated storage is used between classes.
- Building and handling costs between dedicated and randomized.

# Individual vs Aggregate SKUs



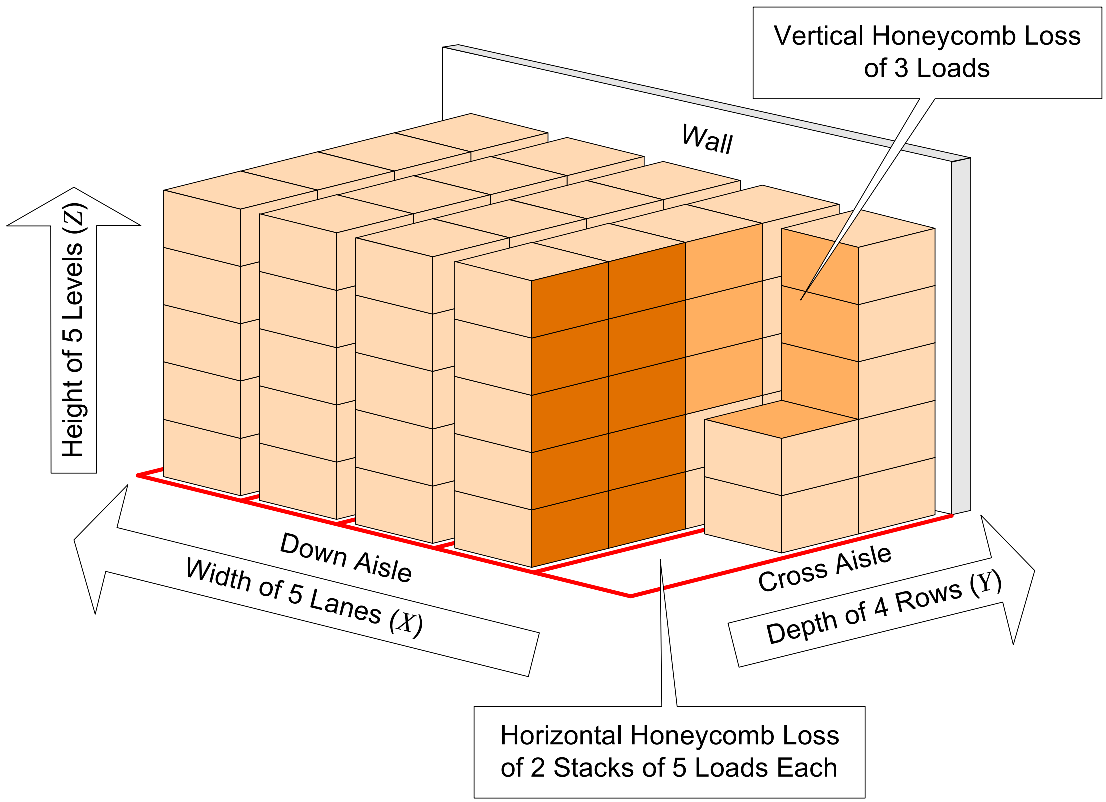

| | Dedicated | | | Random | Class-Based | | |
|---|---|---|---|---|---|---|---|
| **Time** | A | B | C | ABC | AB | AC | BC |
| 1 | 4 | 1 | 0 | 5 | 5 | 4 | 1 |
| 2 | 1 | 2 | 3 | 6 | 3 | 4 | 5 |
| 3 | 4 | 3 | 1 | 8 | 7 | 5 | 4 |
| 4 | 2 | 4 | 0 | 6 | 6 | 2 | 4 |
| 5 | 0 | 5 | 3 | 8 | 5 | 3 | 8 |
| 6 | 2 | 5 | 0 | 7 | 7 | 2 | 5 |
| 7 | 0 | 5 | 3 | 8 | 5 | 3 | 8 |
| 8 | 3 | 4 | 1 | 8 | 7 | 4 | 5 |
| 9 | 0 | 3 | 0 | 3 | 3 | 0 | 3 |
| 10 | 4 | 2 | 3 | 9 | 6 | 7 | 5 |
| $M_i$ | 4 | 5 | 3 | 9 | 7 | 7 | 8 |

# Cube Utilization

- *Cube utilization* is percentage of the total space (or “cube”) required for storage actually occupied by items being stored.
- There is usually a trade-off between cube utilization and material accessibility.
- Bulk storage using block stacking can result in the minimum cost of storage, but material accessibility is low since only the top of the front stack is accessible.
- Storage racks are used when support and/or material accessibility is required.

# Honeycomb Loss

- *Honeycomb loss*, the price paid for accessibility, is the unusable empty storage space in a lane or stack due to the storage of only a single SKU in each lane or stack



# Estimating Cube Utilization

- The (3-D) cube utilization for dedicated and randomized storage can estimated as follows:

$$\text{Cube utilization} = \frac{\text{item space}}{\text{total space}} = \frac{\text{item space}}{\text{item space} + \begin{pmatrix}\text{honeycomb}\\ \text{loss}\end{pmatrix} + \begin{pmatrix}\text{down aisle}\\ \text{space}\end{pmatrix}}$$

$$CU(\text{3-D}) = \begin{cases} \dfrac{x \cdot y \cdot z \cdot \sum_{i=1}^{N} M_i}{TS(D)}, & \text{dedicated} \\ \dfrac{x \cdot y \cdot z \cdot M}{TS(D)}, & \text{randomized} \end{cases}$$

$$CU(\text{2-D}) = \begin{cases} \dfrac{x \cdot y \cdot \sum_{i=1}^{N} \left\lceil \dfrac{M_i}{H} \right\rceil}{TA(D)}, & \text{dedicated} \\ \dfrac{x \cdot y \cdot \left\lceil \dfrac{M}{H} \right\rceil}{TA(D)}, & \text{randomized} \end{cases}$$

where

$x$ = lane/unit-load width

$y$ = unit-load depth

$z$ = unit-load height

$M_i$ = maximum number of units of SKU $i$

$M$ = maximum number of units of all SKUs

$N$ = number of different SKUs

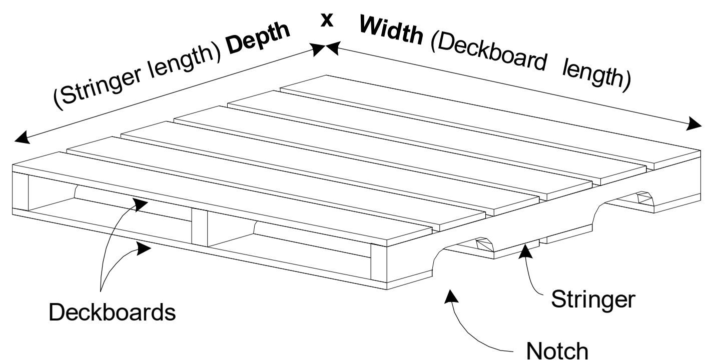
$D$ = number of rows

$TS(D)$ = total 3-D *space* (given $D$ rows of storage).

$TA(D)$ = total 2-D *area* (given $D$ rows of storage).

# Unit Load

- *Unit load*: single unit of an item, or multiple units restricted to maintain their integrity
- Linear dimensions of a unit load:

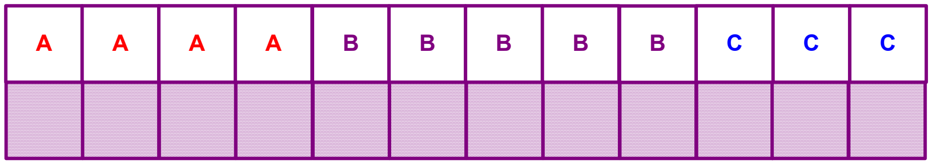
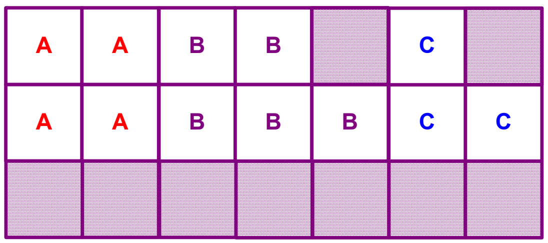
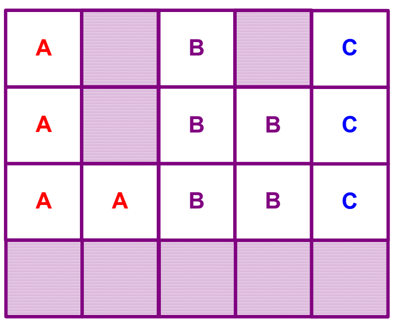



*Depth* (stringer length) × *Width* (deckboard length)

$$y \times x$$

- Pallet height (5 in.) + load height gives $z$: $y \times x \times z$

# Cube Utilization for Dedicated Storage

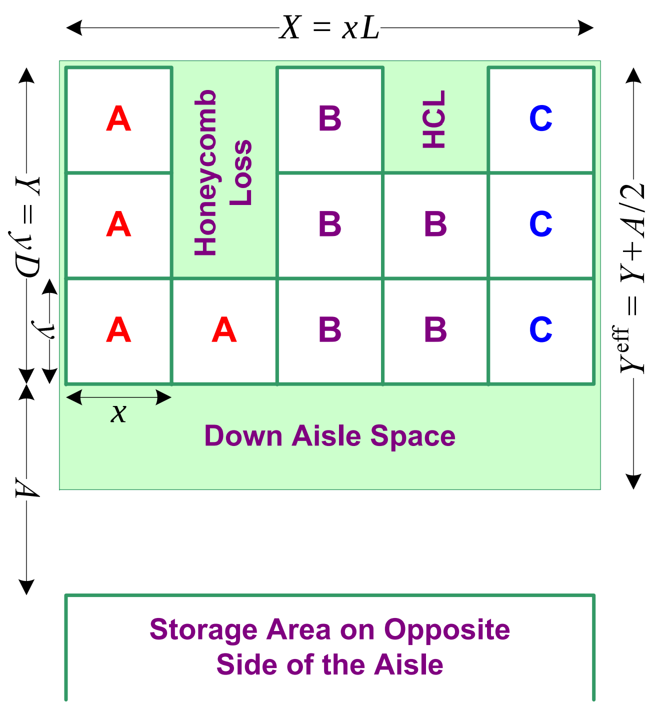
| Storage Area at Different Lane Depths | Item Space | Lanes | Total Space | Cube Util. |
|---|---|---|---|---|
| $D = 1$<br>$A/2 = 1$<br> | 12 | 12 | 24 | 50% |
| $D = 2$<br>$A/2 = 1$<br> | 12 | 7 | 21 | 57% |
| $D = 3$<br>$A/2 = 1$<br> | 12 | 5 | 20 | 60% |

# Total Space/Area

- The total space required, as a function of lane depth *D*:

$$\text{Total space (3-D):}\quad TS(D) = X \cdot \underbrace{\left(Y + \frac{A}{2}\right)}_{\text{Eff. lane depth}} \cdot Z = xL(D) \cdot \left(yD + \frac{A}{2}\right) \cdot zH$$

$$\text{Total area (2-D):}\quad TA(D) = \frac{TS(D)}{Z} = X \cdot Y^{\text{eff}} = xL(D) \cdot \left(yD + \frac{A}{2}\right)$$




where

$X =$ width of storage region (row length)

$Y =$ depth of storage region (lane depth)

$Z =$ height of storage region (stack height)

$A =$ down aisle width

$L(D) =$ number of lanes (given $D$ rows of storage)

$H =$ number of levels.

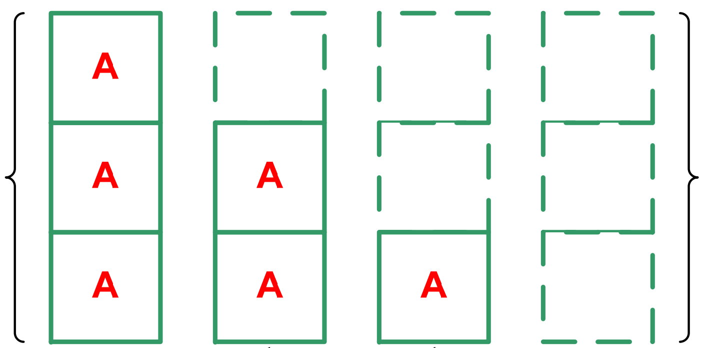
# Number of Lanes

- Given $D$, estimated total number of lanes in region:

$$\text{Number of lanes:}\quad L(D) = \begin{cases} \displaystyle\sum_{i=1}^{N} \left\lceil \frac{M_i}{DH} \right\rceil, & \text{dedicated} \\ \left\lceil \dfrac{M + NH\left(\dfrac{D-1}{2}\right) + N\left(\dfrac{H-1}{2}\right)}{DH} \right\rceil, & \text{randomized } (N > 1) \end{cases}$$

- Estimated HCL:

$D = 3$




doesn't occur because slots are used by another SKU

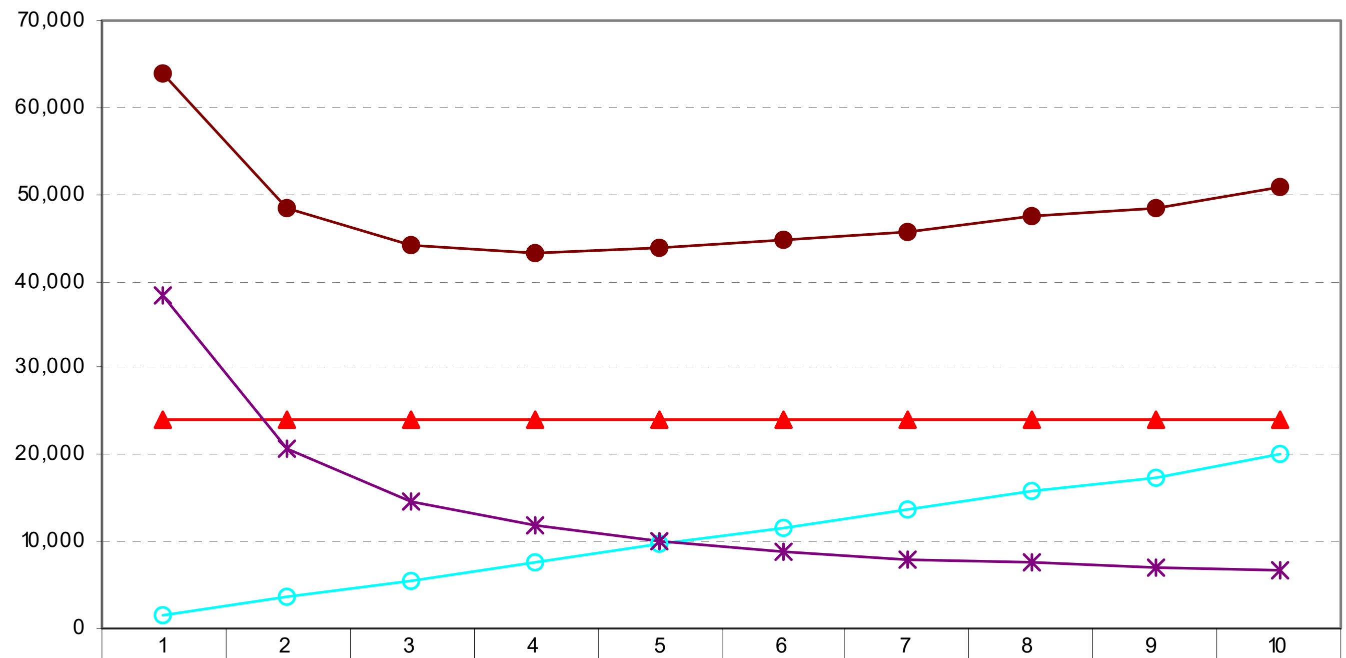
$$\begin{array}{lccccc} \text{Probability:} & \dfrac{1}{D} & & \dfrac{1}{D} & & \dfrac{1}{D} \\ & \times & + & \times & + & \times \\ \text{Unit Honeycomb Loss:} & 0 & & (D-2) & & (D-1) \end{array}$$

$$\text{Expected Loss:} \frac{1}{D}\big[(D-2)+(D-1)\big] = \frac{1}{D}(1+2) = \frac{1}{D}\sum_{i=1}^{D-1} i = \frac{1}{D}\left(\frac{(D-1)D}{2}\right) = \frac{D-1}{2} = 1$$

# Optimal Lane Depth

- Solving for $D$ in $dTS(D)/dD = 0$ results in:

Optimal lane depth for randomized storage (in rows): $$D^* = \left\lfloor \sqrt{\frac{A(2M-N)}{2NyH}} + \frac{1}{2} \right\rfloor$$




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Item Space | 24,000 | 24,000 | 24,000 | 24,000 | 24,000 | 24,000 | 24,000 | 24,000 | 24,000 | 24,000 |
| Honeycomb Loss | 1,536 | 3,648 | 5,376 | 7,488 | 9,600 | 11,712 | 13,632 | 15,936 | 17,472 | 20,160 |
| Aisle Space | 38,304 | 20,736 | 14,688 | 11,808 | 10,080 | 8,928 | 8,064 | 7,488 | 6,912 | 6,624 |
| Total Space | 63,840 | 48,384 | 44,064 | 43,296 | 43,680 | 44,640 | 45,696 | 47,424 | 48,384 | 50,784 |

**Lane Depth (in Rows)**

# Max Aggregate Inventory Level

- Usually can determine max inventory level for each SKU:
  - $M_i$ = maximum number of units of SKU *i*
- Since usually don't know $M$ directly, but can estimate it **if**
  - SKUs' inventory levels are uncorrelated
  - Units of each item are either stored or retrieved at a constant rate

$$M = \left\lfloor \sum_{i=1}^{N} \frac{M_i}{2} + \frac{1}{2} \right\rfloor$$

- Can add include safety stock for each item, $SS_i$
  - For example, if the order size of three SKUs is 50 units and 5 units of each item are held as safety stock

