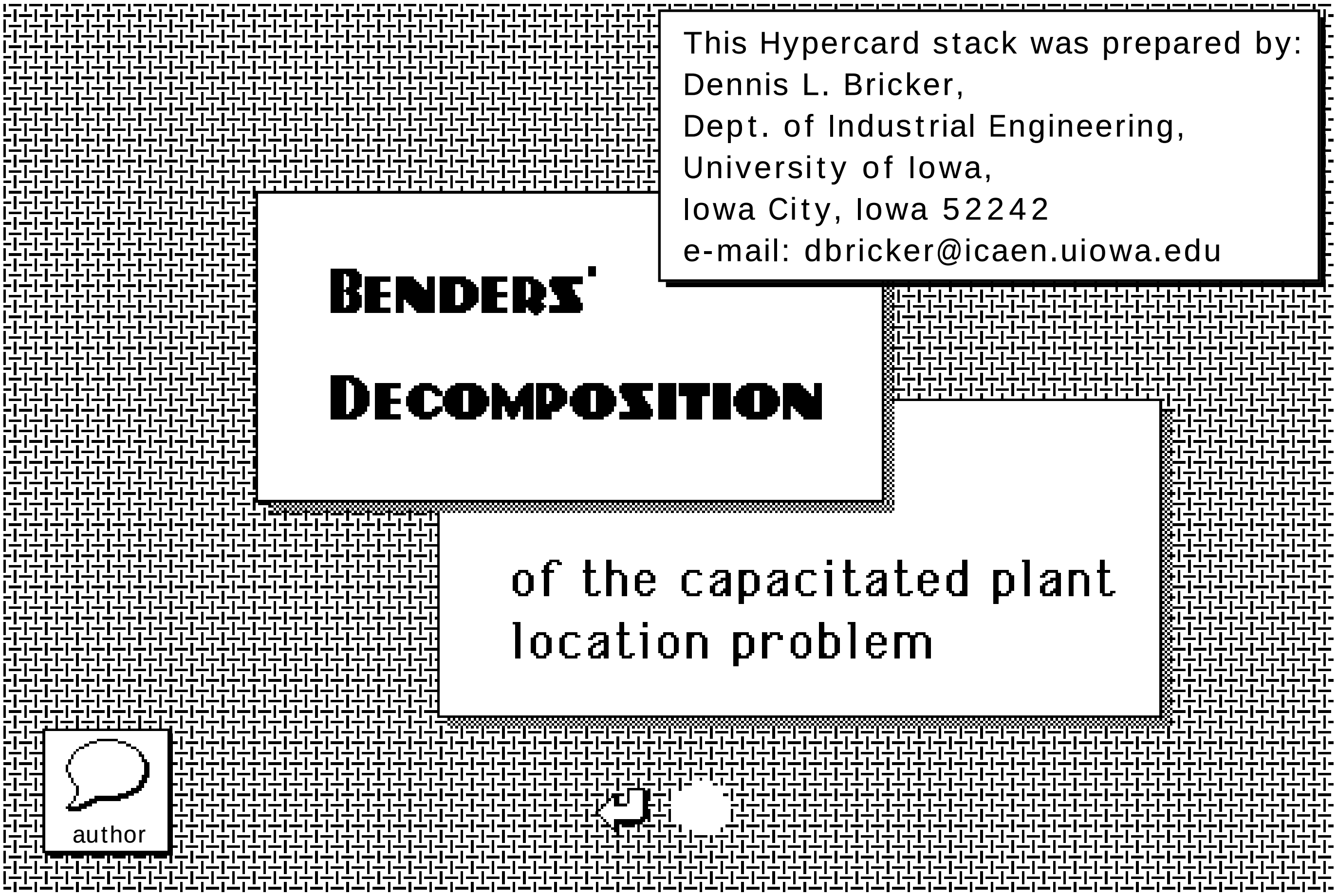


BENDERS' DECOMPOSITION

of the capacitated plant location problem

This Hypercard stack was prepared by:
Dennis L. Bricker,
Dept. of Industrial Engineering,
University of Iowa,
Iowa City, Iowa 52242
e-mail: dbricker@icaen.uiowa.edu

The Problem

Given: a set of **N** demand points, with

D_j = annual demand of customer #**j**

a set of M potential plant sites, with

S_i = annual capacity of plant #**i** (if built)

F_i = annual fixed cost of building & operating plant #**i**

C_{ij} = unit cost of production at plant #**i**, plus cost of shipping to customer #**j**

- Which plant(s) should be built? *"location"*
- Which customers should be supplied by each plant? *"allocation"*

Define the variables:

X_{ij} = annual quantity shipped from plant #i to customer #j

$$Y_i = \begin{cases} 1 \text{ if a plant is built at site \#i} \\ 0 \text{ otherwise} \end{cases}$$

The mathematical model

$$\text{Minimize} \quad \sum_{i=1}^{M} F_i Y_i + \sum_{i=1}^{M} \sum_{j=1}^{N} C_{ij} X_{ij}$$

subject to:

$$\sum_{j=1}^{N} X_{ij} \leq S_i Y_i \quad \textbf{for all i}$$

if no plant is built at site #i, the total shipments from site #i must be zero!

$$\sum_{i=1}^{M} X_{ij} \geq D_j \quad \textbf{for all j}$$

$$X_{ij} \geq 0, \quad Y_i \, \varepsilon \, \{0,1\} \ \textbf{for all i,j}$$

⇦ ⇨

Notice that If we had selected values for each variable Y_i, the problem of selecting X_{ij} is the classical

transportation problem!

