BRANCHING STRATEGIES

FOR A

QUADRATIC ASSIGNMENT PROBLEM

BRANCH AND BOUND ALGORITHM

Peter M. Hahn, Sci-Tech. Services, Inc.
William L. Hightower, Elon College
***Terri Anne Johnson, Elon College**

*** This research is supported in part by the National Science Foundation under Grant DMS-9631994.**

Impact of Bound Choice

- **Pruning the Tree**
- **Guiding Branching Choices**
- **Improvement of Later Bound Computations**

DUAL PROCEDURE BRANCH-AND-BOUND VERSUS COMPETING ENUMERATION ALGORITHM

(Run times in equivalent CPU minutes.)

	DPB&B NODES	TIME	CLAUSEN NODES	TIME
Nug 15	2,203	3.8	105,773	2.3
Had 16	13,808	22.3	18,770,885	7,140
Had 18	197,487	561.1	761,452,218	6,188
Rou 20	2,090,862	16,501	2,161,665,137	26,908
Nug 20	724,289	2,665	360,148,026	13,524
Nug 21	3,192,565	10,375	3,631,929,368	44,982
Nug 22	10,768,366	30,207	48,538,844,413	805,140
Nug 24	49,542,338	80,999	217.6 billion	1,370,880

THE QUADRATIC ASSIGNMENT PROBLEM

$$\underset{\pi \quad \Pi}{\text{minimize}} \sum_{i} \sum_{j} f_{i,j} \cdot d_{\pi(i),\pi(j)},$$

THE QUADRATIC ASSIGNMENT PROBLEM

$$\text{minimize} \quad \sum_i \sum_j \sum_k \sum_l c_{ijkl} x_{ij} x_{kl} + \sum_k \sum_l b_{kl} x_{kl}$$

$$\text{subject to} \quad \sum_k x_{kl} = 1 \qquad l = 1, \ldots, m$$

$$\sum_l x_{kl} = 1 \qquad k = 1, \ldots, m$$

x binary

REFORMULATION LINEARIZATION TECHNIQUE (RLT)

FOR PURE 0-1 AND MIXED 0-1 QUADRATIC PROGRAMS

Adams and Sherali (1986, 1990)
Sherali and Adams (1990, 1994-manuscript 1989)

LEVEL-1 RLT FORMULATION OF THE QUADRATIC ASSIGNMENT PROBLEM

Johnson (Dissertation 1992)
Adams and Johnson (1994)

LEVEL-1 RLT FORMULATION OF THE QAP

$$\text{minimize} \quad \sum_{i \neq k} \sum_{j \neq l} \sum_{k} \sum_{l} c_{ijkl} y_{ijkl} + \sum_{k} \sum_{l} b_{kl} x_{kl}$$

subject to

$$\sum_{i \neq k} y_{ijkl} = x_{kl} \qquad (j,k,l),\ j \neq l$$

$$\sum_{j \neq l} y_{ijkl} = x_{kl} \qquad (i,k,l),\ i \neq k$$

$$y_{ijkl} = y_{klij} \qquad (i,j,k,l),\ i < k,\ j \neq l$$

$$\sum_{k} x_{kl} = 1 \qquad l = 1,\ldots,m$$

$$\sum_{l} x_{kl} = 1 \qquad k = 1,\ldots,m$$

$$y_{ijkl} \geq 0 \qquad (i,j,k,l),\ i \neq k,\ j \neq l$$

x binary

Set of Problem Coefficients

(for a problem of size n)

$$a,\ b_{ij},\ c_{ijkl} \qquad\qquad 1 \le i,j,k,l \le n$$

$$a,\ b_{ij},\ c_{ijkl} \ge 0$$

Cost of a Feasible Solution x_{ij}, y_{ijkl}

$$a + \sum b_{ij}\, x_{ij} + \sum c_{ijkl}\, y_{ijkl}$$

Level 1 Transformation

- **Level 1 Transformational Bound**

- **Level 1 Transformational Bounds**
- **Gilmore (1962) - Lawler (1963)**
- **Burkard & Stratmann (1978)**
- **Roucairol (1979,1987)**
- **Burkard & Derigs (1980)**
- **Christofides et al. (1980)**
- **Edwards (1980)**
- **Frieze & Yadegar (1983)**
- **Assad and Xu (1985)**
- **Li et al. (1992)**
- **Carraresi and Malucelli (1992)**
- **Adams and Johnson (1992)**
- **Hahn and Grant (1998)**
- **Karisch et al. (1998)**

