

## 8 Experimentos e Resultados

Este capítulo abordará a metodologia experimental e os resultados dos experimentos propostos neste trabalho. Os resultados do resolvidor CPLEX serão apresentados primeiro e parte das instâncias discutidas na seção 6.4 serão descartadas. As seções seguintes descrevem os resultados e comparam os outros algoritmos no conjunto restante de instâncias.

Quando presente, a coluna com os melhores resultados da literatura tem seus valores retirados do trabalho de Yagiura et al. [39]. Alguns dos resultados vêm deste mesmo trabalho, enquanto vários outros vêm do trabalho em metaheurísticas paralelas de Ishibashi [22]. Resultados de destaque em cada uma das tabelas serão comentados no texto.

### 8.1 Metodologia

Os algoritmos deste trabalho foram implementados em C++, usando o compilador MSVC 2008 em sistema operacional Windows. Como resolvidor MIP em todos os algoritmos, foi utilizada a versão 11 do resolvidor ILOG CPLEX. A interface com o resolvidor foi feita utilizando a API em linguagem C provida pela ILOG. Os testes foram executados em um computador com processador Intel com 2 cores de 2.66GHz, memória RAM de 8Gb (onde os algoritmos só possuem acesso a 3Gb, devido às limitações do sistema operacional).

Cada algoritmo foi executado uma vez para cada instância e os limites de tempo utilizados foram os mesmos utilizados em [39]: 3000 segundos para as instâncias com  $N$  igual a 100 e 400, 6000 segundos para as instâncias de  $N$  igual a 200, 10000 segundos para as instâncias onde  $N$  é igual a 900 e, finalmente, 50000 segundos para as instâncias onde  $N$  é 1600. O melhor resultado encontrado até o tempo limite é considerado.

Alguns dos métodos exatos encerraram sua execução antes do limite de tempo estabelecido (conseguindo provar o ótimo da instância). Estes casos serão destacados na discussão dos resultados. Quando possível, também será destacado o tempo que o algoritmo demorou para encontrar sua melhor solução

- este dado será útil para analisar algoritmos que rapidamente conseguem uma solução de qualidade, mas não conseguem depois encontrar uma solução melhor.

## 8.2 CPLEX

As tabelas 8.1, 8.2 e 8.3 mostram os resultados do resolvidor CPLEX para as instâncias das classes C, D e E, respectivamente. As duas primeiras colunas de cada tabela identificam a instância (onde a coluna “Código” representa a concatenação do tipo da instância, o número de máquinas  $M$  e o número de tarefas  $N$ ), enquanto as três últimas identificam, respectivamente, o *gap* absoluto ( $valor\_encontrado - melhor$ ), o *gap* relativo ( $(valor\_encontrado - melhor)/melhor$ ) e o tempo gasto pelo algoritmo.

Os resultados marcados em negrito enfatizam instâncias onde a solução ótima foi atingida. Analisando estas tabelas é fácil perceber que o CPLEX por si só já consegue excelentes resultados: 20 de 45 instâncias são resolvidas à otimalidade, muitas delas em poucos segundos. A maior parte das instâncias da classe C é resolvida, com exceção das maiores instâncias, onde  $n = 1600$  e uma onde  $n = 900$ .

Dados estes resultados do CPLEX, todas as instâncias que foram resolvidas à otimalidade (de todas as classes) serão ignoradas nos próximos testes, resultando em um conjunto de testes de 25 instâncias. Existem alguns motivos para isso: muitas delas são resolvidas em poucos segundos, sem uso excessivo de memória, tornando qualquer outro algoritmo desnecessário. Além disso, alguns dos algoritmos usam o CPLEX para encontrar a solução inicial (ou o conjunto de soluções inicial). Em suma, essas instâncias já podem ser consideradas “fáceis” dado o poder computacional do resolvidor MIP e do hardware utilizado.

De maneira geral, o desempenho do CPLEX é, como esperado, muito bom. Em várias instâncias, ele termina prematuramente por falta de memória mas, ainda assim, consegue bons resultados. Para as instâncias e10400, e20400 e e15900 (destacadas na tabela 8.3), o CPLEX consegue pela primeira vez encontrar soluções comprovadamente ótimas. Para a instância e30900 (também destacada na tabela 8.3), uma nova melhor solução é encontrada. O pior desempenho nas instâncias da classe D é esperado, principalmente em vista dos resultados descritos nas próximas seções - a classe D possui, sem sombra de dúvida, as instâncias mais difíceis do GAP.

