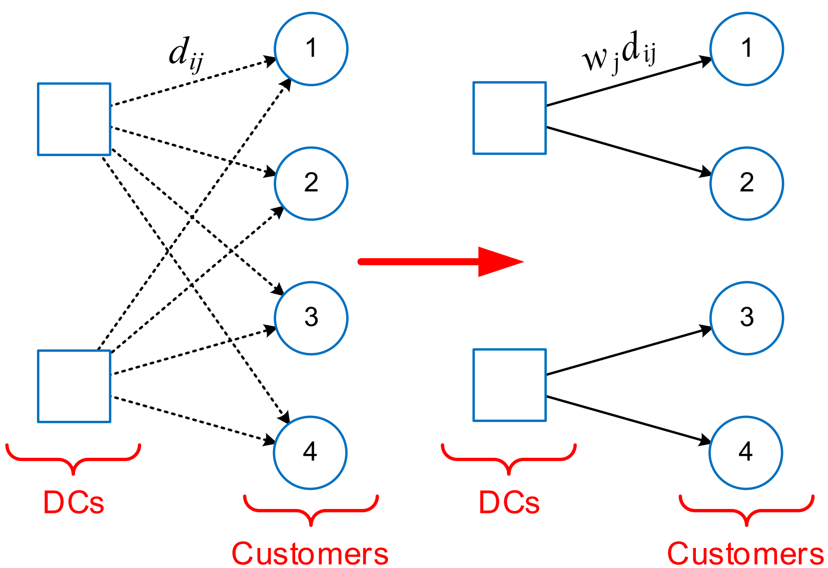
Allocation

- **Example:** given n DCs and m customers, with customer j receiving w_j TLs per week, determine the total distance per week assuming each customer is served by its *closest* DC



$$w = \begin{bmatrix} 2 & 4 & 6 & 8 \end{bmatrix}$$

$$D = \begin{bmatrix} 10 & 20 & 30 & 40 \\ 45 & 35 & 25 & 15 \end{bmatrix}$$

$$TD = 2(10) + 4(20) + 6(25) + 8(15)$$
$$= 370$$

Pseudocode

- Different ways of representing how allocation and TD can be calculated
  - High-level pseudocode most concise, but leaves out many implementation details (sets don't specify order, initial starting points)
  - Low-level pseudocode includes more implementation details, which can hide/obscure the core idea, and are usually not essential

Low-level Pseudocode

```
TD = 0
for j = 1:m
   dj = D(1,j)
   for i = 2:n
      if D(i,j) < dj
         dj = D(i,j)
      end
   end
   TD = TD + w(j)*dj
end
```

High-level Pseudocode

$$N = \{1,\ldots,n\}, \quad n = |N|$$

$$M = \{1,\ldots,m\}, \quad m = |M|$$

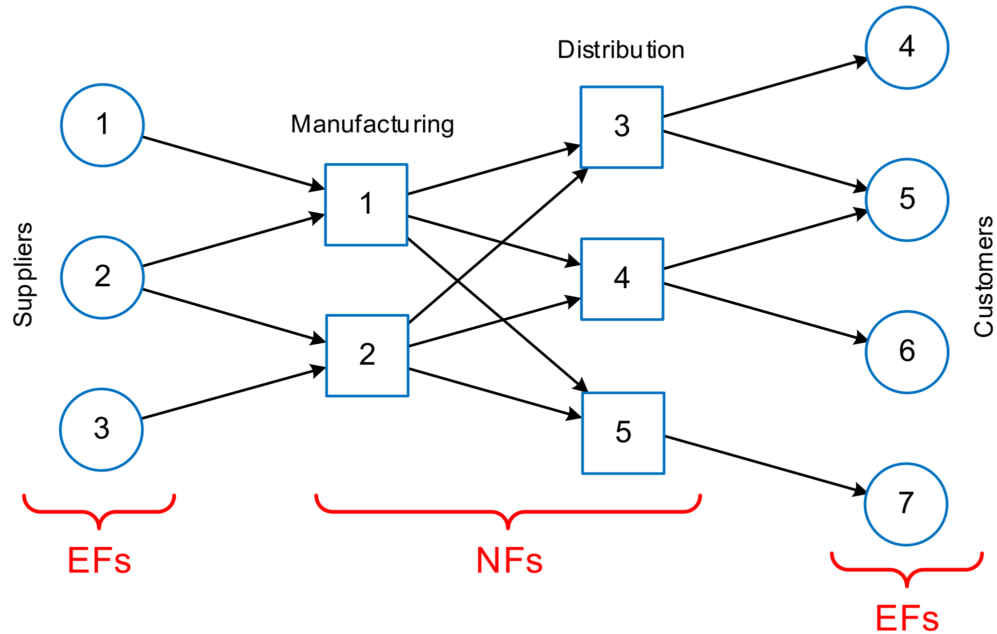
$$\alpha = [\alpha_j] = \arg\min_{i \in N} d_{ij}$$

$$TD = \sum_{j \in M} w_j d_{\alpha_j, j}$$

Matlab/Matlog

```
a = argmin(D);
W = sparse(a,1:m,w,n,m)
TD = sum(sum(W.*D))
```

