イロン イボン イヨン イヨン

Some Quadratic Assignment Formulations and their impact on Gurobi

Leo Lopes¹

¹Monash University, School of Mathematical Sciences, Melbourne Australia



Monash University School of Mathematics

Problem Definition

Integer Programming Formulations The QAP A simple formulation

Branch and Bound

Diagnosing and Improving our Formulation

Step 0: bound variables whenever possible Step 1: Strengthen cuts by adding more variables to each constraint Step 2: Smaller constants and more terms

Conclusions

Integer Programming Formulations

The basic principle: strong and weak formulations



Integer Programming Formulations

The basic principle: formulations and branch-and-cut



Lopes

Diagnosing and Improving our Formulation

The Quadratic Assignment Problem (Input)



Flow	/S:				
	1	2	3	4	5
1		2	0	2	4
2			4	4	1
3				3	2
4					3
5					

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへで

Lopes

The Quadratic Assignment Problem (Output)



Diagnosing and Improving our Formulation

イロン イボン イヨン イヨン

Permutation Formulation of the QAP

Let:

The QAP

► F be a set of facilities. Between each pair (f₁, f₂) of facilities, the flow volume is v_{f1f2}.

• L be a set of locations. The distance between location l_1 and l_2 is $d_{l_1 l_2}$.

Find an assignment $g: F \rightarrow L$ of each facility to a distinct location that minimizes the total distance in the network weighted by flow:

$$\min_{g(x):F\to L} \sum_{f_i\in F} \sum_{f_2\in F} v_{f_1f_2} d_{g(f_1),g(f_2)}$$

Basic Mixed Integer Programming Formulation

Let:

Lopes

- $x_{fl} = \begin{cases} 1 & \text{if facility f is assigned to location I} \\ 0 & \text{otherwise} \end{cases}$
- $y_{f_1 f_2} \ge 0$ be the distance between facilities f_1 and f_2 .

► The objective:
$$\min_{x,y} \sum_{(f_1,f_2) \in F \times F} v_{f_1 f_2} y_{f_1 f_2}$$

Since the objective is lower when y is lower, a constraint that bounds the distance between f₁ and f₂ from below to d_{l1l2} if and only if f₁ is assigned to l₁ and f₂ is assigned to l₂ will set y to the correct value:

►
$$y_{f_1f_2} \ge (x_{f_1l_1} + x_{f_2l_2} - 1)d_{l_1l_2}, \forall f_1, f_2 : f_1 \ne f_2, l_1, l_2 : l_1 \ne l_2$$

In AMPL (1 of 2)

param n > 0; set FAC := 1..n; set LOC := 1..n;

```
param d{LOC,LOC} >= 0;
param v{FAC,FAC} >= 0;
```

```
var Assign{FAC,LOC} binary;
var Dist{FAC,FAC} >= 0;
```

Branch and Bound

Diagnosing and Improving our Formulatio

A simple formulation

In AMPL (2 of 2)

```
minimize VolumeWeightedDistance:
  sumf1 in FAC, f2 in FAC: f2 <> f1 v[f1, f2]*Dist[f1, f2];
subject to Permutelf in FAC:
  suml in LOC Assign[f,1] = 1;
subject to Permute21 in LOC:
  sumf in FAC Assign [f, 1] = 1;
subject to ComputeDistf1 in FAC, f2 in FAC, l1 in LOC, l2 in LOC:
                        f1 \iff f2 and l1 \iff l2:
  Dist[f1, f2] >= (Assign[f1, 11] + Assign[f2, 12] - 1)*d[11, 12];
```

In AMPL (2 of 2)

```
minimize VolumeWeightedDistance:
  sumf1 in FAC, f2 in FAC: f2 <> f1 v[f1, f2]*Dist[f1, f2];
subject to Permutelf in FAC:
  suml in LOC Assign[f,1] = 1;
subject to Permute21 in LOC:
  sumf in FAC Assign [f, 1] = 1;
subject to ComputeDistf1 in FAC, f2 in FAC, l1 in LOC, l2 in LOC:
                        f1 \iff f2 and l1 \iff l2:
  Dist[f1, f2] >= (Assign[f1, 11] + Assign[f2, 12] - 1)*d[11, 12];
```

In AMPL (2 of 2)

```
minimize VolumeWeightedDistance:
  sumf1 in FAC, f2 in FAC: f2 <> f1 v[f1, f2]*Dist[f1, f2];
subject to Permutelf in FAC:
  suml in LOC Assign[f,1] = 1;
subject to Permute21 in LOC:
  sumf in FAC Assign [f, 1] = 1;
subject to ComputeDistf1 in FAC, f2 in FAC, l1 in LOC, l2 in LOC:
                        f1 \iff f2 and l1 \iff l2:
  Dist[f1, f2] >= (Assign[f1, 11] + Assign[f2, 12] - 1)*d[11, 12];
```

Solver Results

The math is correct, but it "doesn't work" "in practice"!

```
option solver gurobix;
option gurobi_options $gurobi_options 'outlev 1';
reset;
model qapNaive.ampl;
read n,
    {f in FAC, l in LOC} d[f,l],
    {f in FAC, l in LOC} v[f,l]
< instances/nug12.dat;</pre>
```

The branch-and-bound tree will grow until it consumes all memory available (2010).

Producing good bounds is essential

Let's see the details.