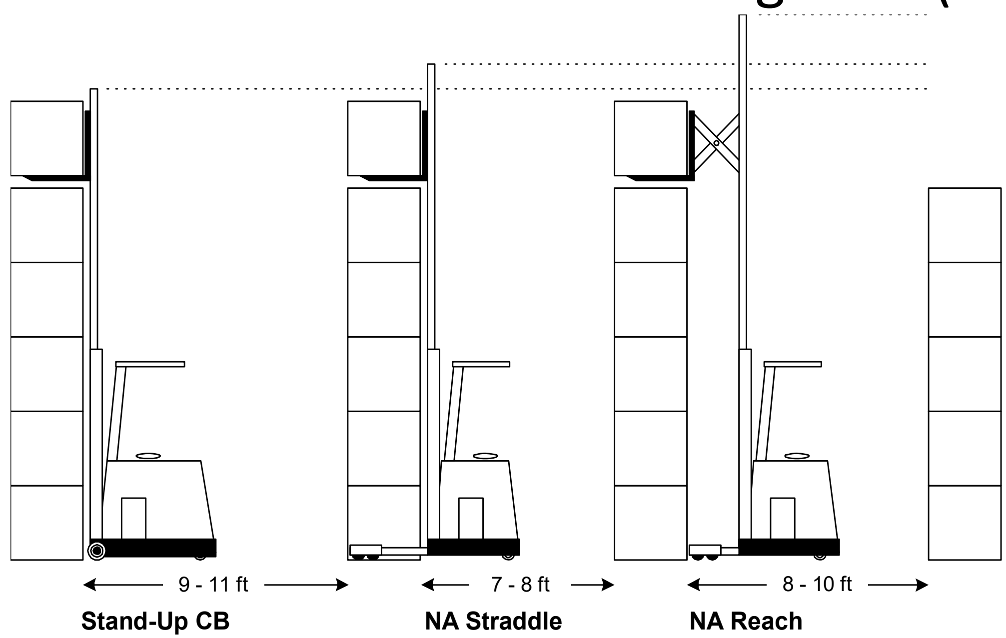
$$M = \left\lfloor \sum_{i=1}^{N} \left( \frac{M_i - SS_i}{2} + SS_i \right) + \frac{1}{2} \right\rfloor = \left\lfloor 3\left( \frac{50}{2} + 5 \right) + \frac{1}{2} \right\rfloor = 90$$

# Steps to Determine Area Requirements

1. For randomized storage, assumed to know $N$, $H$, $x$, $y$, $z$, $A$, and all $M_i$
   - Number of levels, $H$, depends on building clear height (for block stacking) or shelf spacing
   - Aisle width, $A$, depends on type of lift trucks used
2. Estimate maximum aggregate inventory level, $M$
3. If $D$ not fixed, estimate optimal land depth, $D^*$
4. Estimate number of lanes required, $L(D^*)$
5. Determine total 2-D area, $TA(D^*)$

# Aisle Width Design Parameter

- Typically, $A$ (and sometimes $H$) is a parameter used to evaluate different overall design alternatives
- Width depends on type of lift trucks used, a narrower aisle truck
  - reduces area requirements (building costs)
  - costs more and slows travel and loading time (handling costs)



# Example 1: Area Requirements

Units of items A, B, and C are all received and stored as 42 × 36 × 36 in. ($y \times x \times z$) pallet loads in a storage region that is along one side of a 10-foot-wide down aisle in the warehouse of a factory. The shipment size received for each item is 31, 62, and 42 pallets, respectively. Pallets can be stored up to three deep and four high in the region.

$$x = \frac{36}{12} = 3' \qquad M_A = 31 \qquad A = 10'$$

$$y = 3.5' \qquad M_B = 62 \qquad D = 3$$

$$z = 3' \qquad M_C = 42 \qquad H = 4$$

$$N = 3$$

# Example 1: Area Requirements

1. If a dedicated policy is used to store the items, what is the 2-D cube utilization of this storage region?

$$L(D) = L(3) = \sum_{i=1}^{N} \left\lceil \frac{M_i}{DH} \right\rceil = \left\lceil \frac{31}{3(4)} \right\rceil + \left\lceil \frac{62}{3(4)} \right\rceil + \left\lceil \frac{42}{3(4)} \right\rceil = 3 + 6 + 4 = 13 \text{ lanes}$$

$$TA(3) = xL(D) \cdot \left( yD + \frac{A}{2} \right) = 3(13) \cdot \left( 3.5(3) + \frac{10}{2} \right) = 605 \text{ ft}^2$$

$$CU(3) = \frac{\text{item space}}{TA(3)} = \frac{x \cdot y \cdot \sum_{i=1}^{N} \left\lceil \frac{M_i}{H} \right\rceil}{TA(3)} = \frac{3 \cdot 3.5 \cdot \left( \left\lceil \frac{31}{4} \right\rceil + \left\lceil \frac{62}{4} \right\rceil + \left\lceil \frac{42}{4} \right\rceil \right)}{605} = 61\%$$

# Example 1: Area Requirements

2. If the shipments of each item are uncorrelated with each other, no safety stock is carried for each item, and retrievals to the factory floor will occur at a constant rate, what is an estimate the maximum number of units of all items that would ever occur?

$$M = \left\lfloor \sum_{i=1}^{N} \frac{M_i}{2} + \frac{1}{2} \right\rfloor = \left\lfloor \frac{31+62+42}{2} + \frac{1}{2} \right\rfloor = 68$$

# Example 1: Area Requirements

3. If a randomized policy is used to store the items, what is total 2-D area needed for the storage region?

$$D = 3$$

$$L(3) = \left\lceil \frac{M + NH\left(\frac{D-1}{2}\right) + N\left(\frac{H-1}{2}\right)}{DH} \right\rceil$$

$$= \left\lceil \frac{68 + 3(4)\left(\frac{3-1}{2}\right) + N\left(\frac{4-1}{2}\right)}{3(4)} \right\rceil = 8 \text{ lanes}$$

$$TA(3) = xL(D) \cdot \left( yD + \frac{A}{2} \right) = 3(8) \cdot \left( 3.5(3) + \frac{10}{2} \right) = 372 \text{ ft}^2$$

# Example 1: Area Requirements

4. What is the optimal lane depth for randomized storage?

$$D^* = \left\lfloor \sqrt{\frac{A\left(2M-N\right)}{2NyH}} + \frac{1}{2} \right\rfloor = \left\lfloor \sqrt{\frac{10\left(2(68)-3\right)}{2(3)3.5(4)}} + \frac{1}{2} \right\rfloor = 4$$

5. What is the change in total area associated with using the optimal lane depth as opposed to storing the items three deep?

$$D = 4 \Rightarrow L(4) = \left\lceil \frac{68 + 3(4)\left(\frac{4-1}{2}\right) + N\left(\frac{4-1}{2}\right)}{3(4)} \right\rceil = 6 \text{ lanes}$$

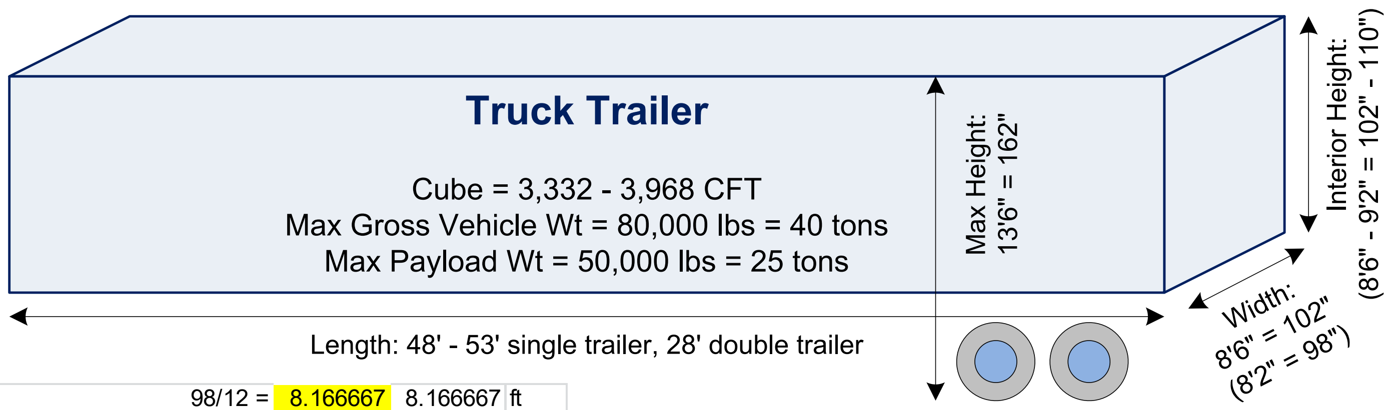
$$\Rightarrow TA(4) = 3(6) \cdot \left(3.5(4) + \frac{10}{2}\right) = 342 \text{ ft}^2$$

$$D = 3 \Rightarrow TA(3) = 372 \text{ ft}^2$$

# Example 2: Trailer Loading

How many identical 48 × 42 × 36 in. four-way containers can be shipped in a full truckload? Each container load:

1. Weighs 600 lb
2. Can be stacked up to six high without causing damage from crushing
3. Can be rotated on the trucks with respect to their width and depth.

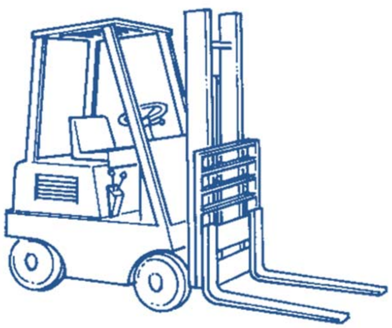



| | | | | |
|---|---|---|---|---|
| X | 98/12 = | 8.166667 | 8.166667 | ft |
| Y | | 53 | 53 | ft |
| Z | 110/12 = | 9.166667 | 9.166667 | ft |
| x | [48,42]/12 = | 4 | 3.5 | ft |
| y | [42,48]/12 = | 3.5 | 4 | ft |
| z | 30/12 = | 2.5 | 2.5 | ft |
| L | floor(X/x) = | 2 | 2 | |
| D | floor(Y/y) = | 15 | 13 | |
| H | min(6,floor(Z/z)) = | 3 | 3 | |
| LDH | L*D*H = | 90 | 78 | units |
| wt | | 600 | 600 | lb |
| unit/TL | min(LDH, floor(50000/wt)) = | 83 | 78 | |

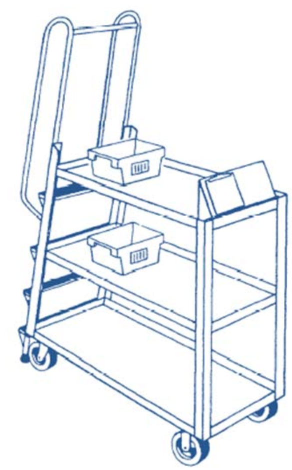
Max of 83 units per TL

# Storage and Retrieval Cycle

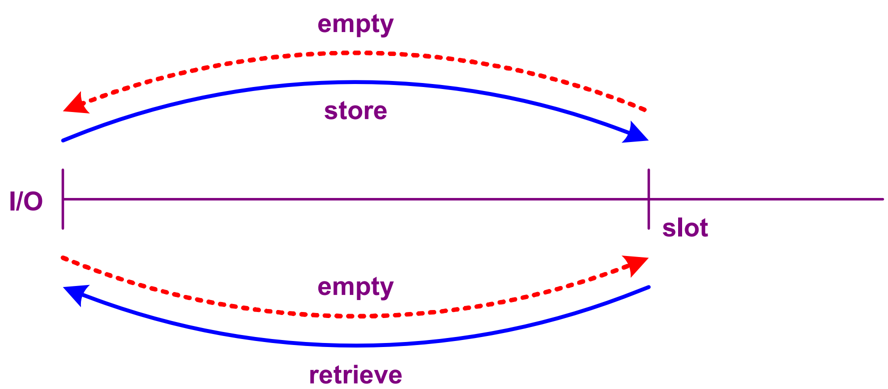
- A storage and retrieval (S/R) cycle is one complete roundtrip from an I/O port to slot(s) and back to the I/O
- Type of cycle depends on load carrying ability:
  - Carrying one load at-a-time (load carried on a pallet):
    - Single command
    - Dual command



  - Carrying multiple loads (order picking of small items):
    - Multiple command



# Single-Command S/R Cycle




Expected time for each SC S/R cycle:

$$t_{SC} = \frac{d_{SC}}{v} + t_L + t_U = \frac{d_{SC}}{v} + 2t_{L/U}$$

where

$d_{SC}$ = expected distance per SC cycle

$v$ = average travel speed (e.g.: 2 mph = 176 fpm walking; 7 mph = 616 fpm riding)

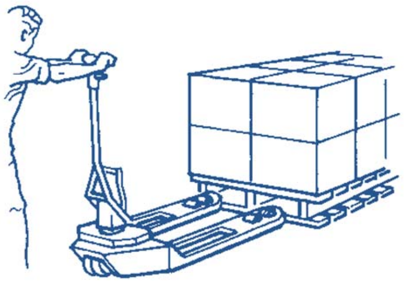
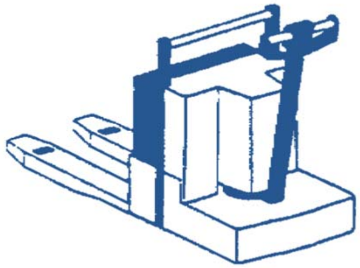
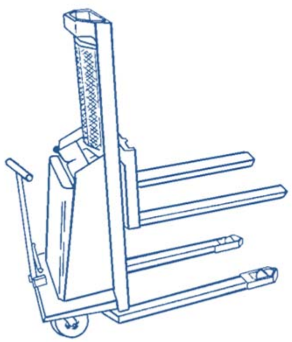
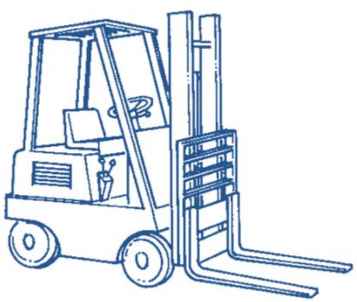
$t_L$ = loading time

$t_U$ = unloading time

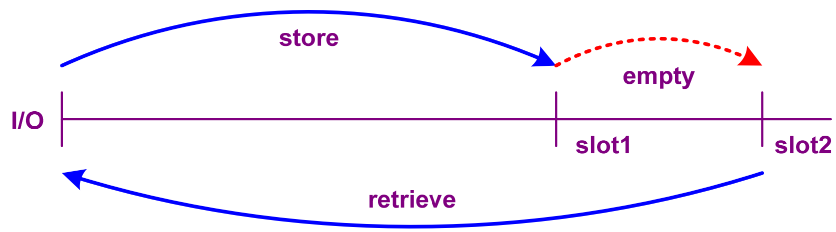
$t_{L/U}$ = loading/unloading time, if same value

- Single-command (SC) cycles:
  - Storage: carry one load to slot for storage and return empty back to I/O port, or
  - Retrieval: travel empty to slot to retrieve load and return with it back to I/O port

# Industrial Trucks: Walk vs. Ride

| Walk (2 mph = 176 fpm) | Ride (7 mph = 616 fpm) |
|---|---|
|  |  |
| Pallet Jack | Pallet Truck |
|  |  |
| Walkie Stacker | Sit-down Counterbalanced Lift Truck |

# Dual-Command S/R Cycle

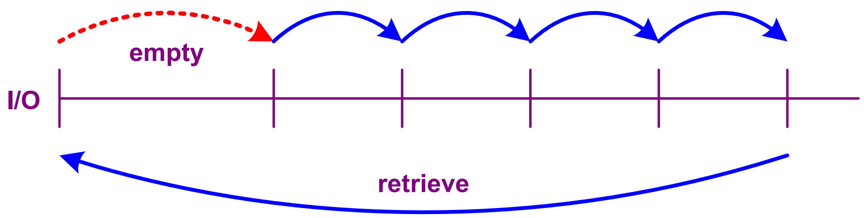



Expected time for each SC S/R cycle:

$$t_{DC} = \frac{d_{DC}}{v} + 2t_L + 2t_U = \frac{d_{DC}}{v} + 4t_{L/U}$$

- Dual-command (DC):
- Combine storage with a retrieval:
  - store load in slot 1, travel empty to slot 2 to retrieve load
- Can reduce travel distance by a third, on average
- Also termed task “interleaving”

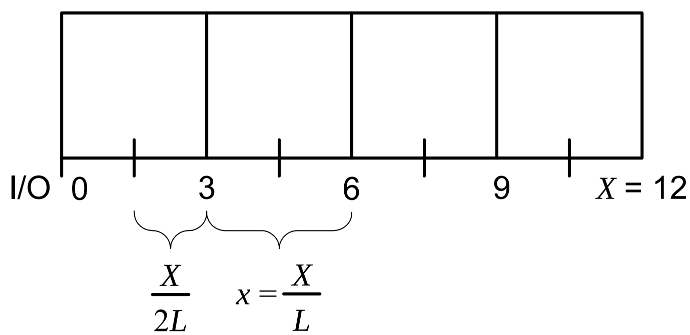
# Multi-Command S/R Cycle




- Multi-command: multiple loads can be carried at the same time
- Used in case and piece order picking
- Picker routed to slots
  - Simple VRP procedures can be used

