Constraint Branching and Disjunctive Cuts for Mixed Integer Programs

Michael Perregaard

Dash Optimization

Small Example

$$\begin{aligned} \min \quad & z \\ \text{s.t.} \quad & 2x_1 + 2x_2 + z \geq 99 \\ & x_1, x_2, z \in \mathbb{Z}_+ \end{aligned}$$

Optimal LP solution: $z = 0$

Optimal MIP solution: $z = 1$

Consider pure branch-and-bound.

Will alternately branch on fractional x_1 or x_2.

Requires exhaustive search of

$(x_1, x_2) = (0,49.5), (0.5,49), (1,48.5), \ldots, (49.5, 0)$

100 solutions to search. 100 times more with new x_3.

Alternatively, branch on $x_1 + x_2 \leq 49 \vee x_1 + x_2 \geq 50$

Branching from Disjunctive Cuts

- Branching is imposing a disjunction valid for all (feasible) integer solutions, but not the current LP solution.
- Disjunctive cuts are derived from some base disjunction and often a strengthening argument.
  - Gomory's Mixed Integer cuts.
  - Lift-and-Project cuts.
  - Reduce and Split cuts of Andersen, Cornuéjols and Li (2003).
- The strengthening of the cut can be transformed into a strengthening of the base disjunction.
- Use the strengthened base disjunction for branching.

Basic Mixed Integer Program

We consider solving:

$$
\begin{aligned}
\min \quad & cx \\
s.t. \quad & Ax = b \\
& x \geq 0 \\
& x_j \in \mathbb{Z} \quad \text{for } j \in I
\end{aligned}
$$

- Solve using branch-and-bound.
- Standard branching selects a single fractional variable x_j and imposes disjunction

$$x_j \leq \lfloor \bar{x}_j \rfloor \vee x_j \geq \lceil \bar{x}_j \rceil$$

- Can we find a better disjunction?

Disjunctive Normal Form

$$\bigvee_{q \in Q} D^q x \geq d_0^q$$

Example

For constraint

$$\sum_{j=1}^{k} x_j = 1$$

where e.g. x_1 and x_2 are fractional, we can create a disjunction

$$x_1 \geq 1 \;\vee\; x_2 \geq 1 \;\vee\; x_1 + x_2 \leq 0$$

Split Disjunctions

$$\bigwedge_{q\in Q} \left(d_I^q x_I \le d_0^q \vee d_I^q x_I \ge d_0^q + 1\right)$$

where $(d_I^q, d_0^q) \in Z^{|I|+1}$ for $q \in Q$.

<u>Example</u>

If x_1, x_2 and x_3 are fractional binaries, we can consider the disjunction

$$(x_1 \le 0 \vee x_1 \ge 1) \wedge (x_2 \le 0 \vee x_2 \ge 1) \wedge (x_3 \le 0 \vee x_3 \ge 1)$$

Leads to $2^3 = 8$ branches.

Basic Disjunctive (Intersection) Cut

Given disjunction (in nonbasic space)

$$\bigvee_{q \in Q} \bar{d}_N^q x_N \geq \bar{d}_0^q$$

where $\bar{d}_0^q > 0$, then

$$\alpha_N x_N \geq 1$$

with

$$\alpha_j = \max_{q \in Q} \left\{ \frac{\bar{d}_j^q}{\bar{d}_0^q} \right\}$$

is a valid inequality that cuts off the LP solution $\bar{x}_N = 0$.

Strengthening Disjunctions

(**Balas, Jeroslaw 1980**)

Let f_q be the largest value for which

$$\bar{d}_N^q x_N \geq f_q$$

is valid for (MIP). Set $h_q = (\bar{d}_0^q - f_q)$.

Let $m_j^q \in Z^q$ for $j \in I$, $q \in Q$, be any set of integers that satisfies $\sum_{q\in Q} m_j^q \geq 0$ for all $j \in I$. Then

$$\bar{\alpha}_N x_N \geq 1$$

with

$$\bar{\alpha}_j = \max_{q\in Q}\{(\bar{d}_j^q + m_j^q h_q)/\bar{d}_0^q\} \text{ for } j \in I$$

$$\bar{\alpha}_j = \max_{q\in Q}\{\bar{d}_j^q/\bar{d}_0^q\} \text{ for } j \in N \setminus I$$

is a valid inequality for (MIP)

Strengthening Disjunctions [continued]