⇦ ⇨

Define an optimal value function of this transportation problem:

$$V(Y) = \sum_{i=1}^{M} F_i Y_i + \textbf{minimum} \sum_{i=1}^{M} \sum_{j=1}^{N} C_{ij} X_{ij}$$

$$\text{s.t.} \quad \sum_{j=1}^{N} X_{ij} \le S_i Y_i \quad \textbf{for all i}$$

$$\sum_{i=1}^{M} X_{ij} \ge D_j \quad \textbf{for all j}$$

$$X_{ij} \ge 0$$

⇦ ⇨

That is, given a value for each Y_i, indicating whether a plant is to be built there, you can then solve a transportation problem to determine the quantities to be shipped from each of the plants to each customer.

⇦ ⇨

The total annual fixed cost of the plants, plus the optimal transportation costs, is the value of the function V at the point Y.

⇦ ⇨

Our original problem is therefore equivalent to

Minimize **v(Y)**

Unfortunately, the function V is difficult to characterize!

⇦ ⇨

By Linear Programming duality theory, the optimal value of the transportation problem is equal to that of its ***dual*** *LP:*

$$v(Y) = \sum_{i=1}^{M} F_i Y_i + \text{maximum} \sum_{i=1}^{M} S_i Y_i u_i + \sum_{j=1}^{N} D_j v_j$$

$$\text{s.t.} \quad u_i + v_j \leq C_{ij} \quad \forall\ i \& j$$

$$u_i \geq \qquad v_j \geq 0$$

⇦ ⇨

Suppose that all the basic solutions of the dual LP are enumerated, with $(\hat{u}^k, \hat{v}^k)$ *denoting basic solution number k. Then v(Y) might be computed by evaluating the dual objective at each extreme point, and selecting that producing the largest value:*

$$v(Y) = \sum_{i=1}^{M} F_i Y_i + \underset{k}{\text{maximum}} \left\{ \sum_{i=1}^{M} S_i Y_i \hat{u}_i^k + \sum_{j=1}^{N} D_j \hat{v}_j^k \right\}$$

⇦ ⇨

Define, for each dual basic solution $(\hat{\mathbf{u}}^{\mathbf{k}}, \hat{\mathbf{v}}^{\mathbf{k}})$,

$$\alpha_i^k = \mathbf{F}_i + \mathbf{S}_i \hat{\mathbf{u}}_i^k$$

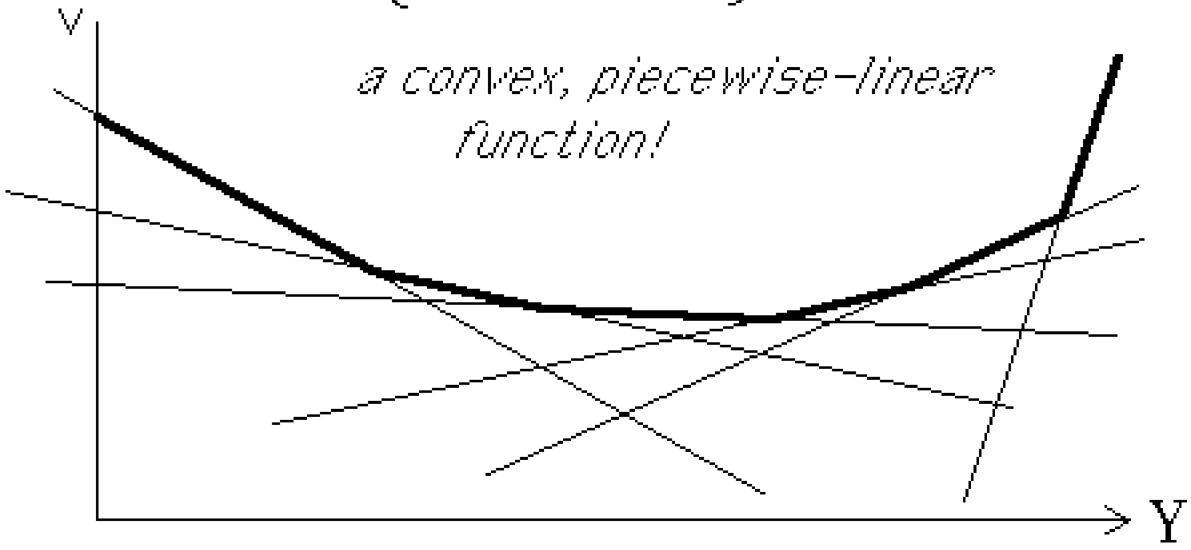
$$\beta^k = \sum_{j=1}^{N} \mathbf{D}_j \hat{\mathbf{v}}_j^k$$

so that $v(Y) = \text{maximum}\left\{ \sum_{i=1}^{M} \alpha_i^k \mathbf{Y}_\mathbf{i} + \beta^k \right\}$

Thus, v(Y) is the maximum of a large number of linear functions of Y.

