Universidade Federal do Ceará
Pró-Reitoria de Pesquisa e Pós-Graduação
Programa de Mestrado e Doutorado em Ciência da Computação - MDCC

Dissertação de Mestrado

Título

Escalonamento UET-UCT de tarefas em múltiplos processadores

Autor

ELIEZER TOMÉ DE PAULA NETO

Orientador

Prof. Dr. Manoel Bezerra Campêlo Neto

Coorientador

Prof. Dr. Carlos Diego Rodrigues

Grupo ParGO – Paralelismo, Grafos e Otimização

2014 — Fortaleza — Ceará

Sumário

1 Introdução **1**

2 Escalonamento UET-UCT **4**

2.1 Principais características 4

2.2 Notação $\alpha|\beta|\gamma$ 6

2.3 Definição formal do problema 7

2.4 Caracterizando uma solução viável 8

2.5 Complexidade Computacional 10

3 Limites inferiores e superiores **14**

3.1 Limites inferiores 14

3.1.1 Limite inferior $LB_{i,j}$ 14

3.1.2 Limite inferior $IFB_{i,j}$ 21

3.1.3 Limite inferior $FFB_{i,j}$ 23

3.2 Limites superiores 25

3.2.1 Heurística $CP/MISF$ adaptada 26

3.2.2 Heurística ISH adaptada 28

3.2.3 Heurística PSO 31

3.3 Comparações entre os limites 35

3.3.1 Limites Inferiores 35

3.3.2 Limites Superiores 46

4 Formulações matemáticas **58**

4.1 Formulação X 58

4.1.1 Incrementos à formulação original 63

4.2 Formulação Y 68

4.3 Formulação Y reforçada 73

4.4 Formulação Z 80

5 Estudo comparativo entre as formulações 84

5.1 Estudo comparativo teórico 84

5.2 Estudo comparativo computacional 87

5.2.1 Resultados computacionais 88

5.2.2 Influência dos limites sobre a Formulação Z 110

6 Conclusão e Trabalhos Futuros 124

A Instâncias e Ambiente Computacional 126

A.1 Instâncias utilizadas 126

A.2 Ambiente Computacional 136

Capítulo 1

Introdução

Problemas de escalonamento ocupam, sem dúvida, um local de destaque no conjunto de problemas clássicos de otimização. Isto se deve sobretudo a suas aplicações diretas e extensas. Na indústria, por exemplo, onde o tempo e os custos de produção são determinantes para a obtenção de maiores lucros, existem grandes aplicações para problemas de escalonamento em suas diversas variações.

Abordamos neste trabalho um problema de escalonamento restrito, porém fica evidenciada a importância do estudo desse caso por ele estar intimamente relacionado com processamento paralelo, um paradigma computacional cada vez mais utilizado e que, em comparação ao paradigma sequencial, pode ser consideravelmente mais eficiente em alguns casos. Apesar deste argumento favorável ao paradigma paralelo, este traz consigo problemas que não existem no paradigma sequencial, como, por exemplo, o particionamento do programa em tarefas, comunicação entre núcleos, sincronização e escalonamento das tarefas nos vários núcleos. Dessa forma, nosso objeto de estudo é uma das bases para este paradigma. O problema que passamos a descrever agora é um problema de escalonamento de tarefas em múltiplas máquinas ou processadores.

Problemas de escalonamento em máquinas podem ser descritos genericamente da seguinte maneira. Possuímos um conjunto de tarefas a serem executadas e um conjunto de máquinas que poderão processá-las. Um escalonamento deve estabelecer um agendamento das tarefas nas máquinas disponíveis, ou seja, ele deverá especificar uma máquina e um intervalo de tempo onde cada tarefa será executada.

Neste trabalho, abordamos o escalonamento *UET-UCT* (*Unit Execution - Unit Communication Time*) em máquinas idênticas com restrições de execução e comunicação. Trabalhamos com duas variações do problema, dependentes do número de processadores disponíveis. Na *versão ilimitada*, o número de processadores seria suficiente para executar todas as tarefas simultaneamente, de modo que diremos por simplicidade que o múmero de processadores é ilimitado. Na *versão limitada*, o número de processadores não é suficiente para tal execução.

Nos problemas que abordamos, recebemos como entrada um grafo direcionado, onde cada vértice representa uma tarefa e um arco entre duas tarefas implica uma ordem de precedência entre elas. Este grafo será chamado *grafo de tarefas*. Devemos indicar um agendamento para a execução das tarefas em um conjunto de processadores, obedecendo duas restrições. Uma tarefa i só pode ser executada após suas antecessoras terem finalizado suas execuções e ainda transmitido os resultados de seus processamentos para o processador que executará i, respeitando o tempo de execução e comunicação entre as tarefas. O tempo de comunicação dos resultados entre duas tarefas i e j é considerado zero, quando a antecessora j é executada no mesmo processador que executará i. O objetivo principal do problema é escalonar as tarefas no conjunto de processadores, respeitando as restrições impostas e minimizando o *makespan*, ou seja, minimizando o tempo final de execução do conjunto de tarefas.

Especificamente, trabalhamos com tempo de execução e comunicação unitários para todas as tarefas. Em relação ao número de processadores abordaremos as duas mencionadas variações. Tomaremos a quantidade de processadores limitada e pertencente a entrada do problema e posteriormente tomaremos um número ilimitado de processadores.

Ambos os problemas foram provados ser *NP-Difíceis* para um grafo de tarefas arbitrário, respectivamente, em [Ull75] e [Pic92]. Mesmo sendo difíceis em geral, ou seja, quando não se conhece uma estrutura definida para o grafo de entrada, existem casos em que os problemas são mais tratáveis. Por exemplo, a variante limitada do problema é linear para grafos de intervalo ordenado [Sta04]. Para os casos em que os problemas são *NP-Difíceis*, evidenciamos algoritmos aproximativos que existem na literatura. As melhores aproximações existentes para o caso limitado e ilimitado são, respectivamente, (7/3 - (4/m)) e (4/3), que se encontram em [MH01] e [MK97], onde m representa o número de processadores.

Do ponto de vista de programação matemática, existem algumas formulações diponíveis na literatura para os dois problemas, como em [MK97] para a versão ilimitada e em [CCMP01] para a versão restrita. No caso de processadores limitados, para certas instâncias não se têm bons resultados com estas formulações do ponto de vista computacional, mesmo a estrutura do grafo sendo conhecida e de fácil resolução. Dispensamos esforços nesta pesquisa no sentido de melhorar ou incrementar as formulações existentes e estudar comparativamente outras formulações para os problemas apresentados.

Primeiro, apresentamos três heurísticas para os problemas abordados e ainda três cálculos diferentes para limites inferiores. Desenvolvemos então mecanismos de melhoria para a resolução computacional da formulação apresentada em [CCMP01] por Campêlo et al,. como a priorização de variáveis durante as ramificações e introdução de novas restrições baseadas nos limites inferiores desenvolvidos. Diante da ineficiência apresentada pela formulação existente na literatura, [CCMP01], para alguns tipos de instâncias, desenvolvemos três outras formulações para ambos os problemas, limitado e ilimitado. Fizemos comparações entre elas no âmbito teórico e computacional que evidenciam a superioridade das novas formulações desenvolvidas.

Esta dissertação está organizada em cinco capítulos. O presente capítulo mostra um panorama geral do trabalho, contextualizando-o, explicitando seus objetivos e contribuições e informando a estrutura da dissertação. No Capítulo 2, apresentamos os principais aspectos relativos aos problemas com os quais trabalhamos, formalizando sua definição, caracterizando uma solução viável e fazendo uma revisão bibliográfica dos principais resultados existentes para a classe de problemas em questão. Os diversos limites superiores e inferiores desenvolvidos são apresentados no Capítulo 3. Dentre os limites superiores estão duas heurísticas gulosas (CP/MISF e ISH) e uma metaheurística baseada em enxame de partículas (PSO). Os limites inferiores são fortalecimentos daqueles propostos em [CCMP01], que se baseiam em uma decomposição do grafo de tarefas. Já no Capítulo 4, apresentamos as principais contribuições do trabalho. Neste capítulo, detalhamos as formulações desenvolvidas. São elas: formulação Y, que se assemelha àquela apresentada em [MK97]; formulação Y_{ref}, notadamente um fortalecimento de Y; e formulação Z, equivalente a Y_{ref}, porém com redução significativa no número de variáveis. De posse das formulações, no Capítulo 5, fazemos um estudo comparativo teórico e computacional entre elas, demonstrando em ambas as análises a predominância das formulações desenvolvidas sobre as existentes na literatura.

Capítulo 2

Escalonamento UET-UCT

Apresentaremos nesse capítulo, formalmente, os principais aspectos necessários ao bom entendimento do problema ao longo do texto. Formularemos matematicamente as duas versões do problema abordado, com finitos e infinitos processadores. Utilizaremos, sempre que possível, uma notação padrão e comumente usada em textos da área de Otimização Combinatória e Teoria dos Grafos.

2.1 Principais características

Um problema de escalonamento em máquinas pode ser definido por três aspectos: ambiente de execução, especificações das tarefas e um objetivo ou critério de otimalidade. Tais aspectos estão bem definidos em [CPW98]. Na literatura, constam variações nesses aspectos e assim as características do problema originado são modificadas, ocasionalmente mudando inclusive a sua complexidade. Os problemas abordados neste trabalho adotam as seguintes suposições:

- **Ambiente de execução**
  - **Estágio único:** Cada tarefa exige apenas uma única operação. E no caso de múltiplos processadores, cada máquina possui a mesma função. Assim, basta que uma tarefa seja delegada a uma única máquina.
  - **Máquinas paralelas idênticas:** Todas as máquinas possuem as mesmas características, não importando em qual máquina uma tarefa será executada.
  - **Número de máquinas:** Limitado ou ilimitado.
  - **Tempo de comunicação uniforme:** Quaisquer duas máquinas demandam um tempo idêntico para transmitirem entre si os resultados de suas computações.
- **Especificações das tarefas**

- **Tempo de processamento uniforme:** As tarefas possuem tempo de processamento definido e idêntico, ou seja, para qualquer tarefa j seu tempo de execução será p, independente da máquina em que será executada. Isto se deve ao fato de supormos máquinas paralelas idênticas.
- **Restrição de precedência:** Uma tarefa, possivelmente, depende de um conjunto de outras tarefas, e ela só pode ser executada quando todas suas predecessoras finalizarem sua execução e transmitirem os resultados de suas computações para a máquina na qual a tarefa sucessora será executada, ocasionando portanto um atraso, devido ao tempo de comunicação. Em nosso caso, independente das tarefas e máquinas envolvidas, todos os tempos de comunicação serão iguais.
- **Não-preempção:** Quando uma tarefa inicia sua execução em alguma máquina, esta execução não poderá ser interrompida até que todo seu processamento tenha sido finalizado.
- **Não-duplicação:** Não é permitida a duplicação de tarefas. Esse artifício facilita, em alguns casos, a resolução do problema.

- **Objetivo**
  - **Tempo final de processamento:** O objetivo dos problemas abordados é que o tempo de término da última tarefa finalizada seja o menor possível. Dessa forma, o tempo de execução do conjunto de tarefas, comumente chamada de *makespan*, deve ser o mínimo.

Além das características já citadas, abordaremos o tipo estático de escalonamento, ou seja, em tempo de compilação temos disponíveis todas as informações necessárias sobre os componentes do escalonamento, como o tempo de processamento das tarefas, bem como as dependências entre elas e a quantidade de processadores disponíveis para o escalonamento.

Como trabalhamos com duas variações do problema principal, utilizaremos a notação apresentada por Graham, Lawler, Lenstra e Rinnooy, em [GLLK79], para que possamos, de forma abreviada, fazer referência às variações e também a problemas semelhantes citados ao longo do texto.

2.2 Notação $\alpha|\beta|\gamma$

A notação $\alpha|\beta|\gamma$ faz referência aos três aspectos que descrevem um problema de escalonamento em máquinas. São eles, respectivamente, o ambiente de execução, as especificações das tarefas e o objetivo do problema. Esta notação é bastante abrangente e, como não temos necessidade de todos os seus detalhes, utilizaremos um subconjunto desta notação.

Denotaremos o conjunto vazio por $\emptyset$, como de costume. O primeiro campo α é da forma $\alpha = \alpha_1\alpha_2$, onde os símbolos α_1 e α_2 podem ser entendidos como:

- $\alpha_1 \in \{\emptyset, P\}$:

  $\alpha_1 = \emptyset$: Máquina única;

  $\alpha_1 = P$: Máquinas paralelas idênticas;

- $\alpha_2 \in \{\emptyset, m, \infty\}$:

  $\alpha_2 = \emptyset$: Existe um número m de máquinas, onde m é parte da entrada.

  $\alpha_2 = m$: Existe um número fixo m de máquinas que faz parte da especificação do problema;

  $\alpha_2 = \infty$: O número de máquinas é arbitrário, podendo ser tão grande quanto necessário;

O segundo campo é denotado por $\beta = \beta_1\beta_2\beta_3$, onde β_1, β_2 e β_3 são respectivamente:

- $\beta_1 \in \{prec, intree, outree, tree, ...\}$, descreve a estrutura do grafo. Se a soubermos de antemão, essa estrutura será especificada; senão será tida como arbitrária ($prec$);

- $\beta_2 \in \{c_{ij}, c_{ij} = 1, c_{ij} = c\}$, refere-se aos custos de comunicação:

  $\beta_2 = c_{ij}$: Os custos de comunicação são arbitrários;

  $\beta_2 = (c_{ij} = 1)$: Os custos de comunicação são unitários;

  $\beta_2 = (c_{ij} = c)$: Neste caso os custos de comunicação são iguais a uma constante c.