Instead of strengthening cut, as in

$$\overline{\alpha}_j = \max_{q \in Q}\{(\overline{d}_j^q + m_j^q h_q) / \overline{d}_0^q\} \quad \text{for } j \in I$$

$$\overline{\alpha}_j = \max_{q \in Q}\{\overline{d}_j^q / \overline{d}_0^q\} \quad \text{for } j \in N \setminus I$$

modify the disjunction directly, as in

$$\bigvee_{q \in Q} (\overline{d}_I^q + m_I^q h_q) x_I + \overline{d}_{N \setminus I}^q x_{N \setminus I} \geq \overline{d}_0^q \qquad (*)$$

Basic disjunctive cut from (*) identical to strengthened cut.

Strengthening Conjunctions

Given valid disjunction for (MIP)

$$\bigwedge_{q\in Q} (d_I^q x \leq d_0^q \vee d_I^q x \geq d_0^q + 1)$$

Let $m_j^q \in \mathbb{Z}^q$ for $j \in I$, $q \in Q$, be any set of integers. Then

$$\bigwedge_{q\in Q} \left((d_I^q + m_I^q)x_I \leq d_0^q \bigvee (d_I^q + m_I^q)x_I \geq d_0^q + 1\right)$$

is a valid disjunction for (MIP) since $m_I^q x_I$ must be integer.

- Gomory's Mixed Integer cuts and Lift-and-Project cuts strengthens in nonbasics.
- Andersen, Cornuéjols, Li cuts iteratively strengthens individual basics and all nonbasics.