# 1-D Expected Distance

1-D Storage Region




$$TD_{1-way} = \sum_{i=1}^{L}\left( i\frac{X}{L} - \frac{X}{2L} \right) = \frac{X}{L}\sum_{i=1}^{L} i - \frac{X}{2L}(1)$$

$$= \frac{X}{L}\left( \frac{L(L+1)}{2} \right) - \frac{X}{2L}(L)$$

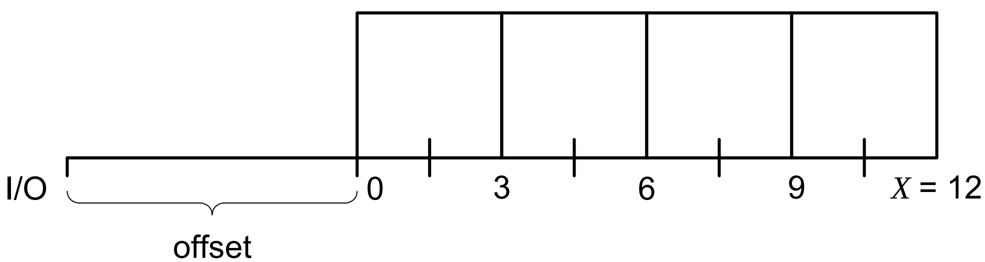
$$= \frac{XL + X - X}{2} = \frac{XL}{2}$$

$$ED_{1-way} = \frac{TD_{1-way}}{L} = \frac{X}{2}$$

$$d_{SC} = 2(ED_{1-way}) = X$$

- Assumptions:
  - All single-command cycles
  - Rectilinear distances
  - Each slot is region used with equal frequency (i.e., randomized storage)
- Expected distance is the average distance from I/O port to midpoint of each slot
  - e.g., [2(1.5) + 2(4.5) + 2(6.5) + 2(10.5)]/4 = 12

# Off-set I/O Port




$$d_{SC} = 2(d_{\text{offset}}) + X$$

- If the I/O port is off-set from the storage region, then 2 times the distance of the offset is added the expected distance within the slots

# 2-D Expected Distances

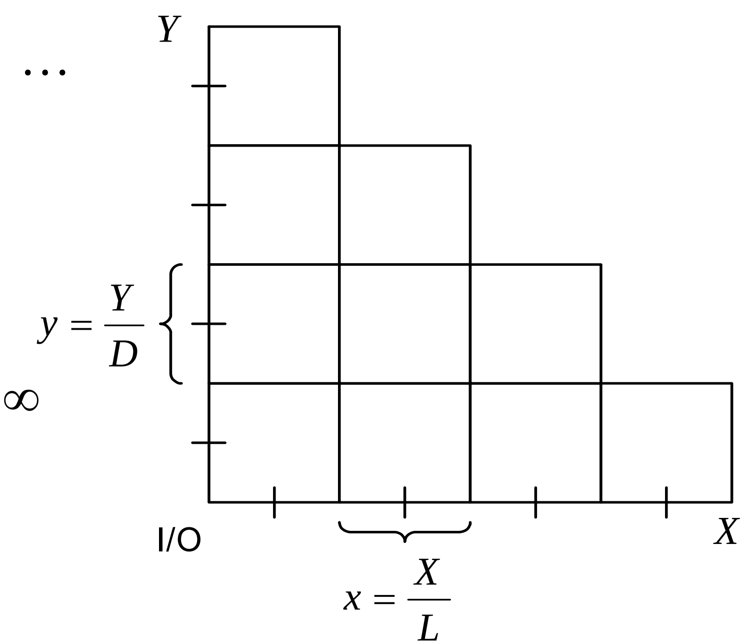
- Since dimensions $X$ and $Y$ are independent of each other for rectilinear distances, the expected distance for a 2-D rectangular region with the I/O port in a corner is just the sum of the distance in $X$ and in $Y$: $d_{SC}^{rect} = X + Y$
- For a triangular region with the I/O port in the corner:

$$TD_{\text{1-way}} = \sum_{i=1}^{L} \sum_{j=1}^{L-i+1} \left[ \left( i\frac{X}{L} - \frac{X}{2L} \right) + \left( j\frac{X}{L} - \frac{X}{2L} \right) \right] = \cdots$$

$$= \frac{X}{6}\left(2L^2 + 3L + 1\right)$$

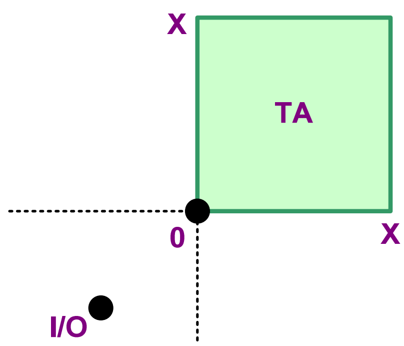
$$ED_{\text{1-way}} = \frac{TD_{\text{1-way}}}{\frac{L(L+1)}{2}} = \frac{2}{3}X + \frac{X}{3L} = \frac{2}{3}X, \quad \text{as } L \to \infty$$

$$d_{SC}^{tri} = 2\left(\frac{2}{3}X\right) = 2\left(\frac{1}{3}X + \frac{1}{3}Y\right) = \frac{2}{3}(X + Y)$$



# I/O-to-Side Configurations

## Rectangular

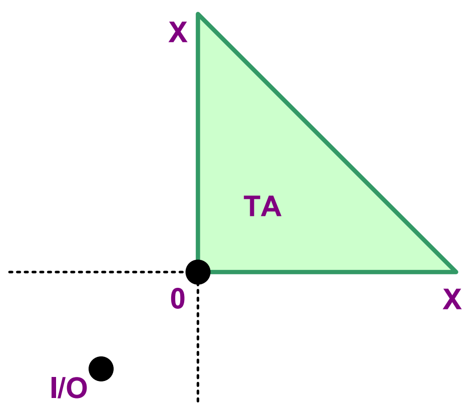



$$TA = X^2$$

$$\Rightarrow X = \sqrt{TA}$$

$$\Rightarrow d_{SC} = 2\sqrt{TA}$$

## Triangular



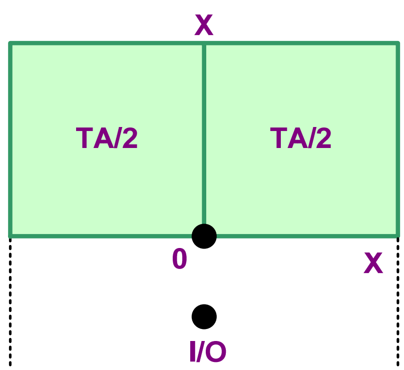

$$TA = \frac{1}{2} X^2$$

$$\Rightarrow X = \sqrt{2TA} = \sqrt{2}\sqrt{TA}$$

$$\Rightarrow d_{SC} = \frac{4}{3}\sqrt{2}\sqrt{TA} = 1.886\sqrt{TA}$$

# I/O-at-Middle Configurations

## Rectangular

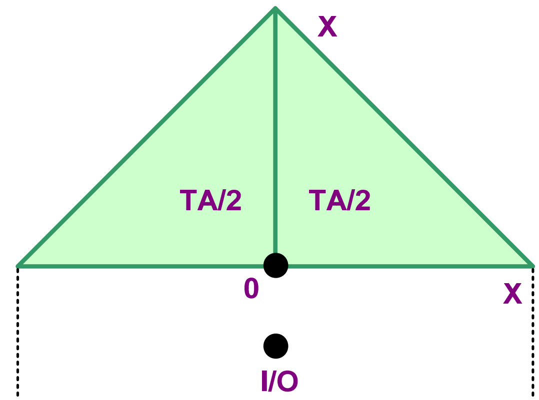



$$\frac{TA}{2} = X^2$$

$$\Rightarrow X = \sqrt{\frac{TA}{2}} = \frac{\sqrt{TA}}{\sqrt{2}}$$

$$\Rightarrow d_{SC} = \sqrt{2}\sqrt{TA} = 1.414\sqrt{TA}$$

## Triangular




$$\frac{TA}{2} = \frac{1}{2}X^2$$

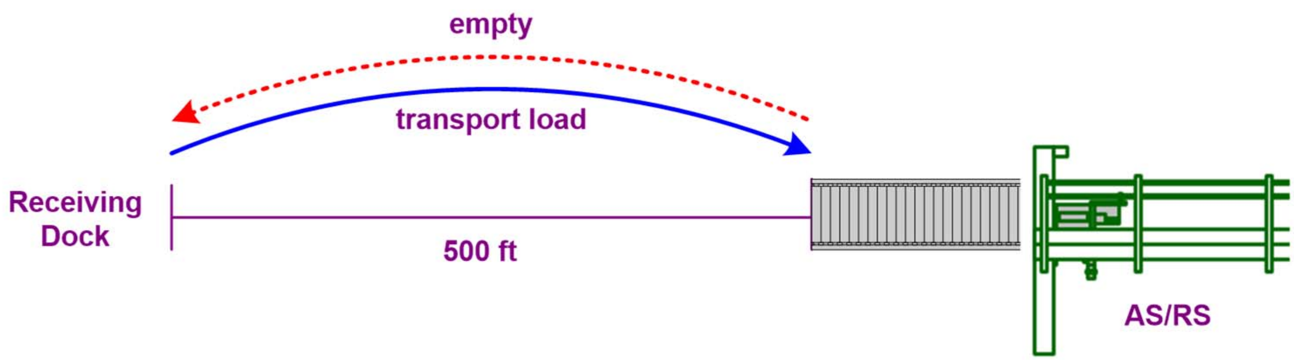
$$\Rightarrow X = \sqrt{TA}$$

$$\Rightarrow d_{SC} = \frac{4}{3}\sqrt{TA} = 1.333\sqrt{TA}$$

# Example 3: Handling Requirements

Pallet loads will be unloaded at the receiving dock of a warehouse and placed on the floor. From there, they will be transported 500 feet using a dedicated pallet truck to the in-floor induction conveyor of an AS/RS. Given

a. It takes 30 sec to load each pallet at the dock
b. 30 sec to unload it at the induction conveyor
c. There will be 80,000 loads per year on average
d. Operator rides on the truck (because a pallet truck)
e. Facility will operate 50 weeks per year, 40 hours per week



# Example 3: Handling Requirements

1. Assuming that it will take 30 seconds to load each pallet at the dock and 30 seconds to unload it at the induction conveyor, what is the expected time required for each single-command S/R cycle?

$$d_{SC} = 2(500) = 1000 \text{ ft/mov}$$

$$t_{SC} = \frac{d_{SC}}{v} + 2t_{L/U} = \frac{1000 \text{ ft/mov}}{616 \text{ ft/min}} + 2\left(\frac{30}{60}\right) \text{min/mov}$$

$$= 2.62 \text{ min/mov} \quad = \frac{2.62}{60} \text{ hr/mov}$$

(616 fpm because operator rides on a pallet truck)

# Example 3: Handling Requirements

2. Assuming that there will be 80,000 loads per year on average and that the facility will operate for 50 weeks per year, 40 hours per week, what is the minimum number of trucks needed?

$$r_{avg} = \frac{80,000 \text{ mov/yr}}{50(40) \text{ hr/yr}} = 40 \text{ mov/hr}$$

$$m = \lfloor r_{avg} t_{SC} + 1 \rfloor$$

$$= \left\lfloor 40\left(\frac{2.62}{60}\right) + 1 \right\rfloor = \lfloor 1.75 + 1 \rfloor$$

$$= 2 \text{ trucks}$$

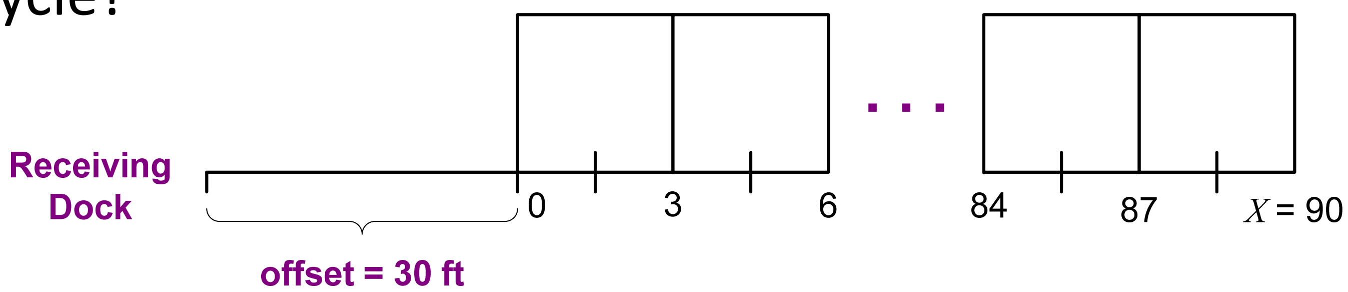
# Example 3: Handling Requirements

3. How many trucks are needed to handle a peak expected demand of 80 moves per hour?

$$
\begin{aligned}
r_{peak} &= 80 \text{ mov/hr} \\
m &= \lfloor r_{peak} t_{SC} + 1 \rfloor \\
&= \left\lfloor 80\left(\frac{2.62}{60}\right) + 1 \right\rfloor = \lfloor 3.50 + 1 \rfloor \\
&= 4 \text{ trucks}
\end{aligned}
$$

# Example 3: Handling Requirements

4. If, instead of unloading at the conveyor, the 3-foot-wide loads are placed side-by-side in a staging area along one side of 90-foot aisle that begins 30 feet from the dock, what is the expected time required for each single-command S/R cycle?




$$d_{SC} = 2(d_{\text{offset}}) + X = 2(30) + 90 = 150 \text{ ft}$$

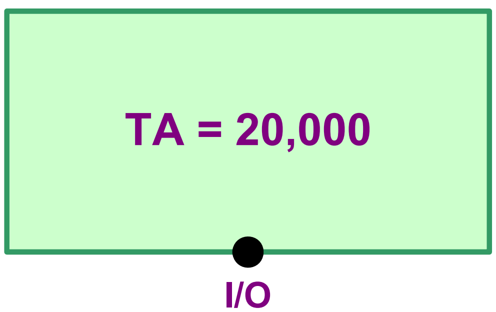
$$t_{SC} = \frac{d_{SC}}{v} + 2t_{L/U} = \frac{150 \text{ ft/mov}}{616 \text{ ft/min}} + 2\left(\frac{30}{60}\right) \text{min/mov}$$

$$= 1.24 \text{ min/mov} \quad = \frac{1.24}{60} \text{ hr/mov}$$

# Estimating Handling Costs

- Warehouse design involves the trade-off between building and handling cost.
- Maximizing the cube utilization of a storage region will help minimize building costs.
- Handling costs can be estimated by determining:
  1. Expected time required for each move based on an average of the time required to reach each slot in the region.
  2. Number of vehicles needed to handle a target *peak demand* for moves, e.g., moves per hour.
  3. *Operating costs per hour of vehicle operation*, e.g., labor, fuel (assuming the operators can perform other productive tasks when not operating a truck)
  4. Annual operating costs based on *annual demand* for moves.
  5. Total handling costs as the sum of the annual capital recovery costs for the vehicles and the annual operating costs.

# Example 4: Estimating Handing Cost




Expected Distance: $d_{SC} = \sqrt{2}\sqrt{TA} = \sqrt{2}\sqrt{20,000} = 200 \text{ ft}$

Expected Time: $$t_{SC} = \frac{d_{SC}}{v} + 2t_{L/U}$$

$$= \frac{200 \text{ ft}}{200 \text{ fpm}} + 2(0.5 \text{ min}) = 2 \text{ min per move}$$

Peak Demand: $r_{\text{peak}} = 75 \text{ moves per hour}$

Annual Demand: $r_{\text{year}} = 100,000 \text{ moves per year}$

Number of Trucks: $$m = \left\lfloor r_{\text{peak}} \frac{t_{SC}}{60} + 1 \right\rfloor = \lfloor 3.5 \rfloor = 3 \text{ trucks}$$

Handling Cost: $$TC_{\text{hand}} = mK_{\text{truck}} + r_{\text{year}} \frac{t_{SC}}{60} C_{\text{labor}}$$

$$= 3(\$2,500 / \text{tr-yr}) + 100,000 \frac{2}{60} (\$10 / \text{hr})$$

$$= \$7,500 + \$33,333 = \$40,833 \text{ per year}$$

# Dedicated Storage Assignment (DSAP)

- The assignment of items to slots is termed *slotting*
  - With randomized storage, all items are assigned to all slots
- DSAP (dedicated storage assignment problem):
  - Assign $N$ items to slots to minimize total cost of material flow
- DSAP solution procedure:
  1. *Order Slots:* Compute the expected cost for each slot and then put into nondecreasing order
  2. *Order Items:* Put the flow density (flow per unit of volume) for each item $i$ into nonincreasing order

$$\frac{f_{[1]}}{M_{[1]}s_{[1]}} \geq \frac{f_{[2]}}{M_{[2]}s_{[2]}} \geq \cdots \geq \frac{f_{[N]}}{M_{[N]}s_{[N]}}$$

  3. *Assign Items to Slots:* For $i = 1, \ldots, N$, assign item $[i]$ to the first slots with a total volume of at least $M_{[i]}s_{[i]}$

# 1-D Slotting Example

| | | A | B | C |
|---|---|---|---|---|
| Max units | M | 4 | 5 | 3 |
| Space/unit | s | 1 | 1 | 1 |
| Flow | f | 24 | 7 | 21 |
| Flow Density | f/(M x s) | 6.00 | 1.40 | 7.00 |

| Flow Density | 1-D Slot Assignments | Expected Distance | | Flow | | Total Distance |
|---|---|---|---|---|---|---|
| $\frac{21}{3} = 7.00$ | I/O: C C C (0 to 3) | 2(0) + 3 = 3 | $\times$ | 21 | = | 63 |
| $\frac{24}{4} = 6.00$ | I/O: A A A A (-3; 0 to 4) | 2(3) + 4 = 10 | $\times$ | 24 | = | 240 |
| $\frac{7}{5} = 1.40$ | I/O: B B B B B (-7; 0 to 5) | 2(7) + 5 = 19 | $\times$ | 7 | = | 133 |
| | I/O: C C C A A A A B B B B B (0, 3, 7, 12) | | | | | 436 |