- $\beta_3 \in \{p_j, p_j = 1, p_j = c\}$, refere-se aos custos de processamento das tarefas:

  $\beta_3 = p_j$: Os custos de processamento são arbitrários;

  $\beta_3 = (p_j = 1)$: Os custos de processamento são unitários;

  $\beta_3 = (p_j = c)$: Neste caso os custos de processamento são iguais a uma constante c.

O último campo se refere ao objetivo do problema de escalonamento. Observe que abordamos apenas problemas com o objetivo de minimizar o *makespan* (C_{max}).

Os problemas tratados nessa dissertação, em notação $\alpha|\beta|\gamma$, são portanto $P|prec, c_{ij} = 1, p_j = 1|C_{max}$ e $P\infty|prec, c_{ij} = 1, p_j = 1|C_{max}$, respectivamente denominados, em vários momento ao longo do texto, por versão limitada e ilimitada.

2.3 Definição formal do problema

Podemos definir o problema de escalonamento de tarefas UET-UCT em múltiplos processadores com base em três elementos: um conjunto finito de tarefas $N = \{1, \ldots, n\}$, uma relação binária $\prec$ que define uma ordem parcial em N e ainda um conjunto $Q \subset \mathbb{N}$ de processadores nos quais as tarefas devem ser executadas. Denotaremos a cardinalidade de Q por m e observamos que podemos ter ($m = \infty$).

A ordem parcial $\prec$ em N define a relação de precedência entre as tarefas. Escrever $i \prec j$ implica que $(i, j) \in \prec$ e ainda que a tarefa j necessita dos resultados de i para começar sua execução; caso isso não ocorra escrevemos $i \not\prec j$. Observe que se $i \not\prec j$ e $j \not\prec i$, as duas tarefas são incomparáveis ou concorrentes, implicando que uma pode ser escalonada independentemente da outra. Escrerevemos $i \parallel j$, quando isso ocorre. A tarefa i é minimal (resp. maximal) caso $j \not\prec i$ (resp. $i \not\prec j$) para todo $j \in N$.

Comumente a ordem parcial $\prec$ é representada por um grafo acíclico e direcionado (DAG) $G = (N, \prec)$, ou de forma simplificada pelo grafo $R = (N, \vdash)$, onde $\vdash$ é a redução transitiva da ordem parcial $\prec$. Observe o seguinte exemplo:

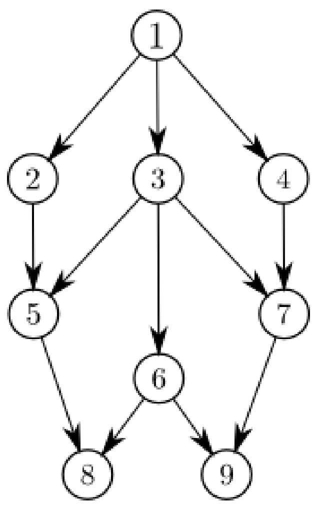


Figura 2.1: Exemplo de tarefas ($R = (N, \vdash)$)

Neste caso, os grafos mencionados possuem as seguintes configurações de arestas:

- $\vdash = \{(1,2);(1,3);(1,4);(2,5);(3,5);(3,6);(3,7);(4,7);(5,8);(6,8);(6,9);(7,9);\}$
- $\prec = \{(1,2);(1,3);(1,4);\mathbf{(1{,}5)};\mathbf{(1{,}6)};\mathbf{(1{,}7)};\mathbf{(1{,}8)};\mathbf{(1{,}9)};(2,5);\mathbf{(2{,}8)};(3,5);$
  $(3,6);(3,7);\mathbf{(3{,}8)};\mathbf{(3{,}9)};(4,7);\mathbf{(4{,}9)};(5,8);(6,8);(6,9);(7,9)\}$

Denominaremos indistintamente por grafo de tarefas quaisquer dos dois grafos mencionados, porém claramente a relação que representa somente as precedências essenciais é a relação $\vdash$.

Como dito anteriormente, supomos que os processadores são idênticos e não há preempção. Portanto, quando uma tarefa inicia sua execução, ela não pode ser interrompida. Assim, cada tarefa executará em apenas um processador. O tempo de execução de cada tarefa é unitário e ainda os custos de comunicação são igualmente unitários, independente do par de tarefas considerado. Note que o custo de comunicação é considerado quando duas tarefas são dependentes ($i \prec j$) e elas são executadas em processadores distintos.

O objetivo do problema de escalonamento $UET - UCT$ em múltiplos processadores é, portanto, agendar o conjunto de tarefas nos processadores, respeitando as restrições de precêdencia, comunicação e recurso (os processadores). Cada tarefa só pode executar quando todas as suas predecessoras forem executadas e enviarem seus resultados ao processador que a executará. Se este processador for o mesmo que executou sua predecessora, o tempo de comunicação com essa predecessora deve ser desconsiderado. O agendamento das tarefas deve fornecer, ao final da execução do conjunto de tarefas, o menor tempo possível. Note que qualquer agendamento das tarefas que respeite as restrições mencionadas é uma solução viável. Detalharemos agora o conjunto de possíveis soluções.

2.4 Caracterizando uma solução viável

Definiremos agora o significado de um agendamento válido. Lembre que uma instância do problema é definida pela dupla (G, Q), com $G = (N, \prec)$. A execução de uma tarefa pode ser expressa em termos do tempo de início, tempo de término e do processador no qual ela é executada. Como o tempo de execução é unitário, podemos definir a execução de uma tarefa $i \in N$ apenas pelo par (t_i, p_i), onde $t_i \in \mathbb{N}$ é o tempo de início da tarefa e $p_i \in Q$, o processador em que ela será executada. De forma geral, um escalonamento é definido pelo par (T, P), onde $T = \{t_1, \ldots, t_n\}$ e $P = \{q_1, \ldots, q_n\}$.

Para que um escalonamento (T, P) seja viável para o problema, ele deve obedecer às restrições estruturais e de recurso. Dividiremos as restrições de acordo com o relacionamento entre duas tarefas i e j, com $i, j \in N$.

As restrições estruturais são decorrentes diretamente do grafo de tarefas. Suponha que a tarefa j depende da tarefa i, portanto $i \prec j$. Claramente, $t_j > t_i$. Se as duas tarefas são executadas no mesmo processador, a diferença dos tempos de início das duas deve ser pelo menos um $(t_j \geq t_i + 1)$, pois este é exatamente o tempo de processamento da tarefa i. Caso sejam executadas em processadores diferentes, o tempo de execução deve ser acrescido do tempo de comunicação (unitário), portanto a diferença entre os tempos de início é pelo menos dois $(t_j \geq t_i + 2)$. Observe que as restrições estruturais devem ser satisfeitas em ambas as variações do problema abordado, pois decorrem apenas do grafo considerado.

Em relação às restrições de recurso, se duas tarefas i, j são concorrentes, apesar de elas poderem ser executadas de forma independente, se não existem processadores disponíveis, esta execução simultânea não poderá acontecer. Ou seja, mesmo que as tarefas sejam independentes, os recursos podem não ser suficientes para que elas sejam efetivamente executadas concorrentemente. Portanto, se $i \parallel j$, temos dois casos: se $p_i \neq p_j$, não é preciso estabelecer relação entre t_i e t_j; senão $(p_i = p_j)$ temos que criar uma relação de precedência artificial entre as tarefas, fazendo com que $t_i \geq t_j + 1$ ou $t_j \geq t_i + 1$. Observe que se o número de processadores é ilimitado, não teremos necessidade de impor este tipo de restrição, pois sempre existirá um processador disponível para executar i e j ao mesmo tempo.

Em resumo, o escalonamento (T, P) é viável se satisfaz as seguintes restrições, para qualquer par de tarefas $i, j \in N$:

- Para $(i \prec j)$: $t_j \geq t_i + 1$, mas se $p_i \neq p_j$, então $t_j \geq t_i + 2$;
- Para $(i \parallel j)$, no caso limitado: se $p_i = p_j$, então $t_j \geq t_i + 1$ ou $t_i \geq t_j + 1$; se $p_i \neq p_j$, então os tempos de início não tem relação definida;

Apresentaremos agora soluções viáveis para escalonamentos do grafo de tarefas da Figura 2.2. As soluções são respectivamente para as variações $P|prec, c_{ij} = 1, p_j = 1|C_{max}$ e $P\infty|prec, c_{ij} = 1, p_j = 1|C_{max}$. Supomos que, para a primeira variação, o número de processadores é 2, ou seja, $m = 2$.

Observe que, em ambos os casos, as restrições estruturais são satisfeitas, pois se duas tarefas dependentes são executadas em um mesmo processador a diferença dos tempos de início é pelo menos um; e caso sejam executadas em processadores diferentes, a diferença é no mínimo 2. Já as restrições de recurso são necessárias apenas no primeiro exemplo, onde o

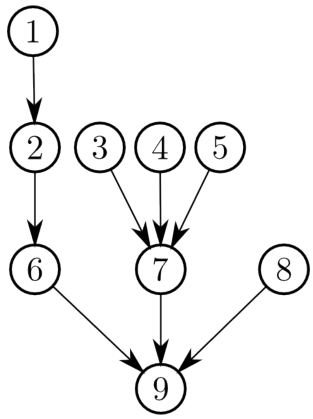


Figura 2.2: Grafo de entrada para o problema.

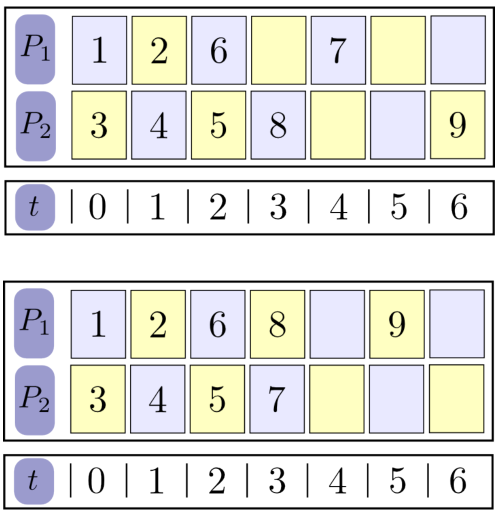


Figura 2.3: Soluções viável e ótima para o problema $P|prec, c_{ij} = 1, p_j = 1|C_{max}$

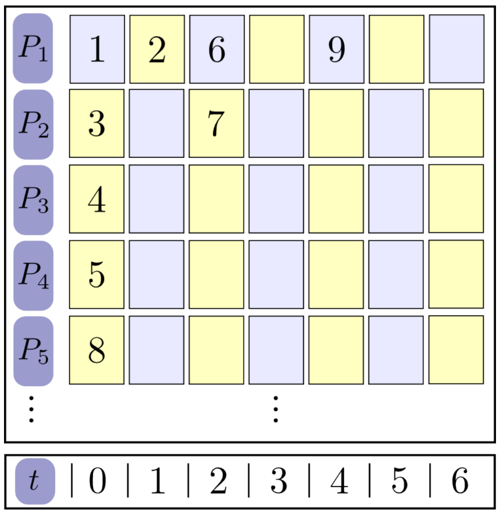


Figura 2.4: Solução ótima para o problema $P\infty|prec, c_{ij} = 1, p_j = 1|C_{max}$

número de processadores é limitado. Mesmo no caso de duas tarefas serem concorrentes, elas podem não ser executadas ao mesmo tempo. Isto ocorre por exemplo com as tarefas 3 e 4 que são completamente independentes, mas não são executadas concorrentemente, já que 1 é alocada no mesmo tempo que 3 e, portanto, não sobra processador para 4 no instante $t = 0$. Claramente, este fato não ocorre com infinitos processadores.

2.5 Complexidade Computacional

Nesta seção abordaremos aspectos relacionados à complexidade computacional do problema. Mostraremos a dificuldade das variações abordadas em seus casos gerais, onde não se conhece antecipadamente a estrutura do grafo, bem como classes de grafos onde o problema é sabidamente fácil. Abordaremos os limites de aproximação existentes na literatura para os casos mencionados anteriormente.

Vamos primeiramente analisar os aspectos relativos à variação $P|prec, c_{ij} = 1, p_j = 1|C_{max}$ do problema. Em [Pic92] e [RS87] este problema é demonstrado ser NP-Difícil para grafos de tarefas arbitrários. Por outro lado, podemos considerar classes específicas de grafos, a fim de que possamos fazer uso de informações sobre a estrutura do grafo na anáilse de complexidade do problema e, possivelmente, conseguir reduzi-la. Para grafos de intervalo ordenado, por exemplo, foi mostrado em [Sta04] um algoritmo $O(n)$. Lenstra et al. [LVV96] mostraram que, mesmo sendo o grafo de tarefas uma coleção de árvores, o problema, denotado por $P|tree, c_{ij} = 1, p_j = 1|C_{max}$, continua NP-Difícil.