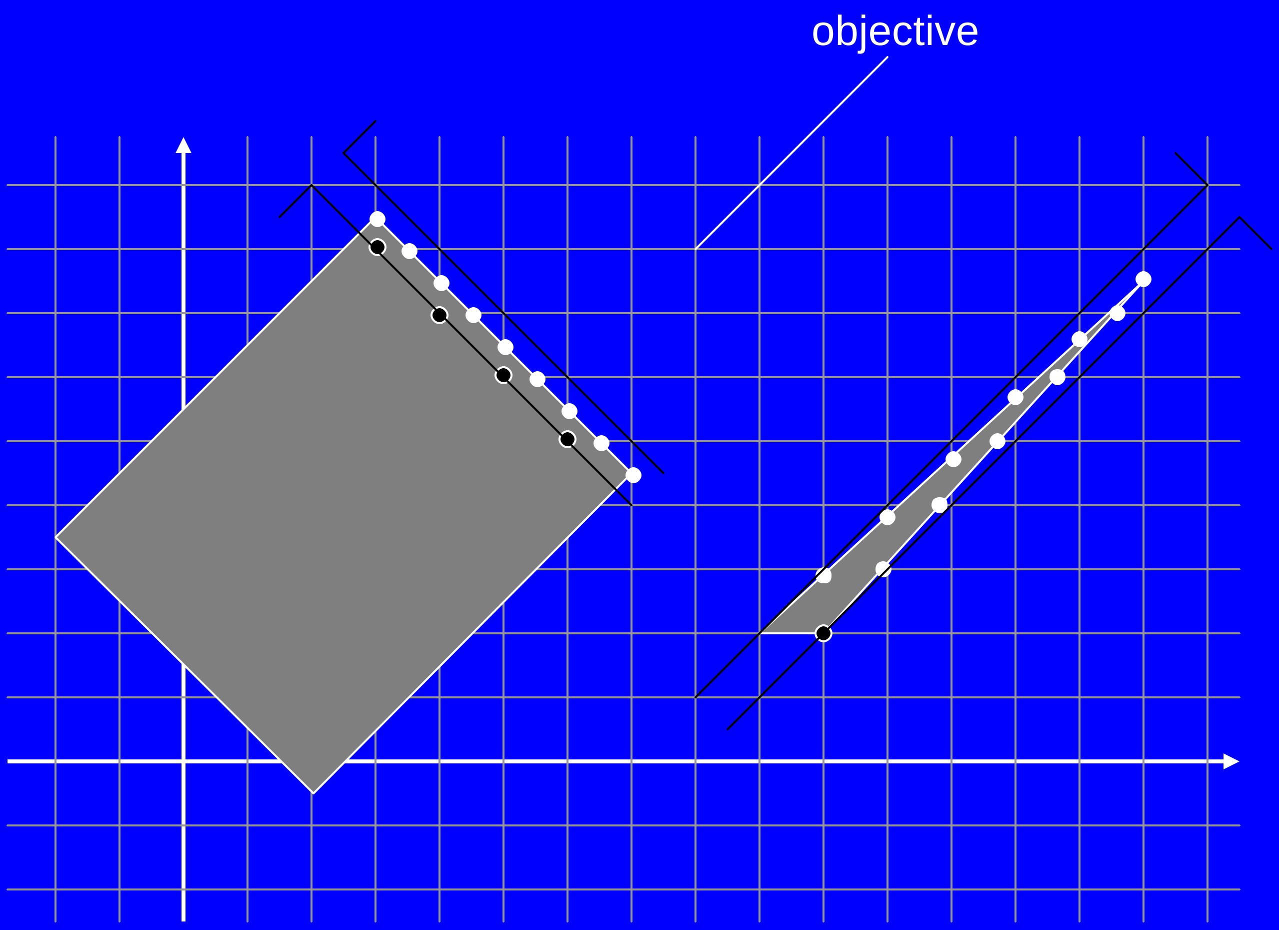
General Branching Alternatives

- Ryan-Foster for Set Packing and Set Partitioning.
  - B.A. Foster and D.M. Ryan (1981).
  - Specifically designed for Set Partitioning constraints:
- Basis Reduction
  - H.W. Lenstra (1983)
  - Polynomial algorithm for solving integer programs for fixed number of variables.
- General Branching of Mehrotra, Owen (2001)
  - Tests each variable using LP reoptimization to determine "best" coefficient.

Even more Alternatives

- General Branching of Karamanov, Cornjuéjols (Monday)
  - Branches on Gomory cut related disjunction.
- Column Basis Reduction of Pataki (Thursday)
- Generalized Branching Methods of Mehrotra (Friday)

Small Examples



General vs. 0-1 branching

General branching		0-1 branching	
+	Branch on any linear disjunction.	-	Branch on 0-1 disjunctions only
-	Adds new constraints $\Rightarrow$ matrix size grows.	+	Changes bounds $\Rightarrow$ matrix size unchanged.
-	More difficult to get implications.	+	Easy to get implications (bound propagation).
-	More basic integers $\Rightarrow$ less reduced cost tightening.	+	Branched variables will be non-basic $\Rightarrow$ allows reduced cost tightening.
?	Heaps of choices	?	Easy to find "best" choice.

Evaluating a Disjunction

Work in space of *nonbasic* variables

$$A_B x_B + A_N x_N = b \quad \Rightarrow \quad x_B = A_B^{-1} b - A_B^{-1} A_N x_N$$

Measure the quality of a disjunction

$$\bigvee_{q \in Q} \bar{d}_N^q x_N \geq \bar{d}_0^q$$

through that of the implied disjunctive cut $\alpha x \geq 1$, with

$$\alpha_j = \max_{q \in Q} \left\{ \bar{d}_j^q \middle/ \bar{d}_0^q \right\}$$

Evaluating a Cut

$$\alpha_N x_N \geq 1$$

Andersen, Cornuéjols, Li (2003) suggests minimizing the L_2-norm of cut coefficients for continuous variables.

What about scaling and cost? Consider reduced costs $\bar{c}_N$.

Cost to satisfy the cut by increasing non-basic variable x_j is at least $\bar{c}_j / \alpha_j$.

Make cut expensive to satisfy $\Rightarrow$ maximize $\bar{c}_j / \alpha_j$, or minimize $\alpha_j / \bar{c}_j$.

Since $\bar{c}_j$ can be zero, we estimate a cut by

$$g(\alpha) = \sum_{j \in N} \frac{1}{1 + \bar{c}_j / \delta} \alpha_j$$

Improving a Disjunction - Nonbasics

Express disjunction in nonbasics x_N

$$\bigvee_{q\in Q} d_B^q x_B + d_N^q x_N \geq d_0^q \quad \Rightarrow \quad \bigvee_{q\in Q} \bar{d}_N^q x_N \geq \bar{d}_0^q$$

Strengthened cut coefficients in nonbasics are

$$\bar{\alpha}_j = \max_{q\in Q}\{(\bar{d}_j^q + m_j^q h_q)/\bar{d}_0^q\} \text{ for } j \in N \cap I$$

$$\bar{\alpha}_j = \max_{q\in Q}\{\bar{d}_j^q / \bar{d}_0^q\} \qquad \text{ for } j \in N \setminus I$$

Find optimal m_j^q for each j independently $\Rightarrow$ easy.