# 1-D Slotting Example (cont)

| | | Dedicated | | | Random | Class-Based | | |
|---|---|---|---|---|---|---|---|---|
| | | A | B | C | ABC | AB | AC | BC |
| Max units | M | 4 | 5 | 3 | 9 | 7 | 7 | 8 |
| Space/unit | s | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Flow | f | 24 | 7 | 21 | 52 | 31 | 45 | 28 |
| Flow Density | f/(M x s) | 6.00 | 1.40 | 7.00 | 5.78 | 4.43 | 6.43 | 3.50 |

| | | 1-D Slot Assignments | | | | | | | | | | | | Total Distance | Total Space |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dedicated (flow density) | I/O | C | C | C | A | A | A | A | B | B | B | B | B | 436 | 12 |
| Dedicated (flow only) | I/O | A | A | A | A | C | C | C | B | B | B | B | B | 460 | 12 |
| Class-based | I/O | C | C | C | AB | AB | AB | AB | AB | AB | AB | | | 466 | 10 |
| Randomized | I/O | ABC | ABC | ABC | ABC | ABC | ABC | ABC | ABC | ABC | | | | 468 | 9 |

# 2-D Slotting Example

| | | A | B | C |
|---|---|---|---|---|
| Max units | M | 4 | 5 | 3 |
| Space/unit | s | 1 | 1 | 1 |
| Flow | f | 24 | 7 | 21 |
| Flow Density | f/(M x s) | 6.00 | 1.40 | 7.00 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 8 | 7 | 6 | 5 | 4 | 5 | 6 | 7 | 8 |
| 7 | 6 | 5 | 4 | 3 | 4 | 5 | 6 | 7 |
| 6 | 5 | 4 | 3 | 2 | 3 | 4 | 5 | 6 |
| 5 | 4 | 3 | 2 | 1 | 2 | 3 | 4 | 5 |
| 4 | 3 | 2 | 1 | 0 | 1 | 2 | 3 | 4 |

Distance from I/O to Slot

| | | | | |
|---|---|---|---|---|
| | C | C | B | |
| C | A | A | B | B |
| A | A | I/O | B | B |

Original Assignment (TD = 215)

| | | | | |
|---|---|---|---|---|
| | B | B | B | |
| B | A | C | A | B |
| A | C | I/O | C | A |

Optimal Assignment (TD = 177)

# DSAP Assumptions

1. All SC S/R moves
2. For item *i*, probability of move to/from each slot assigned to item is the same
3. The *factoring assumption*:
   a. Handling cost and distances (or times) for each slot are identical for all items
   b. Percent of S/R moves of item stored at slot *j* to/from I/O port *k* is identical for all items

- Depending of which assumptions not valid, can determine assignment using other procedures

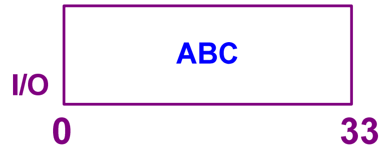
$$\left(c_i x_{ij}\right) DSAP \subset \underset{\left(c_{ij} x_{ij}\right)}{LAP} \subset LP \subset \underset{\underset{TSP}{\cup}}{QAP} \left(c_{ijkl} x_{ij} x_{kl}\right)$$

# Example 5: 1-D DSAP

- What is the change in the minimum expected total distance traveled if dedicated, as compared to randomized, block stacking is used, where
  a. Slots located on one side of 10-foot-wide down aisle
  b. All single-command S/R operations
  c. Each lane is three-deep, four-high
  d. $40 \times 36$ in. two-way pallet used for all loads
  e. Max inventory levels of SKUs A, B, C are 94, 64, and 50
  f. Inventory levels are uncorrelated and retrievals occur at a constant rate
  g. Throughput requirements of A, B, C are 160, 140, 130
  h. Single I/O port is located at the end of the aisle

# Example 5: 1-D DSAP

- Randomized:




$$M = \left\lfloor \frac{M_A + M_B + M_C}{2} + \frac{1}{2} \right\rfloor = \left\lfloor \frac{94 + 64 + 50}{2} + \frac{1}{2} \right\rfloor = 104$$

$$L_{rand} = \left\lceil \frac{M + NH\left(\frac{D-1}{2}\right) + N\left(\frac{H-1}{2}\right)}{DH} \right\rceil$$

$$= \left\lceil \frac{104 + 3(4)\left(\frac{3-1}{2}\right) + N\left(\frac{4-1}{2}\right)}{3(4)} \right\rceil = 11 \text{ lanes}$$

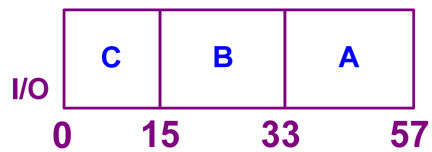
$$X = xL_{rand} = 3(11) = 33 \text{ ft}$$

$$d_{SC} = X = 33 \text{ ft}$$

$$TD_{rand} = \left( f_A + f_B + f_C \right) X = \left(160 + 140 + 130\right) 33 = 14{,}190 \text{ ft}$$

# Example 5: 1-D DSAP

- Dedicated:




$$\frac{f_A}{M_A} = \frac{160}{94} = 1.7, \ \frac{f_B}{M_B} = \frac{140}{64} = 2.19, \ \frac{f_C}{M_C} = \frac{130}{50} = 2.6 \quad \Rightarrow \quad C < B < A$$

$$L_A = \left\lceil \frac{M_A}{DH} \right\rceil = \left\lceil \frac{160}{3(4)} \right\rceil = 8, \ L_B = \left\lceil \frac{M_B}{DH} \right\rceil = \left\lceil \frac{140}{3(4)} \right\rceil = 6, \ L_C = \left\lceil \frac{M_C}{DH} \right\rceil = \left\lceil \frac{130}{3(4)} \right\rceil = 5$$

$$X_C = xL_C = 3(5) = 15, \ X_B = xL_B = 3(6) = 18, \ X_A = xL_A = 3(8) = 24$$

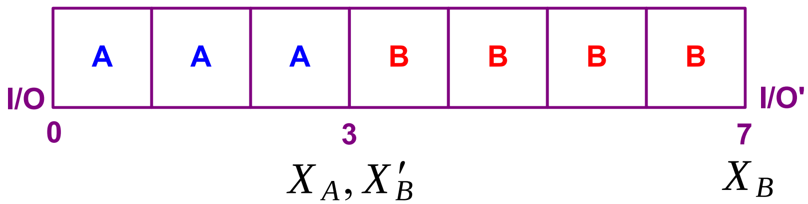
$$d_{SC}^{C} = X_C = 3(5) = 15 \text{ ft}$$

$$d_{SC}^{B} = 2(X_C) + X_B = 2(15) + 18 = 48 \text{ ft}$$

$$d_{SC}^{A} = 2(X_C + X_B) + X_A = 2(15 + 18) + 24 = 90 \text{ ft}$$

$$TD_{ded} = f_A d_{SC}^{A} + f_B d_{SC}^{B} + f_C d_{SC}^{C} = 160(90) + 140(48) + 130(15) = 23,070 \text{ ft}$$

# 1-D Multiple Region Expected Distance




- In 1-D, easy to determine the offset
- In 2-D, no single offset value for each region

$$d_{SC}^{A} = d_A = X_A = 3$$

$$d_B = 2d_{offset} + \left(X_B - X_A\right) = 2X_A + \left(X_B - X_A\right) = X_A + X_B = 10$$

$$= 2\left(d_{I/\mathrm{O\ to\ }I/O'}\right) - X'_B = 2\left(7\right) - 4 = 10$$

$$d_{AB} = 7$$

$$TA_A = X_A = 3, \quad TA_B = X_B - X_A = 4, \quad TA_{AB} = TA_A + TA_B = 7$$

$$TM_A = TA_A d_A, \quad TM_B = TA_B d_B$$

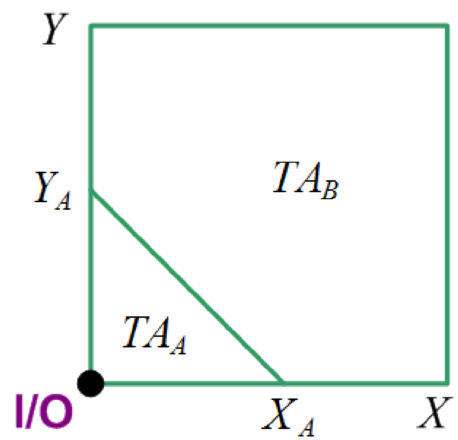
$$TM_{AB} = TA_{AB} d_{AB} = X_B^2 = \left(X_A + X_B - X_A\right)^2 = X_A^2 + \left(X_B - X_A\right)\left(2X_A + X_B - X_A\right)$$

$$= TA_A d_A + TA_B d_B = TM_A + TM_B$$

$$d_B = \frac{TM_B}{TA_B} = \frac{TM_{AB} - TM_A}{TA_B} = \frac{TA_{AB} d_{AB} - TA_A d_A}{TA_B} = \frac{7(7) - 3(3)}{4} = 10$$

# 2-D Multiple Region Expected Distance

**Case:** $\boxed{TA_A \le \frac{1}{2} TA_{AB}}$




Let $X = Y, X_A = Y_A, d = d_{sc}$

$$d_A = \frac{2}{3}(X_A + Y_A) = \frac{4}{3} X_A$$

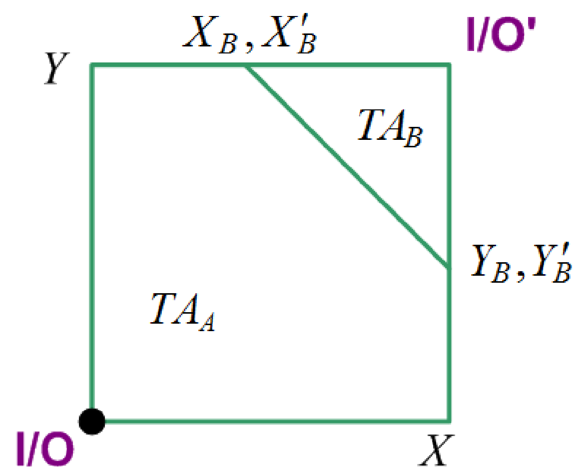
$$d_B = \frac{TM_B}{TA_B} = \frac{TM_{AB} - TM_A}{TA_B} = \frac{TA_{AB}\, d_{AB} - TA_A\, d_A}{TA_B} = \frac{XY(X+Y) - \frac{1}{2} X_A Y_A \frac{2}{3}(X_A + Y_A)}{XY - \frac{1}{2} X_A Y_A}$$

$$= \frac{2}{3} \frac{3X^2Y + 3XY^2 - X_A^2 Y_A - X_A Y_A^2}{2XY - X_A Y_A} = \frac{4}{3} \frac{3X^3 - X_A^3}{2X^2 - X_A^2}$$

If $X = X_A \Rightarrow d_B = \frac{4}{3} \frac{X^2(2X)}{X^2} = \frac{8}{3} X$

# 2-D Multiple Region Expected Distance

**Case:** $\boxed{TA_A \geq \frac{1}{2} TA_{AB}}$



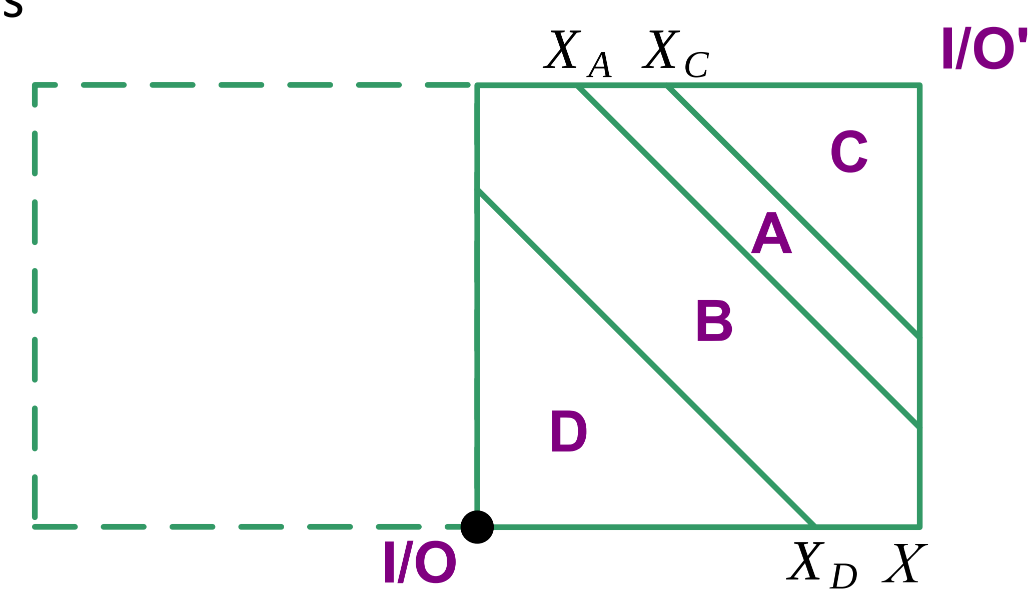

Let $X'_B = X - X_B, Y'_B = Y - Y_B$

$$\begin{aligned}
d_B &= 2\left(d_{I/O \text{ to } I/O'}\right) - \frac{2}{3}\left(X'_B + Y'_B\right) \\
&= 2(X+Y) - \frac{2}{3}\left[(X - X_B) + (Y - Y_B)\right] \\
&= \frac{4}{3}(X+Y) + \frac{2}{3}(X_B + Y_B) \\
&= \frac{8}{3}X + \frac{4}{3}X_B, \text{ where } X = Y
\end{aligned}$$

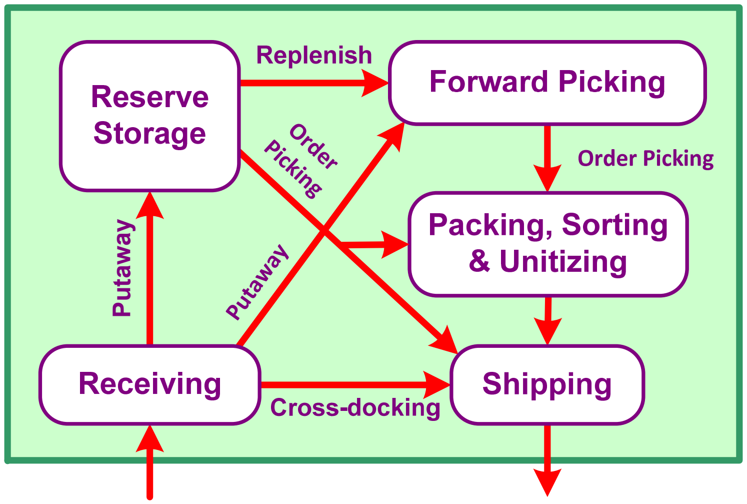
If $X_B = 0 \Rightarrow d_B = \frac{8}{3}X$

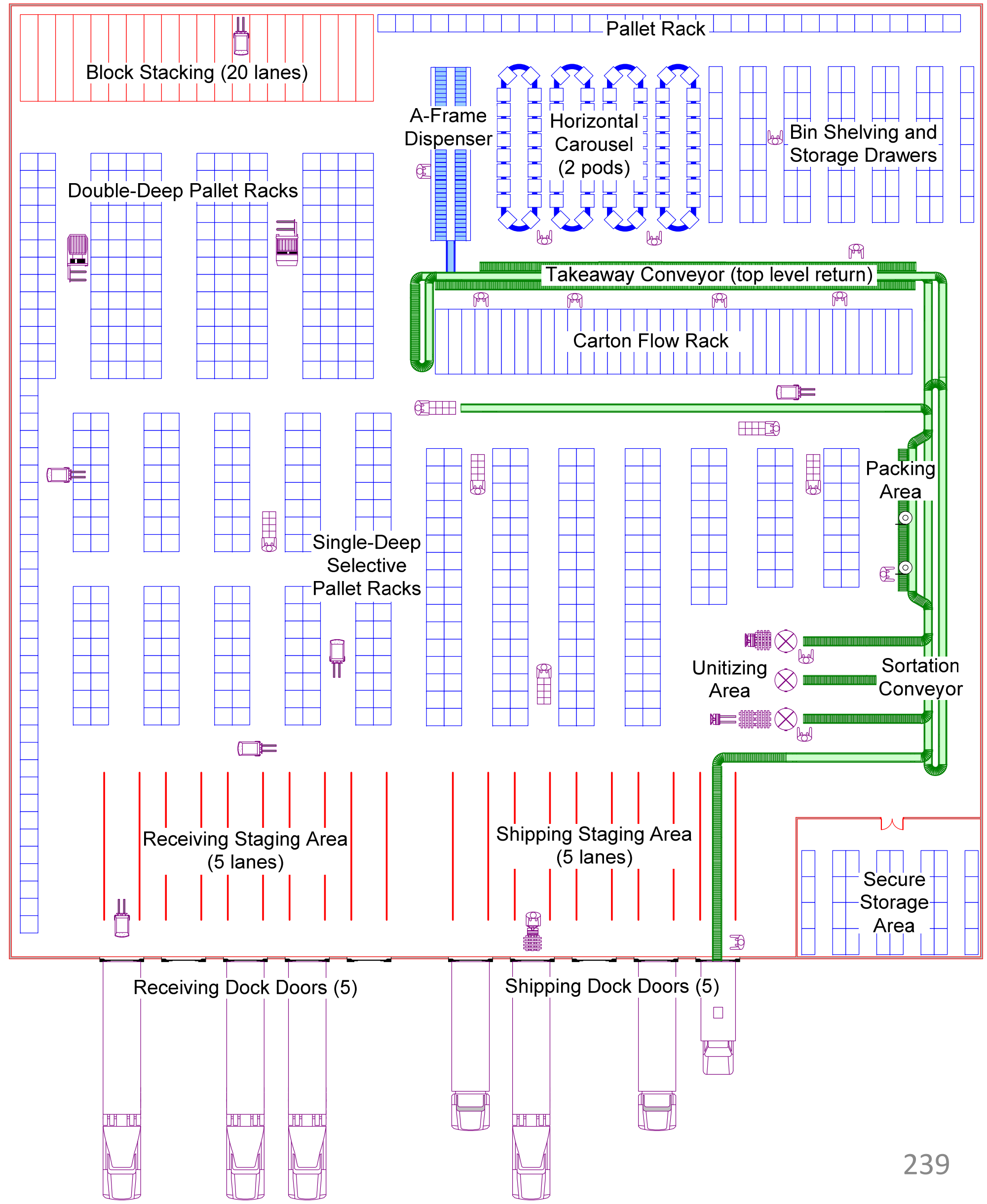
# 2-D Multiple Region Expected Distance

- If more than two regions
  - For regions below diagonal (D), start with region closest to I/O
  - For regions above diagonal (A+C), start with regions closest to I/O’ (C)
  - For region in the middle (B), solve using whole area less other regions