Um dado interessante das tabelas 8.1, 8.2 e 8.3 são as instâncias em que o CPLEX terminou antes do limite de tempo, mas não conseguiu atingir a

Tabela 8.1: Resultados do algoritmo CPLEX para as instâncias classe C

Instâncias		CPLEX		
Código	Melhor	gap abs	gap %	Tempo Total
c05100	1931	0	0.00E+00	0.106
c10100	1402	0	0.00E+00	0.797
c20100	1243	0	0.00E+00	0.417
c05200	3456	0	0.00E+00	0.633
c10200	2806	0	0.00E+00	9.207
c20200	2391	0	0.00E+00	5.925
c10400	5597	0	0.00E+00	6.167
c20400	4782	0	0.00E+00	964.953
c40400	4244	0	0.00E+00	43.811
c15900	11340	0	0.00E+00	1250.74
c30900	9982	2	2.00E-02	1955.68
c60900	9327	0	0.00E+00	3871.3
c201600	18802	1	5.32E-03	2023.35
c401600	17145	1	5.83E-03	3783.73
c801600	16285	1	6.14E-03	50071.3

Tabela 8.2: Resultados do algoritmo CPLEX para as instâncias classe D

Instâncias		CPLEX		
Código	Melhor	gap abs	gap %	Tempo Total
d05100	6353	0	0.00E+00	51.717
d10100	6348	3	4.73E-02	3000.06
d20100	6190	23	3.72E-01	3000.07
d05200	12742	1	7.85E-03	6000.06
d10200	12432	5	4.02E-02	6000.08
d20200	12241	18	1.47E-01	6000.12
d10400	24963	4	1.60E-02	3000.12
d20400	24574	20	8.14E-02	1593.71
d40400	24392	60	2.46E-01	3000.32
d15900	55409	10	1.80E-02	2229.01
d30900	54852	30	5.47E-02	3884.11
d60900	54568	262	4.80E-01	5940.65
d201600	97832	47	4.80E-04	4571.28
d401600	97105	209	2.15E-03	8031.44
d801600	97035	396	4.08E-03	9721.61

Tabela 8.3: Resultados do algoritmo CPLEX para as instâncias classe E

Instâncias		CPLEX		
Código	Melhor	gap abs	gap %	Tempo Total
e05100	12681	0	0.00E+00	3.361
e10100	11577	0	0.00E+00	23.133
e20100	8436	0	0.00E+00	1924.53
e05200	24930	0	0.00E+00	1.294
e10200	23307	0	0.00E+00	19.151
e20200	22379	0	0.00E+00	91.653
e10400	45746	0	0.00E+00	45.479
e20400	44877	0	0.00E+00	301.434
e40400	44562	4	8.98E-03	2751.39
e15900	102421	0	0.00E+00	130.88
e30900	100429	-1	-1.00E-03	4240.87
e60900	100153	8	7.99E-03	10001
e201600	180646	22	1.22E-04	22268.9
e401600	178294	0	0.00E+00	22166.9
e801600	176828	13	7.35E-05	22871.7

melhor solução. Nestas instâncias, o algoritmo encerrou-se prematuramente devido a falta de memória. Este caso é bastante frequente e, como esperado, diretamente relacionado ao tamanho da instância em questão. Estas instâncias podem ser indentificadas pelo tempo total tomado pelo algoritmo, que é menor que o tempo limite estabelecido.

### 8.3

#### PostProcessing

Os resultados do algoritmo PostProcessing encontram-se na tabela 8.4, juntamente com os resultados do algoritmo CPLEX, para fins de comparação. A coluna “Pós?” indica se o resultado de PostProcessing foi conseguido na fase de pós-processamento ou não.

Estes resultados indicam que a fase de pós-processamento do algoritmo não foi eficaz. Em somente uma instância o pós-processamento conseguiu encontrar uma solução melhor que a fase inicial. Um detalhe interessante destes resultados é que, devido à ênfase dada à primeira fase do algoritmo (onde o CPLEX é executado por si só) e ao uso de múltiplas *threads* pelo resolvidor, o PostProcessing tem resultados melhores que o CPLEX em algumas instâncias. É importante observar que, mesmo nestas instâncias, o que produz o melhor resultado é somente a fase inicial, ou seja, o resolvidor MIP em si e não a fase de pós-processamento. Da mesma maneira, na única instância em que o pós-processamento é eficaz e uma solução melhor que a inicial é encontrada, a melhor solução ainda é pior que a solução encontrada pelo algoritmo CPLEX.