Note: For simple split disjunction $x_k \geq \lceil \bar{x}_k \rceil \vee x_k \leq \lfloor \bar{x}_k \rfloor$ optimal m_j^q gives Gomory's Mixed Integer cut.

Improving a Disjunction - Basics

Same strengthening applies to basic variable x_k

$$\bigvee_{q \in Q} m_k^q h_q x_k + \bar{d}_N^q x_N \geq \bar{d}_0^q$$

Use the row i of the *simplex tableau* in which x_k is basic:

$$x_k + \sum_{j \in N} \bar{a}_{ij} x_j = \bar{b}_i$$

to re-express the disjunction in nonbasics:

$$\bigvee_{q \in Q} (\bar{d}_N^q - m_k^q h_q \bar{a}_{iN}) x_N \geq \bar{d}_0^q - m_k^q h_q \bar{a}_{i0}$$

Problem: Find optimal discrete amount $-m_k^q h_q$ to add simplex tableau row i (without basic x_k) to each term q of the disjunction.

Procedure

1. Convert Xpress selected branching variable x_k into a simple disjunction
2. Apply Gomory-esque strengthening to coefficients of non-basics in *D*.
3. Are there more basic, integer variables to use for strengthening? If not, stop.
4. Select basic, integer variable x_i. Calculate optimal continuous coefficient m_i in *D*. Update *D* with the better of $\lfloor m_i \rfloor$ or $\lceil m_i \rceil$. Repeat from 2.

Test Set

- Miplib 3
  - http://www.caam.rice.edu/~bixby/miplib/miplib.html
  - 65 instances, 15 with general integers
- Miplib 2003
  - http://miplib.zib.de/
  - 61 instances, 15 with general integers
- H. Mittelmann's test set
  - http://plato.la.asu.edu/bench.html
  - 63 instances, 6 with general integers

146 unique instances, 30 with general integers.

Instances with General Integers

Name	Rows	Columns	Binaries	Integers	Int.Gap
arki001	1049	1388	415	123	5
atlanta-ip	21732	48738	46667	106	2.127e9
bell3a	***124***	***133***	***39***	***32***	***1000***
bell5	***92***	***104***	***30***	***28***	***3379***
blend2	275	353	231	33	2
flugpl	***19***	***18***	***0***	***11***	***18***
gen	781	870	144	6	24
gesa2	1393	1224	240	168	3
gesa2_o	1249	1224	384	336	3
gesa3	1369	1152	216	168	3
gesa3_o	1225	1152	336	336	3
gt2	***30***	***188***	***24***	***164***	***7***
manna81	6481	3321	18	3303	7
momentum2	24238	3732	1808	1	100
momentum3	56822	13532	6598	1	150

Name	Rows	Columns	Binaries	Integers	Int.Gap
msc98-ip	15851	21143	20237	53	2.106e9
mzzv11	9500	10240	9989	251	35
mzzv42z	10461	11717	11482	235	26
neos7	1994	1556	434	20	380
neos8	46324	23228	23224	4	900
neos10	46793	23489	23484	5	900
neos16	1019	377	336	41	9
neos20	2446	1165	937	30	267
noswot	183	128	75	25	100000
qnet1	504	1541	1288	129	5
qnet1_o	457	1541	1288	129	5
roll3000	2296	1166	246	492	2
rout	292	556	300	15	2
timtab1	171	397	64	107	2
timtab2	295	675	113	181	3

- Instances not suited for general integer branching.

Computational Settings

- Implemented in C using Xpress 2005B optimizer library.
- Uses Xpress callbacks to override default branches with new constraint branches.
- No in-tree cutting.
- No heuristics.
- Best-first search.
- Run on a dual processor Opteron 246 system (2GHz, 4GB RAM, Linux OS).