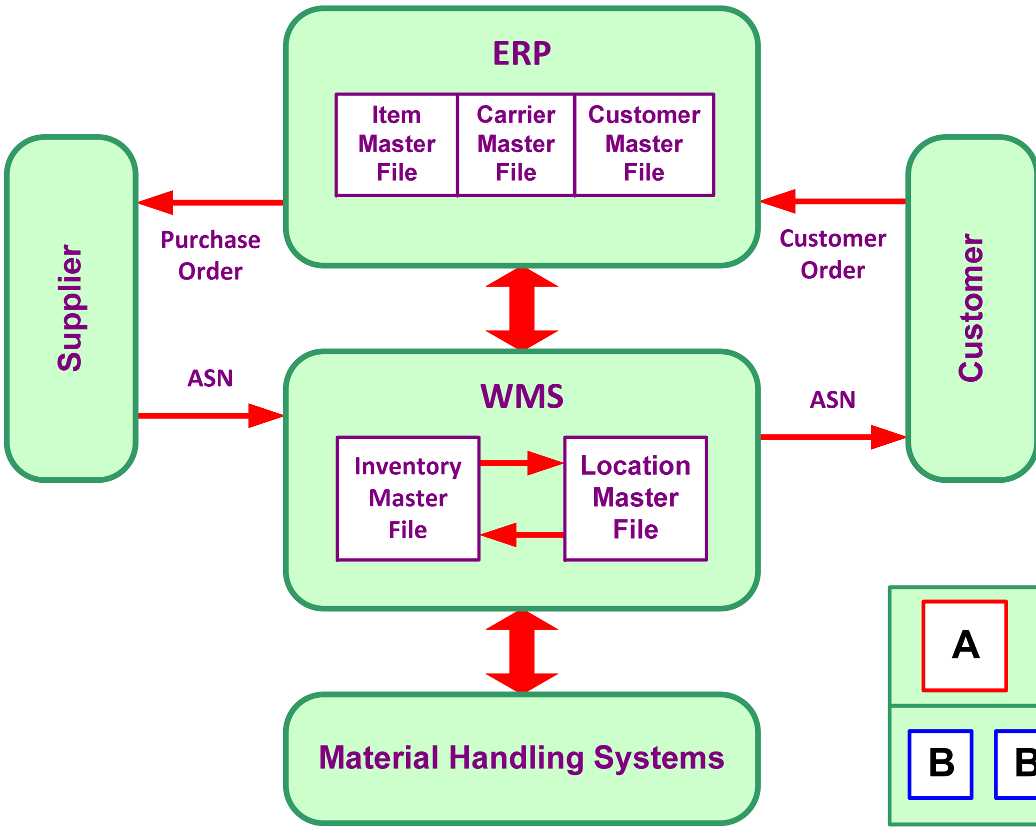
# Warehouse Operations



# Warehouse Management System

- WMS interfaces with a corporation's enterprise resource planning (ERP) and the control software of each MHS

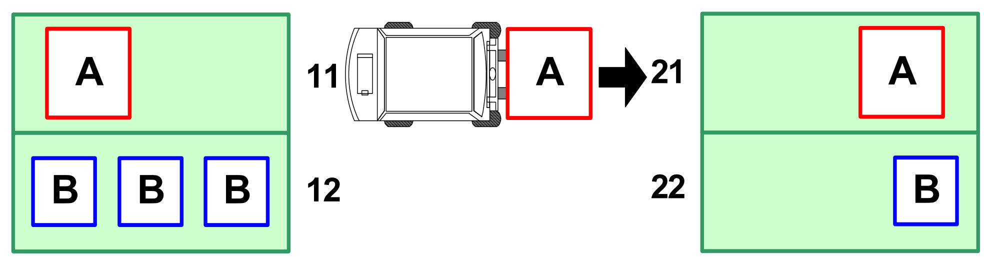



**Inventory Master File**

| Item | On-Hand Balance | In-Transit Qty. | Locations |
|---|---|---|---|
| A | 2 | 1 | 11,21 |
| B | 4 | 0 | 12,22 |

**Location Master File**

| Location | Item | On-Hand Balance | In-Transit Qty. |
|---|---|---|---|
| 11 | A | 1 | 0 |
| 12 | B | 3 | 0 |
| 21 | A | 1 | 1 |
| 22 | B | 1 | 0 |



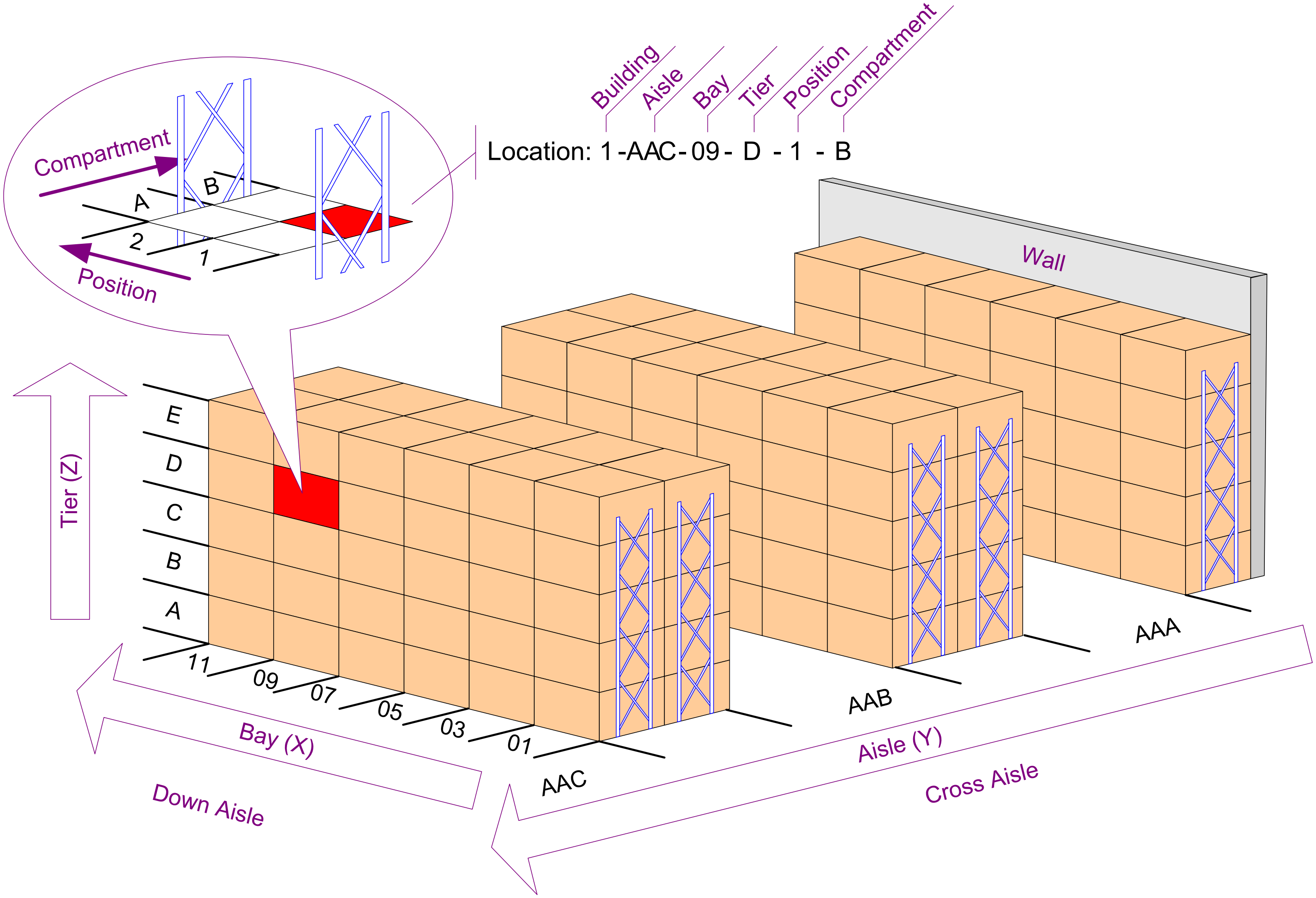

- Advance shipping notice (ASN) is a standard format used for communications

# Logistics-related Codes

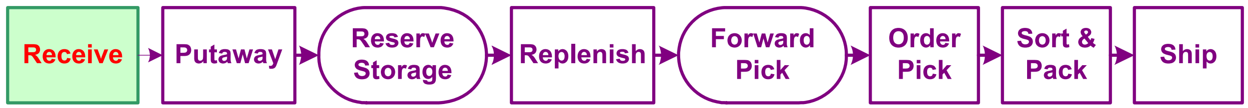
| | Commodity Code | Item Code | Unit Code |
|---|---|---|---|
| Level | Category | Class | Instance |
| Description | Grouping of similar objects | Grouping of identical objects | Unique physical object |
| Function | Product classification | Inventory control | Object tracking |
| Names | — | Item number, Part number, SKU, SKU + Lot number | Serial number, License plate |
| Codes | UNSPSC, GPC | GTIN, UPC, ISBN, NDC | EPC, SSCC |

UNSPSC: United Nations Standard Products and Services Code
GPC: Global Product Catalogue
GTIN: Global Trade Item Number (includes UPC, ISBN, and NDC)
UPC: Universal Product Code
ISBN: International Standard Book Numbering
NDC: National Drug Code
EPC: Electronic Product Code (globally unique serial number for physical objects identified using RFID tags)
SSCC: Serial Shipping Container Code (globally unique serial number for identifying *movable units* (carton, pallet, trailer, etc.))

# Identifying Storage Locations



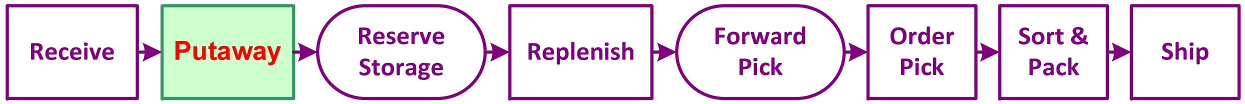
# Receiving




- Basic steps:

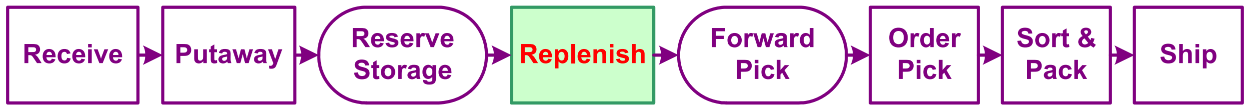
1. Unload material from trailer.
2. Identify supplier with ASN, and associate material with each moveable unit listed in ASN.
3. Assign inventory attributes to movable unit from item master file, possibly including repackaging and assigning new serial number.
4. Inspect material, possibly including holding some or all of the material for testing, and report any variances.
5. Stage units in preparation for putaway.
6. Update item balance in inventory master and assign units to a receiving area in location master.
7. Create receipt confirmation record.
8. Add units to putaway queue

# Putaway

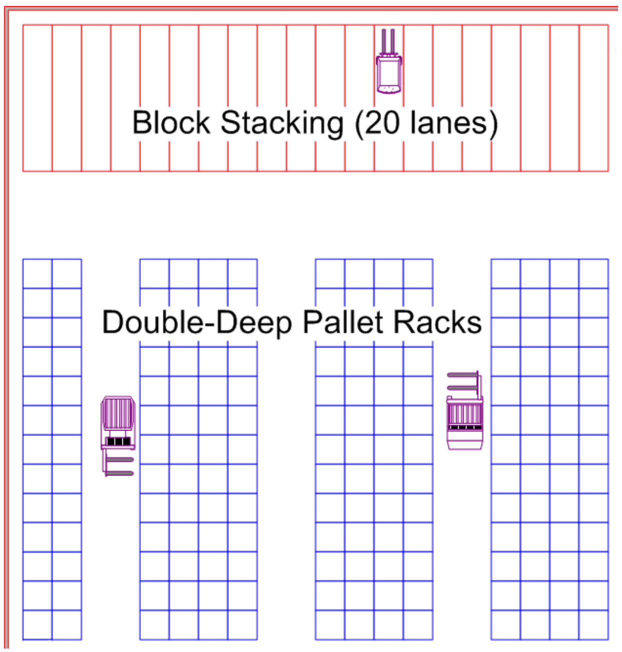



- A putaway algorithm is used in WMS to search for and validate locations where each movable unit in the putaway queue can be stored
- Inventory and location attributes used in the algorithm:
  - *Environment* (refrigerated, caged area, etc.)
  - *Container type* (pallet, case, or piece)
  - *Product processing type* (e.g., floor, conveyable, nonconveyable)
  - *Velocity* (assign to A, B, C based on throughput of item)
  - Preferred putaway zone (item should be stored in same zone as related items in order to improve picking efficiency)

# Replenishment

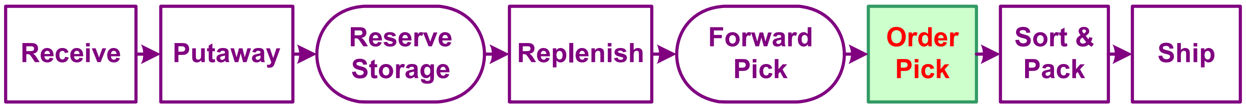



- Replenishment is the process of moving material from reserve storage to a forward picking area so that it is available to fill customer orders efficiently
- Other types of in-plant moves include:
  - Consolidation: combining several partially filled storage locations of an item into a single location
  - Rewarehousing: moving items to different storage locations to improve handling efficiency