Alguns motivos possíveis para a falta de sucesso deste algoritmo seriam:

- os resultados gerados pela primeira fase são muito semelhantes, assim a vizinhança elipsoidal não consegue “fugir” desta região do espaço de soluções;
- a estratégia de escolher sempre as soluções mais distantes não é eficaz;
- as vizinhanças elipsoidais não têm sucesso em melhorar as soluções do GAP.

Os resultados do algoritmo PathRelinkMIP mostrarão que a terceira afirmação não é verdadeira - quando utilizada corretamente, a vizinhança elipsoidal consegue encontrar soluções significativamente melhores. O motivo dos maus resultados de PostProcessing pode ser atribuído, então, a uma mistura das duas primeiras justificativas.

### 8.4

#### VNSBra

A tabela 8.5 apresenta os resultados do algoritmo VNSBra. Uma observação interessante que pode ser extraída destes resultados é que o algoritmo tem bastante dificuldade em fugir de soluções ótimo local. Isso é evidenciado pela coluna “Tempo %”, que mostra em que momento de sua execução o VNSBra encontrou a sua melhor solução. A percentagem é dada em função do tempo total de execução, ou seja, 0% significaria que a solução foi encontrada

Tabela 8.4: Comparação dos algoritmos PostProcessing e CPLEX

Instâncias		PostProcessing			CPLEX	
Código	Melhor	gap abs	gap %	Pós?	gap abs	gap %
c30900	9982	1	2.00E-04	não	2	2.00E-04
c60900	9327	1	1.07E-04	não	0	0.00E+00
c201600	18802	2	1.06E-04	não	1	5.32E-05
c401600	17145	1	5.83E-05	não	1	5.83E-05
c801600	16285	1	6.14E-05	não	1	6.14E-05
d10100	6348	2	3.15E-04	não	3	4.73E-04
d20100	6190	19	3.07E-03	não	23	3.72E-03
d05200	12742	3	2.35E-04	não	1	7.85E-05
d10200	12432	4	3.22E-04	não	5	4.02E-04
d20200	12241	18	1.47E-03	não	18	1.47E-03
d10400	24963	12	4.81E-04	não	4	1.60E-04
d20400	24574	22	8.95E-04	não	20	8.14E-04
d40400	24392	85	3.48E-03	não	60	2.46E-03
d15900	55409	10	1.80E-04	não	10	1.80E-04
d30900	54852	35	6.38E-04	não	30	5.47E-04
d60900	54568	466	8.54E-03	não	262	4.80E-03
d201600	97832	47	4.80E-04	não	47	4.80E-04
d401600	97105	228	2.35E-03	não	209	2.15E-03
d801600	97035	468	4.82E-03	não	396	4.08E-03
e40400	44562	5	1.12E-04	sim	4	8.98E-05
e30900	100429	1	9.96E-06	não	-1	-9.96E-06
e60900	100153	8	7.99E-05	não	8	7.99E-05
e201600	180646	22	1.22E-04	não	22	1.22E-04
e401600	178294	12	6.73E-05	não	0	0.00E+00
e801600	176828	27	1.53E-04	não	13	7.35E-05

no primeiro segundo e 100% significaria que a solução foi encontrada no último segundo. Na maioria das instâncias o VNSBra encontra a melhor solução nos primeiros 20% do seu tempo disponível e depois não consegue mais melhorar aquela solução - ficando assim “preso” àquele ótimo local. Essa característica seria menos alarmante se estas soluções fossem de alta qualidade, o que, como será visto nas seções a seguir, não é o caso.

É importante mencionar também que o VNSBra é o único dos algoritmos que utiliza o CPLEX mas não utiliza o resolvidor para conseguir uma primeira solução de alta qualidade. Ao invés disso, o VNSBra invoca o CPLEX para que este o retorne simplesmente a primeira solução viável encontrada. Como essa primeira solução em geral é de péssima qualidade, o VNSBra acaba gastando bastante tempo tentando melhorar esta solução. Apesar da escolha de só conseguir a primeira solução viável através do resolvidor MIP ser louvável na

teoria, na prática é uma péssima escolha - o CPLEX consegue normalmente soluções razoáveis em poucos segundos. Em troca desses segundos a mais, o VNSBra acaba tendo que começar com uma solução de má qualidade.