⇦ ⇨

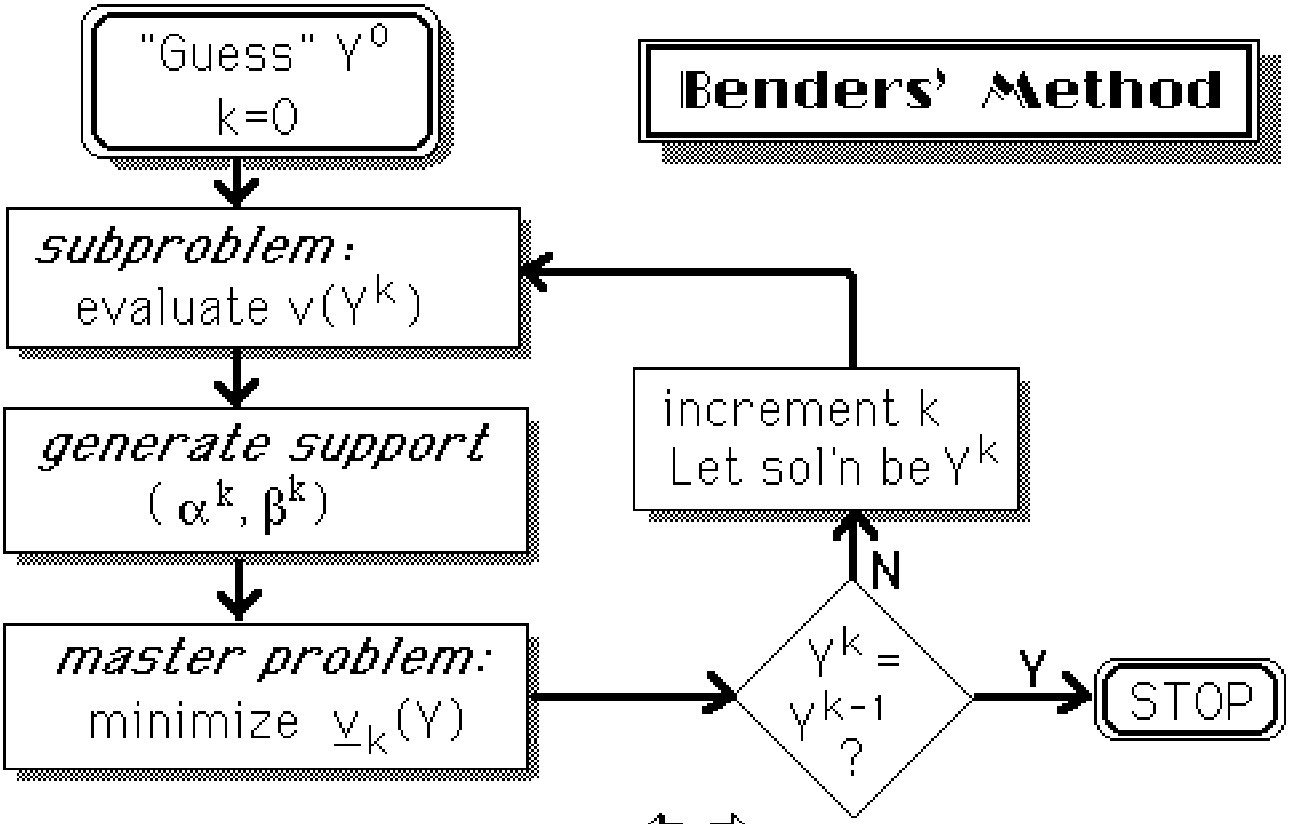
$$v(Y) = \text{maximum}\left\{ \sum_{i=1}^{M} \alpha_i^k \mathbf{Y}_\mathbf{i} + \beta^k \right\}$$



a convex, piecewise-linear function!

⇦ ⇨

Benders' Method



"Guess" Y^0
$k=0$

subproblem:
evaluate $v(Y^k)$

generate support
(α^k, β^k)

master problem:
minimize $\underline{v}_k(Y)$

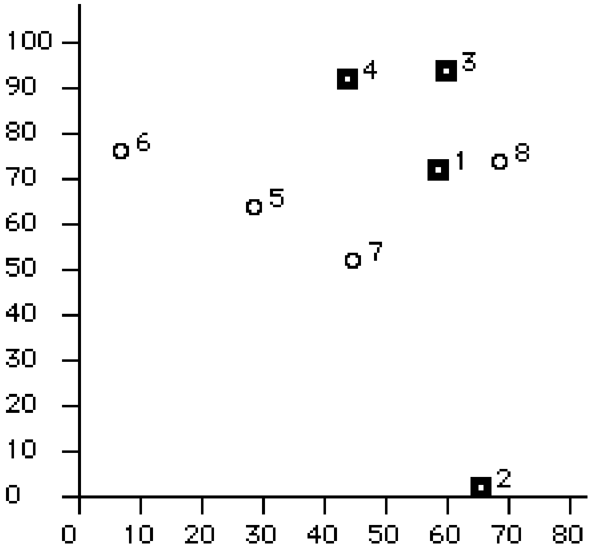
$Y^k = Y^{k-1}$?

Y: STOP

N: increment k
Let sol'n be Y^k

Example

Plant Location Problem



4 plant sites,
8 demand points

(4 demand points
are also potential
plant sites)

▣ plant site & demand pt

○ demand pt only

Random Problem (Seed = 94294)

Number of sources = M = 4
Number of destinations = N = 8
Total demand: 29

Costs, Supplies, Demands

i\j	1	2	3	4	5	6	7	8	K	F
1	0	70	22	25	31	52	24	10	13	300
2	70	0	92	93	72	95	54	72	13	400
3	22	92	0	16	43	56	45	22	10	250
4	25	93	16	0	32	40	40	31	9	200
Demand	4	2	10	5	1	1	5	1	45	