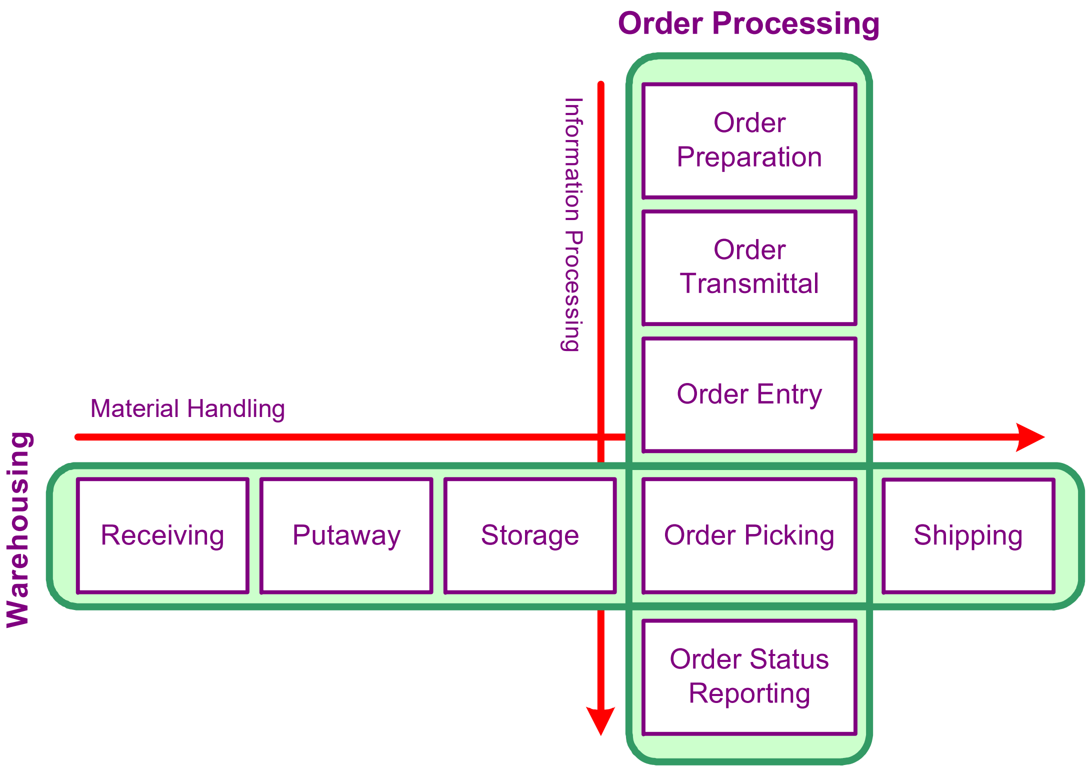


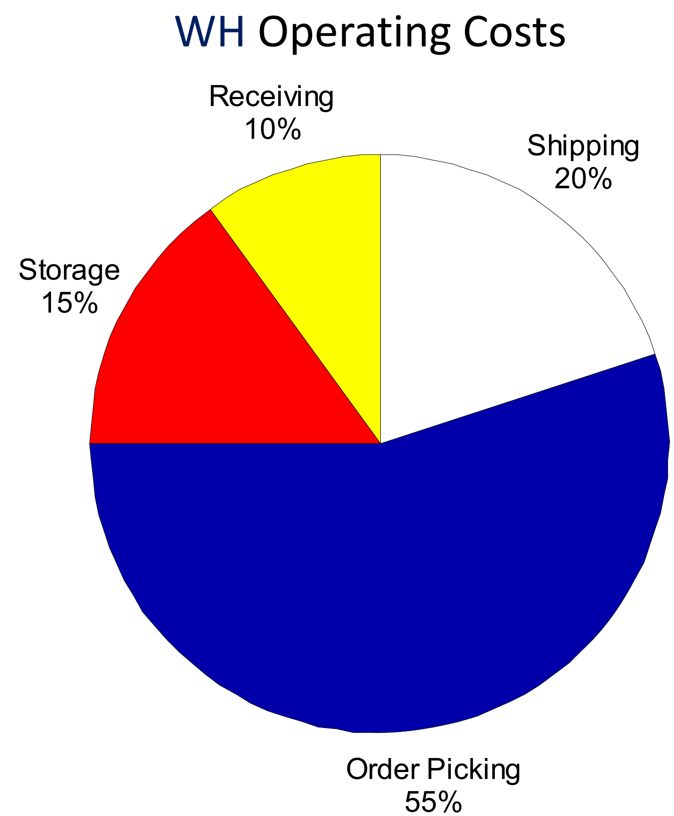

Reserve Storage Area

# Order Picking

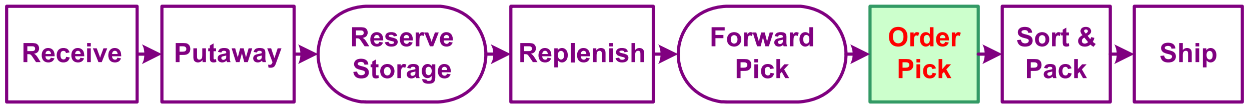



- Order picking is at the intersection of warehousing and order processing

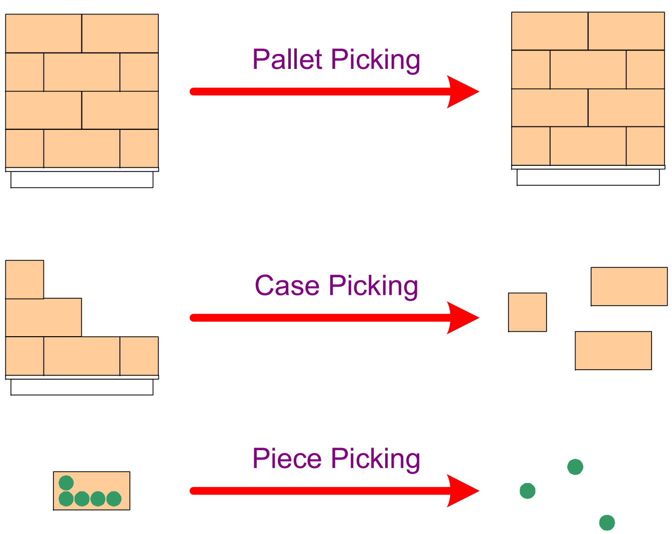


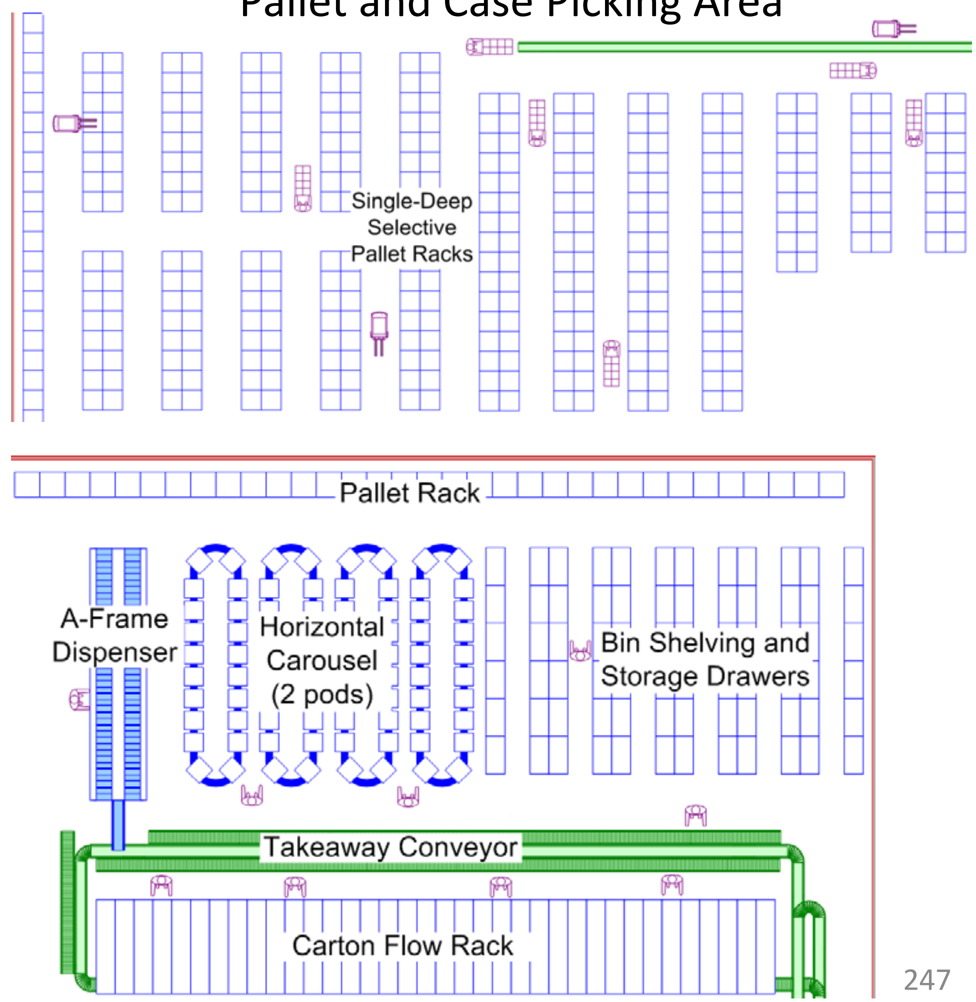
# Order Picking




## Levels of Order Picking

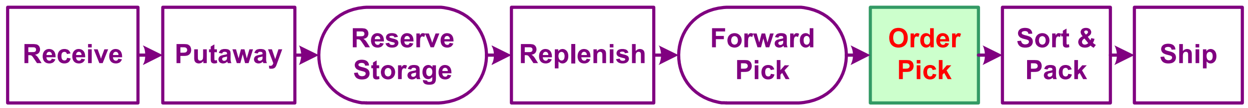



Pallet and Case Picking Area

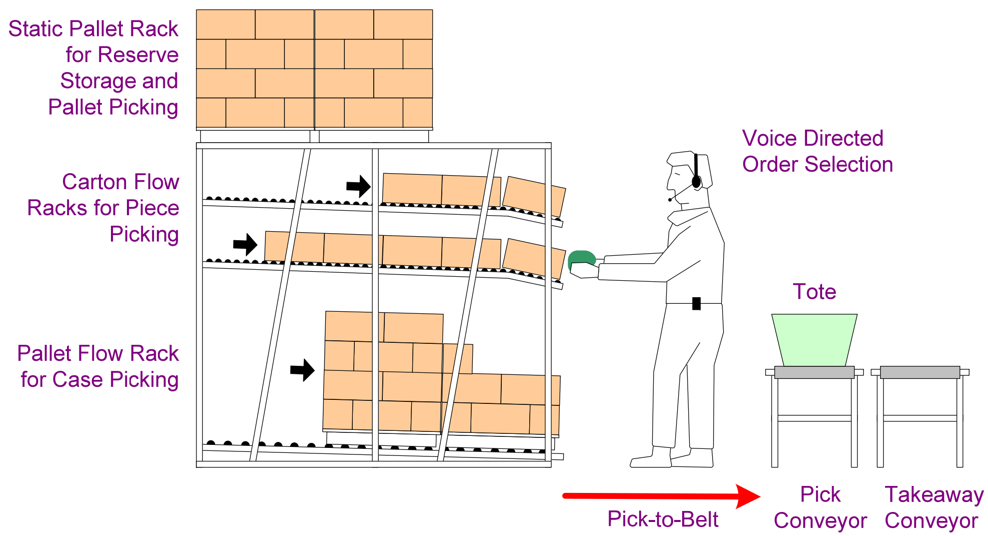



Forward Piece Picking Area

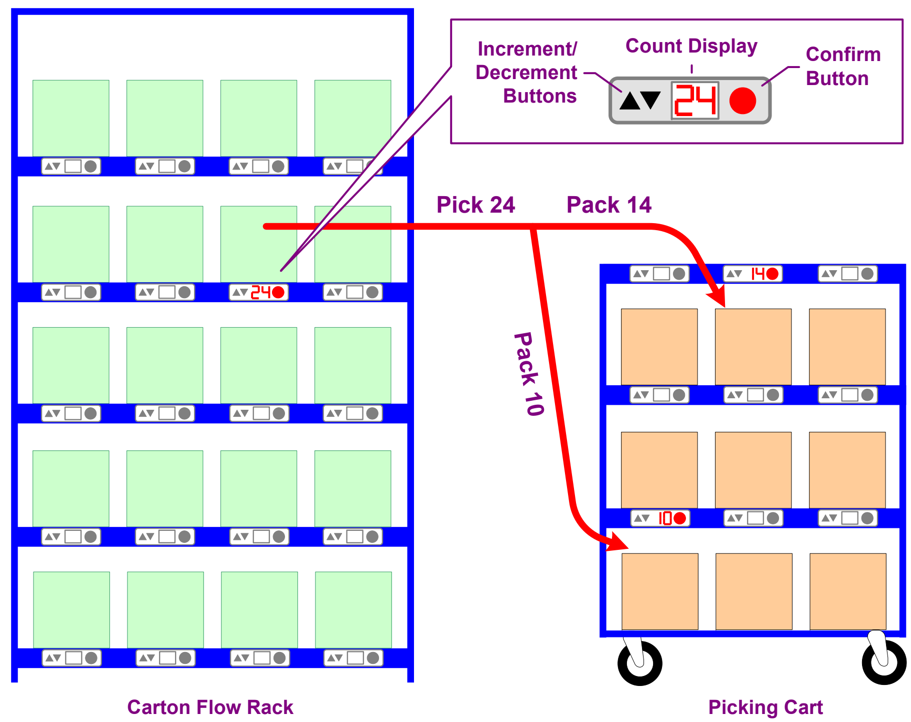
# Order Picking



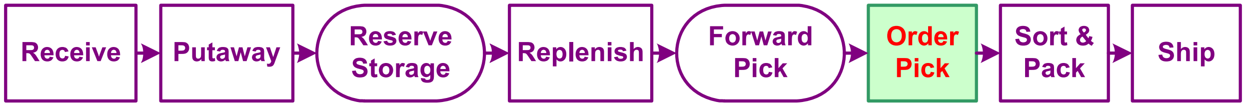

## Voice-Directed Piece and Case Picking




## Pick-to-Light Piece Picking



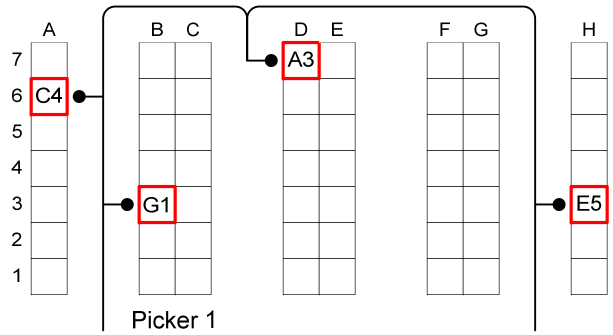
# Order Picking



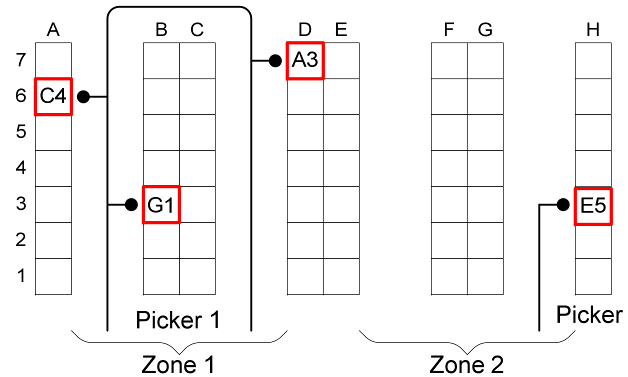

## Methods of Order Picking

| Method | Pickers per Order | Orders per Picker |
|---|---|---|
| Discrete | Single | Single |
| Zone | Multiple | Single |
| Batch | Single | Multiple |
| Zone-Batch | Multiple | Multiple |

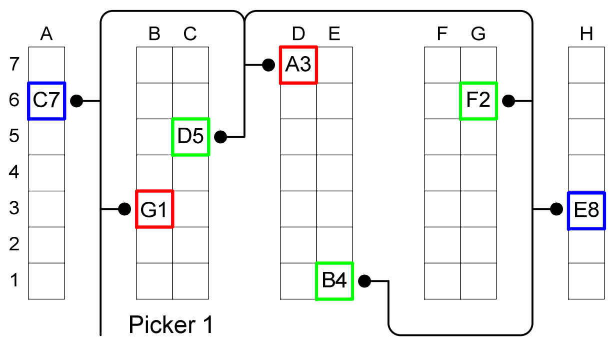
### Discrete



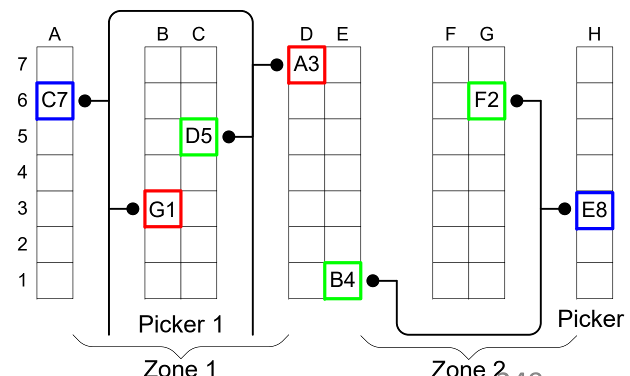

### Zone




### Batch

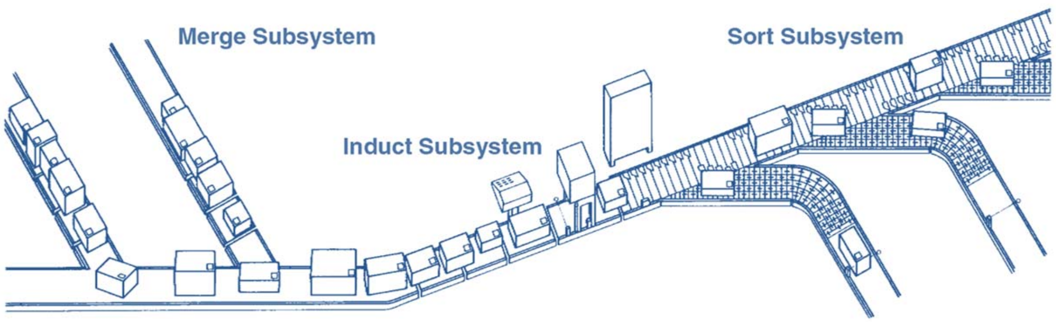



### Zone-Batch

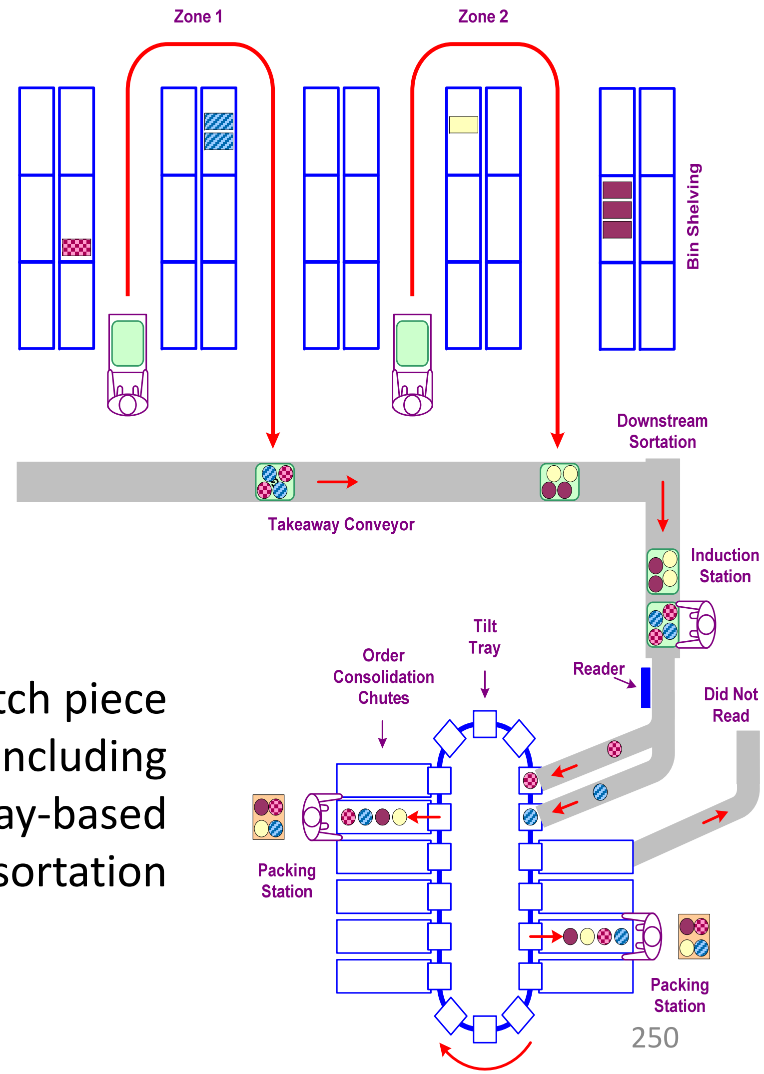


# Sortation and Packing

Receive → Putaway → Reserve Storage → Replenish → Forward Pick → Order Pick → **Sort & Pack** → Ship