Monash University

-

イロン イボン イヨン イヨン

Lopes

A Branch and Bound Tree



After	Lower	Upper
node	Bound	Bound
0	2	∞
1	2	∞
2		
3	4	∞
4	5	∞
5	5	∞
6		∞
7*		
8	6	8
9	6	7
10*		

◆□▶ ◆□▶ ◆ 三▶ ◆ 三▶ ・ 三 ・ の Q ()

Gurobi 3.0.0: outlev 1 Optimize a model with 17448 Rows, 276 Columns and 52560 NonZeros Presolve removed 5544 rows and 42 columns Presolve time: 0.57s Presolved: 11904 Rows, 234 Columns, 47808 Nonzeros Objective GCD is 1 Found heuristic solution: objective 868.0000000 Found heuristic solution: objective 724.0000000

Root relaxation: objective 0.000000e+00, 768 iterations, 0.05 seconds

Nodes		Current Node			Obje	5	Work		
Expl Un	expl	Obj Depth	n IntInf		Incumbent	BestBd	Gap	It/N	ode Time
• • •									
311942	84869	cutoff	42		582.00000	499.14837	14.2%	20.9	230s
321368	85757	575.74359	48	46	582.00000	500.51202	14.0%	20.8	235s
331149	86576	523.66648	43	65	582.00000	501.93019	13.8%	20.7	240s
*332324	85008		39	57	78.0000000	502.07250	13.1%	20.6	240s

. . .

◆□▶ ◆□▶ ◆ 三▶ ◆ 三▶ ・ 三 ・ の Q ()

Lopes

Gurobi 3.0.0: outlev 1 Optimize a model with 17448 Rows, 276 Columns and 52560 NonZeros Presolve removed 5544 rows and 42 columns Presolve time: 0.57s Presolved: 11904 Rows, 234 Columns, 47808 Nonzeros Objective GCD is 1 Found heuristic solution: objective 868.0000000 Found heuristic solution: objective 724.000000

Root relaxation: objective 0.000000e+00, 768 iterations, 0.05 seconds

Nodes			Current Node				Obje	ctive B		Work		
Expl (Jnexp)	Obj	Depth	IntInf		Incumbent	Be	stBd	Gap	It/	Node Time
• • •												
31194	42 84	1869	cuto	off	42		582.00000	499.14	837	14.2%	20.9	230s
32136	68 85	5757	575.	74359	48	46	582.00000	500.51	202	14.0%	20.8	235s
33114	49 86	5576	523.	66648	43	65	582.00000	501.93	019	13.8%	20.7	240s
*33232	24 85	5008			39	57	78.0000000	502.07	250	13.1%	20.6	240s

. . .

◆ロト ◆母 ▶ ◆ ヨ ▶ ◆ ヨ ・ ク へ ()・

Lopes

Gurobi 3.0.0: outlev 1 Optimize a model with 17448 Rows, 276 Columns and 52560 NonZeros Presolve removed 5544 rows and 42 columns Presolve time: 0.57s Presolved: 11904 Rows, 234 Columns, 47808 Nonzeros Objective GCD is 1 Found heuristic solution: objective 868.0000000 Found heuristic solution: objective 724.000000

Root relaxation: objective 0.000000e+00, 768 iterations, 0.05 seconds

Nodes		Current Node				Objec	5	Work				
Expl U	nexpl		Obj	Depth	IntInf		Incumbent	Best	tBd	Gap	It/N	ode Time
• • •												
31194	2 84869	9	cuto	ff	42		582.00000	499.148	37	14.2%	20.9	230s
32136	8 85757	7	575.	74359	48	46	582.00000	500.512	02	14.0%	20.8	235s
33114	9 86576	5	523.	66648	43	65	582.00000	501.930	19	13.8%	20.7	240s
*33232	4 85008	3			39	57	78.0000000	502.072	50	13.1%	20.6	240s

. . .

◆ロト ◆母 ▶ ◆ ヨ ▶ ◆ ヨ ・ ク へ ()・

Lopes

Gurobi 3.0.0: outlev 1 Optimize a model with 17448 Rows, 276 Columns and 52560 NonZeros Presolve removed 5544 rows and 42 columns Presolve time: 0.57s Presolved: 11904 Rows, 234 Columns, 47808 Nonzeros Objective GCD is 1 Found heuristic solution: objective 868.0000000 Found heuristic solution: objective 724.000000

Root relaxation: objective 0.000000e+00, 768 iterations, 0.05 seconds

Nodes			Cu	rrent l	Node		Objec	ctive	Bounds	5		Work
Expl U	Inexpl		Obj	Depth	IntInf		Incumbent	E	BestBd	Gap	It/N	Node Time
• • •												
31194	2 8486	9	cuto	ff	42		582.00000	499.1	4837	14.2%	20.9	230s
32136	8 8575	7	575.	74359	48	46	582.00000	500.5	1202	14.0%	20.8	235s
33114	9 8657	6	523.	66648	43	65	582.00000	501.9	3019	13.8%	20.7	240s
*33232	4 8500	8			39	57	78.0000000	502.0	7250	13.1%	20.6	240s

◆ロト ◆聞 ▶ ◆ ヨ ▶ ◆ ヨ ・ の Q ()