Minisum Multifacility Location



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

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

$\mathbf{V}_{n \times n} =$

NF-NF	1	2	3	4	5
1	0	0	+	+	+
2	0	0	+	+	+
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0

$\mathbf{W}_{n \times m} =$

NF-EF	1	2	3	4	5	6	7
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	+

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

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

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

Majority Theorem for Minisum Location

- Single-facility: Locate NF at EFj if $w_j \geq \frac{W}{2}$, where $W = \sum_{i=1}^{m} w_i$
- Multifacility: can be used to reduce and sometimes solve

Given m EF and n NF, let $\mathbf{V} \leftarrow \mathbf{V} + \mathbf{V}'$

1. While any $v_{ik} \geq \frac{1}{2}\left(\sum_{j=1}^{m} w_{ij} + \sum_{j=1}^{n} v_{ij} \right)$, co-locate NF$i$ and NFk and

   (a) add row k to row i of $\mathbf{W}$, remove row k from $\mathbf{W}$

   (b) add row k to row i and column k to column i of $\mathbf{V}$

   (c) remove row k and column k from $\mathbf{V}$, and set $0 \leftarrow v_{ii}$

2. Locate all NFi at EFk if $w_{ik} \geq \frac{1}{2}\left(\sum_{j=1}^{m} w_{ij} + \sum_{j=1}^{n} v_{ij} \right)$,

   where any NFj co-located with NFi are also located at EFk.

Ex 6: Multifacility Majority Theorem

3 EF, 2 NF, $\mathbf{V} = \begin{bmatrix} 0 & 2 \\ 0 & 0 \end{bmatrix} \Rightarrow \mathbf{V} = \begin{bmatrix} 0 & 2 \\ 2 & 0 \end{bmatrix}$:

1. No solutions or reductions possible

$$[\mathbf{W}\ \mathbf{V}] = \left[\begin{array}{ccc:cc} 2 & 1 & 0 & 0 & 2 \\ 4 & 0 & 5 & 2 & 0 \end{array}\right] \begin{array}{l} \sum = 5 \\ \sum = 11 \end{array}$$

2. Modified $\mathbf{V} \Rightarrow$ solution

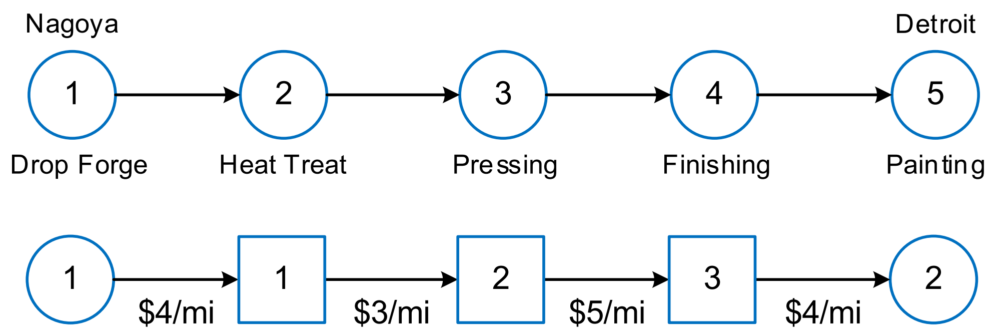
$$[\mathbf{W}\ \mathbf{V}] = \left[\begin{array}{ccc:cc} 2 & 1 & 0 & 0 & 0.5 \\ 4 & 0 & 5 & 0.5 & 0 \end{array}\right] \begin{array}{l} \sum = 3.5 \Rightarrow w_{1,1} = 2 > 1.75 \Rightarrow \text{ NF1 at EF1} \\ \sum = 9.5 \Rightarrow w_{2,3} = 5 > 4.75 \Rightarrow \text{ NF2 at EF3} \end{array}$$

3. Modified $\mathbf{V} \Rightarrow$ reduction $\Rightarrow$ solution

$$[\mathbf{W}\ \mathbf{V}] = \left[\begin{array}{ccc:cc} 2 & 1 & 0 & 0 & 4 \\ 4 & 0 & 5 & 4 & 0 \end{array}\right] \begin{array}{l} \sum = 7 \Rightarrow v_{1,2} = 4 > 3.5 \Rightarrow \text{ NF1 and NF2 co-located} \\ \sum = 13 \Rightarrow \text{all } w_{ij}, v_{ij} < 6.5 \end{array}$$