Note que nessas análises o número de processadores m é limitado e é parte da entrada do problema. Todavia, tomando m como um valor fixo, é possível que o problema passe a ter resolução polinomial. Por exemplo, Vargarious et al., em [VRKL96], mostraram que $Pm|tree, c_{ij} = 1, p_j = 1|C_{max}$, onde m é fixo, pode ser resolvido em tempo polinomial por programação dinâmica. Um algoritmo linear para o caso $P2|tree, c_{ij} = 1, p_j = 1|C_{max}$ também é dado em [LVV96]. Para este mesmo problema, porém restrito à classe de grafos Série-Paralelo, Finta et al. [FLMB96] mostraram que $P2|serie - paralelo, c_{ij} = 1, p_j = 1|C_{max}$ é polinomial. Uma outra classe importante nessa discussão são as In-forests/Out-forests. Em [VRKL96] foi mostrado um algoritmo $O(n^{2m-2})$ para esta classe. Apesar desse algoritmo ser complexo, observe que, se m é fixo e não faz parte da entrada, o algoritmo é polinomial.

Finalmente Picouleau [Pic92] mostrou que o problema de decidir quando uma instância de $P|prec, c_{ij} = 1, p_j = 1|C_{max}$ tem um escalonamento de tamanho no máximo 3 é resolvido em tempo polinomial. Por outro lado, Hoogeveen et al. [HLV94] mostraram que decidir se uma instância desse mesmo problema possui um escalonamento de tamanho no máximo 4 é $NP - Completo$, até mesmo se o grafo de precedência é bipartido.

A partir destes dois resultados temos um limite de aproximabilidade. Deduzimos que não pode existir um algoritmo polinomial com aproximação menor que 5/4, a menos que $P = NP$. Do contrário, suponha que existisse um algoritmo com fator de aproximação inferior a 5/4, e sejam v o valor da solução viável retornada por esse algoritmo e OPT o valor ótimo do problema. Então, temos $\frac{4}{5}v < OPT \leq v$, de modo que $OPT \leq 4$ (caso $v \leq 4$) ou $OPT > 4$ (caso $v \geq 5$). Assim, decidiríamos, em tempo polinomial, se existe ou não uma solução ótima de valor menor ou igual a 4. Essa é a técnica do GAP, que é descrita em detalhes em [APMS+99].

Temos então um limite inferior para o fator de um algoritmo aproximativo para este problema. Por outro lado, o melhor limite superior, ou seja, a melhor aproximação para esta variação é obtida em [MH01]. Esta aproximação é baseada na melhor aproximação do problema irrestrito $P\infty|prec, c_{ij} = 1, p_j = 1|C_{max}$. O algoritmo aproximativo obtido possui aproximação de $7/3 - 4/m$.

Vamos considerar agora a variação irrestrita, ou seja, com infinitos processadores. Picouleau estabeleceu a NP-Completude para o problema de decidir quando uma instância de $P\infty|prec, c_{ij} = 1, p_j = 1|C_{max}$ possui um escalonamento de tamanho no máximo 8 em [Pic92]. Esse número foi diminuido por Hoogeveen et al. em [HLV94]. Eles demonstraram que também é NP-Completo decidir se uma instância possui um escalonamento de tamanho no máximo 6. E para um escalonamento de tamanho 5, decidir é polinomial.

Com esses resultados, temos também um limite inferior para o fator de aproximabilidade para a versão ilimitada, novamente pela técnica do GAP. Não existe então um algoritmo aproximativo com aproximação menor que 7/6 para essa variação do problema, a menos que $P = NP$. Para esta variação, a melhor aproximação descrita na literatura é 4/3, e foi obtida em [MK97] por Munier e König. A aproximação é baseada na relaxação linear de uma formulação de programação inteira que será abordada posteriormente nesse texto. Diferentemente da versão abordada anteriormente para processadores limitados, a variação ilimitada possui algoritmo linear para a estrutura de árvore (intree), ou seja, $P\infty|intree, c_{ij} = 1, p_j = 1|C_{max}$ possui um algoritmo linear.

Na literatura, existem muitas outras variantes para este mesmo problema, que diferem das que trabalhamos em apenas pequenos detalhes, como, por exemplo, alterações nos tempos de execução e comunicação. Não é nosso objetivo tratar dessas variantes, porém como parte integrante do trabalho de investigação de problemas correlatos e complementação da revisão bibliográfica feita, forneceremos a seguir a Tabela 2.5 com as seguintes colunas: o problema em notação $\alpha|\beta|\gamma$, a complexidade ou informação sobre algoritmo aproximativo e a referência em que podem ser encontrados mais detalhes.

Os casos gerais, quer dizer, para grafos de tarefas arbitrários, das variações abordadas nesse texto são ambos NP-Difíceis, ou seja, $P|prec, c_{ij} = 1, p_j = 1|C_{max}$ e $P\infty|prec, c_{ij} = 1, p_j = 1|C_{max}$ são NP-Difíceis. Neste trabalho, abordaremos tais problemas via Programação Inteira. Para isso, no próximo capítulo, apresentaremos algumas formulações matemáticas.

Complexidade computacional de problemas relacionados

Identificador	Notação $\alpha\|\beta\|\gamma$	Complexidade/Aproximação	Referência
1	$P\|prec, c_{ij} = 1, p_j = 1\|Cmax <= D$	NP-Completo	[Ull75]
2	$P2\|prec, c_{ij} = 0, p_j = 1\|Cmax$	Polinomial	[FKN69]
3	$P\|\text{bipartite}, c_{ij} = 1, p_j = 1\|Cmax <= 4$	NP-Completo	[HLV94]
4	$P\|prec, c_{ij} = 1, p_j = 1\|Cmax <= 3$	Polinomial	[Pic92]
5	$P\infty\|prec, c_{ij} = 1, p_j = 1\|Cmax <= 6$	NP-Completo	[HLV94]
6	$P\infty\|prec, c_{ij} = 1, p_j = 1\|Cmax <= 5$	Polinomial	[HLV94]
7	$P\infty\|\text{tree of depth} \leq 2, c_{ij}\|Cmax <= D$	NP-Completo	[PC95]
8	$P\infty\|tree, c_{ij} <= min_j p_j, p_j\|Cmax <= D$	$O(n)$	[Chr89]
9	$Pm\|\text{in-out forest}, c_{ij} = 1, p_j = 1\|Cmax$	$O(n^{2m-2})$	[VRKL96]
10	$P\|\text{in-out forest}, c_{ij} = 1, p_j = 1\|Cmax$	$\leq OPT + m - 2$	[VRKL96]
11	$P\|\text{in-out forest}, c_{ij} = 0, p_j = 1\|Cmax$	Polinomial	[Hu61]
12	$P\|\text{opposing forest}, c_{ij} = 0, p_j = 1\|Cmax$	NP-Completo	[MRGY83]
13	$Pm\|prec, c_{ij} = 0, p_j = 1\|Cmax$	Polinomial	[MRGY83]
14	$Pm\|\text{level orders}, c_{ij} = 0, p_j = 1\|Cmax$	Polinomial	[DW85]
15	$P2\|\text{serie-paralelo}, c_{ij} = 1, p_j = 1\|Cmax$	Polinomial	[FLMB96]
16	$P\infty\|prec, c_{ij} = 1, p_j = 1\|Cmax$	4/3	[MK97]
17	$P\|prec, c_{ij} = 1, p_j = 1\|Cmax$	2	[MH97]
18	$P\|prec, c_{ij} = 1, p_j = 1\|Cmax$	$7/3 - (4/m)$	[MH01]
19	$P\|\text{k-ary intree}, c_{ij} = c(1 \leq c \leq j-1), p_j = 1\|Cmax$	3	[FHa]
20	$P\infty\|\text{complete binary tree}, c_{ij}, p_j = 1\|Cmax$	NP-Difícil	[JR92]
21	$P\infty\|\text{binary tree}, c_{ij} = c, p_j = 1\|Cmax$	NP-Difícil	[JR92]
22	$P\infty\|prec, c_{ij} = c\|Cmax$	$O(n^{(c+1)})$	[JKS89]
23	$P\infty\|\text{complete k-ary intree}, c_{ij} = c\|Cmax$	$O(n^2 logn)$	[JR92]
24	$P\infty\|\text{complete k-ary intree}, c_{ij} = c(1 \leq c \leq j-1), p_j = 1\|Cmax$	$O(n)$	[FHb]
25	$P\infty\|\text{complete k-ary intree}, c_{ij}(j-1 \leq c_{ij} \leq j), p_j = 1\|Cmax$	$O(n)$	[FHb]
26	$P\infty\|prec, c_{ij} = c, p_j = 1\|Cmax$	2	[PY88]
27	$P\|\text{interval order}, c_{ij} = 1, p_j = 1\|Cmax$	$O(n)$	[Sta04]

Capítulo 3

Limites inferiores e superiores

O desenvolvimente de bons limites inferiores e superiores é decisivo para a qualidade e eficiência de métodos exatos. Neste capítulo, nosso objeto de estudo será portanto esses limites. Iniciaremos mostrando alguns algoritmos ou formas de se obter limites inferiores para os dois problemas de nosso interesse (escalonamento UET-UCT com número limitado ou ilimitado de processadores). Mostraremos dois algoritmos, um baseado em programação dinâmica, que generaliza um método presente em [CCMP02], e outro que se mostra mais satisfatório em relação ao mencionado anteriormente em alguns aspectos que serão abordados ao longo do capítulo. Para o limite superior, mostraremos três heurísticas que fornecem cotas superiores para nossos problemas, sendo as duas primeiras heurísticas gulosas e a terceira, uma meta-heurística iterativa. Por simplicidade de apresentação, consideraremos ao longo deste capítulo que 1 e n são, respectivamente, as únicas tarefas minimal e maximal no grafo de tarefas.

3.1 Limites inferiores

3.1.1 Limite inferior $LB_{i,j}$

Nesta seção mostraremos um método para a obtenção de um limite inferior para os problemas abordados. A base para esse método é a decomposição do grafo de tarefas em redes e o uso de programação dinâmica [CCMP02]. Tal estratégia foi originalmente aplicada por Chrétienne, quando a relação de precedência define uma árvore e o número de processadores é ilimitado [Chr89]. Em [CCMP02], o método é estendido para grafos de tarefas quaisquer. Aqui, introduzimos modificações na estratégia de [CCMP02], melhorando a qualidade do limite obtido. Para a sua apresentação, é necessária a introdução de novos conceitos e notações específicas.

Decomposição em cadeias

Para $N' \subseteq N$, $\prec_{N'}$ ($\vdash_{N'}$) é o subconjunto de $\prec$ ($\vdash$) restrito a pares de elementos pertencentes a N'; $\prec_{N'}$ também é transitiva, não-reflexiva e antissimétrica. Se $\prec_{N'}$ não contém tarefas incomparáveis, então esta relação é denominada uma *cadeia* de $\prec$ e diz-se que N' define ou induz tal cadeia (por abuso de linguagem, diremos frequentemente que N' é a cadeia). Já uma *anticadeia* de $\prec$ é um conjunto de tarefas incomparáveis duas a duas, e a *largura* de $\prec$, denotada $\omega(\prec)$, é a cardinalidade da maior anticadeia de $\prec$. Uma *extensão* $\prec'$ de $\prec$ é uma relação transitiva, não-reflexiva e antissimétrica em N tal que $\prec \subseteq \prec'$. No grafo de tarefas da Figura 3.1, $N' = \{1, 2, 5\}$ define uma cadeia, enquanto $N' = \{2, 3, 4\}$ é uma anticadeia. Uma extensão pode ser obtida incluindo a relação $(2, 3)$, entre as tarefas incomparáveis 2 e 3.

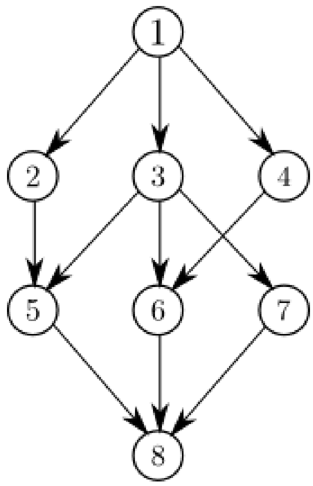


Figura 3.1: Exemplo de grafo de tarefas

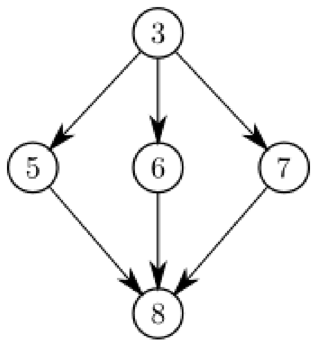


Figura 3.2: Rede $G_{3,8}$ do grafo da Figura 3.1

Dados $i, j \in N$, com $i \prec j$, defina $N_{i,j}$ como o conjunto de tarefas que são simultaneamente sucessoras de i e antecessoras de j, incluindo i e j. Por exemplo, na Figura 3.1, $N_{3,8} = \{3, 5, 6, 7, 8\}$. A *rede* $G_{i,j}$ é o subgrafo de $G = (N, \prec)$ induzido por $N_{i,j}$. Em paticular, note que $G_{1,n} = G$. Observe a rede $G_{3,8}$ do grafo da Figura 3.2. Por simplicidade, a relação $\prec_{N_{i,j}}$ ($\vdash_{N_{i,j}}$) será denotada por $\prec_{i,j}$ ($\vdash_{i,j}$).