Lopes

Gurobi 3.0.0: outlev 1 Optimize a model with 17448 Rows, 276 Columns and 52560 NonZeros Presolve removed 5544 rows and 42 columns Presolve time: 0.57s Presolved: 11904 Rows, 234 Columns, 47808 Nonzeros Objective GCD is 1 Found heuristic solution: objective 868.0000000 Found heuristic solution: objective 724.000000

Root relaxation: objective 0.000000e+00, 768 iterations, 0.05 seconds

Nodes		L	Current Node				Objec	ds	Work			
Expl	Uner	kpl		Obj	Depth	IntInf		Incumbent	BestB	d Gap	It/N	ode Time
• • •												
3119	42 8	34869		cuto	ff	42		582.00000	499.14837	14.2%	20.9	230s
3213	68 8	35757		575.	74359	48	46	582.00000	500.51202	14.0%	20.8	235s
3311	49 8	36576		523.	66648	43	65	582.00000	501.93019	13.8%	20.7	240s
*3323	24 8	35008				39	57	78.0000000	502.07250	13.1%	20.6	240s

. . .

◆ロト ◆聞 ▶ ◆ ヨ ▶ ◆ ヨ ・ の Q ()

Lopes

Gurobi 3.0.0: outlev 1 Optimize a model with 17448 Rows, 276 Columns and 52560 NonZeros Presolve removed 5544 rows and 42 columns Presolve time: 0.57s Presolved: 11904 Rows, 234 Columns, 47808 Nonzeros Objective GCD is 1 Found heuristic solution: objective 868.0000000 Found heuristic solution: objective 724.000000

Root relaxation: objective 0.000000e+00, 768 iterations, 0.05 seconds

Nodes		Current	Node		Objec	5	Work		
Expl Une	expl	Obj Depth	IntInf		Incumbent	BestBd	Gap	It/N	lode Time
• • •									
311942	84869	cutoff	42		582.00000	499.14837	14.2%	20.9	230s
321368	85757	575.74359	48	46	582.00000	500.51202	14.0%	20.8	235s
331149	86576	523.66648	43	65	582.00000	501.93019	13.8%	20.7	240s
*332324	85008		39	51	78.0000000	502.07250	13.1%	20.6	240s

. . .

Lopes

Gurobi 3.0.0: outlev 1 Optimize a model with 17448 Rows, 276 Columns and 52560 NonZeros Presolve removed 5544 rows and 42 columns Presolve time: 0.57s Presolved: 11904 Rows, 234 Columns, 47808 Nonzeros Objective GCD is 1 Found heuristic solution: objective 868.0000000 Found heuristic solution: objective 724.000000

Root relaxation: objective 0.000000e+00, 768 iterations, 0.05 seconds

Nodes		Cu	rrent 1	Node		Objec	ctive B		Work			
Expl Un	expl	0}	эj	Depth	IntInf		Incumbent	Be	estBd	Gap	It/N	ode Time
• • •												
311942	84869	Cl	ito	ff	42		582.00000	499.14	4837	14.2%	20.9	230s
321368	85757	5	75.	74359	48	46	582.00000	500.51	L202	14.0%	20.8	235s
331149	86576	52	23.	66648	43	65	582.00000	501.93	3019	13.8%	20.7	240s
*332324	85008				39	57	78.0000000	502.07	7250	13.1%	20.6	240s

. . .

Lopes

Gurobi 3.0.0: outlev 1 Optimize a model with 17448 Rows, 276 Columns and 52560 NonZeros Presolve removed 5544 rows and 42 columns Presolve time: 0.57s Presolved: 11904 Rows, 234 Columns, 47808 Nonzeros Objective GCD is 1 Found heuristic solution: objective 868.0000000 Found heuristic solution: objective 724.000000

Root relaxation: objective 0.000000e+00, 768 iterations, 0.05 seconds

Nodes			Current Node				Objec	5	Work			
Expl U	Jnexpl		Obj	Depth	IntInf		Incumbent	Best	Bd	Gap	It/	Node Time
31194	12 8486	59	cuto	ff	42		582.00000	499.1483	37	14.2%	20.9	230s
32136	58 8575	57	575.	74359	48	46	582.00000	500.5120)2	14.0%	20.8	235s
33114	19 865	76	523.	66648	43	65	582.00000	501.9301	9	13.8%	20.7	240s
*33232	24 8500	8			39	57	78.0000000	502.0725	50	13.1%	20.6	240s

. . .

◆□ ▶ ◆□ ▶ ◆ 三 ▶ ◆ 三 ● ● の Q ()

Lopes

Gurobi 3.0.0: outlev 1 Optimize a model with 17448 Rows, 276 Columns and 52560 NonZeros Presolve removed 5544 rows and 42 columns Presolve time: 0.57s Presolved: 11904 Rows, 234 Columns, 47808 Nonzeros Objective GCD is 1 Found heuristic solution: objective 868.0000000 Found heuristic solution: objective 724.000000

Root relaxation: objective 0.000000e+00, 768 iterations, 0.05 seconds

Nodes		Current Node				Objec	5	Work			
Expl Un	expl	Obj 1	Depth	IntInf		Incumbent	I	BestBd	Gap	It/1	Node Time
• • •											
311942	84869	cutof	f	42		582.00000	499.3	14837	14.2%	20.9	230s
321368	85757	575.7	4359	48	46	582.00000	500.5	51202	14.0%	20.8	235s
331149	86576	523.6	6648	43	65	582.00000	501.9	93019	13.8%	20.7	240s
*332324	85008			39	57	78.0000000	502.0	07250	13.1%	20.6	240s

. . .