Nonbasic Strengthening

	No Strengthening		Simple Strengthening	
Instance	Time	Nodes (Bound)	Time	Nodes (Bound)
arki001	*	(7580565)	*	(7580295)
atlanta-ip	*	(82.88)	*	(82.92)
bell3a	24	**52105**	0	**413**
bell5	446	**384977**	1	**1357**
blend2	6	7563	5	6695
dsbmip	1	65	1	65
flugpl	0	801	0	329
gesa2_o	1	195	1	229
gesa2	0	25	0	29
gesa3_o	2	129	2	165
gesa3	1	87	1	101
gt2	0	**1335**	0	**33**
msc98-ip	*	(19702878)	*	(19702878)

	No Strengthening		Simple Strengthening	
Instance	Time	Nodes (Bound)	Time	Nodes (Bound)
mzzv11	*	(-21728)	*	(-21728)
mzzv42z	1060	9017	1099	9017
neos10	265	421	271	421
neos16	*	(434)	*	(432)
neos20	*	(-461)	*	(-468)
neos7	*	(709934)	*	(713934)
qnet1_o	1	29	3	135
qnet1	2	59	4	103
roll3000	*	(12453)	*	(12456)
rout	574	228537	*	(1047)
timtab1	*	(644157)	*	(570727)
timtab2	*	(695712)	*	(660029)

* Not finished in 1800 seconds
gen, manna81: solved on root (excluded).
noswot: can't raise bound (excluded).
atlanta-ip, dsbmip, msc98-ip, mzzv11, mzzv42z, neos10: very few branches on integers

Full Strengthening

	No Strengthening		Full Strengthening	
Instance	Time	Nodes (Bound)	Time	Nodes (Bound)
arki001	*	(7580565)	*	(7580052)
atlanta-ip	*	(82.88)	*	(82.89)
bell3a	24	**52105**	0	**195**
bell5	446	**384977**	0	**611**
blend2	6	7563	5	6879
dsbmip	1	65	1	65
flugpl	0	**801**	0	**31**
gesa2_o	1	195	1	121
gesa2	0	25	0	35
gesa3_o	2	129	2	129
gesa3	1	87	2	105
gt2	0	**1335**	0	**39**
msc98-ip	*	(19702878)	*	(19702878)

	No Strengthening		Simple Strengthening	
Instance	Time	Nodes (Bound)	Time	Nodes (Bound)
mzzv11	*	(-21728)	*	(-21728)
mzzv42z	1060	9017	1003	9017
neos10	265	421	266	421
neos16	*	(434)	*	(432)
neos20	*	(-461)	*	(-468)
neos7	*	(709934)	*	(713718)
qnet1_o	1	29	6	65
qnet1	2	59	5	39
roll3000	*	(12453)	*	(12459)
rout	574	228537	*	(1053)
timtab1	*	(644157)	*	(552455)
timtab2	*	(695712)	*	(642602)

* Not finished in 1800 seconds.
bell3a, bell5: Half the number of nodes of Nonbasic Strengthening.
flugpl: reduced from 329 to 31 nodes.

Branching on Binaries

Nonbasic Strengthening	No Strengthening	Strengthening on integer branches
#Better	4	3
#Worse	18	17

Comparing results from Nonbasic Strengthening on all Binary/Integer branches against previous results.

Full Strengthening	No Strengthening	Strengthening on integer branches
#Better	7	5
#Worse	17	15

Comparing results from Full Strengthening on all Binary/Integer branches against previous results.