Um *corte-cadeia* C de $G_{i,j}$ é uma cadeia da relação $\prec_{i,j}$ tal que, se $l \in C$ e $k \parallel l$, então $k \notin N_{i,j}$. Como toda tarefa de C (em particular a minimal e a maximal) ou é predecessora ou é sucessora de cada outra tarefa de $N_{i,j}$, temos que C define um corte de vértices de $G_{i,j}$ e que as tarefas de uma componente de $G_{i,j} \setminus C$ são todas ou sucessoras ou antecessoras de cada tarefa em C. Observe a Figura 3.3. Um *pedaço* de $G_{i,j}$ relativo a C é uma rede induzida pelas tarefas de uma componente B de $G_{i,j} \backslash C$ e ainda uma tarefa s pertencente ao corte-cadeia C. Se as tarefas de C são sucessoras de todas as de B, então s é minimal de C; caso contrário s será maximal em C.

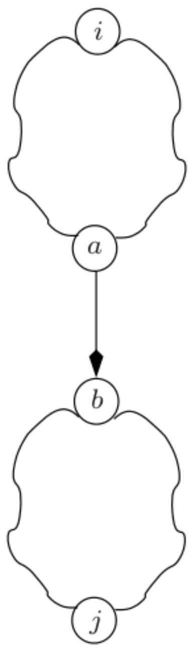


Figura 3.3: Ilustração de corte-cadeia $\prec_{a,b}$ de $G_{i,j}$

Podemos decompor o grafo $G_{i,j}$ em pedaços relativos a cortes-cadeia de $G_{i,j}$. Seja $\mathcal{C}$ um conjunto de cadeias disjuntas de G e $\mathcal{C}_{i,j}$, com $i \prec j$ e $i, j \in N$, um subconjunto maximal de $\mathcal{C}$ contendo apenas cortes-cadeia de $G_{i,j}$. Um pedaço de $G_{i,j}$ relativo a $\mathcal{C}_{i,j}$ é um pedaço de $G_{i,j}$ relativo a algum corte-cadeia de $\mathcal{C}_{i,j}$.

Uma *decomposição em cadeias* do grafo $G = (N, \prec)$, relativa a $\mathcal{C}$, é um grafo direcionado, com coloração nos arcos, $D = (V, E)$. A definição dos conjuntos de vértices e arcos de D é dada recursivamente da seguinte forma:

1. Definimos trivialmente que $[1, n] \in V$ é a raiz de D.

2. Seja $[k,l]$ um par de tarefas, com $k \prec l$, e $[i,j] \in V$, tal que $\prec_{i,j}$ não é uma cadeia. Existem três possibilidades para que este par pertença a V e para que $a = ([i,j],[k,l]) \in E$:

   - **$\mathcal{C}_{i,j} \neq \emptyset$ e $G_{k,l}$ é um pedaço de $G_{i,j}$:** Damos a cor **branca** ao arco a.
   - **$\mathcal{C}_{i,j} = \emptyset$ e $(i = k$ e $l \vdash j)$:** O arco a receberá a cor **azul**.
   - **$\mathcal{C}_{i,j} = \emptyset$ e $(j = l$ e $i \vdash k)$:** Neste caso, o arco a receberá a cor **vermelha**.

Um vértice $[i,j]$ de D é denominado folha se possui vizinhança positiva vazia, ou seja, se não existir $[k,l] \in V$, tal que $([i,j],[k,l]) \in E$. Veja que as folhas do grafo são cadeias e, portanto, o tempo mínimo de execução das folhas é o tamanho das cadeias. Perceba também que D é um grafo direcionado acíclico (DAG – *Directed Acyclic Graph*).

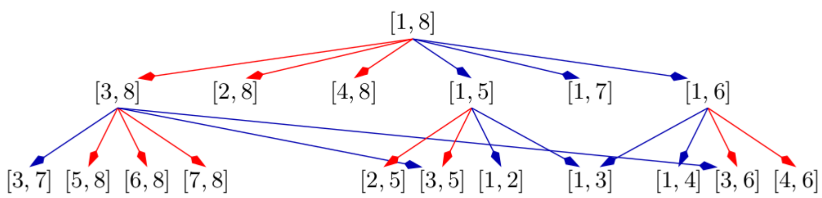


Figura 3.4: Exemplo de decomposição do grafo da Figura 3.1

Limite inferior a partir de D

Descreveremos agora como o grafo de decomposição $D = (V,E)$ de $G = (N,\prec)$ pode ser usado para a obtenção de um limite inferior para o escalonamento de $(N,\prec,Q)$. Seja a ordem $\{v_1 = [i_1,j_1],\ldots,v_{|V|} = [i_{|V|},j_{|V|}]\}$, com $v_i \in V$, obtida pela inversão de uma ordem topológica de V. O grafo D não possui ciclos orientados e, portanto, tal ordem existe, podendo ser obtida por uma busca em profundidade. Através de um processo de programação dinâmica o limite inferior para o par $[i,j]$ (ou melhor, para o *makespan* de $(N_{i,j},\prec_{i,j},Q)$) pode ser obtido em função do limite dos elementos de sua vizinhança positiva em D, que pela ordem anterior já devem ter sido calculados ao serem requisitados.

Dados $i \prec j$, vamos denotar por $LB_{i,j}$ um limite inferior para a diferença mínima entre os tempos de início das tarefas j e i em $(N,\prec,Q)$, ou seja, $t_j - t_i \geq LB_{i,j}$. Assim, $LB_{i,j}+1$ é um limite inferior para o *makespan* de $(N_{i,j},\prec_{i,j},Q)$. O cálculo de $LB_{i,j}$ segue a definição de D acima. O caso mais simples ocorre quando $[i,j]$ é uma folha do grafo D e, portanto, $\prec_{i,j}$ é uma cadeia. O limite inferior é o tamanho da própria cadeia, portanto temos que

$$LB_{i,j}^{folha} = |N_{i,j}| - 1$$

Considere agora um vértice $[i, j] \in V$ que não é uma folha do grafo D. Utilizaremos a classificação dos arcos apresentada anteriormente que se baseia no conjunto de cortes-cadeia, $\mathcal{C}_{i,j}$. Iniciamos com o caso em que $\mathcal{C}_{i,j} \neq \emptyset$. Portanto, sejam $[i'_1, j'_1], [i'_2, j'_2], \ldots, [i'_r, j'_r]$, $r \geq 1$, os pedaços de $G_{i,j}$, relativos a $\mathcal{C}_{i,j} = \{C_2^1, C_3^2 \ldots, C_r^{r-1}\}$, onde j'_t e i'_{t+1} são os vértices minimal e maximal, respectivamente, do corte-cadeia $C_{t+1}^t = G_{j'_t, i'_{t+1}}$, para $t \in \{1, \ldots, r-1\}$. Observe que o corte-cadeia C_{t+1}^t conecta os pedaços $[i'_t, j'_t]$ e $[i'_{t+1}, j'_{t+1}]$ e que $i'_1 = i$ e $j'_r = j$. Dessa forma, percebemos que $G_{i,j}$ é uma sequência de pedaços e cadeias (cortes-cadeia) intercalados e essa deve ser a exata ordem de execução no escalonamento, ou seja, a ordem de execução sempre será um pedaço $[i'_t, j'_t]$, o corte-cadeia C_{t+1}^t e o pedaço $[i'_{t+1}, j'_{t+1}]$. Observe a Figura 3.5.

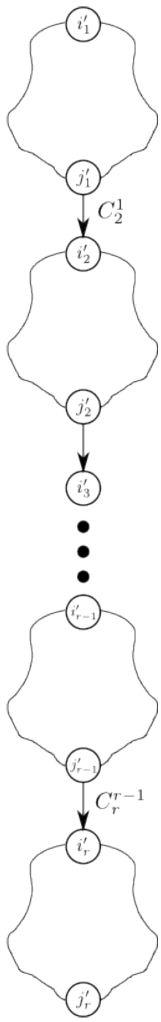


Figura 3.5: Exemplo de corte-cadeia e seus pedaços

Isto posto, um limite inferior para o subgrafo $G_{i,j}$ é:

$$LB_{i,j}^{branco} = \sum_{t=1}^{r} LB_{i'_t,j'_t} + \sum_{t=1}^{r-1}(|C_{t+1}^t| - 1)$$

Na primeira parcela somamos os limites de cada pedaço de $G_{i,j}$, relativos a $\mathcal{C}_{i,j}$, enquanto na segunda parcela somamos os limites das cadeias (cortes-cadeia), porém não adicionamos ao limite de cada cadeia C_{t+1}^t o tempo de execução do vértice maximal i'_{t+1}, pois este já está incluído no limite $LB_{i'_{t+1},j'_{t+1}}$.

Trataremos agora o caso em que $\mathcal{C}_{i,j} = \emptyset$, considerando os vértices da vizinhança positiva de $[i,j]$ em D, conforme sejam ligados por arcos azuis ou vermelhos. Separaremos estes vizinhos de acordo com a cor do arco e determinaremos um limite inferior para os dois grupos de vizinhos ($LB_{i,j}^{azul}$ e $LB_{i,j}^{vermelho}$). O limite inferior de $[i,j]$ será obtido como o melhor dos dois limites. Os cálculos de $LB_{i,j}^{azul}$ e $LB_{i,j}^{vermelho}$ são análogos, portanto mostraremos apenas a obtenção do limite para a cor azul.

Chamemos de vizinhos azuis aqueles ligados por um arco azul. Lembre que, se $[k,l]$ é vizinho positivo azul de $[i,j]$, então $k = i$ e $l \vdash j$. Inicialmente, tome os *diferentes* valores de limites inferiores dos vizinhos positivos azuis de $[i,j]$, $lb_1 < lb_2 < \ldots < lb_z$. Formalmente, fixados $i, j \in$, $i \prec j$, temos

$$\begin{aligned} lb_1 &= \min\{LB_{k,l} : ([i,j],[k,l]) \in E^{azul}\} = \min\{LB_{i,l} : i \prec l \vdash j\} \\ lb_{t+1} &= \min\{LB_{k,l} : ([i,j],[k,l]) \in E^{azul}, LB_{k,l} > lb_t\} \\ &= \min\{LB_{i,l} : i \prec l \vdash j, LB_{i,l} > lb_t\}, \quad \text{com } t = \{1, \ldots, z-1\} \end{aligned}$$

Perceba que mais de um vizinho pode possuir o mesmo valor de limite inferior. Denotaremos por Λ_t, com $t \in \{1, \ldots, z\}$, o cojunto definido pelos vizinhos positivos azuis de $[i,j]$ que possuem limite maior ou igual a lb_t. Formalmente:

$$\Lambda_t = \{l : i \prec l \vdash j, LB_{i,l} \geq lb_t\}, \quad \text{com } t = \{1, \ldots, z\}$$

Observação 1. *Se $l \in \Lambda_t$, então a tarefa l deve ser executada pelo menos lb_t unidades de tempo depois do início da execução de i.*

A partir da definição do conjunto Λ_t e da observação acima, podemos concluir que pelo menos $|\Lambda_t|$ tarefas devem ser executadas lb_t unidades de tempo após o início da tarefa i. Sabemos que a tarefa j deve ser executada após a execução de todas as tarefas $l \in \Lambda_t$. Portanto, a execução de j deve iniciar pelo menos $lb_t + \delta(\Lambda_t)$ unidades de tempo após o início da execução de i, sendo $\delta()$ uma função que associa às tarefas que precisam ser executadas, entre i e j, o mínimo de unidades de tempo que elas demandarão, considerando o número de processadores e as restrições de comunicação. Pelas restrições de precedência e comunicação, no máximo uma tarefa em Λ_t poderá ser executada exatamente uma unidade de tempo antes do início de j (pois todas são antecessoras de j); além de disso, as outras $|\Lambda_t| - 1$ restantes demandam no mínimo $\left\lceil \frac{|\Lambda_t|-1}{m} \right\rceil$ unidades de tempo para serem executadas. Em outras palavras, podemos fazer $\delta(\Lambda_t) = \left\lceil \frac{|\Lambda_t|-1}{m} \right\rceil + 1$. Mais detalhadamente, a definição da função segue a Tabela 3.1.

$\lvert\Lambda_t\rvert$ tarefas	$\delta(\Lambda_t)$, com $m < \infty$	$\delta(\Lambda_t)$, com $m = \infty$
$\lvert\Lambda_t\rvert = 1$	1	1
$2 \leq \lvert\Lambda_t\rvert \leq m+1$	2	2
$m+2 \leq \lvert\Lambda_t\rvert \leq 2m+1$	3	2
$2m+2 \leq \lvert\Lambda_t\rvert \leq 3m+1$	4	2
$\vdots$	$\vdots$	$\vdots$
$(\alpha-1)m+2 \leq \lvert\Lambda_t\rvert \leq \alpha m+1$	$\alpha+1$	2
$\vdots$	$\vdots$	$\vdots$
$\lvert\Lambda_t\rvert = \beta$	$\left\lceil \frac{\beta-1}{m} \right\rceil + 1$	$\min\{2, \beta\}$

Tabela 3.1: Definição da função δ

Portanto, quando $m < \infty$, o limite inferior para $[i, j]$, baseado nos vizinhos positivos azuis é:

$$LB^{azul}_{i,j} = \max_{t=1,\ldots,z} \left\{ lb_t + \left\lceil \frac{|\Lambda_t| - 1}{m} \right\rceil + 1 \right\}, \tag{3.1}$$

enquanto, para $m = \infty$, temos:

$$LB^{azul}_{i,j} = lb_z + \min\{2, |\Lambda_t|\} \tag{3.2}$$

Note que a expressão (3.2) corresponde exatamente a (3.1) se usarmos a convenção de que $\left\lceil \frac{a}{\infty} \right\rceil$, será igual a 0 ou 1, dependendo se $a = 0$ ou $a > 0$. Assim, o máximo em (3.1) com $m = \infty$ ocorre quando $t = z$.