Reduced $\mathbf{W}$, no $\mathbf{V}$: $\mathbf{W} = \begin{bmatrix} 6 & 1 & 5 \end{bmatrix} \sum = 12 \Rightarrow w_1 = 6 \geq 6 \Rightarrow$ NF1 (and NF2) at EF1

Ex 7: Location of Production Processes

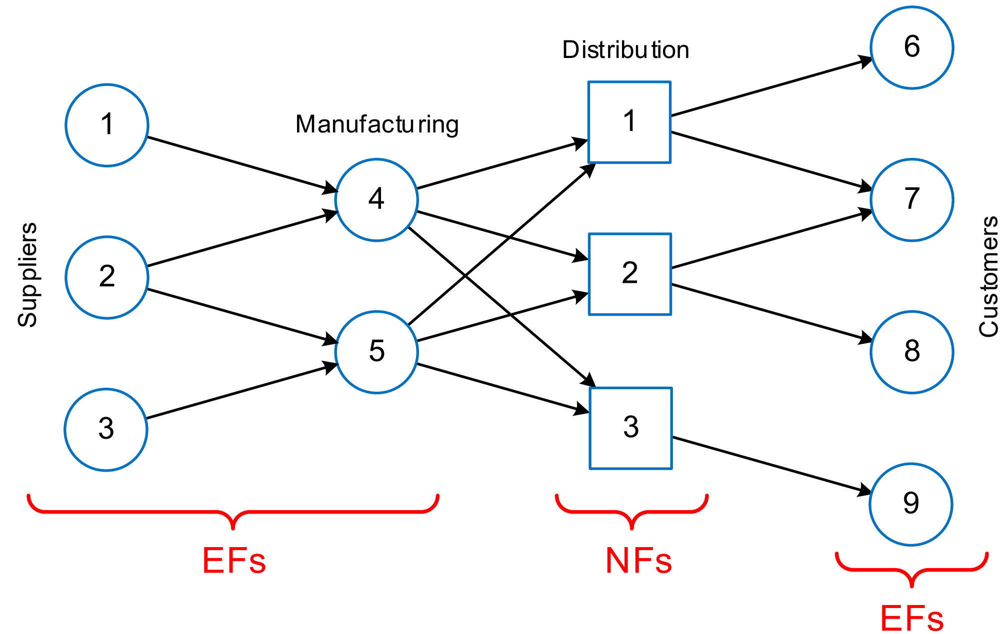


$$2 \text{ EF, } 3 \text{ NF, } \mathbf{W} = \begin{bmatrix} 4 & 0 \\ 0 & 0 \\ 0 & 4 \end{bmatrix}, \mathbf{V} = \begin{bmatrix} 0 & 3 & 0 \\ 0 & 0 & 5 \\ 0 & 0 & 0 \end{bmatrix} \Rightarrow \mathbf{V} = \begin{bmatrix} 0 & 3 & 0 \\ 3 & 0 & 5 \\ 0 & 5 & 0 \end{bmatrix}$$

$$1.\ [\mathbf{W}\ \mathbf{V}] = \left[\begin{array}{cc:ccc} 4 & 0 & 0 & 3 & 0 \\ 0 & 0 & 3 & 0 & 5 \\ 0 & 4 & 0 & 5 & 0 \end{array}\right] \sum = 8 \Rightarrow v_{2,3} = 5 > 4 \Rightarrow \text{NF2 and NF3 co-located}$$

$$2.\ [\mathbf{W}\ \mathbf{V}] = \left[\begin{array}{cc:cc} 4 & 0 & 0 & 3 \\ 0 & 4 & 3 & 0 \end{array}\right] \begin{array}{l} \sum = 7 \Rightarrow w_{1,1} = 4 > 3.5 \Rightarrow \text{NF1 at EF1} \\ \sum = 7 \Rightarrow w_{2,2} = 4 > 3.5 \Rightarrow \text{NF2 (and NF3) at EF2} \end{array}$$