Lopes

Diagnosing weak bounds

•	••									
	Nod	es	Cur	rrent No	de	Obje	ctive Bound	s	Wo	rk
]	Expl U	nexpl	Obj	Depth I	ntInf	Incumben	t BestBd	Gap	It/Nod	e Time
	0	0	0.000	0 000	37	724.00000	0.00000	100%	-	1s
	0	0	0.000	0 000	40	724.00000	0.00000	100%	-	1s
	••									
	0	0	0.000	0 000	31	724.00000	0.00000	100%	-	4s
Н	0	0				682.000000	0.00000	100%	-	4s
	0	0	0.000	0 000	31	682.00000	0.00000	100%	-	4s
Н	1238	974				618.000000	15.97409	97.4%	53.8	32s
	1320	1020	142.411	196 15	107	618.00000	27.18269	95.6%	53.1	35s
Н	1369	1048				606.000000	27.18269	95.5%	52.7	35s
	1470	1131	377.685	530 33	40	606.00000	27.75623	95.4%	52.3	41s
	1473	1135	27.756	623 16	49	606.00000	27.75623	95.4%	53.6	47s
	1474	1135	27.756	623 16	96	606.00000	27.75623	95.4%	53.7	50s

. . .

Diagnosing weak bounds

•	••									
	Noc	les	Cu:	rrent No	ode	Obje	ctive Bounds	- I	Woi	ck
	Expl (Jnexpl	Obj	Depth I	IntInf	Incumbent	t BestBd	Gap	It/Node	e Time
	0	0	0.000	000 0) 37	724.00000	0.00000	100%	-	1s
	0	0	0.000	000 C	40	724.00000	0.00000	100%	-	1s
	••									
	0	0	0.000	000 0) 31	724.00000	0.00000	100%	-	4s
Н	0	0				682.0000000	0.00000	100%	-	4s
	0	0	0.000	000 0) 31	682.00000	0.00000	100%	-	4s
•										
Н	1238	974				618.0000000	15.97409	97.4%	53.8	32s
	1320	1020	142.41	1 <mark>96</mark> 15	5 107	618.00000	27.18269	95.6%	53.1	35s
Н	1369	1048				606.0000000	27.18269	95.5%	52.7	35s
	1470	1131	377.68	<mark>530</mark> 33	3 40	606.00000	27.75623	95.4%	52.3	41s
	1473	1135	27.75	623 16	5 49	606.00000	27.75623	95.4%	53.6	47s
	1474	1135	27.75	623 16	5 96	606.00000	27.75623	95.4%	53.7	50s

. . .

▲□▶▲□▶▲□▶▲□▶ ■ のQ@

Diagnosing and Improving our Formulation

Step 0: bound variables whenever possible

Observation on variable bounds

No matter what facility gets assigned where, the distance between any two facilities must be at least the minimal distance between any two locations:

- $y_{f_1 f_2} \ge \min_{(l_1, l_2) \in L \times L: l_1 \neq l_2} d_{l_1 l_2}$
- ► In AMPL:

```
var Dist{FAC,FAC} >=
  min{l1 in LOC, l2 in LOC: l1 <> l2} d[l1,l2];
```

Lopes

E ∽ Q ↔ Monash University

Solver Results

At least it solves, but...

Explored 10,180,956 nodes (160,414,000 simplex iterations) in 13,664.80 seconds Thread count was 4 (of 4 available processors)

Let's see the details.

・ロ・・聞・・ 声・・ 聞・ うらの

Initial bounds vs. the muddled middle

• • •							
Nodes	Current	Node	Object	tive Bounds	1	Wor	k
Expl Unexpl	Obj Depth	IntInf	Incumbent	BestBd	Gap	It/Node	Time
0 0	348.00000	0 27	724.00000	348.00000	51.9%	-	0s
н 5117 3068			622.0000000	353.58269	43.2%	28.2	61s
* 6157 3790		38	612.0000000	354.34464	42.1%	27.6	64s
6179 3811	382.04491	29 78	612.00000	354.34464	42.1%	27.6	65s
146583 88509	447.72035	31 6	6 590.00000	402.72441	31.7%	23.6	600s
147934 89260	549.35810	41 6	5 590.00000	402.91520	31.7%	23.6	605s
149758 90263	3 516.43027	44 6	7 590.00000	403.18118	31.7%	23.6	610s
1660356 6856	538 563.64870	59	55 586.000	00 469.6273	71 19.9	19.7	3000s
1664829 6870)24 cutoff	48	586.000	00 469.7104	17 19.8	19.7	3005s