K = capacity,
F = fixed cost

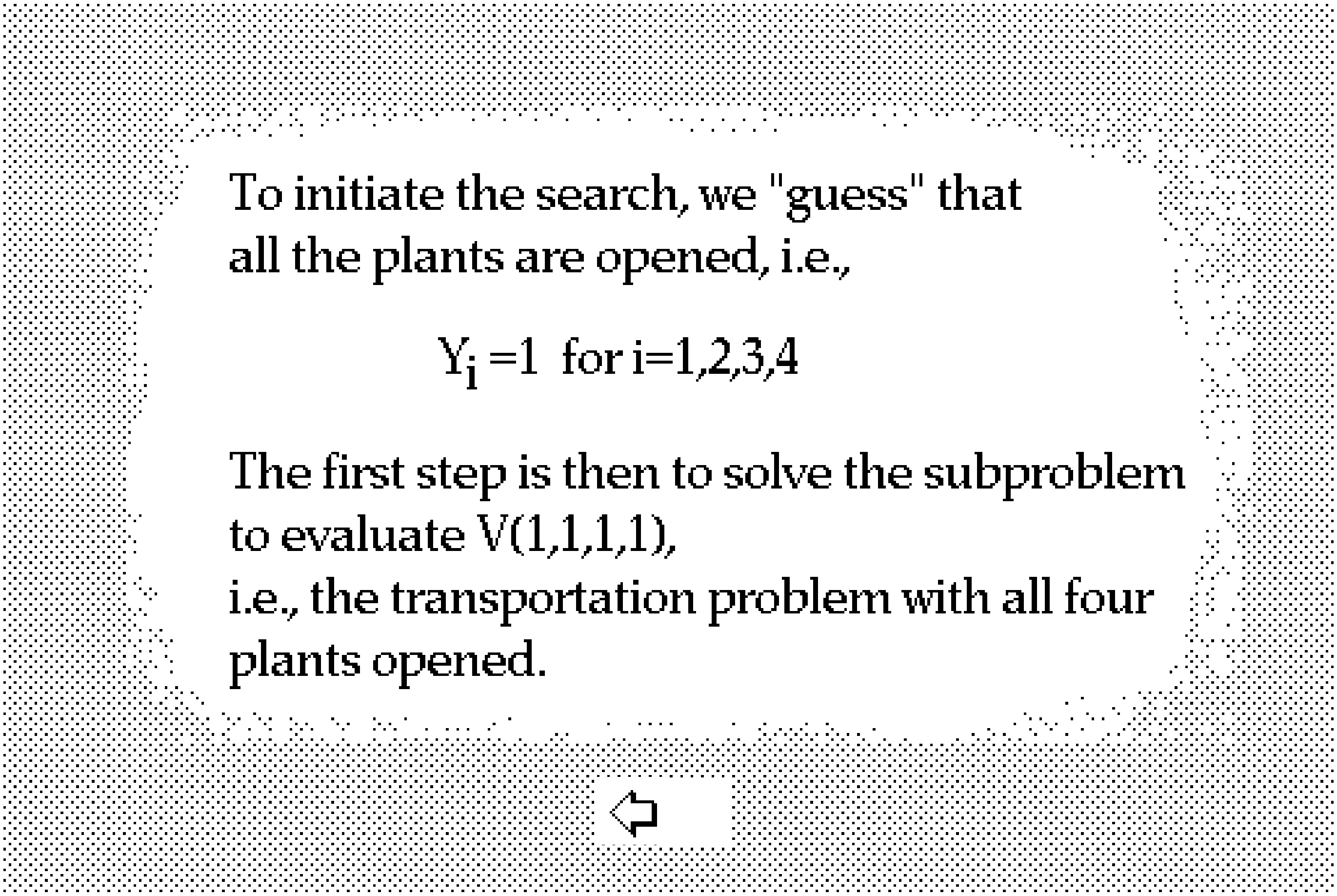
Options

☞ Optimizing the Master Problem

Each Master Problem minimizes v(Y), requiring a complete search of the enumeration tree.

☞ Suboptimizing the Master Problem

A solution Y with v(Y) < incumbent is found by the Master Problem; only one "pass" through the enumeration tree is required.



To initiate the search, we "guess" that all the plants are opened, i.e.,

$$Y_i = 1 \quad \text{for } i=1,2,3,4$$

The first step is then to solve the subproblem to evaluate V(1,1,1,1),
i.e., the transportation problem with all four plants opened.

Subproblem Solution

```
Plants opened: # 1 2 3 4

Minimum transport cost = 201
Fixed cost of plants = 1150
      Total = 1351

CPU time = 9.05 sec.

Generated support is αY+b, where
 α = 300 400 250 200
 & b = 201
That is, v(Y) ≥ αY+b

This is support # 1
```

*** New incumbent! ***

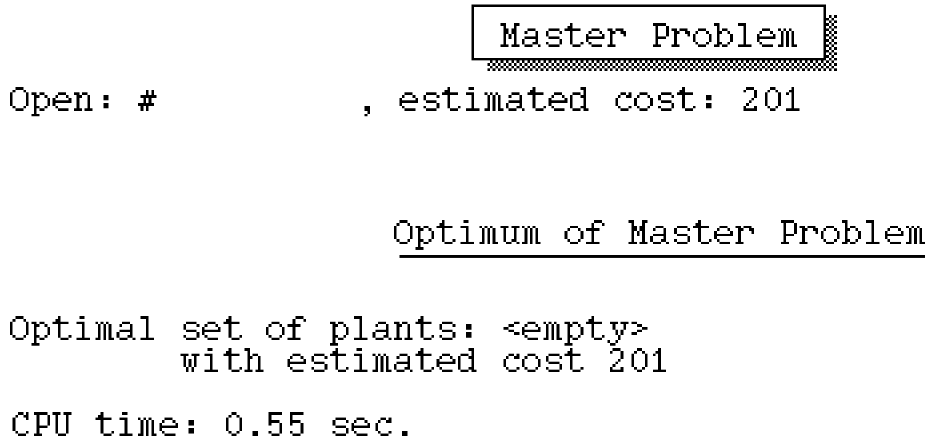
The cost of (1,1,1,1) is 1351, our initial "incumbent"