Case Sortation System

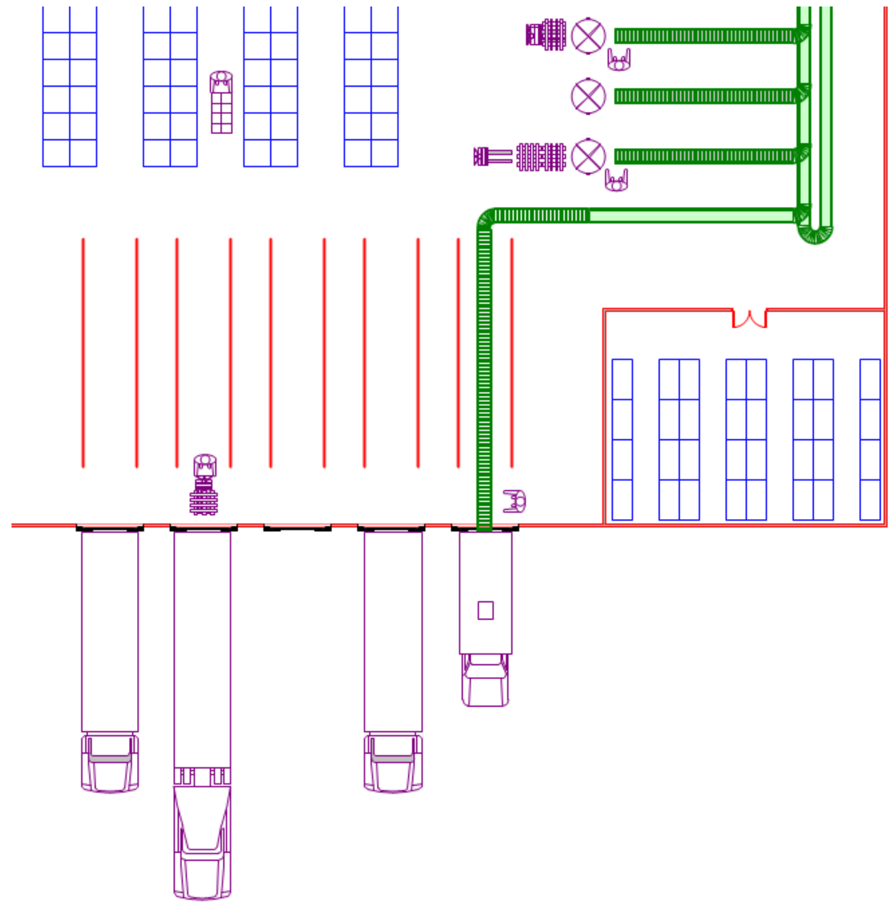



Wave zone-batch piece picking, including downstream tilt-tray-based sortation

# Shipping

Receive → Putaway → Reserve Storage → Replenish → Forward Pick → Order Pick → Sort & Pack → **Ship**

- Staging, verifying, and loading orders to be transported
  - ASN for each order sent to the customer
  - Customer-specific shipping instructions retrieved from customer master file
  - Carrier selection is made using the rate schedules contained in the carrier master file



Shipping Area

# Activity Profiling

- *Total Lines*: total number of lines for all items in all orders
- *Lines per Order*: average number of different items (lines/SKUs) in order
- *Cube per Order*: average total cubic volume of all units (pieces) in order
- *Flow per Item*: total number of S/R operations performed for item
- *Lines per Item* (*popularity*): total number of lines for item in all orders
- *Cube Movement*: total unit demand of item time x cubic volume
- *Demand Correlation*: percent of orders in which both items appear

**Customer Orders**

| Order: 1 | |
|---|---|
| SKU | Qty |
| A | 5 |
| B | 3 |
| C | 2 |
| D | 6 |

| Order: 2 | |
|---|---|
| SKU | Qty |
| A | 4 |
| C | 1 |

| Order: 3 | |
|---|---|
| SKU | Qty |
| A | 2 |

| Order: 4 | |
|---|---|
| SKU | Qty |
| B | 2 |

| Order: 5 | |
|---|---|
| SKU | Qty |
| C | 1 |
| D | 12 |
| E | 6 |

**Item Master**

| SKU | Length | Width | Depth | Cube | Weight |
|---|---|---|---|---|---|
| A | 5 | 3 | 2 | 30 | 1.25 |
| B | 3 | 2 | 4 | 24 | 4.75 |
| C | 8 | 6 | 5 | 180 | 9.65 |
| D | 4 | 4 | 3 | 32 | 6.35 |
| E | 6 | 4 | 5 | 120 | 8.20 |

**Total Lines = 11**

**Lines per Order = 11/5 = 2.2**

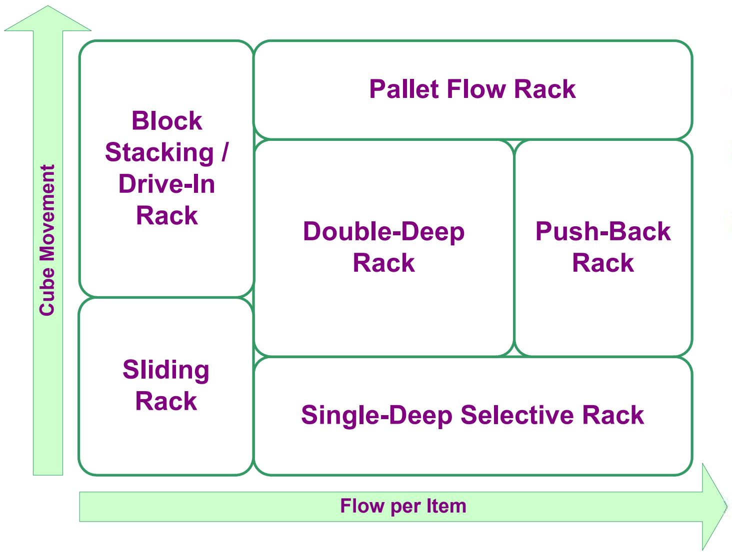
**Cube per Order = 493.2**

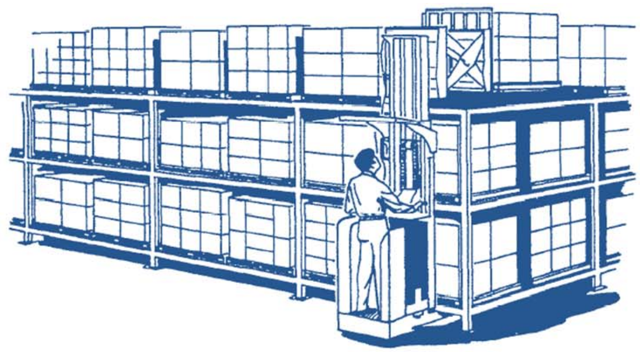
| SKU | Flow per Item | Lines per Item | Cube Movement |
|---|---|---|---|
| A | 11 | 3 | 330 |
| B | 5 | 2 | 120 |
| C | 4 | 3 | 720 |
| D | 18 | 2 | 576 |
| E | 6 | 1 | 720 |

**Demand Correlation Distibution**

| SKU | A | B | C | D | E |
|---|---|---|---|---|---|
| A | | 0.2 | 0.4 | 0.2 | 0.0 |
| B | | | 0.2 | 0.2 | 0.0 |
| C | | | | 0.4 | 0.2 |
| D | | | | | 0.2 |
| E | | | | | |