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ ○臣 - のへ⊙

Lopes

Initial bounds vs. the muddled middle

• • •							
Nodes	Cur	rent Node	Ob-	ective Bounds		Wor	k
Expl Unex	pl Obj	Depth IntIn	f Incumbe	ent BestBd	Gap	It/Node	Time
• • •							
0	0 348.000	00 0 2	7 724.0000	0 348.00000	51.9%	-	0s
H 5117 30	68		622.000000	0 353.58269	43.2%	28.2	61s
* 6157 37	90	38	612.000000	0 354.34464	42.1%	27.6	64s
6179 38	11 382.044	91 29 7	8 612.0000	0 354.34464	42.1%	27.6	65s
146583 88	509 447.72	035 31	66 590.000	00 402.72441	31.7%	23.6	600s
147934 89	260 549.35	810 41	65 590.000	00 402.91520	31.7%	23.6	605s
149758 90	263 516.43	027 44	67 590.000	00 403.18118	31.7%	23.6	610s
1660356 6	85638 563.	64870 59	55 586.0	0000 469.627	71 19.9	19.7	3000s
1664829 6	87024 c	utoff 48	586.0	0000 469.710	47 19.8	19.7	3005s

▲ロト▲聞を▲目を▲目を 目 めんぐ

Lopes

Initial bounds vs. the muddled middle

• • •							
Nodes	Current	Node	Object	tive Bounds		Wor	k
Expl Unexpl	Obj Depth	IntInf	Incumbent	BestBd	Gap	It/Node	Time
•••							
0 0	348.00000	0 27	724.00000	348.00000	51.9%	-	0s
Н 5117 3068			622.0000000	353.58269	43.2%	28.2	61s
* 6157 3790		38	612.0000000	354.34464	42.1%	27.6	64s
6179 3811	382.04491	29 78	612.00000	354.34464	42.1%	27.6	65s
146583 88509	447.72035	31 6	6 590.00000	402.72441	31.7%	23.6	600s
147934 89260	549.35810	41 6	5 590.00000	402.91520	31.7%	23.6	605s
149758 90263	516.43027	44 6	7 590.00000	403.18118	31.7%	23.6	610s
1660356 68563	38 563.64870	59	55 586.000	00 469.627	71 19.9	18 19.7	3000s
1664829 68702	24 cutoff	48	586.000	00 469.7104	47 19.8	3% 19.7	3005s

◆ロト ◆聞 ▶ ◆ 臣 ▶ ◆ 臣 ・ の Q ()・

Initial bounds vs. the muddled middle

• • •							
Nodes	Current	Node	Objec	tive Bounds	1	Wor	k
Expl Unexpl	Obj Depth	IntInf	Incumbent	BestBd	Gap	It/Node	e Time
0 0	348.00000	0 27	724.00000	348.00000	51.9%	-	0s
H 5117 3068			622.0000000	353.58269	43.2%	28.2	61s
* 6157 3790		38	612.0000000	354.34464	42.1%	27.6	64s
6179 3811	382.04491	29 78	612.00000	354.34464	42.1%	27.6	65s
146583 88509	447.72035	31 6	56 590.00000	402.72441	31.7%	23.6	600s
147934 89260	549.35810	41 6	55 590.00000	402.91520	31.7%	23.6	605s
149758 90263	516.43027	44 6	57 590.00000	403.18118	31.7%	23.6	610s
1660356 6856	38 563.64870	59	55 586.000	00 469.627	71 19.9)% 19.7	3000s
1664829 6870	24 cutoff	48	586.000	00 469.7104	17 19.8	3% 19.7	3005s

▲ロト▲聞を▲目を▲目を 目 めんぐ

Lopes

Initial bounds vs. the muddled middle

• • •									
Node	s	Current	Node	1	Object	ive Bounds	1	Wor	ck
Expl Une	expl	Obj Deptl	n IntI	nf	Incumbent	BestBd	Gap	It/Node	e Time
• • •									
0	0	348.00000	0	27 '	724.00000	348.00000	51.9%	-	0s
H 5117	3068			622	2.0000000	353.58269	43.2%	28.2	61s
* 6157	3790		38	613	2.0000000	354.34464	42.1%	27.6	64s
6179	3811	382.04491	29	78	612.00000	354.34464	42.1%	27.6	65s
146583	88509	447.72035	31	66	590.00000	402.72441	31.7%	23.6	600s
147934	89260	549.35810	41	65	590.00000	402.91520	31.7%	23.6	605s
149758	90263	516.43027	44	67	590.00000	403.18118	31.7%	23.6	610s
1660356	68563	88 563.64870	0 59	5.	5 586.0000	0 469.6277	71 19.9	98 19.7	7 3000s
1664829	68702	4 cutof:	f 48	3	586.0000	0 469.7104	47 19.8	3% 19.7	7 3005s

▲ロト▲聞を▲目を▲目を 目 めんぐ

Lopes

Rule of thumb: use more variables in each constraint

If we use more variables in the constraints, when some of them get fixed, the resulting LP bounds tend to be tigher.

Observation:

If we assume that f_1 was assigned to l_1 , then for each f_2 : the distance between f_1 and f_2 must be at least the sum of the distances out of l_1 minus the sum of all the other distances except that between l_1 and l_2 when f_2 is assigned to l_2 :

►
$$y_{f_1f_2} \ge \sum_{l_2 \neq l_1} d_{l_1l_2} x_{f_1l_1} - \sum_{f_2 \neq f_1} (\sum_{l_3 \notin \{l_1, l_2\}} d_{l_1l_3}) x_{f_2l_2}, \forall f_1, f_2, l_1 : f_1 \neq f_2$$

In AMPL:

Monash University

イロト 不得 とくほ とくほ とうほう

Diagnosing and Improving our Formulation

Step 1: Strengthen cuts by adding more variables to each constraint

Solver Results

Much better...