Tabela 8.5: Resultados do algoritmo VNSBra

Instâncias		VNSBra			
Código	Melhor	gap abs	gap %	Tempo	Tempo %
c30900	9982	11	1.10E-01	203.139	02.03
c60900	9327	5	5.36E-02	1099.96	10.99
c201600	18802	7	3.72E-04	121.968	0.24
c401600	17145	6	3.50E-04	515.624	1.03
c801600	16285	7	4.30E-04	965.65	1.93
d10100	6348	46	7.25E-01	283.175	09.43
d20100	6190	62	1.00E+00	527.064	17.56
d05200	12742	15	1.18E-01	418.514	06.97
d10200	12432	34	2.73E-01	183.095	03.05
d20200	12241	72	5.88E-01	1080.95	18.01
d10400	24963	79	3.16E-01	636.844	21.22
d20400	24574	159	6.47E-01	2212.33	73.74
d40400	24392	198	8.12E-01	1560.46	52.01
d15900	55409	67	1.21E-01	776.554	07.76
d30900	54852	156	2.84E-01	600.22	06.00
d60900	54568	335	6.14E-01	7472.56	74.72
d201600	97832	80	8.18E-02	240.062	00.48
d401600	97105	248	2.55E-01	720.339	01.44
d801600	97035	454	4.68E-03	48829.4	97.65
e40400	44562	24	5.39E-02	2946.25	98.20
e30900	100429	286	2.85E-01	1397.18	13.97
e60900	100153	775	7.74E-01	9999.53	99.99
e201600	180646	0	0.00E+00	7254.6	14.50
e401600	178294	312	1.75E-03	5070.65	10.14
e801600	176828	979	5.54E-03	6867.5	13.73

## 8.5

### PathRelinkMIP

Os resultados do algoritmo PathRelinkMIP encontram-se na tabela 8.6. Além das colunas com o *gap* absoluto e relativo, com o tempo até a melhor solução e o percentual de tempo utilizado para encontrar a melhor solução, os resultados também incluem em quais instâncias a busca EllipsoidalSearch e a busca local KOPT, respectivamente, encontraram a melhor solução. Em algumas poucas instâncias, nenhuma das duas buscas encontra a melhor

solução e esta acaba vindo da geração inicial de soluções pelo resolvidor CPLEX.

Como os resultados do algoritmo PostProcessing não tiveram o sucesso esperado (já que a fase de pós-processamento não melhorou os resultados), uma das primeiras questões a ser respondida para este algoritmo é se a vizinhança elipsoidal tem sucesso ou não em melhorar as soluções. Os resultados da tabela 8.6 mostram que, das vinte e cinco instâncias testadas, a busca EllipsoidalSearch consegue encontrar a melhor solução em doze, a busca KOPT em oito e ambas falham em melhorar a solução inicial em cinco instâncias. Estabelecido o fato que EllipsoidalSearch consegue, de fato, melhorar as soluções iniciais, as seções seguintes mostrarão que as soluções encontradas são de alta qualidade.

Outro fato interessante retratado na tabela 8.6 é que, ao contrário do algoritmo VNSBra, PathRelinkMIP utiliza o tempo total de maneira mais balanceada. Em boa parte das instâncias a solução é encontrada somente na metade final da busca e, para várias instâncias, somente nos últimos 30% do tempo restante.

## 8.6

### Algoritmos MIP: PathRelinkMIP vs. VNSBra vs. CPLEX

Neste ponto, é interessante fazer uma comparação dos algoritmos MIP testados neste trabalho. A grande vantagem destes algoritmos é que eles são, em certo ponto, agnósticos ao problema que está sendo resolvido. Em outras palavras, estes algoritmos funcionarão para qualquer problema que possa ser modelado como um MIP.

Na tabela 8.7 encontram-se os resultados de três dos algoritmos MIP testados (o algoritmo PostProcessing, devido à sua fraca performance, foi excluído desta comparação). Nesta tabela, podemos perceber que o algoritmo PathRelinkMIP consegue os melhores resultados para doze instâncias, o algoritmo CPLEX consegue os melhores resultados para catorze instâncias e o algoritmo VNSBra consegue o melhor resultado em somente uma instância. Os melhores resultados para cada instância foram marcados nesta tabela com um asterisco (\*).