Mesmo tratanto o caso limitado, em [CCMP02], o limite proposto é (3.2), ou seja, não usa o número de processadores em sua definição. Sendo assim, o limite $LB^{azul}_{i,j}$ aqui desenvolvido constitui uma melhoria em termos de qualidade daquele apresentado em [CCMP02].

Conforme mencionado anteriormente e como pode ser inferido a partir do desenvolvimento acima, o limite inferior baseado nos vizinhos positivos de $[i, j]$ ligados por arcos vermelhos, que denotaremos $LB^{vermelho}_{i,j}$, pode ser definido de forma análoga.

Ainda para o caso onde $\mathcal{C}_{i,j} = 0$ e $[i, j]$ não é folha, podemos definir um outro limite inferior para a diferença $t_j - t_i$ entre o tempo de início, t_j, de execução de j e o tempo de início de execução t_i de i, observando que $i \prec l \prec j$, $\forall l \in N_{i,j}$. Note que $|N_{i,j}| \geq 4$ e que no máximo uma tarefa começa sua execução em $t_i + 1$, o mesmo ocorrendo com $t_j - 1$. Sendo assim, as outras $|N_{i,j}| - 4$ (retirando estas duas além de i e j) demandam pelo menos $\left\lceil \frac{|N_{i,j}|-4}{m} \right\rceil$ unidades de tempo entre $t_i + 2$ e $t_j - 2$. Em outras palavras, $t_j - t_i \geq LB^{preto}_{i,j}$, onde:

$$LB^{preto}_{i,j} = \left\lceil \frac{|N_{i,j}| - 4}{m} \right\rceil + 3 \tag{3.3}$$

Reunindo as diferentes formas de calcular o limite $LB_{i,j}$, podemos formular a seguinte proposição:

Proposição 1. *Seja $\mathcal{C}$ um conjunto de cadeias disjuntas do grafo de tarefas $G = (N, \prec)$. Seja $D = (V, E)$ o grafo de decomposição de G relativo a $\mathcal{C}$. Para $[i, j] \in V(D)$, defina:*

$$LB_{i,j} = \begin{cases} LB^{folha}_{i,j}, & [i,j] \text{ é uma folha} \\ LB^{branco}_{i,j}, & \mathcal{C}_{i,j} \neq \emptyset \\ max\{LB^{azul}_{i,j}, LB^{vermelho}_{i,j}, LB^{preto}_{i,j}\}, & \mathcal{C}_{i,j} = \emptyset \end{cases}$$

Então, $LB_{i,j} + 1$ é um limite inferior para o makespan de $(N_{i,j}, \prec_{i,j}, Q)$.

Através da implementação desse limite é possível perceber que para algumas instâncias, quando o número de processadores é muito menor que a largura do grafo, ou seja, ($m \ll \omega(\prec)$), o valor obtido pelo algoritmo não é satisfatório, ficando distante do ótimo e assim dificultando a obtenção do mesmo. Ao final do capítulo apresentamos os resultados computacionais obtidos da implementação desse e dos outros limites. Para resolver os casos citados, onde não obtemos um limite adequado, buscamos outras formas para calcular limites inferiores para o problema. São os limites $IFB_{i,j}$ e $FFB_{i,j}$ que serão apresentados a seguir, ambos baseados no método original de [FB73].

3.1.2 Limite inferior $IFB_{i,j}$

Baseando-nos na proposta de [CCMP02], que melhora o limite obtido por [FB73], podemos buscar um novo limite, mais forte, para nosso problema. Para apresentar este novo limite, vamos revisitar os conceitos introduzidos em [FB73]. Iremos considerar $LB_{k,k} = 0$, para todo $k \in N$.

Sejam θ_1 e θ_2 dois inteiros, $\theta_2 > \theta_1$, tal que o intervalo $[\theta_1, \theta_2) \subseteq [0, LB_{i,j})$. Podemos definir o conjunto de tarefas de $G_{i,j}$ que obrigatoriamente devem ser executadas totalmente dentro do intervalo $[\theta_1, \theta_2)$, para cumprir os limites inferiores derivados na subseção anterior. Denotaremos este conjunto por $R_{i,j}(\theta_1, \theta_2)$. Em outros termos, defina:

$$R_{i,j}(\theta_1, \theta_2) = \{k \in N_{i,j} \mid [LB_{i,k}, LB_{i,j} + 1 - LB_{k,j}) \subseteq [\theta_1, \theta_2)\}.$$

Note que $R_{i,j}(\theta_1, \theta_2)$ não contém j, pois $\theta_2 \leq LB_{i,j}$, e portanto compreende sempre tarefas que precisam ser executadas antes de j. Na próxima subseção apresentaremos uma forma eficiente de calcular este conjunto.

A propriedade a ser explorada encontra-se no fato de que, para qualquer tarefa $k \in R_{i,j}(\theta_1, \theta_2)$, se $t_k \geq \theta_2$ em um escalonamento, o makespan de $(N_{i,j}, \prec_{i,j}, Q)$ nesse escalonamento será *maior* que $LB_{i,j} + 1$. Sabendo que $m * (\theta_2 - \theta_1)$ é o número máximo de tarefas que podem ser executadas no intervalo $[\theta_1, \theta_2)$, podemos melhorar o limite $LB_{i,j}$, quando:

$$S_{i,j}(\theta_1, \theta_2) = |R_{i,j}(\theta_1, \theta_2)| - m * (\theta_2 - \theta_1) > 0$$

Precisamente, para cumprir os limites inferiores $LB_{i,k}$ e $LB_{k,j}$, para todo $k \in R_{i,j}(\theta_1, \theta_2)$, podemos incrementar $LB_{i,j}$ conforme o seguinte teorema.

Proposição 2. *(Fernández and Bussel, 1973 [FB73]) Defina*

$$FB_{i,j} = \max_{\substack{[\theta_1,\theta_2) \subseteq [0, LB_{i,j}): \\ S_{i,j}(\theta_1,\theta_2) > 0}} \left\{ \left\lceil \frac{S_{i,j}(\theta_1, \theta_2)}{m} \right\rceil \right\}$$

Então, $LB_{i,j} + FB_{i,j} + 1$ é um limite inferior para o makespan de um escalonamento de $(N_{i,j}, \prec_{i,j}, Q)$.

Derivamos a seguir uma melhoria no limite dado acima. Note que, na definição de $FB_{i,j}$, nada é observado sobre as relações entre as tarefas em $R_{i,j}(\theta_1, \theta_2)$, apenas a cardinalidade desse conjunto. A nossa estratégia é considerar os limites $LB_{k,j}$, para $k \in R_{i,j}(\theta_1, \theta_2)$.

Proposição 3. *Seja $[\theta_1, \theta_2) \subseteq [0, LB_{i,j})$. Considere que as tarefas $l \in R_{i,j}(\theta_1, \theta_2)$ estejam ordenadas segundo os valores $LB_{l,j}$ em ordem não-crescente. Seja $r_l(\theta_1, \theta_2) \geq 1$ a posição da tarefa $l \in R_{i,j}(\theta_1, \theta_2)$ nesta ordem. Defina*

$$IFB_{i,j}(\theta_1, \theta_2) = \max_{l \in R_{i,j}(\theta_1, \theta_2)} \left\{ \theta_1 + \left\lceil \frac{r_l(\theta_1, \theta_2)}{m} \right\rceil + LB_{l,j} \right\} \tag{3.4}$$

Então,

$$IFB_{i,j} = \max_{[\theta_1,\theta_2) \subseteq [0, LB_{i,j})} IFB_{i,j}(\theta_1, \theta_2)$$

é um limite inferior para o makespan de um escalonamento $(N_{i,j}, \prec_{i,j}, Q)$.

Demonstração. Primeiro, lembre que todas as tarefas em $R_{i,j}(\theta_1, \theta_2)$ precisam ser escalonadas antes de j, a partir do tempo θ_1, e que $LB_{l,j}$ é o mínimo de tempo transcorrido entre os inícios de l e j. Sendo assim, um limite inferior para o makespan de $(N_{i,j}, \prec_{i,j}, Q)$ é dado pelo valor ótimo do problema:

$$\begin{aligned}
\min\ & t_j + 1 \\
\text{s.a:}\ & t_j - t_l \geq LB_{l,j}, \quad \forall l \in R_{i,j}(\theta_1, \theta_2) \\
& t_l \geq \theta_1, \quad \forall l \in R_{i,j}(\theta_1, \theta_2) \\
& |\{l \in R_{i,j}(\theta_1, \theta_2) : t_l = \theta\}| \leq m, \quad \forall \theta \in [\theta_1, \theta_2)
\end{aligned}$$

Uma solução ótima para este problema obtida fazendo

$$t_l = \theta_1 + \left\lceil \frac{r_l(\theta_1, \theta_2)}{m} \right\rceil - 1, \quad \forall l \in R_{i,j}(\theta_1, \theta_2),$$

e

$$t_j = \max\{t_l + LB_{l.j} : l \in R_{i,j}(\theta_1, \theta_2)\},$$

o que leva ao resultado desejado. Note que esta solução corresponde a escalonarmos as tarefas de $R_{i,j}(\theta_1, \theta_2)$ segundo a ordem definida no enunciado da proposição, respeitando a disponibilidade de processadores. □

Proposição 4. *O limite fornecido pela Proposição 3 é maior que ou igual àquele dado pela Proposição 2, ou seja,* $IFB_{i,j} \geq LB_{i,j} + FB_{i,j} + 1$.

Demonstração. Primeiro note que

$$IFB_{i,j}(\theta_1, \theta_2) = \max_{l \in R_{i,j}(\theta_1, \theta_2)} \left\{ \theta_2 + \left\lceil \frac{r_l(\theta_1, \theta_2) - m * (\theta_2 - \theta_1)}{m} \right\rceil + LB_{l,j} \right\}.$$

Seja $k \in R_{i,j}(\theta_1, \theta_2)$ a última tarefa na ordenação definida na Proposição 3, ou seja, $r_k(\theta_1, \theta_2) = |R_{i,j}(\theta_1, \theta_2)|$. Então, $S_{i,j}(\theta_1, \theta_2) = r_k(\theta_1, \theta_2) - m * (\theta_2 - \theta_1)$. E como $\theta_2 + LB_{k,j} \geq LB_{i,j} + 1$ (pela definição de $R_{i,j}(\theta_1, \theta_2)$), obtemos a desigualdade desejada. □

3.1.3 Limite inferior $FFB_{i,j}$

O limite introduzido em [FB73] pode ser visto de uma outra forma, baseada no número mínimo de processadores necessário para que possamos executar o escalonamento em até T unidades de tempo. O algoritmo que apresentamos adiante calcula explícita e diretamente um limite inferior para o *makespan* de um grafo de tarefas, uma vantagem sobre o limite anterior, que a cada intervalo percorrido incrementa o *makespan* em função do tamanho do conjunto de tarefas que não podem ser executadas naquele intervalo.

Considere um certo valor T de *makespan* e um intervalo $[\theta_1, \theta_2) \subseteq [0, T-1)$, novamente com θ_1 e θ_2 inteiros. Para $(N_{i,j}, \prec_{i,j}, Q)$, defina

$$m_L(T) = \max_{[\theta_1,\theta_2)\subseteq[0,T-1)} \left\lceil \frac{|R_{i,j}(\theta_1,\theta_2)|}{\theta_2 - \theta_1} \right\rceil$$

Note que $m_L(T)$ é um limite inferior para o número de processadores que possibilita a obtenção do makespan T.

Observe que $m_L(T)$ pode definir um novo parâmetro para decidirmos quando é possível melhorar um limite inferior. Claramente, se lb é um limite inferior do problema e $m < m_L(lb)$, então é possível melhorar este limite com a estratégia de [FB73]. Apresentaremos agora um algoritmo devido a [Fuj11], com algumas modificações e adaptações para se adequar melhor ao nosso problema específico (com custos unitários). Este algoritmo se mostra mais eficiente para calcular o limite de Fernández e Bussell, baseado em $m_L(T)$.

Nesse algoritmo, partimos de um limite inferior trivial, o caminho crítico, de tamanho t_{cp}. Aumentando este limite inferior em parcelas de tamanho Δ, o número de processadores necessários para executar o escalonamento no tempo $t_{cp}+\Delta$ será decrementado até que satisfaça o número inicial de processadores disponíveis para execução das tarefas. O objetivo é encontrar um *makespan* que satisfaça o limite de processadores disponíveis. Após este procedimento, buscaremos o menor tempo que consegue satisfazer a restrição do número de processadores. Vejamos uma descrição deste procedimento no Algoritmo 1.