Explored 2,855,769 nodes (19,376,754 simplex iterations) in 800.96 seconds Thread count was 4 (of 4 available processors)

Much better. Let's see the details.



Monash University

э

イロン イボン イヨン イヨン

Lopes

Diagnosing and Improving our Formulation

Step 1: Strengthen cuts by adding more variables to each constraint

A healthy tree

• • •								
Nodes	Current 1	Node	- 1	Objecti	ve Bounds.		Work	2
Expl Unexpl	Obj Depth	IntIr	nf	Incumbent	BestBd	Gap 3	It/Node	Time
148565 58745	472.38928	42	41	578.00000	426.17068	26.3%	11.0	60s
162715 63069	550.49347	51	41	578.00000	428.39964	25.9%	10.9	65s
176814 67294	493.66237	43	41	578.00000	430.60198	25.5%	10.8	70s
2352964 205490	cutoff	32		578.00000	525.2458	5 9 . 13	8 7.3	600s
2374875 201750	cutoff	47		578.00000	526.1426	1 8.97	8 7.3	605s
2398390 197345	536.17206	41	43	3 578.00000	527.1724	4 8.79	8 7.3	610s

Monash University

æ

・ロ・・ (日・・ (日・・ (日・))

Diagnosing and Improving our Formulation

Step 1: Strengthen cuts by adding more variables to each constraint

A healthy tree

• • •								
Nodes	Current 1	lode		Objecti	ve Bounds		Work	2
Expl Unexpl	Obj Depth	IntIr	nf	Incumbent	BestBd	Gap 1	[t/Node	Time
148565 58745	472.38928	42	41	578.00000	426.17068	26.3%	11.0	60s
162715 63069	550.49347	51	41	578.00000	428.39964	25.9%	10.9	65s
176814 67294	493.66237	43	41	578.00000	430.60198	25.5%	10.8	70s
2352964 205490) cutoff	32		578.00000	525.2458	5 9.13%	s 7.3	600s
2374875 201750) cutoff	47		578.00000	526.1426	1 8.97%	\$ 7.3	605s
2398390 197345	536.17206	41	43	3 578.00000	527.1724	4 8.798	\$ 7.3	610s

Diagnosing and Improving our Formulation

Step 1: Strengthen cuts by adding more variables to each constraint

A healthy tree

• • •								
Nodes	Current 1	Node		Objecti	ve Bounds		Wor}	c
Expl Unexpl	Obj Depth	IntInf		Incumbent	BestBd	Gap	It/Node	Time
• • •								
148565 58745	472.38928	42 43	1	578.00000	426.17068	26.3%	11.0	60s
162715 63069	550.49347	51 43	1	578.00000	428.39964	25.9%	10.9	65s
176814 67294	493.66237	43 43	1	578.00000	430.60198	25.5%	10.8	70s
2352964 205490) cutoff	32		578.00000	525.2458	5 9.13	8 7.3	600s
2374875 201750) cutoff	47		578.00000	526.1426	1 8.97	8 7.3	605s
2398390 197345	536.17206	41	43	578.00000	527.1724	4 8.79	8 7.3	610s

. . .

・ロ・・母・・ヨ・・ヨ・ のへぐ

Lopes

Diagnosing and Improving our Formulatio

Step 1: Strengthen cuts by adding more variables to each constraint

A healthy tree

•••								
Nodes	Current 1	lode		Objecti	ve Bounds.	1	Work	2
Expl Unexpl	Obj Depth	IntIn	E	Incumbent	BestBd	Gap	It/Node	Time
• • •								
148565 58745	472.38928	42 4	41	578.00000	426.17068	26.3%	11.0	60s
162715 63069	550.49347	51 4	41	578.00000	428.39964	25.9%	10.9	65s
176814 67294	493.66237	43 4	41	578.00000	430.60198	25.5%	10.8	70s
2352964 205490) cutoff	32		578.00000	525.2458	5 9.13	8 7.3	600s
2374875 201750) cutoff	47		578.00000	526.1426	1 8.97	8 7.3	605s
2398390 197345	5 536.17206	41	43	578.00000	527.1724	4 8.79	8 7.3	610s

・ロト・部・・ヨト・ヨー りへぐ

Lopes

Lopes

Diagnosing and Improving our Formulatio

Step 1: Strengthen cuts by adding more variables to each constraint

A healthy tree

• • •								
Nodes	Current 1	Node		Objecti	lve Bounds	1	Work	2
Expl Unexpl	Obj Depth	IntIr	nf	Incumbent	BestBd	Gap	It/Node	Time
148565 58745	472.38928	42	41	578.00000	426.17068	26.3%	11.0	60s
162715 63069	550.49347	51	41	578.00000	428.39964	25.9%	10.9	65s
176814 67294	493.66237	43	41	578.00000	430.60198	25.5%	10.8	70s
2352964 205490) cutoff	32		578.00000	525.2458	5 9.13	8 7.3	600s
2374875 201750) cutoff	47		578.00000	526.1426	1 8.97	8 7.3	605s
2398390 197345	5 536.17206	41	43	3 578.00000	527.1724	4 8.79	8 7.3	610s

Monash University

æ

・ロ・・ (日・・ (日・・ (日・))