Next, we must solve the (partial) master problem, namely

$$\boxed{\begin{array}{l}\text{Minimize} \quad \underline{v}_1(Y) \\ Y_i \in \{0,1\}\end{array}}$$

where

$$\underline{v}_1(Y) = 300\,Y_1 + 400Y_2 + 250Y_3 + 200Y_4 + 201$$



```
                    Master Problem
Open: #         , estimated cost: 201


                 Optimum of Master Problem

Optimal set of plants: <empty>
        with estimated cost 201

CPU time: 0.55 sec.
```

Because the approximating function $\underline{v}_1(Y)$ is such a poor approximation, the solution to the master problem is to open NO plants!

(A constraint might have been added to the master problem which would guarantee that only feasible sets of plants were selected.... that is,

$$\sum_{i=1}^{M} S_i Y_i \geq \sum_{j=1}^{N} D_j ,$$

but in this case no such constraint was used.)

```
Subproblem Solution

Plants opened: #                          (none)

Minimum transport cost = 290000
Fixed cost of plants = 0
       Total = 290000

CPU time = 14.45 sec.

Generated support is ∝Y+b, where

 ∝ = ¯129700 ¯129600 ¯99750 ¯89800
 & b = 290000
That is, v(Y) ≥ ∝Y+b

This is support # 2
```

A "dummy" source with very large "shipping" costs was included, to guarantee feasibility.

Master Problem

```
Open: # 1 2 3, estimated cost: 1151
Open: # 1 2 4, estimated cost: 1101
Open: # 1 3 4, estimated cost: 951
```

Optimum of Master Problem

```
Optimal set of plants: 1 3 4
        with estimated cost 951

CPU time: 4.8 sec.
```

$\underline{v}_2$ is minimized at

$$Y=(1,0,1,1)$$

Subproblem Solution

```
Plants opened: # 1 3 4

Minimum transport cost = 341
Fixed cost of plants = 750
       Total = 1091

CPU time = 11.2 sec.

Generated support is αY+b, where
 α = 1210 400 790 830
 & b = ¯1739
That is, v(Y) ≥ αY+b

This is support # 3
```

*** New incumbent! ***
(replaces 1351)

$$\underline{v}_2(1,0,1,1) = 951 < 1091 = v(1,0,1,1)$$

Master Problem

```
Open: # 1 3 4, estimated cost: 1091
Open: # 2 3 4, estimated cost: 1051
```

Optimum of Master Problem

```
Optimal set of plants: 2 3 4
        with estimated cost 1051

CPU time: 4.8 sec.
```

Minimum of $\underline{v}_3$ is 1051, at

$$Y = (0, 1, 1, 1)$$

Subproblem Solution

```
Plants opened: # 2 3 4

Minimum transport cost = 599
Fixed cost of plants = 850
       Total = 1449

CPU time = 21.75 sec.
```

While the estimated cost of Y=(0,1,1,1) was lower than the incumbent's cost, its actual cost is considerably higher!

$$\underline{v}_3(0,1,1,1) = 1051 < 1449 = v(0,1,1,1)$$

```
Generated support is ∝Y+b, where
 ∝ = 300 1310 340 425
 & b = ¯626
That is, v(Y) ≥ ∝Y+b

This is support # 4
```

Master Problem

```
Open: # 1 3 4, estimated cost: 1091
```

Optimum of Master Problem

```
Optimal set of plants: 1 3 4
        with estimated cost 1091
```

the incumbent

```
CPU time: 4.85 sec.
```

$$v(Y) = \underline{v}(Y)$$

termination criterion is satisfied!

The Y which minimizes $\underline{v}_4(Y)$ happens to be the incumbent!

Current List of Supports of v(Y)

```
Current approximation of v(Y) is  Maximum { α[i]Y + b[i] }
where α & b are:
```

α_1	α_2	α_3	α_4	β
300	400	250	200	201
¯129700	¯129600	¯99750	¯89800	290000
1210	400	790	830	¯1739
300	1310	340	425	¯626

```
Current incumbent: 1091
```

Suboptimizing the Master Problem

Again, we begin with the "guess"

Y= (1,1,1,1),

i.e., that all four plants are open.

Initial "guess": all plants open

Subproblem Solution

```
Plants opened: # 1 2 3 4

Minimum transport cost = 201
Fixed cost of plants = 1150
       Total = 1351

CPU time = 9.05 sec.

Generated support is αY+b, where
 α = 300 400 250 200
 & b = 201
That is, v(Y) ≥ αY+b

This is support # 1
```