Algoritmo 1: Limite inferior $FFB_{i,j}$

Chamada: FBBinarySearchMethod
Entrada: Tamanho do caminho crítico: t_{cp}
Resultado: Limite inferior para o *makespan* de $(N_{i,j}, \prec_{i,j}, Q)$: t'

$\Delta \leftarrow 1/2$;
repita
  $\Delta \leftarrow \Delta * 2$;
  $m' \leftarrow m_L(t_{cp} + \Delta)$;
até $m' > m$;
$low \leftarrow t_{cp} + \lfloor \Delta/2 \rfloor$;
$high \leftarrow t_{cp} + \Delta$;
Encontre o menor $t' \in \{low, low+1, \ldots, high\}$ tal que $m_L(t') \leq m$ usando busca binária.
retorne t'

Para que este algoritmo seja eficiente é importante que o cálculo de $m_L(T)$ não seja custoso. Para tanto, utilizamos uma adaptação do método mostrado em [Fuj11]. Observe que não necessariamente devemos iniciar o algoritmo a partir do caminho crítico. Caso tenhamos um limite inferior não trivial já calculado, por exemplo $LB_{i,j}(LB_{1,n})$, podemos partir deste limite; certamente, melhorando a qualidade do limite gerado pelo Algoritmo 1.

3.2 Limites superiores

Encontramos na literatura várias heurísticas para o problema $P|prec, c_{i,j}, p_j|C_{max}$, ou seja, o problema limitado aqui abordado, porém com custos de processamento e comunicação arbitrários. Tomando como base tais trabalhos, apresentaremos, nessa seção, três heurísticas que fornecem cotas superiores para o nosso problema (com custos unitários). Nossas heurísticas podem ser vistas como adaptações de heurísticas para o problema $P|prec, c_{i,j}, p_j|C_{max}$. Para obtermos versões dessas heurísticas relativas ao problema ilimitado, é suficiente executá-las tomando um número de processadores suficientemente grande, por exemplo, com $m = \omega(\prec)$.

Considerando o universo de heurísticas existentes para o problema limitado com custos arbitrários, orientamo-nos pelos estudos comparativos realizados em [JST08] e [BS13]. Dentre as heurísticas avaliadas em [JST08], destacam-se a *ISH - (Insertion Scheduling Heuristic)*, como a que apresenta os melhores resultados dentre as heurísticas gulosas, e a meta-heurística Tabu Search, que obteve um dos melhores resultados nas heurísticas iterativas. A heurística *ISH* tem sua origem em uma outra heurística gulosa *CP/MISF - (Critical Path/Most Immediate Successors First)*. Já no estudo, mais recente, realizado em [BS13], sobressai-se uma metaheurística *PSO - (Particle Swarm Optimization)*. O autor avalia o algoritmo *PSO* em relação às melhores heurísticas encontradas no estudo comparativo feito em [JST08]. É possível concluir de [BS13] que o algoritmo *PSO* obtém resultados melhores ou comparáveis ao Tabu Search.

A partir dessa análise, optamos por investir em adaptações das heurísticas CP/MISF, ISH e PSO. Antes de descrevermos nossos procedimentos, apresentamos brevemente os três tomados como base.

A primeira heurística, *CP/MISF - (Critical Path/Most Immediate Successors First)*, foi introduzida em [KN84] para o problema generalizado, considerando custos quaisquer de processamento e comunicação. A heurística escalona iterativamente as tarefas, respeitando os custos de execução e comunicação, de forma a favorecer a execução primeiramente dos sucessores imediatos que possuem maior caminho crítico até um terminal.

A estratégia da *CP/MISF* pode deixar, porém, espaços ociosos durante o escalonamento. Dessa forma, surge a ideia de um procedimento de inserção que leva em consideração estes espaços e aloca neles tarefas que possam executar naquela unidade de tempo.

Esta estratégia para o problema generalizado foi introduzida em [KB87], levando ao desenvolvimento da heurística *ISH - (Insertion Scheduling Heuristic)*, que, de forma simplificada, constitui-se em uma versão de *CP/MISF* com um esquema de alocação de tarefas livres em tempos ociosos.

Diferentemente das duas anteriores, que empregam estratégias gulosas, a terceira heurística, PSO, evolui a partir de uma população de soluções candidatas, comumente chamadas partículas, que são movidas no espaço de soluções de acordo com fórmulas matemáticas simples, definidas em função da posição corrente e velocidade de cada partícula. Uma implementação dessa meta-heurística para o problema com custos arbitrários foi proposta em [BS13], com bons resultados computacionais.

3.2.1 Heurística $CP/MISF$ adaptada

Apresentaremos agora uma heurística gulosa para o problema, baseada no caminho crítico de cada tarefa. Este tipo de abordagem é recorrente em problemas de escalonamento. Em particular, a heurística similar foi definida em [KN84] para o problema geral. Na verdade, a heurística que apresetamos a seguir foi usada nos experimentos computacionais realizados em [CCMP02] para o problema limitado e com custos unitários, porém agora a descrevemos em maiores detalhes.

Definimos *caminho crítico de uma tarefa* como o maior caminho entre ela e a tarefa maximal no grafo de tarefas. A cada iteração, a heurística $CP/MISF$ *(Critical path, most immediate successors first)* agenda uma tarefa cujos predecessores já foram agendados, respeitando os custos de comunicação e dando preferência àquela tarefa com maior caminho crítico. Adicionalmente, se as duas tarefas possuem caminhos críticos de mesmo tamanho, a heurística fará opção pela tarefa que possui mais sucessores imediatos.

A ideia central da heurística é organizar as tarefas em um heap sob os critérios mencionados anteriormente (maior caminho crítico e maior número de sucessores) e, a cada passo, selecionar uma delas para ser executada. O processador em que ela será executada depende do tempo de término de suas antecessoras, especificamente das predecessoras que executam no tempo mais tarde. Os processadores também são organizados em um heap, de acordo com o tempo de término da última tarefa alocada a ele, de forma crescente. Portanto, preferencialmente são escolhidos processadores que possuem menor tempo de término da última tarefa executada nele. Os detalhes da heurística $CP/MISF$ podem ser vistos no Algoritmo 2.

A entrada do algoritmo são os nós $i, j \in N$, que induzem o subgrafo $G_{i,j}$ para o qual devemos encontrar um escalonamento viável, e ainda o número de processadores, m. Nas linhas $1 - 4$, adicionamos ao heap de tarefas a tarefa minimal i em $G_{i,j}$, pois inicialmente apenas ela pode ser executada, e ainda calculamos o número de predecessores de cada tarefa no subgrafo. A variável $tproc[q]$, para $q \in Q$, indica o tempo de término da última tarefa alocada

Algoritmo 2: Heurística *CP/MISF (Critical path, most immediate successors first)*

Chamada: Cpmisf(i, j, m)
Entrada: Tarefas: i, j; Número de processadores: m
Resultado: Solução viável: (X,P) e *makespan*

```
heapTask.add(i);
para cada k ∈ N hacer
    npred[k] ← |{z|z ≺_ij k}| ;
fin
para cada q ∈ Q hacer
    heapProc.add(q);
    tproc[q] ← 0;
fin
enquanto heapTask ≠ ∅ faça
    elected ← heapTask[0];
    heapTask.remove(elected);
    sortHeap(heapTask);
    tmaxpred ← 0; nmaxpred ← 0;
    para cada k : k ≺_ij elected hacer
        se (taskTime[k] == tmaxpred) então
            nmaxpred ← nmaxpred + 1;
            procpred ← proc[k];
        fim
        se (taskTime[k] > tmaxpred) então
            tmaxpred ← taskTime[k];
            nmaxpred ← 1;
            procpred ← proc[k];
        fim
    fin
    se (nmaxpred == 0) então
        taskTime[elected] ← tproc[heapProc[0]];
        proc[elected] ← heapProc[0];
    senão se (nmaxpred == 1) e (tproc[procpred] ≤ tmaxpred + 1) então
        taskTime[elected] ← tmaxpred + 1;
        proc[elected] ← procpred;
    fim
    senão
        taskTime[elected] ← max{tmaxpred + 2, tproc[heapProc[0]]};
        proc[elected] ← heapProc[0];
    fim
    tproc[proc[elected]] ← taskTime[elected] + 1 ;
    sortHeap(heapProc);
    para cada k : elected ≺_ij k hacer
        npred[k] ← npred[k] − 1;
        se (npred[k] == 0) então
            heapTask.add(k); sortHeap(heapTask);
        fim
    fin
fim
retorne ((taskTime, proc), tproc[proc[j]])
```

a q até o momento. Portanto, inicialmente (linhas $5-8$) adicionamos cada processador ao heap de processadores e zeramos a variável *tproc* associada.

No laço principal da heurística, entre as linhas $10-24$, elegemos a tarefa que atende os critérios mencionados e determinamos seu predecessor que executa no tempo mais tarde (*tmaxpred*) e o processador (*tproc*[*pred*]) onde tal predecessor está sendo executada. O número de predecessores (*nmaxpred*) agendados nesse tempo máximo é também calculado.

Entre as linhas $25-36$ é feito o agendamento da tarefa eleita. Se não existirem predecessores, alocamos a tarefa ao processador obtido do heap de processadores. Se apenas um predecessor é executado no instante máximo (*tmaxpred*) e ele é a última tarefa alocada ao seu processador, alocamos a tarefa eleita ao mesmo processador de seu antecessor. Caso contrário, a tarefa eleita será alocada ao processador obtido do heap de processadores, tendo o cuidado de adequar seu tempo de início de acordo com seus predecessores e ainda a variável *tproc* do processador ao qual ela foi destinada. O heap de processadores é então atualizado, na linha 37, considerando a nova alocação.

Como agendamos a tarefa eleita, é possível que alguns de seus sucessores agora estejam livres para serem executados. Esta verificação é feita nas linhas $38-42$. Se existir um sucessor que possa ser executado, ele será adicionado ao heap de tarefas.

O escalonamento viável (T, P) é então definido pelos vetores *taskTime* e *proc*. O *makespan* $UB_{i,j}$ é obtido pelo valor máximo do instante de término das tarefas, ou seja, pelo valor máximo das variáveis *tproc*$[q]$, $q \in Q$. Este máximo é dado pela tarefa maximal j. Note que $UB_{1,n}$ é um limite superior para para o makespan ótimo do problema em G.

3.2.2 Heurística ISH adaptada

Pode-se verificar que a heurística $CP/MISF$ pode gerar escalonamentos em que existem processadores ociosos em certas unidades de tempo, ou seja, em certas unidades de tempo um processador pode não ser utilizado. Claramente, este fato pode aumentar o tempo total de processamento do conjunto de tarefas, principalmente se existir uma tarefa que poderia ser executada naquele processador na unidade de tempo em que ele permaneceu ocioso. A heurística ISH tenta corrigir essa falha. A estratégia adotada será justamente buscar uma tarefa que, ao ser colocada no processador ocioso, não quebre nenhuma regra, de precedência ou comunicação. O conceito principal desta heurística foi introduzido primeiramente por [KB87]. Descreveremos agora uma adaptação para o nosso problema.

O algoritmo é bastante simples (ver Algoritmo 3). A cada instante de tempo, percorremos o conjunto de processadores e tentamos alocar o maior número de tarefas possível, obedecendo as regras de precedência e comunicação. Usamos um vetor (*proc*) para registrar a tarefa em execução em cada processador no instante corrente. Precisamente, para cada processador $q \in Q$, $proc[q] = 0$, se o processador q está ocioso no instante corrente, e $proc[q] = i > 0$, se a tarefa i está sendo executada no momento.

A estrutura principal do algoritmo é um heap de tarefas livres (*heapTask*), ou seja, tarefas que estão prontas para executarem no atual instante de tempo. Este heap será organizado segundo algum critério. Os critérios que se mostraram mais eficazes foram os mesmos critérios utilizados na heurística $CP/MISF$: maior caminho crítico e maior número de sucessores imediatos.

Para nos auxiliar a determinar as tarefas livres, utilizamos um vetor (*npred*) que indica a situação atual de cada tarefa. Eis o significado de cada possível valor de uma entrada *npred*[k]:

- > 0 - o valor é o número de predecessores de k ainda não executados;
- 0 - tarefa k já pode ser executada em qualquer processador;
- -1 - tarefa k pode ser executada, desde que no mesmo processador que executou por último uma de suas predecessoras imediatas;
- -2 - todas as predecessoras de k foram executadas, mas k aguarda tempo de comunicação para iniciar sua execução.

Uma tarefa i é posta para executar em dois casos: *npred[i]* $= 0$, tarefa livre para executar em qualquer processador nesta unidade de tempo, ou *npred[i]* $= -1$, quando todas as predecessoras da tarefa foram executadas em instantes anteriores e apenas uma no instante imediatamente anterior.

Iniciamos com a tarefa minimal i, pois, como não possui predecessores, pode ser executada de forma irrestrita em relação à precedência e comunicação. Quando colocamos uma tarefa u para executar, devemos atualizar a situação de cada sucessor imediato. Se k for sucessor imediato de u, incrementamos *npred_exec[k]*, que indica o número de predecessores de k excutados na iteração corrente.