Nestes testes tanto o algoritmo CPLEX quanto o algoritmo PathRelinkMIP apresentam ótimos resultados, podendo-se dizer que existe um empate entre os resultados destes algoritmos. O algoritmo PathRelinkMIP consegue resultados expressivos para algumas instâncias da classe D e para uma instância da classe E, destacados na tabela: para estas, a busca elipsoidal consegue encontrar soluções significativamente melhores que as do CPLEX. Para esta



Tabela 8.6: Resultados do algoritmo PathRelinkMIP

Instâncias		PathRelinkMIP					
Código	Melhor	gap abs	gap %	Tempo	Tempo %	Elipsoidal Search?	KOPT?
c30900	9982	2	2.00E-02	6726	67.26	não	sim
c60900	9327	1	1.07E-02	831	27.70	sim	não
c201600	18802	1	5.32E-05	10789	21.57	não	sim
c401600	17145	1	5.83E-05	8998	17.99	não	sim
c801600	16285	2	1.23E-04	28223.6	56.44	sim	não
d10100	6348	2	3.15E-02	360.26	12.01	não	não
d20100	6190	5	8.08E-02	427.05	14.24	sim	não
d05200	12742	1	7.85E-03	1485.54	24.76	não	sim
d10200	12432	9	7.24E-02	360.84	06.01	não	não
d20200	12241	30	2.45E-01	766	12.77	sim	não
d10400	24963	8	3.20E-02	720.16	24.01	não	sim
d20400	24574	30	1.22E-01	600	20.00	não	não
d40400	24392	76	3.12E-01	1887.87	62.93	sim	não
d15900	55409	15	2.71E-02	600	06.00	não	não
d30900	54852	76	1.39E-01	5911	59.11	não	sim
d60900	54568	232	4.25E-01	9657.51	96.58	sim	não
d201600	97832	31	3.17E-04	30535.9	61.07	sim	não
d401600	97105	121	1.25E-03	48162.4	96.32	não	sim
d801600	97035	325	3.35E-03	43308.1	86.61	sim	não
e40400	44562	0	0.00E+00	600.57	20.00	não	não
e30900	100429	3	2.99E-03	2813	28.13	sim	não
e60900	100153	53	5.29E-02	4346	43.46	sim	não
e201600	180646	18	9.96E-05	10547.8	21.09	sim	não
e401600	178294	4	2.24E-05	24865.2	49.73	não	sim
e801600	176828	1	5.66E-06	36589.2	73.17	sim	não

comparação é importante ressaltar que para várias destas instâncias o CPLEX encerra prematuramente devido a falta de memória, que ocorre por conta principalmente do tamanho e da dificuldade inerente destas instâncias (em especial as da classe D). Esses resultados enfatizam o potencial da vizinhança elipsoidal: combinando várias soluções de boa qualidade, ela consegue fugir de ótimos locais e encontrar excelentes soluções.

## 8.7 BPRINS

O algoritmo BPRINS se encontra no meio do caminho entre os algoritmos MIP detalhados nas seções anteriores e os algoritmos metaheurísticos que possuem os melhores resultados para o GAP (seções 6.3.4 e 6.3.5). Apesar do modelo *branch and bound* e dos algoritmos RINS e *guided dives* funcionarem para qualquer MIP, a geração de colunas e toda a lógica do *branch and bound* foi implementada com o GAP em vista e não poderia ser facilmente adaptada a outros problemas. Como foi enfatizado na seção 7.5, o resolvidor CPLEX não permite a implementação nativa de algoritmos *branch and price*.

A tabela 8.8 detalha os resultados do algoritmo BPRINS, comparando com os resultados do CPLEX. Em doze instâncias o algoritmo BPRINS consegue resultados melhores que o CPLEX, enquanto o contrário ocorre em seis instâncias (ambos conseguem resultados iguais nas instâncias restantes). BPRINS parece conseguir melhores resultados em instâncias de tamanho pequeno e médio na difícil classe D, com resultados especialmente bons para as de tamanho 100 a 400. O algoritmo também consegue alguns excelentes resultados para as instâncias grandes ( $n = 1600$ ) para a classe E. Em destaque na tabela estão os resultados para as instâncias d10200, e201600 e e401600, onde um novo melhor resultado foi encontrado.