*** New incumbent! ***

Initial subproblem

Solution of 1st subproblem

Optimal Shipments

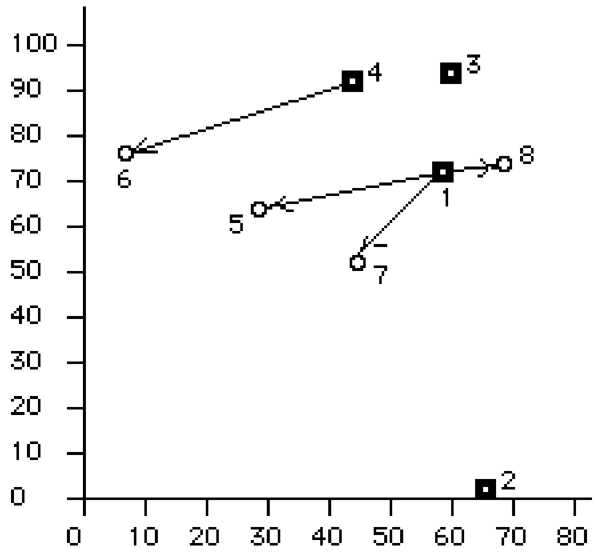
"dummy demand"

from \ to	1	2	3	4	5	6	7	8	9
1	4	0	0	0	1	0	5	1	2
2	0	2	0	0	0	0	0	0	11
3	0	0	10	0	0	0	0	0	0
4	0	0	0	5	0	1	0	0	3

(Demand pt #9 is dummy demand for excess capacity.)

NOTE: Solution is degenerate!

Optimal shipments (to non-local customers)



Dual Solution of Transportation Problem

Supply constraints

```
   i= 1   2   3   4
U[i]= 0   0   0   0
```

Demand constraints

```
   j= 1    2    3    4    5    6    7    8
V[j]= 0    0    0    0   31   40   24   10
```

Reduced costs: COST - U∘.+V

```
 0  70  22  25   0  12   0   0
70   0  92  93  41  55  30  62
22  92   0  16  12  16  21  12
25  93  16   0   1   0  16  21
```

Generating the first support for v(Y)

$$\alpha_i^k = F_i + S_i \hat{u}_i^k$$

$$\beta^k = \sum_{j=1}^{N} D_j \hat{v}_j^k$$

Supply constraints

```
   i=  1    2   3   4
U[i]=  0    0   0   0
S[i]= 13   13  10   9
F[i]=300 400 250 200
```

Demand constraints

```
   j= 1    2    3    4    5    6    7    8
V[j]= 0    0    0    0   31   40   24   10
D[j]= 4    2   10    5    1    1    5    1
```

$$\alpha_i^0 = F_i \Longrightarrow \alpha^0 = (300,400,250,200)$$

$$\beta^0 = 31+40+120+10=201$$

Current List of Supports of v(Y)

Current approximation of v(Y) is Maximum { α[i]Y + b[i] } where α & b are:

α_1	α_2	α_3	α_4	β
300	400	250	200	201

Current incumbent: 1351

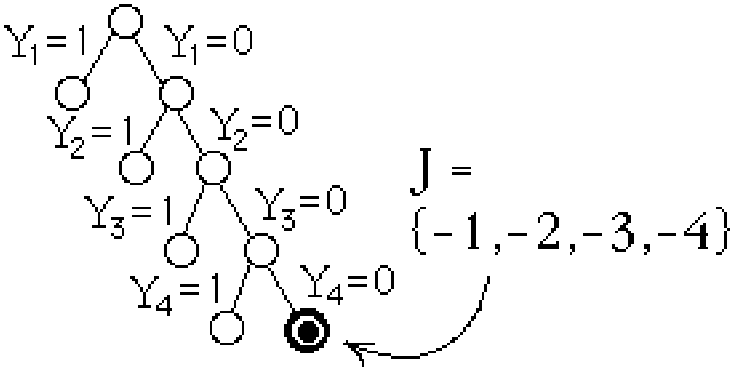
$$\Longrightarrow \quad \underline{v}_1(Y) = 300Y_1 + 400Y_2 + 250Y_3 + 200Y_4 + 201$$

First master problem solution

Master Problem

(suboptimized, i.e., a solution Y such that v(Y) < incumbent.)

```
Trial set of plants : <empty>
with estimated cost 201 < incumbent ( = 1351)

Current status vectors for Balas' additive algorithm:
    j:     ¯1 ¯2 ¯3 ¯4
underline:  0  0  0  0

CPU time: 0.55 sec.
```



Subproblem Solution

```
Plants opened: #   (none)

Minimum transport cost = 290000
Fixed cost of plants = 0
       Total = 290000

CPU time = 14.45 sec.

Generated support is ∝Y+b, where
 ∝ = ¯129700 ¯129600 ¯99750 ¯89800
 & b = 290000
That is, v(Y) ≥ ∝Y+b

This is support # 2
```

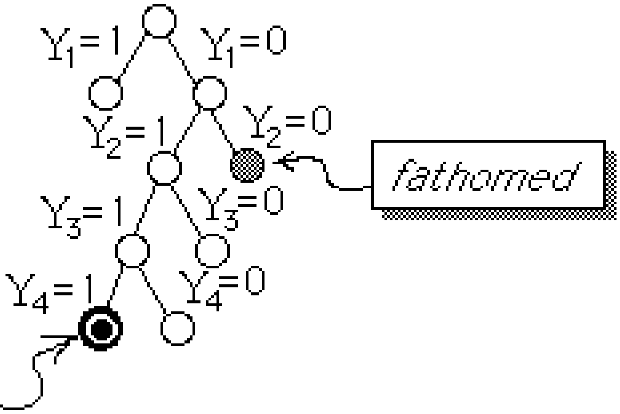
(all demand is
supplied from
dummy plant
with high shipping
cost, 10000/unit)

Master Problem

```
(suboptimized, i.e., a solution Y such that v(Y) < incumbent.)

Trial set of plants: 2 3 4
with estimated cost 1051 < incumbent ( = 1351)

Current status vectors for Balas' additive algorithm:
    j:      ¯1  2  3  4
underline:  0  1  0  0

CPU time: 1.6 sec.
```

$$J = \{-1, \underline{+2}, +3, +4\}$$

The status of the search tree is currently:



$J = \{-1, \underline{+2}, +3, +4\}$

Subproblem Solution

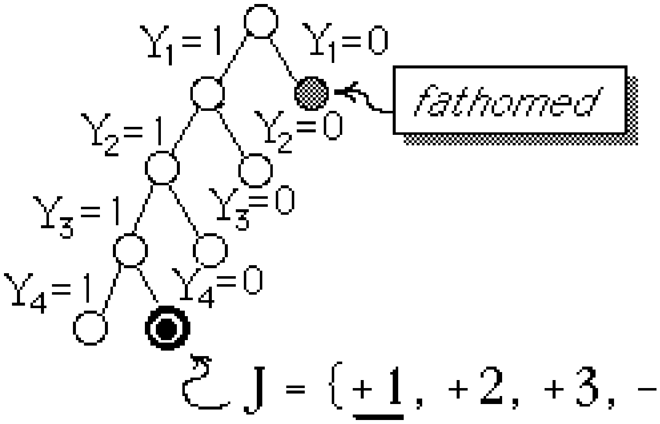
```
Plants opened: # 2 3 4

Minimum transport cost = 599
Fixed cost of plants = 850
        Total = 1449

CPU time = 21.8 sec.

Generated support is αY+b, where
 α = 300 1310 340 425
 & b = ¯626
That is, v(Y) ≥ αY+b

This is support # 3
```

Master Problem

```
(suboptimized, i.e., a solution Y such that v(Y) < incumbent.)


Trial set of plants: 1 2 3
with estimated cost 1324 < incumbent ( = 1351)

Current status vectors for Balas' additive algorithm:
    j:       1  2  3 ¯4
underline:   1  0  0  0

CPU time: 1.7 sec.
```

$$\boxed{J = \{\underline{+1}, +2, +3, -4\}}$$

The status of the search tree is currently:



$J = \{\underline{+1}, +2, +3, -4\}$

Subproblem Solution

```
Plants opened: # 1 2 3

Minimum transport cost = 458
Fixed cost of plants = 950
       Total = 1408

CPU time = 16.3 sec.

Generated support is αY+b, where
 α = 625 1115 280 200
 & b = ¯612
That is, v(Y) ≥ αY+b

This is support # 4
```

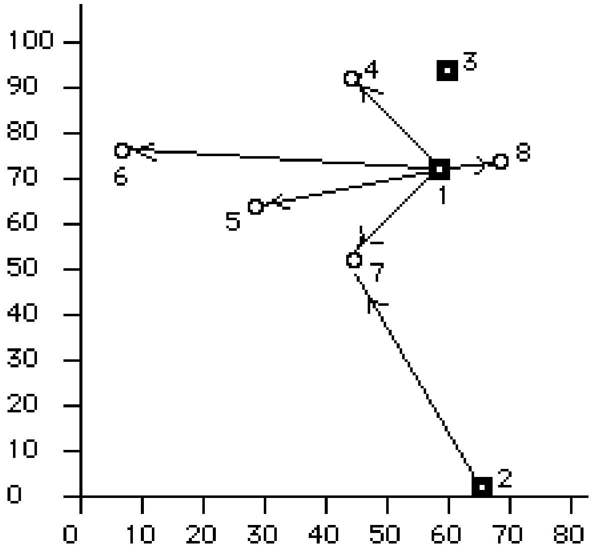
Optimal Shipments

from \ to	1	2	3	4	5	6	7	8	9
1	4	0	0	5	1	1	1	1	0
2	0	2	0	0	0	0	4	0	7
3	0	0	10	0	0	0	0	0	0

(Demand pt #9 is dummy demand for excess capacity.)

NOTE: Solution is degenerate!

Optimal shipments (to non-local customers)



Plant #3 serves only the local customer at that location

Customer #7 is supplied by two different plants!

Dual Solution of Transportation Problem

Supply constraints

```
   i=  1   2   3   4
U[i]= 25  55   3   0
```

Demand constraints

```
   j=  1   2   3   4   5   6   7   8
V[j]= ¯25 ¯55  ¯3   0   6  27  ¯1 ¯15
```

Reduced costs: COST - U∘.+V

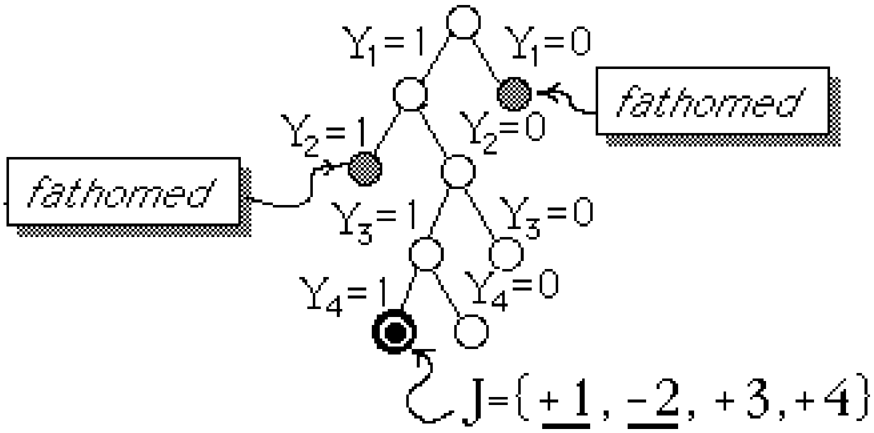
```
 0  100   0   0   0   0   0   0
40    0  40  38  11  13   0  32
44  144   0  13  34  26  43  34
50  148  19   0  26  13  41  46
```