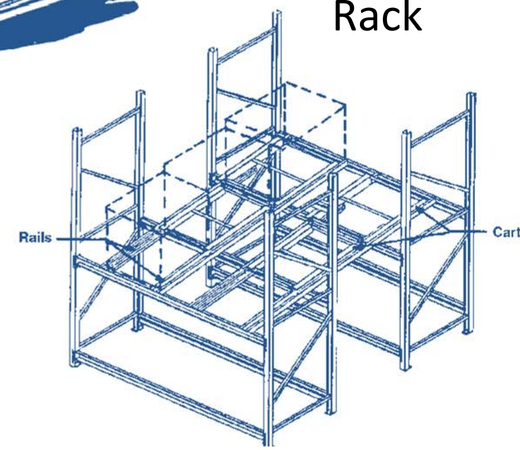
# Pallet Picking Equipment

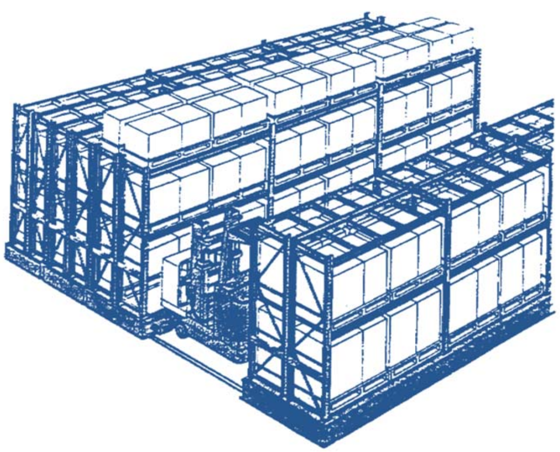


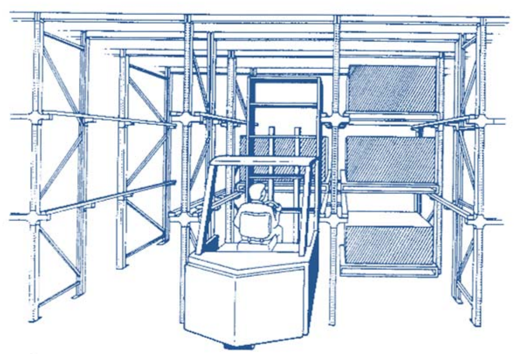
Double-Deep Rack

Push-Back Rack

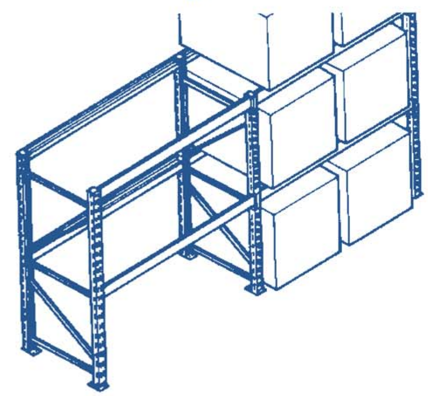


Sliding Rack

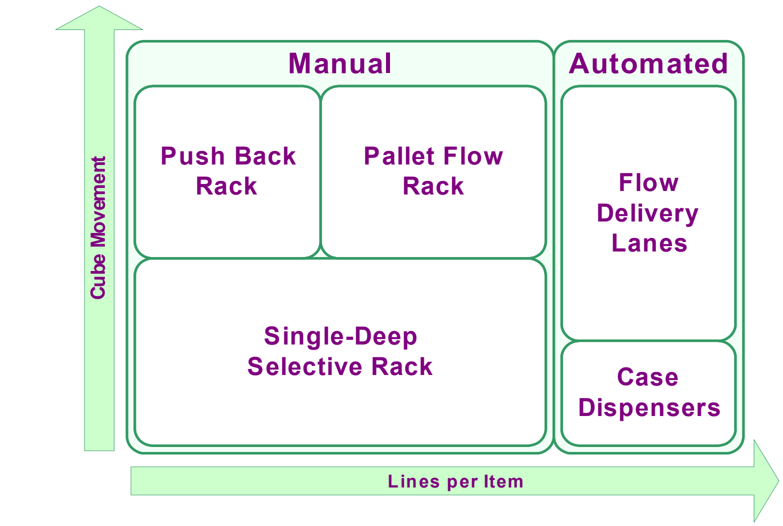


Drive-In Rack

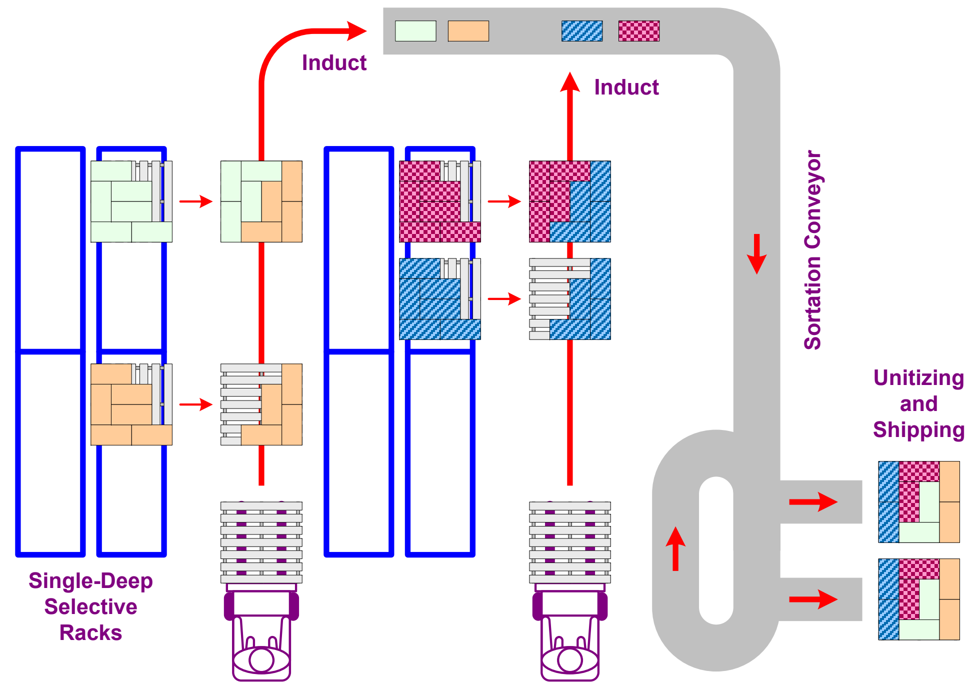


Single-Deep Selective Rack

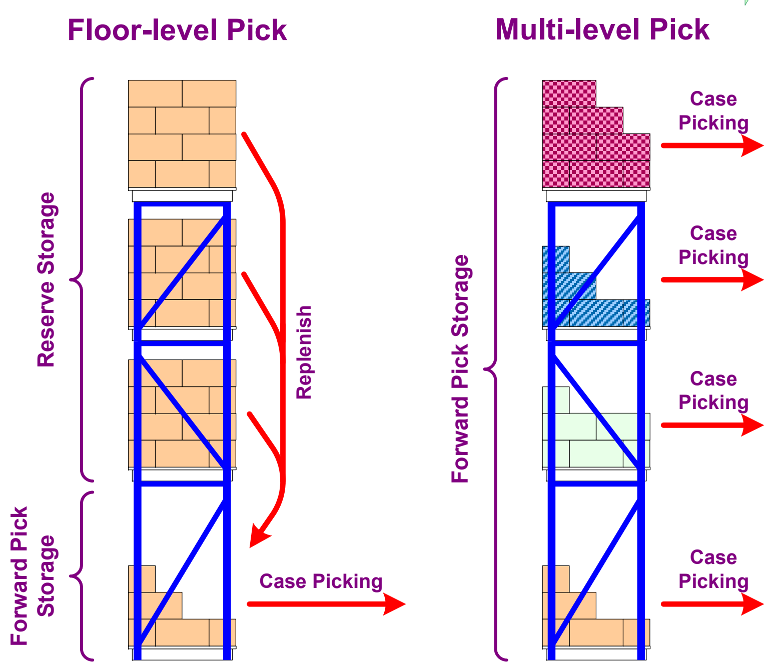
# Case Picking




Case Picking Equipment

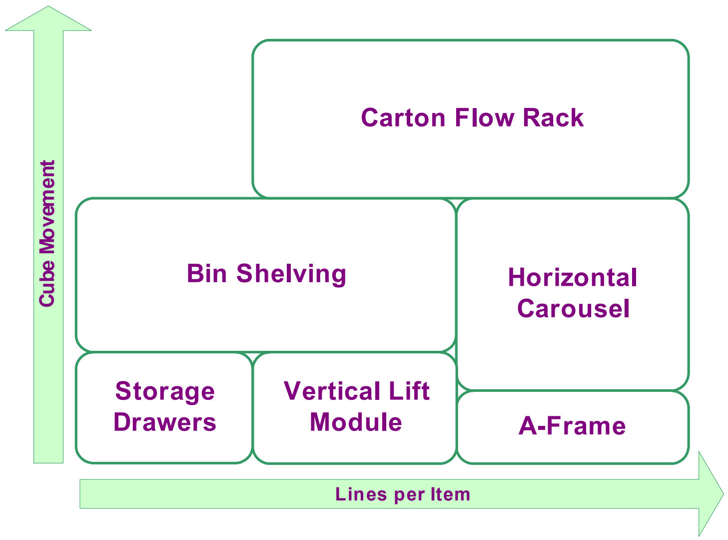


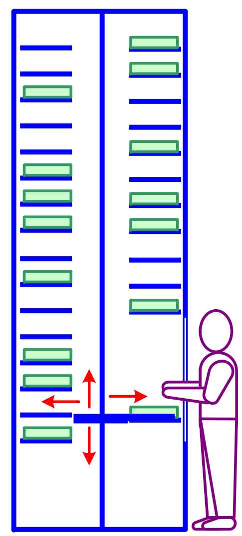

Zone-Batch Pick to Pallet



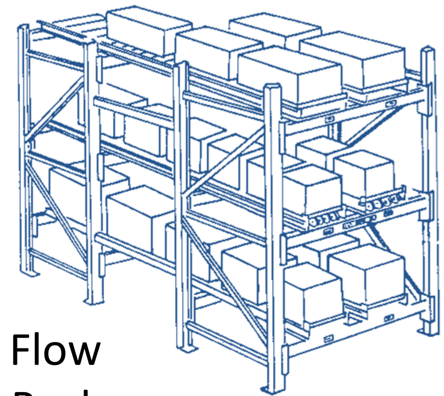

Floor- vs. Multi-level Pick to Pallet

# Piece Picking Equipment

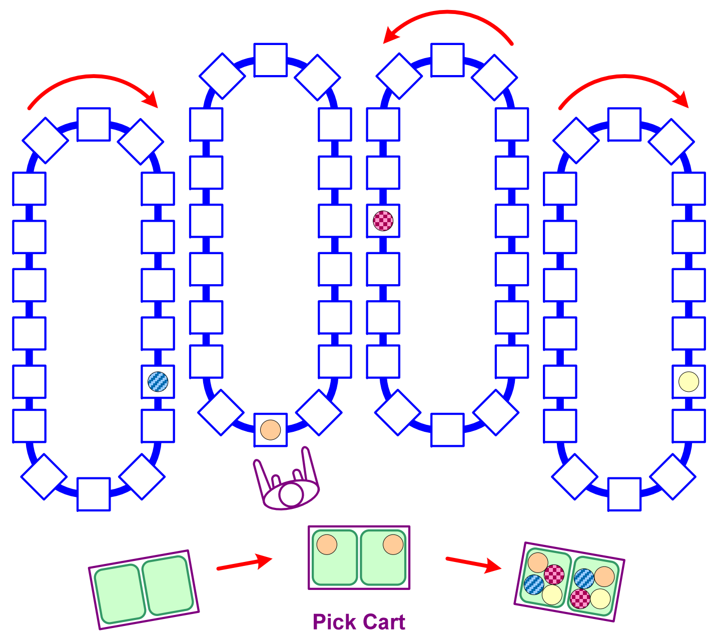


Vertical Lift Module

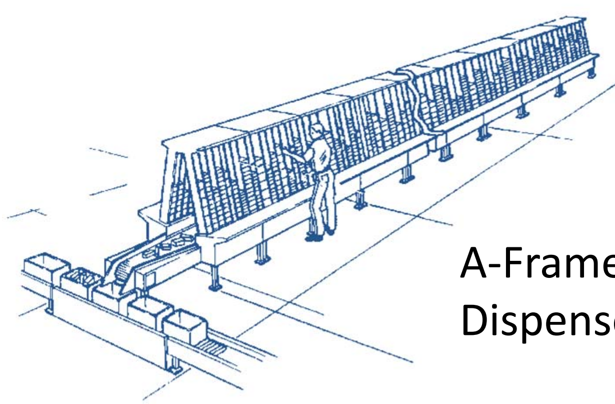


Carton Flow Rack



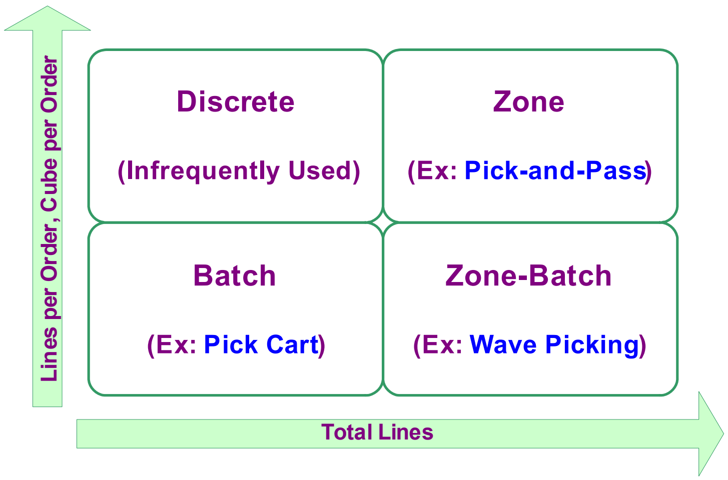

Carousel

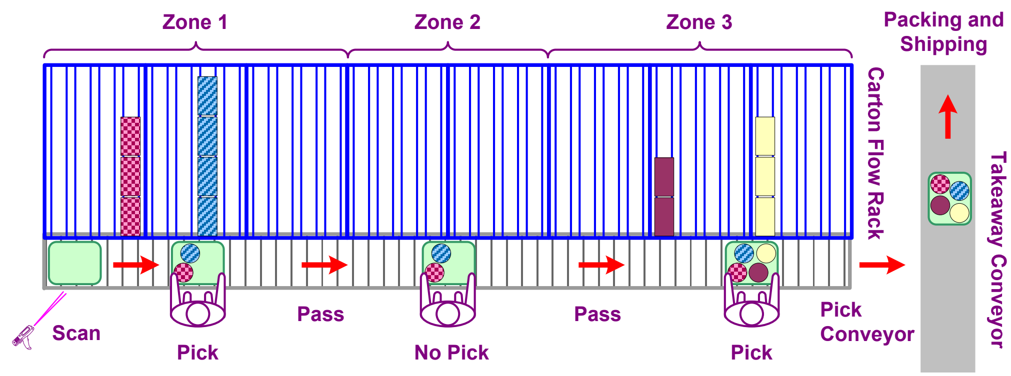
Drawers/Bins



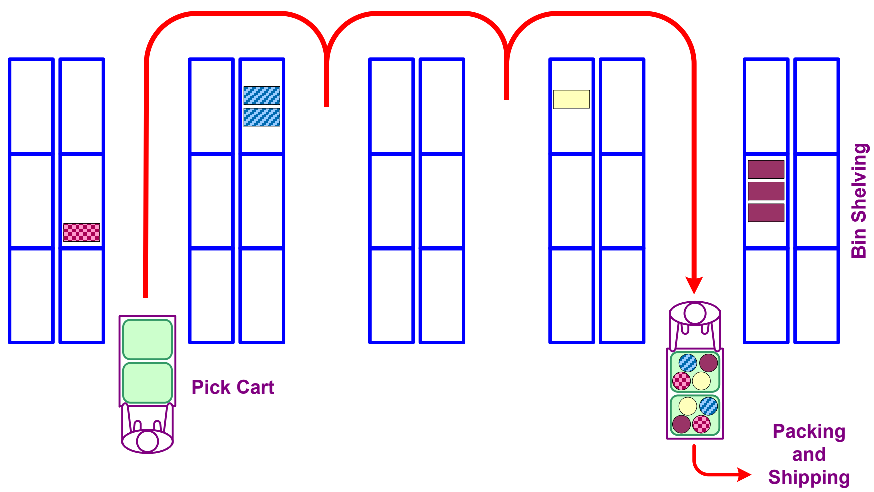
A-Frame Dispenser

# Methods of Piece Picking

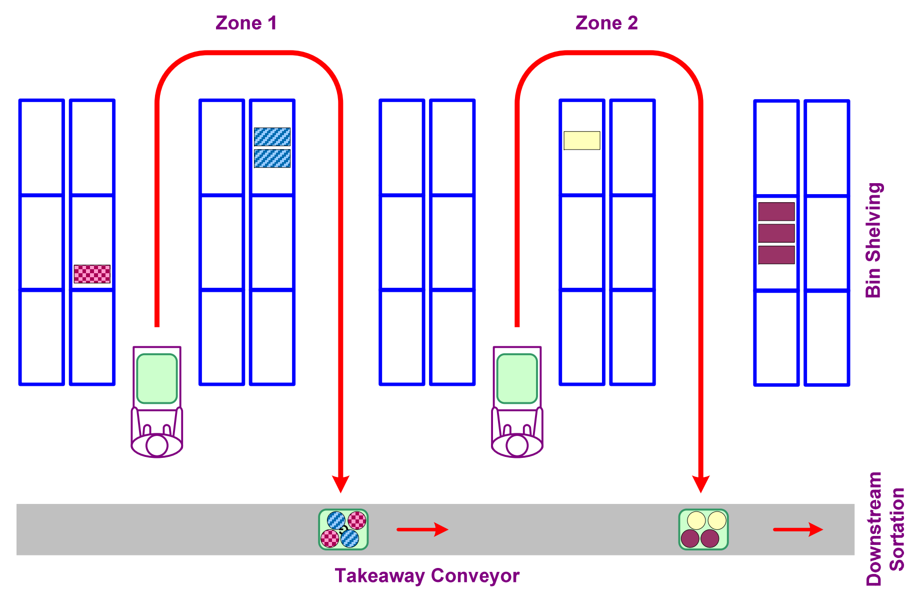



Pick-and-Pass Zone Piece Picking




Pick-cart Batch Piece Picking




Wave Zone-Batch Piece Picking

# Warehouse Automation

- Historically, warehouse automation has been a craft industry, resulting highly customized, one-off, high-cost solutions
- To survive, need to
  - adapt mass-market, consumer-oriented technologies in order to realize to economies of scale
  - replace mechanical complexity with software complexity
- How much can be spent for automated equipment to replace one material handler:

$$\$45,432\left(\frac{1-1.017^{-5}}{1-1.017^{-1}}\right)=\$45,432\left(4.83\right)=\$219,692$$

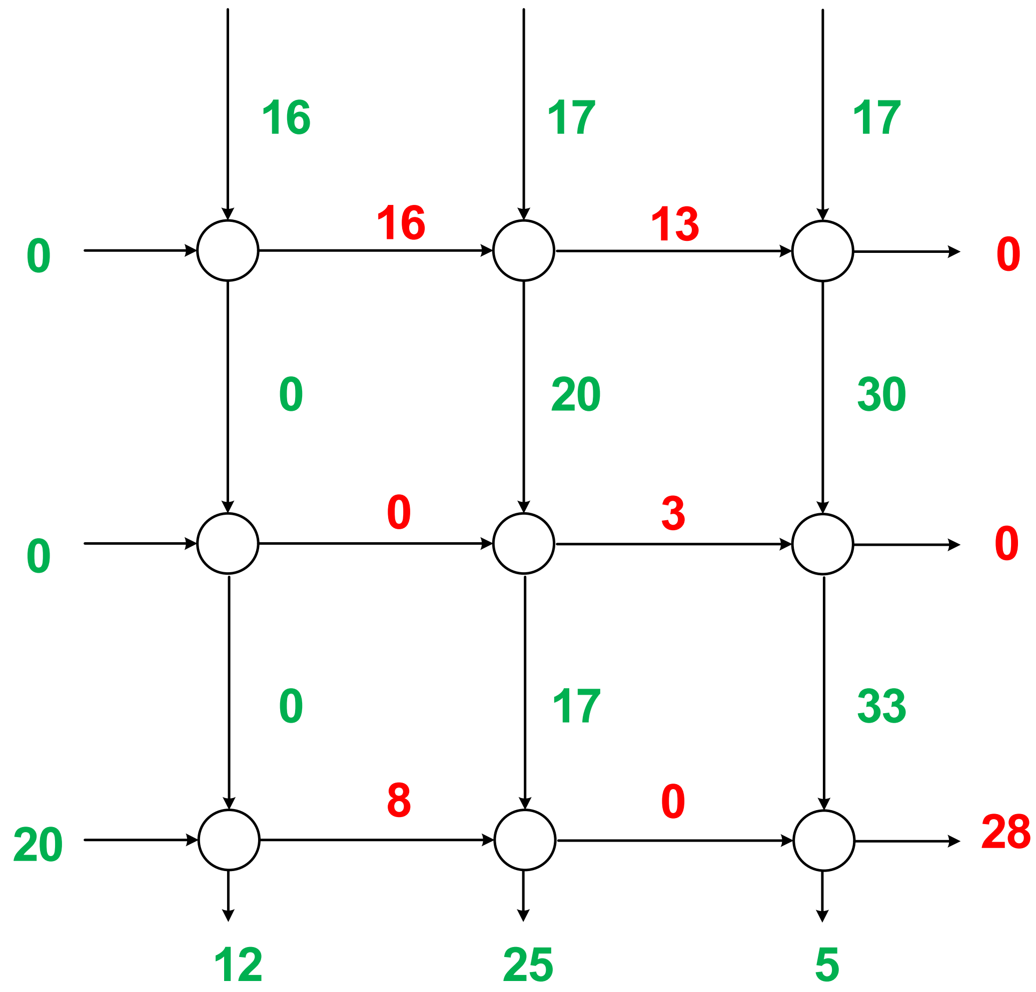
  - \$45,432: median moving machine operator annual wage + benefits
  - 1.7% average real interest rate 2005-2009 (real = nominal – inflation)
  - 5-year service life with no salvage (service life for Custom Software)

# KIVA Mobile-Robotic Fulfillment System

- Goods-to-man order picking and fulfillment system
- Multi-agent-based control
  - Developed by Peter Wurman, former NCSU CSC professor
- Kiva now called Amazon Robotics
  - purchased by Amazon in 2012 for $775 million

# Study Guide for Final Exam

2. Second material: infer all flows from given data



# Study Guide for Final Exam

4. How *minTLC* determines TLC for a route:

$$q_{agg} = \sqrt{\frac{f_{agg} \max\left\{r\,d_{agg}, MC\right\}}{\alpha v_{agg} h}}$$ (no truck capacity constraints, only min charge)

$$q_i = q_{agg} \frac{f_i}{f_{agg}}$$ (allocate based on demand)

$$s_{L_j} = \sum_{i \in L_j} f_i \Bigg/ \sum_{i \in L_j} \frac{f_i}{s_i}$$ (aggregate density of shipments in load-mix $L_j$)

$$k = \min_{L_j} \left\{ 1, \quad \frac{\min\left\{ K_{wt}, \dfrac{s_{L_j} K_{cu}}{2000} \right\}}{\sum_{i \in L_j} q_i} \right\}$$ (min ratio of max payload to size of shipment in load-mix)

$$q_i^* = k q_i$$ (apply truck capacity deduction factor)

$$TLC^* = \frac{f_{agg}}{\sum q_i^*} r\,d_{agg} + \alpha v_{agg} h \sum q_i^*$$ ($d_{agg}$ = distance of entire route)

# Study Guide for Final Exam

5. Order in which *twoopt* considers each sequence:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1: | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 38 |
| 2: | 1 | 3 | 2 | 4 | 5 | 6 | 1 | 39 |
| 3: | 1 | 4 | 3 | 2 | 5 | 6 | 1 | 32 |
| 4: | 1 | 3 | 4 | 2 | 5 | 6 | 1 | 31 |
| 5: | 1 | 4 | 3 | 2 | 5 | 6 | 1 | 32 |
| 6: | 1 | 2 | 4 | 3 | 5 | 6 | 1 | 31 |
| 7: | 1 | 5 | 2 | 4 | 3 | 6 | 1 | 21 |
| 8: | 1 | 2 | 5 | 4 | 3 | 6 | 1 | 21 |
| 9: | 1 | 4 | 2 | 5 | 3 | 6 | 1 | 32 |
| 10: | 1 | 3 | 4 | 2 | 5 | 6 | 1 | 31 |
| 11: | 1 | 5 | 4 | 2 | 3 | 6 | 1 | 12 |
| 12: | 1 | 4 | 5 | 2 | 3 | 6 | 1 | 34 |
| 13: | 1 | 2 | 4 | 5 | 3 | 6 | 1 | 40 |
| 14: | 1 | 3 | 2 | 4 | 5 | 6 | 1 | 39 |
| 15: | 1 | 5 | 2 | 4 | 3 | 6 | 1 | 21 |
| 16: | 1 | 5 | 3 | 2 | 4 | 6 | 1 | 30 |
| 17: | 1 | 5 | 6 | 3 | 2 | 4 | 1 | 31 |
| 18: | 1 | 5 | 4 | 3 | 2 | 6 | 1 | 13 |
| 19: | 1 | 5 | 4 | 6 | 3 | 2 | 1 | 18 |
| 20: | 1 | 5 | 4 | 2 | 6 | 3 | 1 | 20 |

Local optimal sequence

Sequences considered at end to verify local optimum: $n$ nodes $\Rightarrow$

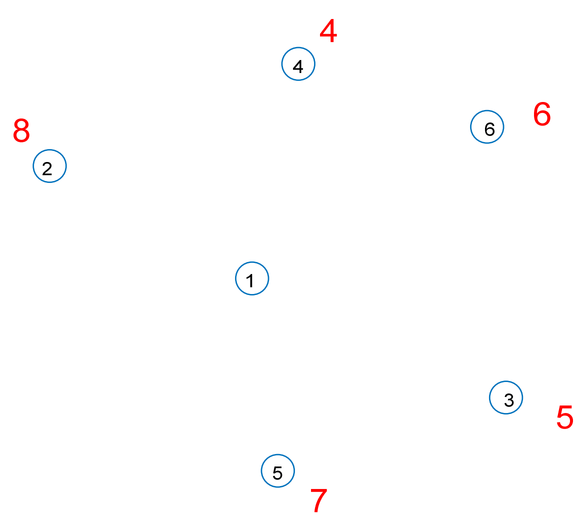
$$\sum_{i=1}^{n-2} \sum_{j=i+2}^{n-1} (1) = \frac{n^2 - 5n}{2} + 3 = 9 \text{ for } n = 6$$

# Study Guide for Final Exam

6. *vrpsavings* implements Clark-Wright procedure (truck capacity = 15)

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | 0 | 40 | 48 | 38 | 33 | 48 |
| 2 | 40 | 0 | 87 | 46 | 65 | 75 |
| 3 | 48 | 87 | 0 | 67 | 41 | 47 |
| 4 | 38 | 46 | 67 | 0 | 70 | 34 |
| 5 | 33 | 65 | 41 | 70 | 0 | 69 |
| 6 | 48 | 75 | 47 | 34 | 69 | 0 |

| $i$ | $j$ | $s_{ij}$ |
|---|---|---|
| 2 | 3 | $40+48-87=1$ |
| 2 | 4 | $40+38-46=32$ |
| 2 | 5 | 8 |
| 2 | 6 | 13 |
| 3 | 4 | 19 |
| 3 | 5 | 40 |
| 3 | 6 | 49 |
| 4 | 5 | 1 |
| 4 | 6 | 52 |
| 5 | 6 | 12 |



# Study Guide for Final Exam

9. $x = 3$, $y = 3$, $M = 100{,}000$, $N = 3600$, $D^* = 4$, $L = 6710$, $TA = 410{,}652$
10. $M = 210$, $D^* = 6$, $L\_ded = 18$, $TD\_ded = 3090$, $L\_rand = 11$, $TD\_rand = 3630$
11. $M = 50{,}000$, $D^* = 3$, $L = 4{,}195$, $TA = 163{,}605$, $TA = 188{,}146$, $d\_SC = 613.43$, $t\_SC = 2.00$

# Study Guide for Final Exam

13.

$$X = \sqrt{TA/2} = 31.62$$

$$X_D = \sqrt{2\left(TA_D / 2\right)} = 24.49$$

$$X'_C = \sqrt{TA_C} = 17.32 \Rightarrow X_C = X - X'_C$$

$$X'_A = \sqrt{TA_A + TA_C} = 24.49 \Rightarrow X_A = X - X'_A$$