A coluna “# nós” detalha quantos nós da árvore de *branch and bound* o BPRINS conseguiu analisar dentro do tempo limite. Pela tabela, percebe-se que, para instâncias grandes (as instâncias de tamanho 1600 e da classe D de tamanho 900), o BPRINS consegue explorar somente pouquíssimos nós da árvore dentro do tempo limite. Para várias instâncias, o algoritmo não explora nenhum nó e retorna somente o resultado da primeira solução encontrada pelo CPLEX durante a fase de inicialização do BPRINS. Nessas instâncias, não há chance de conseguir resultados melhores que os do CPLEX. Entretanto, para as instâncias em que o algoritmo consegue percorrer um número razoável de nós, ele consegue de maneira consistente encontrar boas soluções. Isso prova a força da modelagem por geração de colunas (detalhada em 6.2): como a relaxação linear neste modelo gera soluções de alta qualidade, o algoritmo RINS trabalha

Tabela 8.7: Comparação dos algoritmos PathRelinkMIP, VNSBra e CPLEX

Instâncias		PathRelinkMIP				VNSBra		CPLEX	
Código	Melhor	Ellipsoidal Search?	KOPT?	gap abs	gap %	gap abs	gap %	gap abs	gap %
c30900	9982	não	sim	2*	2.00E-04	11	1.10E-03	2*	2.00E-04
c60900	9327	sim	não	1	1.07E-04	5	5.36E-04	0*	0.00E+00
c201600	18802	não	sim	1*	5.32E-05	7	3.72E-04	1*	5.32E-05
c401600	17145	não	sim	1*	5.83E-05	6	3.50E-04	1*	5.83E-05
c801600	16285	sim	não	2	1.23E-04	7	4.30E-04	1	6.14E-05
d10100	6348	não	não	2*	3.15E-04	46	7.25E-03	3	4.73E-04
d20100	6190	sim	não	5*	8.08E-04	62	1.00E-02	23	3.72E-03
d05200	12742	não	sim	1*	7.85E-05	15	1.18E-03	1*	7.85E-05
d10200	12432	não	não	9	7.24E-04	34	2.73E-03	5*	4.02E-04
d20200	12241	sim	não	30	2.45E-03	72	5.88E-03	18*	1.47E-03
d10400	24963	não	sim	8	3.20E-04	79	3.16E-03	4*	1.60E-04
d20400	24574	não	não	30	1.22E-03	159	6.47E-03	20*	8.14E-04
d40400	24392	sim	não	76	3.12E-03	198	8.12E-03	60*	2.46E-03
d15900	55409	não	não	15	2.71E-04	67	1.21E-03	10*	1.80E-04
d30900	54852	não	sim	76	1.39E-03	156	2.84E-03	30*	5.47E-04
d60900	54568	sim	não	232*	4.25E-03	335	6.14E-03	262	4.80E-03
d201600	97832	sim	não	31*	3.17E-04	80	8.18E-04	47	4.80E-04
d401600	97105	não	sim	121*	1.25E-03	248	2.55E-03	209	2.15E-03
d801600	97035	sim	não	325*	3.35E-03	454	4.68E-03	396	4.08E-03
e40400	44562	não	não	0*	0.00E+00	24	5.39E-04	4	8.98E-05
e30900	100429	sim	não	3	2.99E-05	286	2.85E-03	-1*	-9.96E-06
e60900	100153	sim	não	53	5.29E-04	775	7.74E-03	8*	7.99E-05
e201600	180646	sim	não	18	9.96E-05	0*	0.00E+00	22	1.22E-04
e401600	178294	não	sim	4	2.24E-05	312	1.75E-03	0	1.75E-03
e801600	176828	sim	não	1*	5.66E-06	979	5.54E-03	13	5.54E-03
# melhores			12			1		14	

em um espaço de busca reduzido e consegue encontrar boas soluções.