Para o próximo instante de tempo, precisamos também indicar o próximo estado da variável *npred*[k] para cada tarefa. Temos alguns casos para examinar:

- Se neste instante de tempo *npred*[k] < 0, no próximo a tarefa k poderá executar em qualquer processador; então faremos *npred*[k] $= 0$;

Algoritmo 3: Heurística *ISH (Insertion Scheduling Heuristic)*

Chamada: ISH(i, j, m)
Entrada: Tarefas: i, j; Número de processadores: m
Resultado: Solução viável: *makespan*

```
makespan ← 0; heapTask.add(i);
npred[k] ← |{z|z ⊢_ij k}| ∀k ∈ N ;
proc[q] ← 0 ∀q ∈ Q;
enquanto heapTask ≠ ∅ faça
    npred_exec[k] ← 0 ∀k ∈ N;
    para cada q ∈ Q hacer
        se (proc[q] ≠ 0) então
            se ∃u : (proc[q] ⊢ u) e (npred[u] == −1) então
                proc[q] ← u
                para cada k : u ⊢ k hacer
                    npred_exec[k] ← npred_exec[k] + 1;
                fin
                heapTask.remove(u);
            fim
            senão proc[q] ← 0;
        fim
        se (proc[q] == 0) então
            se (∃u : npred[u] == 0) então
                proc[q] ← u
                para cada k : u ⊢ k hacer
                    npred_exec[k] ← npred_exec[k] + 1;
                fin
                heapTask.remove(u);
            fim
        fim
    fin
    makespan ← makespan + 1;
    para cada k ∈ N hacer
        se (npred[k] < 0) então
            npred[k] ← 0;
        fim
        se (npred[k] > 0) então
            se (npred[k] == npred_exec[k]) então
                se (npred_exec[k] == 1) então
                    npred[k] = −1;
                fim
                senão
                    npred[k] = −2;
                fim
            fim
            senão
                npred[k] = npred[k] − npred_exec[k];
            fim
        fim
    fin
fim
retorne makespan
```

- Se todos os antecessores de k forem executados até esse instante de tempo, atualizaremos a variável $npred[k]$ de acordo com a quantidade total de antecessores que faltavam executar. Se existia apenas 1 para executar, $npred[k]$ será -1; caso contrário teremos que esperar um tempo de comunicação no próximo instante de tempo, $npred[k]$ será -2;
- Se nem todos os antecessores forem executados, atualizamos $npred[k] = npred[k] - npred\_exec[k]$, retirando o número de antecessores executados nesse instante de tempo;

Na descrição do Algoritmo 3 retornamos apenas o *makespan*, porém podemos modificá-lo facilmente para retornar também o escalonamento (T, P) que gera tal makespan.

3.2.3 Heurística PSO

Embora as heurísticas gulosas se mostrem muito eficientes do ponto de vista do esforço computacional demandado, é possível que consigamos melhorar o valor dos resultados obtidos por elas. Com esse objetivo buscamos uma heurística iterativa de base evolucionária. Geralmente, neste tipo de heurística, iniciamos com uma solução viável e, a cada passo do algoritmo, produzimos novas soluções iguais ou melhores que as anteriores.

Escolhemos a meta-heurística Enxame de Partículas *(Particle Swarm)* por se tratar de um método de fácil implementação e que se mostrou comparável e até melhor em algumas instâncias numa comparação feita em [BS13], com relação a outras meta-heurísticas como Algoritmos Genéticos e Tabu Search. Vale observar que os estudos feitos em [JST08] e [BS13] tratam o problema com custos de processamento e comunicação gerais. Fizemos algumas adaptações no algoritmo apresentado em [BS13] para que ele consiga gerar melhores soluções e para tornar as soluções viáveis para nosso problema específico.

O método de otimização por enxame de partículas (PSO) foi instroduzido por [KJY01]. Este método simula o comportamento de criaturas em sociedade no processo de busca por comida. Basicamente, cada indivíduo (partícula) do grupo possui uma posição e uma velocidade de deslocamento. Objetivando encontrar a melhor posição (com maiores suprimentos), cada indivíduo guarda a melhor posição em que já passou (ótimo local) e informa aos outros esta posição. Assim teremos como decidir a melhor posição dentre todas as já percorridas por algum indivíduo (ótimo global). A cada iteração, cada indivíduo seguirá, obedecendo sua velocidade de deslocamento, para a posição indicada por uma combinação entre os dois ótimos disponíveis para ele, seu ótimo local e o ótimo global. Uma função de avaliação da posição deverá ser definida a priori. A seguir exibimos um algoritmo genérico da otimização por enxame de partículas.

Algoritmo 4: *Enxame de partículas genérico*

```
Inicializar parâmetros;
Inicializar população;
para cada partícula hacer
    Avaliar sua posição;
    Inicializar ótimo local;
fin
Inicializar ótimo global; enquanto não terminado faça
    para cada partícula hacer
        Atualizar velocidade e posição;
        Avaliar sua posição;
        Atualizar ótimo local, se necessário;
    fin
    Atualizar ótimo global, se necessário;
fim
```

O método *PSO* foi originalmente desenvolvido para problemas que podem ser representados por vetores de posição e velocidade contínuos. Um vetor contínuo de posições é necessário para que consigamos atingir todo o espaço de posições possíveis. No caso de problemas discretos, será necessário converter, a cada iteração, o vetor de posições para um vetor de valores discretos. Na literatura, são utilizadas diferentes formas de conversão, de acordo com o problema, sendo comuns o arredodamento dos valores das posições ou o uso dos índices do vetor de posições.

Para o problema de escalonamento em multiprocessadores que tratamos, cada indivíduo (partícula) representará um escalonamento e, a cada iteração, este escalonamento será modificado e avaliado pelo makespan associado. Também será necessário converter o vetor de posições para valores discretos, conforme descreveremos a seguir

Em geral, o escalonamento de uma tarefa é definido pelo processador em que ela será executada e também pelo tempo de início da tarefa. Por conveniência na descrição das partículas, representaremos o escalonamento de uma tarefa apenas pelo processador que a executará. De posse do conjunto de tarefas que cada processador executará, é possível determinar os tempos de início das execuções das tarefas baseado nas restrições de precedência e comunicação.

Determinaremos, agora, o significado de cada estrutura usada no método *PSO* feito para o nosso problema.

Para cada partícula i, definiremos:

- Uma matriz de posições $X^i \in \mathbb{R}^{n \times m}$, onde X^i_{jq} é a entrada associada à alocação da tarefa j no processador q;
- Uma matriz de velocidade $V^i \in \mathbb{R}^{n \times m}$ da partícula i;

- Um vetor inteiro de posições $S^i \in \mathbb{Z}^n$, que é a conversão da matriz X^i em uma alocação das tarefas aos processadores. S^i_j descreve o processador em que a tarefa j será alocada na solução i, sendo dado por

$$S^i_j := \text{argmax}\{X^i_{jq} : q = 1, \ldots, m\}$$

Observe que esta definição indicará em qual processador S^i_j será mais "interessante" executar a tarefa j, segundo a linha j da matriz de posições X^i, referente à partícula i.

Como nosso objetivo é melhorar as soluções fornecidas pelas heurísticas de passo único, uma das partículas será iniciada com o escalonamento de menor valor entre as soluções encontradas pelas duas heurísticas de passo único apresentadas. Ou seja, para cada instância observamos o melhor escalonamento gerado por $CP/MISF$ e ISH, e este será atribuído a uma partícula. Dessa forma, a solução da heurística PSO será igual ou melhor que as anteriores.

O Algoritmo 5 descreve nossa heuristica PSO. Primeiro, determinamos aleatoriamente as matrizes X e V iniciais, para cada partícula, conforme linhas 7–17, através da definição dos mínimos e máximos adequados especificamente para posição e velocidade. Os mínimos locais inicais são calculados a partir da matriz X inicial, usando a fórmula de conversão anteriomente definida. Cada uma dessas soluções iniciais, dada por uma alocação das tarefas aos processados, é convertida em um escalonamento através do procedimetno *Produzir-escalonamento*, cujo makespan é calculado pela função *Avaliar*. O mínino global inicial é o melhor dentre as soluções iniciais.

O laço principal se estende entre os passos 19–38, sendo executado por no máximo um número *limite* de iterações ou até uma solução ótima ser encontrada. A cada iteração devemos atualizar o parâmetro ω, denominado peso de inércia, que indica o quão dependente a velocidade de uma iteração é da velocidade da iteração anterior. Devemos definir a priori, o valor inicial de ω.

A fórmula da atualização da velocidade, apesar de complexa, é flexível o suficiente para calibrarmos o algoritmo de acordo com a necessidade. Os parâmetros $c1 * r1$ e $c2 * r2$ permitem que privilegiemos o ótimo local e global, respectivamente. Os parâmetros $c1$ e $c2$ são fixos e definidos a priori, enquanto $r1$ e $r2$ são números aleatórios entre 0 e 1. Após a definição da velocidade das partículas na iteração corrente, precisamos atualizar os vetores de posição. Esta atualização é feita simplesmente somando a velocidade atual à posição da iteração anterior.

A discretização de X é feita como mencionado anteriormente. Assim podemos gerar novas soluções a partir da alocação das tarefas aos processadores, conforme cada S^i. Neste passo,

Algoritmo 5: Heurística *PSO (Particle Swarm Optimization)*

Chamada: PSO(s, t, m, n, p, $limite$)
Entrada: Tarefas: s, t; Número de processadores: m; Número de tarefas em N_{st}: n; Número de partículas: p; Limite de iterações: $limite$
Resultado: *makespan*

$x_{min} \leftarrow 0.0$;
$x_{max} \leftarrow m$;
$v_{min} \leftarrow -m$;
$v_{max} \leftarrow m$;
$k \leftarrow 1$;
$\omega \leftarrow 0.8$;
para cada $i \leftarrow 1...p$ **hacer**
    para cada $j \leftarrow 1...n$ **hacer**
        para cada $q \leftarrow 1...m$ **hacer**
            $X^i_{j,q} \leftarrow x_{min} + (x_{max} - x_{min}) * rand(0...1)$;
            $V^i_{j,q} \leftarrow v_{min} + (v_{max} - v_{min}) * rand(0...1)$;
        fin
        $S^i_j \leftarrow \text{argmax}\{X^i_{jq} : q = 1, \ldots, m\}$;
    fin
    $pB^i \leftarrow X^i$;
    $M_i \leftarrow$ Produzir-escalonamento(S^i, s, t);
    Avalie(M_i);
fin
$gB \leftarrow pB^i$ com melhor avaliação;
enquanto $k < limite$ **e** $makespan \geq LB_{1,n}$ **faça**
    $\alpha \leftarrow rand(0...1)$;
    $\omega \leftarrow \omega * \alpha$;
    para cada $i \leftarrow 1...p$ **hacer**
        $r1 \leftarrow rand(0...1)$;
        $r2 \leftarrow rand(0...1)$;
        $V^i \leftarrow \omega * V^i + c1 * r1 * (pB^i - X^i) + c2 * r2 * (gB - X^i)$;
        $X^i \leftarrow X^i + V^i$;
        para cada $j \leftarrow 1...n$ **hacer**
            $S^i_j \leftarrow \text{argmax}\{X^i_{jq} : q = 1, \ldots, m\}$;
        fin
    fin
    para cada $i \leftarrow 1...p$ **hacer**
        $M_i \leftarrow$ Produzir-escalonamento(S^i, s, t);
        Avaliar(M_i);
        Atualizar mínimo local, (pB^i);
    fin
    Atualizar mínimo global, gB;
    $k \leftarrow k + 1$;
fim
$makespan \leftarrow leght(gB)$;
retorne *makespan*

é importante observar que devemos gerar soluções que sigam as regras de precedência e comunicação, ou seja, soluções viáveis para o problema. Se uma solução gerada na partícula i é boa o suficiente local ou globalmente, armazenamos sua matriz de posições em pB^i ou gB respectivamente.

O algoritmo irá parar quando atingirmos um limite inferior ou o número de iterações for alcançado. O *makespan* será então dado pelo tempo total do escalonamento armazenado como ótimo global (gB).

Como mencionado, é interessante partir de uma boa solução inicial e assim tentar melhorá-la. Assim, em lugar de iniciar todas as partículas aleatoriamente, podemos, de forma simples, atribuir a uma delas o escalonamento obtido de uma das heurísticas anteriores. Basta gerarmos um vetor de posições baseado nesse escalonamento, percorrendo o caminho inverso do apresentado anteriormente.

3.3 Comparações entre os limites

Nesta seção, apresentamos os resultados computacionais das implementações realizadas dos limites apresentados neste capítulo. Implementamos os limites inferiores $LB_{i,j}$ e $FFB_{i,j}$ e os limites superiores $CP/MISF$, ISH e PSO. A não implementação efetiva do limite $FB_{i,j}$ (e sua versão melhorada $IFB_{i,j}$) se deu pelo fato de que os conceitos utilizados nele são os mesmos utilizados em $FFB_{i,j}$, tendo este último uma implementação mais simplificada.

Comparamos o desempenho dos limites para os problemas ilimitado e limitado, este último com 3 versões: $m = \lfloor \omega(\prec)/2 \rfloor$, $m = \lfloor \omega(\prec)/4 \rfloor$ e $m = \lfloor \omega(\prec)/8 \rfloor$, lembrando que $\omega(\prec)$ representa a largura do grafo de tarefas.

Separamos as instâncias em três conjuntos: Instâncias Aleatórias 1, Instâncias Aleatórias 2 e Instâncias Estruturadas. As instâncias deste último grupo possuem uma estrutura definida, por exemplo, as *bin* são árvores binárias. O conjunto de instâncias e o ambiente computacional utilizados estão descritos em detalhes no Apêndice A.