Smaller constants tend to be better

- The sums over all distances are likely making many of the constraints non-binding in the LP relaxations. A formulation with smaller numbers will usually perform better.
- ► Since $\sum_{l} x_{fl} = 1$, then either $x_{fl} = 1$ or $\sum_{l_2 \neq l} x_{fl_2} = 1$. In the former case, $y_{f_1 f_2}$ will be d_{ll_2} . In the latter case, then discounting $\max_{l_2 \neq l} d_{ll_2}$ from $y_{f_1 f_2}$ is just enough to make the constraint non-binding.
- In algebra:

$$y_{f_1f_2} \ge \sum_{l_2 \neq l_1} d_{l_1l_2} x_{f_2l_2} - (\max_{l_2 \neq l_1} d_{l_1l_2}) \sum_{l_2 \neq l_1} x_{f_1l_2}, \forall f_1, f_2, l_1 : f_1 \neq f_2$$

► Rearranging and reorganizing terms: $M_l = \max_{l_2 \neq l} d_{l_2}$ and $y_{f_1 f_2} \ge \sum_{l_2 \neq l} (d_{l_2} x_{f_2 l_2} - M_l x_{f_1 l_2}), \forall f_1, f_2, l : f_1 \neq f_2$ ► In AMPL :

```
param maxd{l in LOC} := max{l2 in LOC: l2 <> l} d[l,l2];
subject to ComputeDist3{f1 in FAC, f2 in FAC,l in LOC: f1 <> f2}:
Dist[f1,f2] >= sum{l2 in LOC: l2 <> l} (
    d[l,l2]*Assign[f2,l2] - maxd[l]* Assign[f1,l2]
);
```

Step 2: Smaller constants and more terms

Solver Results

Total time cut in half again...

Explored 723,627 nodes (12,293,156 simplex iterations) in 428.08 seconds Thread count was 4 (of 4 available processors)

Much better again. Let's see the details.

Lopes

Monash University

э

A healthier tree

Nodes	Curren	t Node		Object	ive Bounds		Wo	rk
Expl Unexpl	l Obj Dep	th IntIn	f	Incumbent	BestBd	Gap	It/Nod	e Time
41971 2062	0 578.17341	42 5	7	586.00000	426.84741	27.2%	26.7	60s
49310 23712	2 506.64911	34 6	8	586.00000	430.28747	26.6%	26.2	65s
56491 2651	3 580.49821	55 6	4	586.00000	433.86992	26.0%	25.9	70s
443823 8592	25 cutoff	35		578.00000	517.55057	10.5%	19.6	300s
454929 852	47 572.52381	38	46	578.00000	518.97396	10.2%	19.5	305s
461098 8493	30 cutoff	29		578.00000	519.80133	10.1%	19.4	310s
676266 341	74 cutoff	39		578.00000	552.77727	4.36%	17.6	410s
687701 271	47 cutoff	50		578.00000	556.03894	3.80%	17.5	415s
699749 188	46 cutoff	30		578.00000	560.31293	3.06%	17.4	420s
713063 888	85 568.11389	37	59	578.00000	566.75240	1.95%	17.2	425s

Explored 723627 nodes (12293156 simplex iterations) in 428.08 seconds Thread count was 4 (of 4 available processors)

E ∽ Q (~ Monash University

A healthier tree

Node	es	Curre	nt Node		Object	tive Bounds		Wo	rk
Expl U	nexpl	Obj Dej	pth Int	Inf	Incumbent	BestBd	Gap	It/Nod	e Time
41971 2	20620	578.17341	42	57	586.00000	426.84741	27.2%	26.7	60s
49310 2	23712	506.64911	34	68	586.00000	430.28747	26.6%	26.2	65s
56491 2	26513	580.49821	55	64	586.00000	433.86992	26.0%	25.9	70s
443823	85925	cutof	f 35		578.00000	517.55057	10.5%	19.6	300s
454929	85247	572.5238	1 38	46	578.00000	518.97396	10.2%	19.5	305s
461098	84930	cutof	f 29		578.00000	519.80133	10.1%	19.4	310s
676266	34174	cutof	f 39		578.00000	552.77727	4.36%	17.6	410s
687701	27147	cutof	f 50		578.00000	556.03894	3.80%	17.5	415s
699749	18846	cutof	£ 30		578.00000	560.31293	3.06%	17.4	420s
713063	8885	568.1138	9 37	59	578.00000	566.75240	1.95%	17.2	425s

Explored 723627 nodes (12293156 simplex iterations) in 428.08 seconds Thread count was 4 (of 4 available processors)

э

A healthier tree

Node	es	Curre	ent Node	Э	Objec	tive Bounds	1	Wc	ork
Expl Ur	nexpl	Obj De	epth Int	tInf	Incumbent	BestBd	Gap	It/Nod	le Time
•••									
41971 2	20620	578.1734	1 42	57	586.00000	426.84741	27.2%	26.7	60s
49310 2	23712	506.6491	1 34	68	586.00000	430.28747	26.6%	26.2	65s
56491 2	26513	580.4982	1 55	64	586.00000	433.86992	26.0%	25.9	70s
443823	85925	cuto	ff 35		578.00000	517.55057	10.5%	19.6	300s
454929	85247	572.523	81 38	46	578.00000	518.97396	10.2%	19.5	305s
461098	84930	cuto	ff 29		578.00000	519.80133	10.1%	19.4	310s
676266	34174	cuto	ff 39		578.00000	552.77727	4.36%	17.6	410s
687701	27147	cuto	ff 50		578.00000	556.03894	3.80%	17.5	415s
699749	18846	cuto	ff 30		578.00000	560.31293	3.06%	17.4	420s
713063	8885	568.113	89 37	59	578.00000	566.75240	1.95%	17.2	425s