LEVEL-1 RLT BOUNDS
for the
QUADRATIC ASSIGNMENT PROBLEM

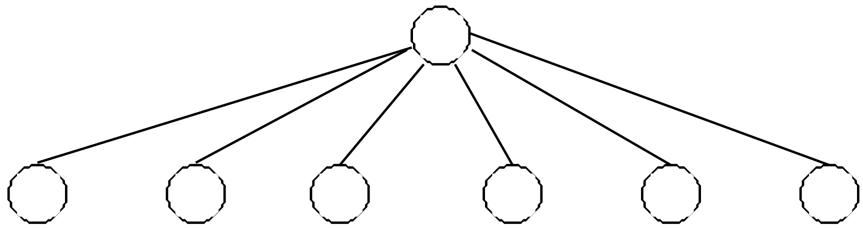
	GL BOUND	LEVEL-1 BOUND	LEVEL-1 OPT	BKV
Nug 5	50		50	50
Nug 6	82		86	86
Nug 7	137		148	148
Nug 8	186	203.4	203.49	214
Nug12	493	522.4	522.89	578
Nug15	963	1039.7	1040.99	1150
Nug20	2057	2179.1	2181.57	2570
Nug30	4539	4788.9	4804.56	6124
Els 19	11,971,949	16,771,926	16,874,205	17,212,548
Kra30a	68,360	75,853.40	76,003	88,900
Kra30b	69,065	76,562.20	76,752	91,420
Rou 20	599,948	641,663	643,346	725,522

** Level-1 Bounds are within at least 99.67% of the Level-1 Optimal Value

Tree Elaboration

- **Depth First**
- **At each node extend by all assignments of an unassigned facility (location) to unassigned locations (facilities)**

Choice of Facility (Location)



- **Most Forbidden**
- **Most Above Threshold**
- **Highest Total**

Single Extension Bound Choice

$$a + \sum b_{ij} x_{ij} + \sum c_{ijkl} y_{ijkl}$$

1.Appropriate element of [b]
(Reduced cost in leader matrix)

2.Solve single extension in [b]

3.Solve single extension in [b] after adding in appropriate submatrix of [c]

4.Gilmore Lawler

5.Some other level 1 transformational bound

Hadley 16 Experiments			
Single Extension Bound Computation	Facility Location Selection Method	Number of Nodes Evaluated	Runtime in Seconds
None	Uppermost Facility	13,549	879
None	Random	17,199	955
1	Max Sum	13,481	614
1	Max Number Above Threshhold	12,892	1,119
3	Max Sum	3,069	206
4	Max Sum	6,121	1,514
5 1 iteration	Max Sum	2,502	1,623
5 1 iteration	Max # > med Sample Every Second	4,045	1,397
5 2 or 3 iterations	Max # > med Sample Every Second	4,021	1,570

Hadley 18 Experiments				
Single Extension Bound Computation	Single Extension Bound Computation	Facility Location Selection Method	Number of Nodes Evaluated	Runtime in Seconds
None		Uppermost Facility	197,487	33,666
1		Max Sum	359,962	22,967
3		Max Sum	53,224	4,342
4		Max # Above Median	53,290	4,921
5 1 iteration sample 6th		Max # Above Median	67,719	12,859
5 1 iteration sample 3rd		Max Above Median	66,288	20,498
5 1 iteration		Max # Above Median	42,607	42,076
5 1 iteration sample 2nd	3 at level 1	Max # Above Median	66,103	5,914

Nugent 20 Experiments				
Single Extension Bound Computation	Single Extension Bound Computation	Facility Location Selection Method	Number of Nodes Evaluated	Runtime in Seconds
None		Uppermost Facility	724,289	48,578
1		Max Sum		
4		Max Sum		
5	3 at level 1	Max Sum	608,258	29,764
5	3 at level 2	Max Sum	239,449	23,645
5	3 at level 3	Max Sum	207,157	27,054

Nugent 22 Experiments				
Single Extension Bound Computation	Single Extension Bound Computation	Facility Location Selection Method	Number of Nodes Evaluated	Runtime in Seconds
None		Uppermost Facility	10,768,366	1,812,100
5	3 at level 4	Max Sum	988,302	293,427

Nugent 24 Experiments

Single Extension Bound Computation	Single Extension Bound Computation	Facility Location Selection Method	Number of Nodes Evaluated	Runtime in Seconds
None		Uppermost Facility	49,542,338	4,859,940
5	3 at level 4	Max Sum	11,674,955	1,135,610 (13.14 days)
5	3 at level 6	Max Sum	5,629,849	2,791,380

Single Extension Bound Informs Branching Strategy

For any bound, quality of information deteriorates with size of problem/closeness to the root

Conclusions

Bound Choice Should Be Informed By Use and Position in the Tree

Expensive Bounds Should Not Be Used Where Cheap Bounds Do What Is Required

Expensive Bounds Are Indispensable In Certain Parts of Particularly Large Problems