Tabela 8.8: Resultados do algoritmo BPRINS, comparados com CPLEX

Instâncias		BPRINS			CPLEX	
Código	Melhor	# nós	gap abs	gap %	gap abs	gap %
c30900	9982	0	3	3.01E-04	2*	2.00E-04
c60900	9327	226	0*	0.00E+00	0*	0.00E+00
c201600	18802	0	1	5.32E-05	1	5.32E-05
c401600	17145	0	2	1.17E-04	1	5.83E-05
c801600	16285	0	3	1.84E-04	1	6.14E-05
d10100	6348	3391	2*	3.15E-04	3	4.73E-04
d20100	6190	5153	8*	1.29E-03	23	3.72E-03
d05200	12742	446	0*	0.00E+00	1	7.85E-05
d10200	12432	1569	-1*	-8.04E-05	5	4.02E-04
d20200	12241	1096	17*	1.39E-03	18	1.47E-03
d10400	24963	4	0*	0.00E+00	4	1.60E-04
d20400	24574	92	28	1.14E-03	20*	8.14E-04
d40400	24392	0	63	2.58E-03	60*	2.46E-03
d15900	55409	0	23	4.15E-04	10*	1.80E-04
d30900	54852	0	82	1.49E-03	30*	5.47E-04
d60900	54568	93	333	6.10E-03	262*	4.80E-03
d201600	97832	0	25*	2.56E-04	47	4.80E-04
d401600	97105	3	209*	2.15E-03	209*	2.15E-03
d801600	97035	61	396*	4.08E-03	396*	4.08E-03
e40400	44562	1851	3*	6.73E-05	4	8.98E-05
e30900	100429	0	-1*	-9.96E-06	-1*	-9.96E-06
e60900	100153	403	1*	9.98E-06	8	7.99E-05
e201600	180646	35	-1*	-5.54E-06	22	1.22E-04
e401600	178294	203	-1*	-5.61E-06	0	0.00E+00
e801600	176828	332	1*	5.66E-06	13	7.35E-05

## 8.8

### Algoritmos MIP vs. BPRINS vs. metaheurísticas

A tabela 8.9 compara os resultados de todos os algoritmos testados. Os resultados retratados nesta tabela mostram um quadro claro: o algoritmo PREC, de Yagiura et al. [39], encontra os melhores resultados entre os algoritmos testados.

A formulação por geração de colunas, representada pelo algoritmo BPRINS, consegue excelentes resultados para instâncias de tamanho pequeno e médio e para as instâncias da classe E. Pode-se conjecturar que, se acoplada ao poderoso *branch and cut* utilizado pelo resolvidor CPLEX, essa formulação conseguiria resultados ainda melhores. Estes resultados do BPRINS também

ênfatizam o fato de que o poder computacional de um MIP está extremamente interligado à formulação sendo utilizada.

O algoritmo PREC, como notado na seção 6.3.5, utiliza uma série de propriedades do GAP para conseguir os melhores resultados entre todos os algoritmos testados. É bom lembrar, entretanto, que é virtualmente impossível adaptar o algoritmo PREC para resolver outros problemas. Algumas das propriedades utilizadas pelo PREC poderiam ser modeladas na formulação MIP do GAP, podendo, assim, também ser utilizadas pelos resolvedores MIP, como a exploração de regiões inviáveis do espaço de busca (utilizando uma função de penalização). Este é material para trabalhos futuros.

Tabela 8.9: Comparação de todos os algoritmos

Instâncias		CPLEX	PathRelinkMIP	VNSBra	BPRINS	PREC	TSEC
Código	Melhor	gap abs	gap abs	gap abs	gap abs	gap abs	gap abs
c30900	9982	2	2	11	3	0*	2
c60900	9327	0*	1	5	0*	1	1
c201600	18802	1*	1*	7	1*	1*	1*
c401600	17145	1	1	6	2	0*	2
c801600	16285	1*	2	7	3	2	6
d10100	6348	3	2	46	2	0*	1
d20100	6190	23	5	62	8	2*	16
d05200	12742	1	1	15	0*	0*	1
d10200	12432	5	9	34	-1*	1	8
d20200	12241	18	30	72	17	4*	36
d10400	24963	4	8	79	0*	4	11
d20400	24574	20	30	159	28	4*	30
d40400	24392	60	76	198	63	17*	64
d15900	55409	10	15	67	23	4*	16
d30900	54852	30	76	156	82	16*	51
d60900	54568	262	232	335	333	5*	88
d201600	97832	47	31	80	25	5*	35
d401600	97105	209	121	248	209	1*	55
d801600	97035	396	325	454	396	12*	62
e40400	44562	4	0*	24	3	3	17
e30900	100429	-1*	3	286	-1*	2	9
e60900	100153	8	53	775	1*	16	24
e201600	180646	22	18	0	-1*	0	1
e401600	178294	0	4	312	-1*	4	17
e801600	176828	13	1*	979	1*	20	28
# melhores		4	3	0	10	15	1