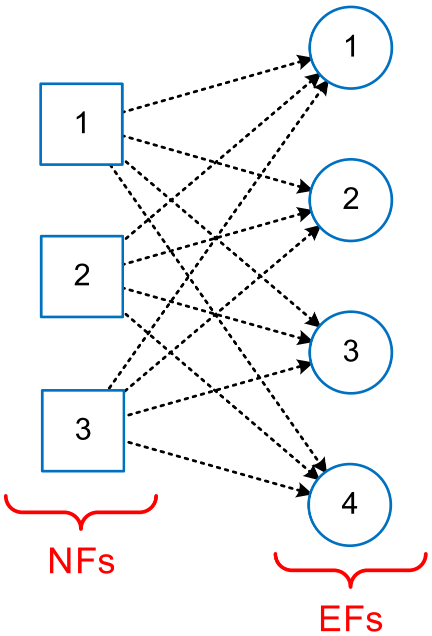
Multiple Single-Facility Location



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

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

Facility Location–Allocation Problem



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

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

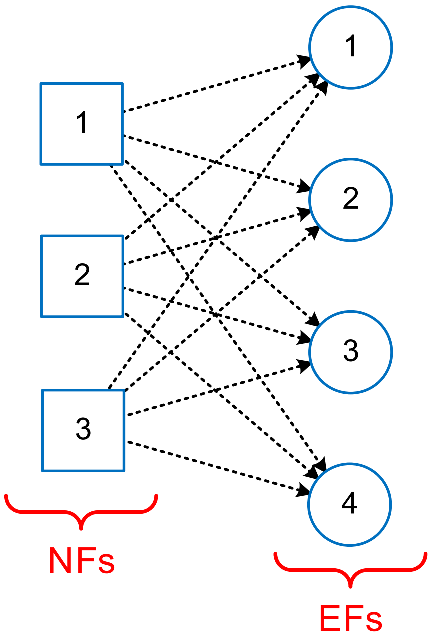
$$w_i = \text{total flow requirememts of EF}i$$

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

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

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

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

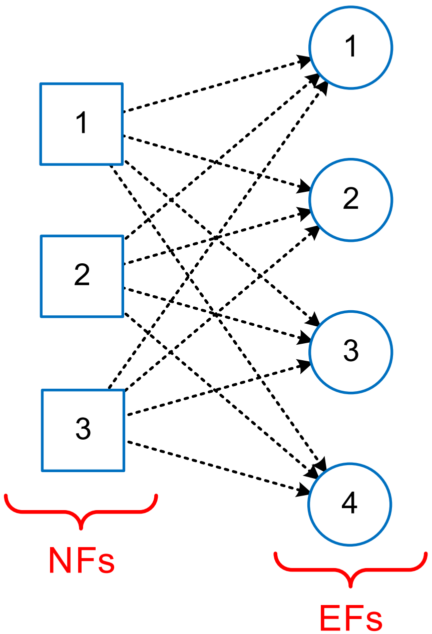
$$\alpha_i(\mathbf{X}) = \arg\min_j d(\mathbf{X}_j, \mathbf{P}_i)$$

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

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

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

Alternating Formulation



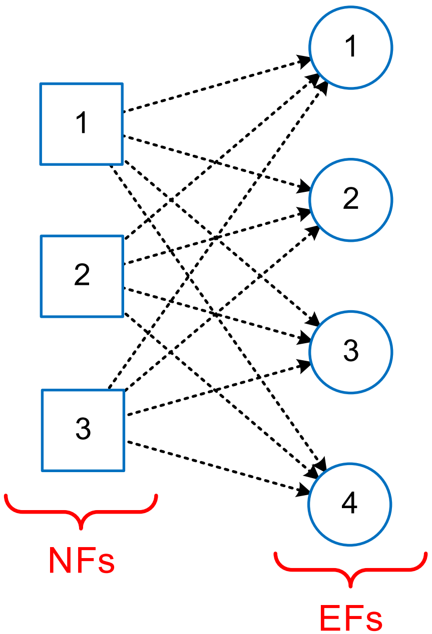
- Alternate between finding locations and finding allocations until no further TC improvement
- Requires *n* *d*-dimensional location searches together with separate allocation procedure
- Separating location from allocation allows other types of location and/or allocation procedures to be used:
  - 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 Meridian, MS	Palmdale; CA	Chicago, IL
5	1.13	Madison, NJ Dallas, TX	Palmdale, CA Macon, GA	Chicago, IL
6	1.08	Madison, NJ Dallas, TX	Pasadena, CA Macon, GA	Chicago, IL Tacoma, WA
7	1.07	Madison, NJ Dallas, TX Lakeland, FL	Pasadena, CA Gainesville, GA	Chicago, IL Tacoma, WA
8	1.05	Madison, NJ Dallas, TX Lakeland, FL	Pasadena, CA Gainesville, GA Denver, CO	Chicago, IL Tacoma, WA
9	1.04	Madison, NJ Dallas, TX Lakeland. FL	Alhambra, CA Gainesville, GA Denver, CO	Chicago, IL Tacoma, WA Oakland, CA
10	1.04	Newark, NJ Palistine, TX Lakeland, FL Mansfield, OH	Alhambra, CA Gainesville, GA Denver, CO	Rockford, IL Tacoma, WA Oakland. CA