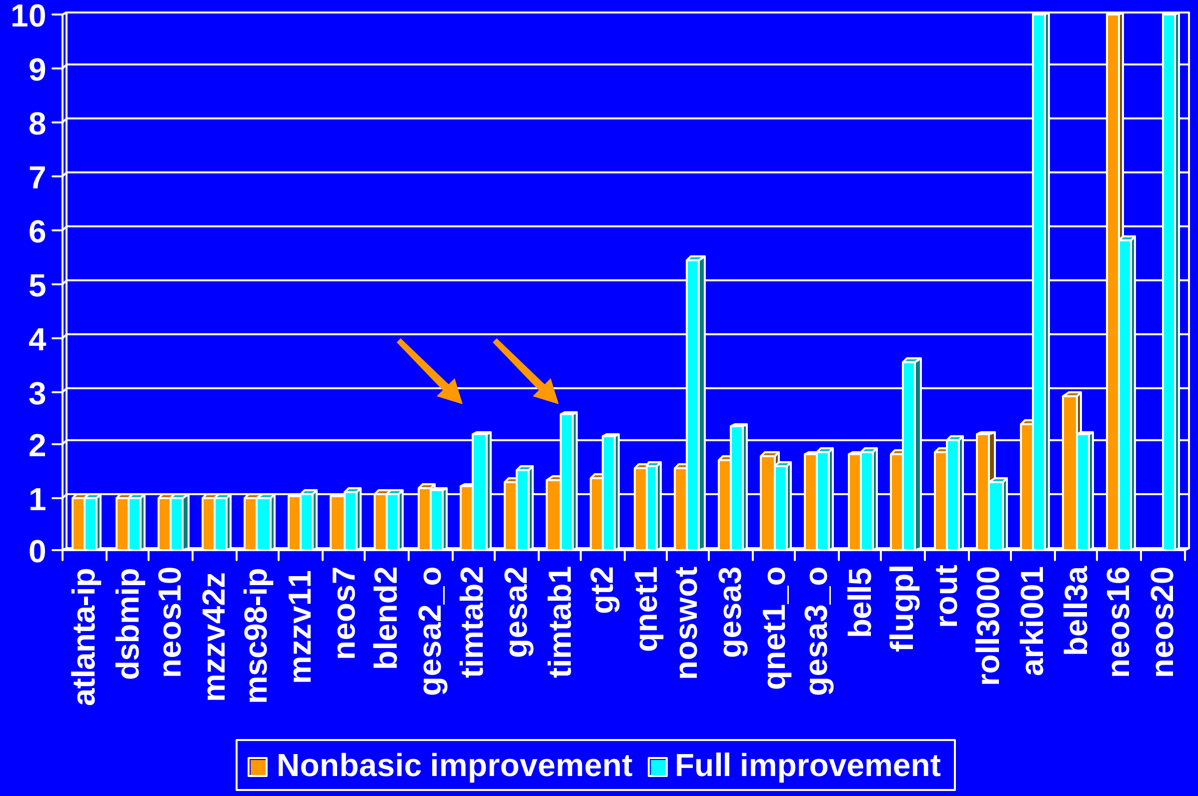
Reduced Cost Scaling of Cut Coefficients

$$f_j = \frac{1}{1 + \bar{c}_j / \delta}$$

	#Best	#Worst
$\delta = \infty$ (no scaling)	6	12
δ = median reduced cost	8	3
δ = 0.1 * median reduced cost.	8	6