Explored 723627 nodes (12293156 simplex iterations) in 428.08 seconds Thread count was 4 (of 4 available processors)

E ∽ Q (~ Monash University

A healthier tree

Nodes	Current	t Node		Object	Work			
Expl Unexpl	. Obj Dept	th IntIn	f	Incumbent	BestBd	Gap	It/Nod	e Time
• • •								
41971 20620	578.17341	42 5	7	586.00000	426.84741	27.2%	26.7	60s
49310 23712	506.64911	34 6	8	586.00000	430.28747	26.6%	26.2	65s
56491 26513	580.49821	55 6	4	586.00000	433.86992	26.0%	25.9	70s
443823 8592	5 cutoff	35		578.00000	517.55057	10.5%	19.6	300s
454929 8524	7 572.52381	38	46	578.00000	518.97396	10.2%	19.5	305s
461098 8493	0 cutoff	29		578.00000	519.80133	10.1%	19.4	310s
676266 3417	4 cutoff	39		578.00000	552.77727	4.36%	17.6	410s
687701 2714	7 cutoff	50		578.00000	556.03894	3.80%	17.5	415s
699749 1884	6 cutoff	30		578.00000	560.31293	3.06%	17.4	420s
713063 888	5 568.11389	37	59	578.00000	566.75240	1.95%	17.2	425s

Explored 723627 nodes (12293156 simplex iterations) in 428.08 seconds Thread count was 4 (of 4 available processors)

э

A healthier tree

Nodes	Current	Node		Object	Work			
Expl Unexpl	Obj Dept	h IntIr	nf	Incumbent	BestBd	Gap	It/Nod	e Time
41971 20620	578.17341	42 5	57	586.00000	426.84741	27.2%	26.7	60s
49310 23712	506.64911	34 6	58	586.00000	430.28747	26.6%	26.2	65s
56491 26513	580.49821	55 (54	586.00000	433.86992	26.0%	25.9	70s
443823 8592	5 cutoff	35		578.00000	517.55057	10.5%	19.6	300s
454929 8524	7 572.52381	38	46	578.00000	518.97396	10.2%	19.5	305s
461098 8493	0 cutoff	29		578.00000	519.80133	10.1%	19.4	310s
676266 3417	4 cutoff	39		578.00000	552.77727	4.36%	17.6	410s
687701 2714	7 cutoff	50		578.00000	556.03894	3.80%	17.5	415s
699749 1884	6 cutoff	30		578.00000	560.31293	3.06%	17.4	420s
713063 888	5 568.11389	37	59	578.00000	566.75240	1.95%	17.2	425s

Explored 723627 nodes (12293156 simplex iterations) in 428.08 seconds Thread count was 4 (of 4 available processors)

э

A healthier tree

Nodes Current Node					le	Objec	1	ork		
	Expl U	nexpl	Obj	Depth Ir	ntInf	Incumbent	BestBd	Gap	It/Noc	de Time
	•••									
	41971 2	20620	578.173	41 42	57	586.00000	426.84741	27.2%	26.7	60s
	49310 2	23712	506.649	11 34	68	586.00000	430.28747	26.6%	26.2	65s
	56491 2	26513	580.498	21 55	64	586.00000	433.86992	26.0%	25.9	70s
	443823	85925	cut	off 35	5	578.00000	517.55057	10.5%	19.6	300s
	454929	85247	572.52	381 38	3 46	5 578.00000	518.97396	10.2%	19.5	305s
	461098	84930	cut	off 29)	578.00000	519.80133	10.1%	19.4	310s
	676266	34174	cut	off 39)	578.00000	552.77727	4.36%	17.6	410s
	687701	27147	cut	off 50)	578.00000	556.03894	3.80%	17.5	415s
	699749	18846	cut	off 30)	578.00000	560.31293	3.06%	17.4	420s
	713063	8885	568.11	389 37	7 59	578.00000	566.75240	1.95%	17.2	425s

Explored 723627 nodes (12293156 simplex iterations) in 428.08 seconds Thread count was 4 (of 4 available processors)

э

イロト イポト イヨト イヨト

Modeling Tools and solvers are powerful – but not automatic *yet*

- I am not endorsing these formulations as practical ways for solving QAPs
 - Even stronger bounds computed using more sophisticated machinery are needed for larger instances
- Modeling Languages allow experimentation with new formulations at very low cost
- The general principle is to define constraints as tightly as possible.
 - Dramatic improvements can be obtained with very little extra programming effort.
 - Your real world "branch" problem likely contains a "core" problem (sequencing, coloring, QAP). Check the academic literature for strong formulations of those problems.
 - Analytical results showing that one constraint is dominated by another are common and appreciated in the academic literature.
 - This formulation can still be improved. How?

Diagnosing and Improving our Formulation

< ロ > < 同 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ >

Thanks for listening!

Parting words: put some thought into the problem and experiment with different formulations. It produces results (and is fun to do!)

Lopes