Master Problem

```
(suboptimized, i.e., a solution Y such that v(Y) < incumbent.)


Trial set of plants: 1 3 4
with estimated cost 951 < incumbent ( = 1351)

Current status vectors for Balas' additive algorithm:
    j:       1 ¯2  3  4
underline:   1  1  0  0

CPU time: 2.3 sec.
```

$$\boxed{J=\{\underline{+1},\ \underline{-2},\ +3,+4\}}$$

The status of the search tree is currently:



$J=\{\underline{+1},\ \underline{-2},\ +3,+4\}$

Subproblem Solution

```
Plants opened: # 1 3 4

Minimum transport cost = 341
Fixed cost of plants = 750
       Total = 1091

CPU time = 11.2 sec.

Generated support is αY+b, where
 α = 1210 400 790 830
 & b = ¯1739
That is, v(Y) ≥ αY+b

This is support # 5
```

Master Problem

```
*** No solution with v(Y) less than incumbent! ***
(Current incumbent: 1091, with plants #1 3 4 open)

CPU time: 0.75 sec.
```

The search tree has been completely enumerated!

One more subproblem was required than in the algorithm which the Master Problem was optimized at each iteration!

When the master problem was optimized at each iteration, a total of FOUR subproblems were necessary, while we required FIVE subproblems when we suboptimized the master problem...

One more subproblem was required than in the algorithm which the Master Problem was optimized at each iteration!

However, the savings in computation in solving the master problem more than compensates for the additional subproblem!

Current List of Supports of v(Y)

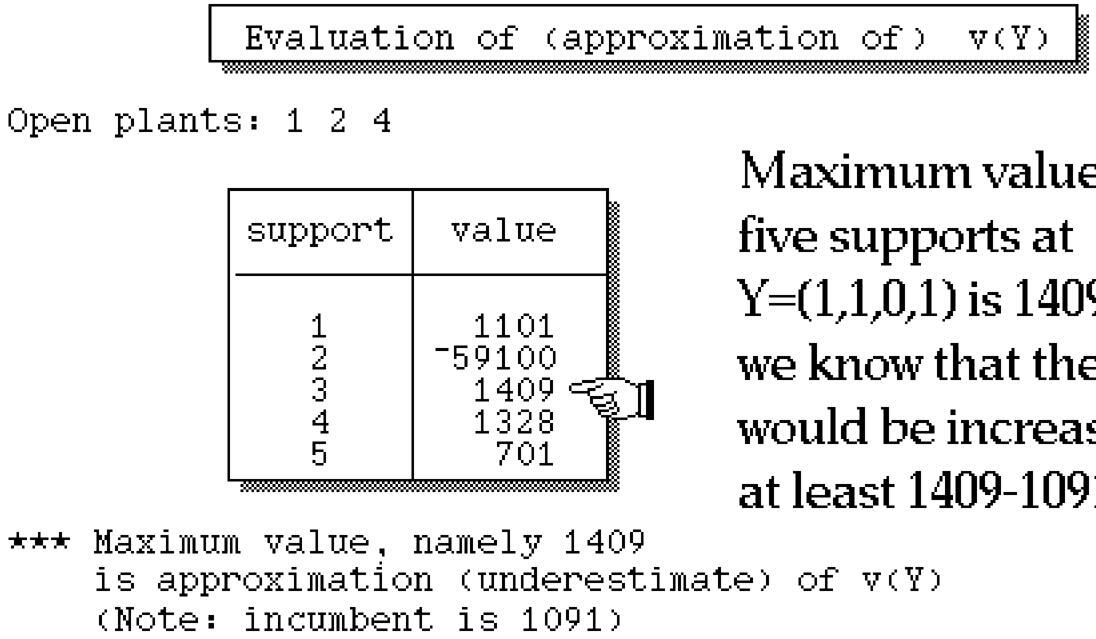
Current approximation of v(Y) is Maximum { α[i]Y + b[i] }
where α & b are:

α_1	α_2	α_3	α_4	β
300	400	250	200	201
¯129700	¯129600	¯99750	¯89800	290000
300	1310	340	425	¯626
625	1115	280	200	¯612
1210	400	790	830	¯1739

Current incumbent: 1091

The approximation $\underline{v}_5(Y)$ which we have computed is useful in answering "what-if" questions, e.g.,

> "Although it is optimal to open plants at locations #1, 3, and 4, what if we were to open a plant at location 2 instead of location 3, i.e., is there a large penalty for choosing location 2 instead of 3?"



```
Evaluation of (approximation of)  v(Y)

Open plants: 1 2 4

  support    value

     1        1101
     2      ⁻59100
     3        1409
     4        1328
     5         701

*** Maximum value, namely 1409
    is approximation (underestimate) of v(Y)
    (Note: incumbent is 1091)
```

Maximum value of the five supports at Y=(1,1,0,1) is 1409, so we know that the cost would be increased by at least 1409-1091=318

trial solution

Subproblem Solution

```
Plants opened: # 1 2 4

Minimum transport cost = 553
Fixed cost of plants = 900
       Total = 1453

CPU time = 16.25 sec.

Generated support is αY+b, where
 α = 586 1076 250 344
 & b = ¯553
That is, v(Y) ≥ αY+b

This is support # 6
```

> 1409 (approximation)

In actuality, the cost is increased by 1453-1091= 361