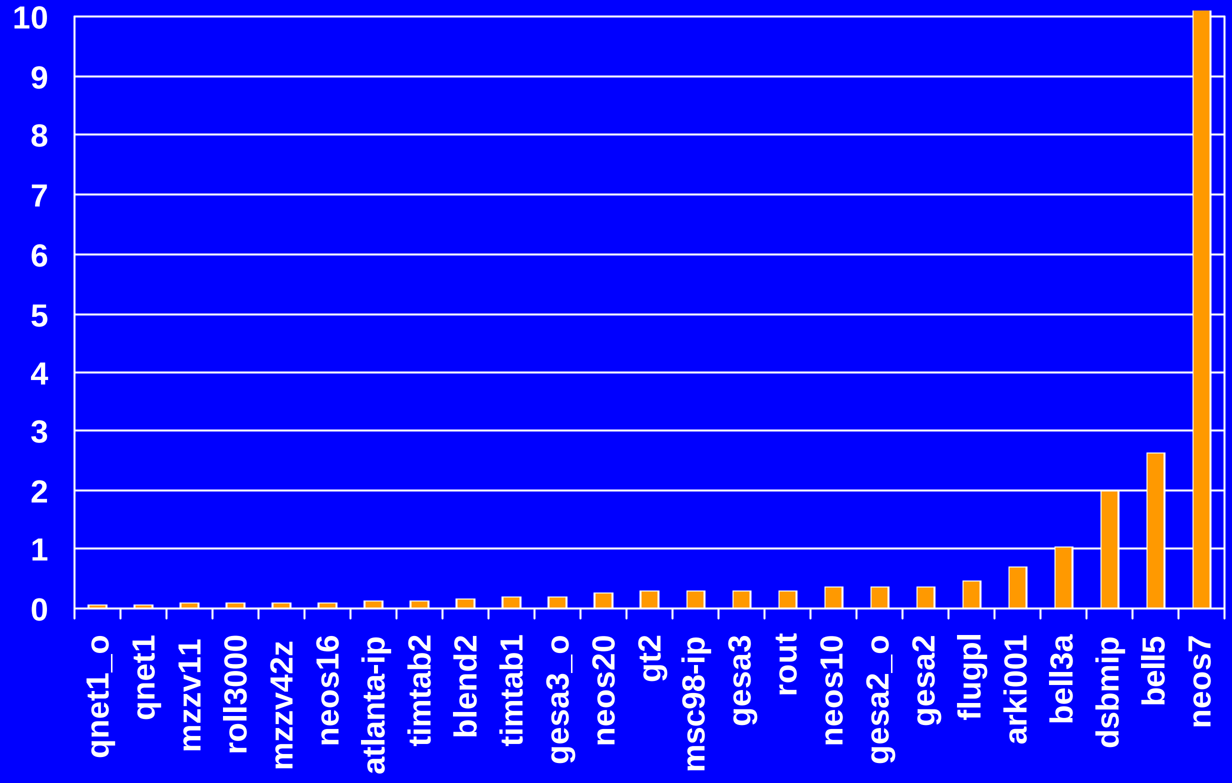
Improvement in Cut Estimate

Average improvement in cut estimate relative to initial disjunction when applying either nonbasic improvement or full improvement.



Basic Improvement Coefficients

Average optimal continuous coefficient for basic integer variables, excluding when zero is optimal.



Results on Full Test Set

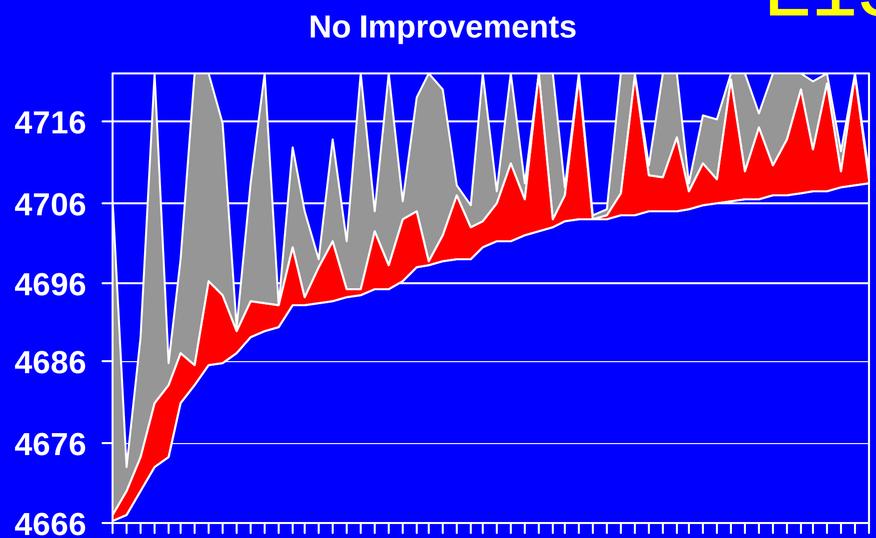
Using full strengthening on both binary and integer branches (137 instances).

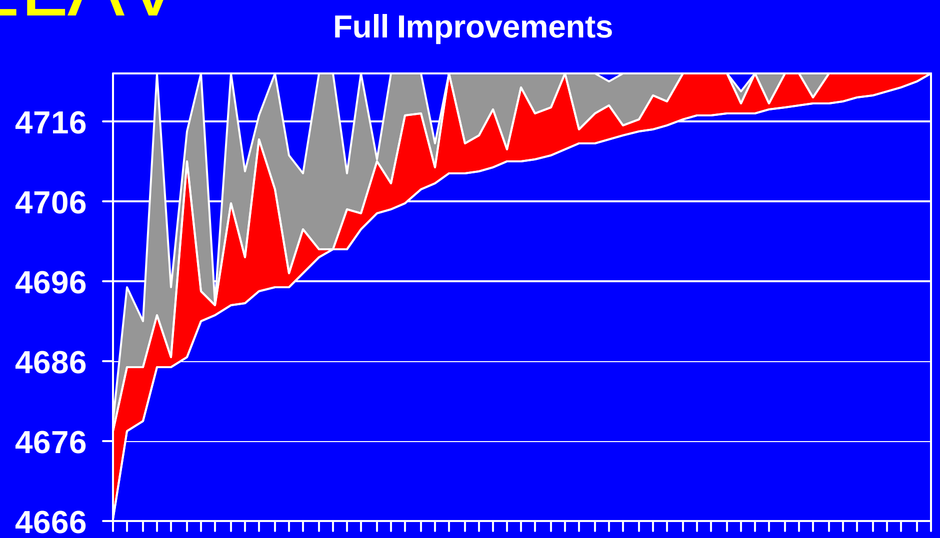
Strengthening?	No	Full
Both finished, least nodes	25	18
One finished	29	2
Both unfinished, best bound	39	0

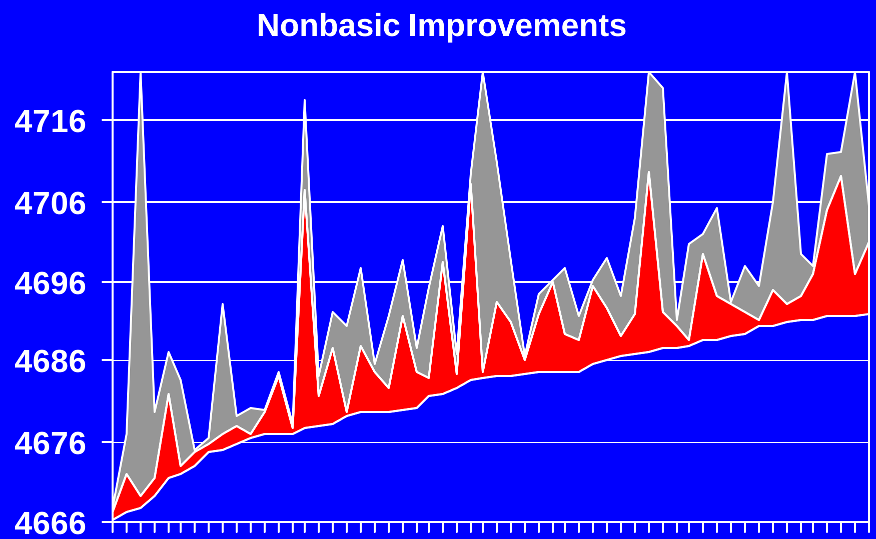
	No strengthening		Full strengthening	
Instance	Time	Nodes	Time	Nodes
bell5	465.42	384977	0.26	531
bell3a	24.45	52105	0.14	183
mod008	0.84	791	0.15	25
gt2	0.48	1335	0.12	43
flugpl	0.15	801	0.01	33
mzzv42z	999.64	9017	681.8	1043
l152lav	5.49	503	4.63	65
lseu	0.52	1267	0.24	187
neos4	1228.9	1167	536.57	203
neos1	47.78	6961	160.39	1971

Top 10 with best performance when applying full strengthening.

L152LAV







Improv.	∑mindeg	∑maxdeg
None	117	512
Nonbas.	134	179
Full	177	255

Sum of min and max degradation over best 25 nodes.

Client Set 1

Small cutting stock problems with general integers.

					Strengthening		
Name	Rows	Cols	Root Objective	Best Sol.	None	Full	Full on cycles only
d3	21	827	7730.5	7732	7730.5 0% (6118006)	7732 100% (37)	7732 100% (227)
d4	20	827	0	75	0 0% (621444)	75 100% (35)	75 100% (10248)
d6	21	827	31266.67	31275	31266.67 0% (903588)	31275 100% (7)	31275 100% (870)

Client Set 2

Lot sizing problems with general integers.

				Strengthening		
Name	Rows	Cols	Root Objective	None	Non-basic	Full
C2	278479	1016708	$1.32206 \cdot 10^{11}$	$1.32235 \cdot 10^{11}$	$1.32207 \cdot 10^{11}$ 0.0008% (561)	$1.32208 \cdot 10^{11}$ 0.0015% (450)
C3	108425	110932	$7.72284 \cdot 10^{8}$	$7.73051 \cdot 10^{8}$	$7.72289 \cdot 10^{8}$ 0.0006% (10071)	$7.72309 \cdot 10^{8}$ 0.0032% (2372)
C4	6530	20365	$2.21516 \cdot 10^{10}$	$2.22042 \cdot 10^{10}$	$2.21876 \cdot 10^{10}$ 0.1621% (39146)	$2.21877 \cdot 10^{10}$ 0.1626% (12213)
C6	2606	5548	$2.50477 \cdot 10^{9}$	$2.50477 \cdot 10^{9}$	$2.50477 \cdot 10^{9}$ 100% (45)	2.50477 100% (45)

Future Directions

- Select initial disjunction independently of Xpress.
- Evaluation of disjunctions.
  Xpress uses e.g. pseudo costs, strongbranch estimates and history values to select a branch candidate. How can this be carried over to general branching?
- Assimilate ideas from/compare against other general branching schemes.
  - Basis reduction
  - LP guided strengthening of disjunction.
  - *IMA general branching presentations..*
- Efficiency (no exploitation of sparsity at the moment).
- Include most promising scheme in future release of Xpress?.