3.3.1 Limites Inferiores

Nas tabelas dos resultados dos limites inferiores temos quatro colunas: a primeira indica a instância do grafo de tarefas (DAG), a seguir temos os valores de cada *makespan* obtido pela resolução exata (IP), limites $LB_{i,j}$, limite $FFB_{i,j}$ a partir do caminho crítico e limite $FFB_{i,j}$ a partir de $LB_{i,j}$.

Aplicamos o limite $FFB_{i,j}$ apenas para o grafo completo, sem levar em cosideração os limites de subgrafos. Já para $LB_{i,j}$ é sabido que implicitamente os limites de cada subgrafo são importantes na composição do limite do grafo principal, tendo como referência o grafo de decomposição D.

Para a produção dos gráficos do limite inferior utilizamos a seguinte estratégia: tomamos o valor ótimo subtraído do limite inferior em questão e este resultado dividimos pelo valor ótimo. O resultado dessas operações, para cada instância, é plotado no gráfico.

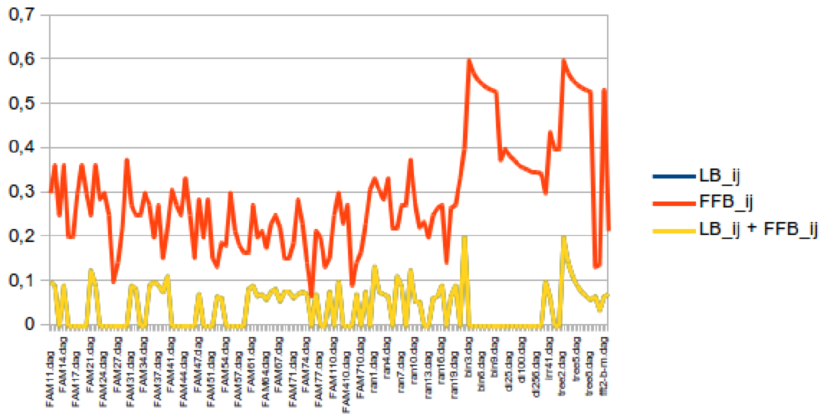


Figura 3.6: Comparação entre os limites inferiores para m ilimitado

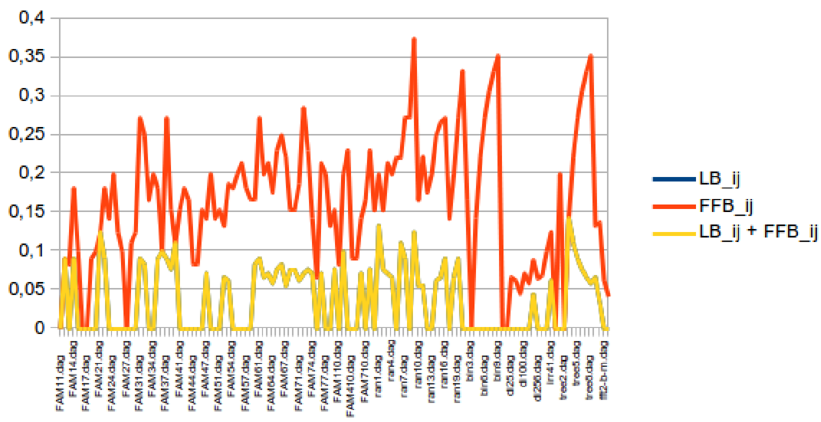


Figura 3.7: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/2 \rfloor$

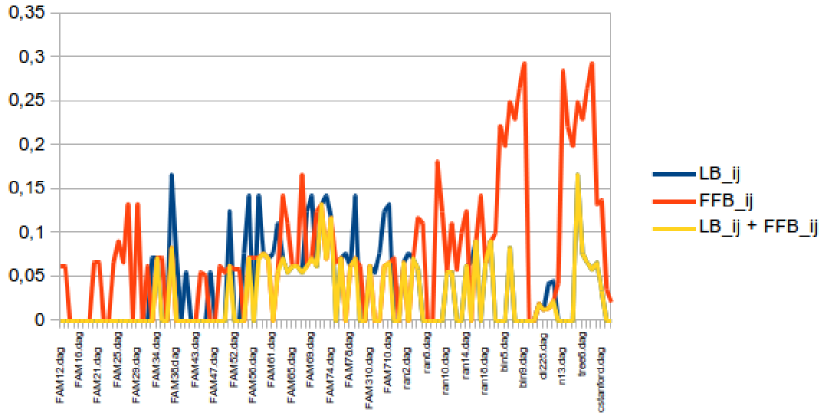


Figura 3.8: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/4 \rfloor$

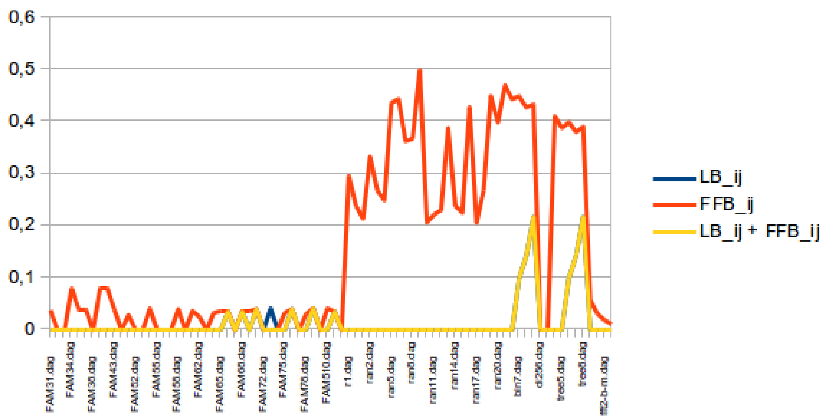


Figura 3.9: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/8 \rfloor$

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
FAM11.dag	10	9	7	9
FAM12.dag	11	10	7	10
FAM13.dag	12	12	9	12
FAM14.dag	11	10	7	10
FAM15.dag	10	10	8	10
FAM16.dag	10	10	8	10
FAM17.dag	10	10	7	10
FAM18.dag	11	11	7	11
FAM19.dag	10	10	7	10
FAM21.dag	8	7	6	7
FAM22.dag	11	10	7	10
FAM23.dag	7	7	5	7
FAM24.dag	10	10	7	10
FAM25.dag	8	8	6	8
FAM26.dag	10	10	9	10
FAM27.dag	7	7	6	7
FAM28.dag	9	9	7	9
FAM29.dag	8	8	5	8
FAM31.dag	11	10	8	10
FAM32.dag	12	11	9	11
FAM33.dag	12	12	9	12
FAM34.dag	10	10	7	10
FAM35.dag	11	10	8	10
FAM36.dag	10	9	8	9
FAM37.dag	11	10	8	10
FAM38.dag	13	12	11	12
FAM39.dag	9	8	7	8
FAM41.dag	13	13	9	13
FAM42.dag	11	11	8	11
FAM43.dag	12	12	9	12
FAM44.dag	12	12	8	12
FAM45.dag	12	12	9	12
FAM46.dag	13	13	11	13
FAM47.dag	14	13	10	13
FAM48.dag	15	15	12	15
FAM49.dag	14	14	10	14
FAM51.dag	13	13	11	13
FAM52.dag	15	14	13	14
FAM53.dag	16	15	13	15
FAM54.dag	11	11	9	11
FAM55.dag	10	10	7	10
FAM56.dag	14	14	11	14
FAM57.dag	11	11	9	11
FAM58.dag	12	12	10	12
FAM59.dag	12	11	10	11
FAM61.dag	11	10	8	10
FAM62.dag	15	14	12	14
FAM63.dag	14	13	11	13
FAM64.dag	17	16	14	16
FAM65.dag	13	12	10	12
FAM66.dag	12	11	9	11
FAM67.dag	18	17	14	17
FAM68.dag	13	12	11	12
FAM69.dag	13	12	11	12
FAM71.dag	16	15	13	15
FAM72.dag	14	13	10	13
FAM73.dag	13	12	10	12
FAM74.dag	14	13	12	13
FAM75.dag	15	15	14	15
FAM76.dag	14	13	11	13
FAM77.dag	10	10	8	10
FAM78.dag	15	15	13	15
FAM79.dag	13	12	11	12
FAM110.dag	12	12	9	12
FAM210.dag	10	9	7	9
FAM310.dag	13	13	10	13
FAM410.dag	11	11	8	11
FAM510.dag	11	11	10	11
FAM610.dag	14	13	12	13
FAM710.dag	12	12	10	12

Tabela 3.2: Comparação entre os limites inferiores para m ilimitado - Grupo de Instâncias Aleatórias 1

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
r1.dag	13	12	10	12
r2.dag	13	13	9	13
ran1.dag	15	13	10	13
ran2.dag	13	12	9	12
ran3.dag	14	13	10	13
ran4.dag	15	14	10	14
ran5.dag	9	9	7	9
ran6.dag	9	8	7	8
ran7.dag	11	10	8	10
ran8.dag	11	11	8	11
ran9.dag	8	7	5	7
ran10.dag	18	17	13	17
ran11.dag	18	17	14	17
ran12.dag	17	17	13	17
ran13.dag	10	10	8	10
ran14.dag	16	15	12	15
ran15.dag	15	14	11	14
ran16.dag	11	10	8	10
ran17.dag	21	21	18	21
ran18.dag	15	14	11	14
ran19.dag	11	10	8	10
ran20.dag	9	9	6	9

Tabela 3.3: Comparação entre os limites inferiores para m ilimitado - Grupo de Instâncias Aleatórias 2

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
arv2.dag	5	4	3	4
bin3.dag	5	5	2	5
bin4.dag	7	7	3	7
bin5.dag	9	9	4	9
bin6.dag	11	11	5	11
bin7.dag	13	13	6	13
bin8.dag	15	15	7	15
bin9.dag	17	17	8	17
dag3.dag	8	8	5	8
di16.dag	10	10	6	10
di25.dag	13	13	8	13
di36.dag	16	16	10	16
di64.dag	22	22	14	22
di100.dag	28	28	18	28
di144.dag	34	34	22	34
di225.dag	43	43	28	43
di256.dag	46	46	30	46
di400.dag	58	58	38	58
gauss.dag	10	9	7	9
irr41.dag	16	15	9	15
n7.dag	5	5	3	5
n13.dag	5	5	3	5
tree2.dag	5	4	2	4
tree3.dag	7	6	3	6
tree4.dag	9	8	4	8
tree5.dag	11	10	5	10
tree6.dag	13	12	6	12
tree7.dag	15	14	7	14
tree8.dag	17	16	8	16
celbow.dag	30	28	26	28
cstanford.dag	29	28	25	28
fft2-b-m.dag	15	14	7	14
iterative2-b-m.dag	14	13	11	13

Tabela 3.4: Comparação entre os limites inferiores para m ilimitado - Grupo de Instâncias Estruturadas

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
FAM11.dag	10	10	10	10
FAM12.dag	11	10	10	10
FAM13.dag	12	12	11	12
FAM14.dag	11	10	9	10
FAM15.dag	10	10	9	10
FAM16.dag	10	10	10	10
FAM17.dag	10	10	10	10
FAM18.dag	11	11	10	11
FAM19.dag	10	10	9	10
FAM21.dag	8	7	7	7
FAM22.dag	11	10	9	10
FAM23.dag	7	7	6	7
FAM24.dag	10	10	8	10
FAM25.dag	8	8	7	8
FAM26.dag	10	10	9	10
FAM27.dag	7	7	7	7
FAM28.dag	9	9	8	9
FAM29.dag	8	8	7	8
FAM31.dag	11	10	8	10
FAM32.dag	12	11	9	11
FAM33.dag	12	12	10	12
FAM34.dag	10	10	8	10
FAM35.dag	11	10	9	10
FAM36.dag	10	9	9	9
FAM37.dag	11	10	8	10
FAM38.dag	13	12	11	12
FAM39.dag	9	8	8	8
FAM41.dag	13	13	11	13
FAM42.dag	11	11	9	11
FAM43.dag	12	12	10	12
FAM44.dag	12	12	11	12
FAM45.dag	12	12	11	12
FAM46.dag	13	13	11	13
FAM47.dag	14	13	12	13
FAM48.dag	15	15	12	15
FAM49.dag	14	14	12	14
FAM51.dag	13	13	11	13
FAM52.dag	15	14	13	14
FAM53.dag	16	15	13	15
FAM54.dag	11	11	9	11
FAM55.dag	10	10	8	10
FAM56.dag	14	14	11	14
FAM57.dag	11	11	9	11
FAM58.dag	12	12	10	12
FAM59.dag	12	11	10	11
FAM61.dag	11	10	8	10
FAM62.dag	15	14	12	14
FAM63.dag	14	13	11	13
FAM64.dag	17	16	14	16
FAM65.dag	13	12	10	12
FAM66.dag	12	11	9	11
FAM67.dag	18	17	14	17
FAM68.dag	13	12	11	12
FAM69.dag	13	12	11	12
FAM71.dag	16	15	13	15
FAM72.dag	14	13	10	13
FAM73.dag	13	12	10	12
FAM74.dag	14	13	12	13
FAM75.dag	15	15	14	15
FAM76.dag	14	13	11	13
FAM77.dag	10	10	8	10
FAM78.dag	15	15	13	15
FAM79.dag	13	12	11	12
FAM110.dag	12	12	11	12
FAM210.dag	10	9	8	9
FAM310.dag	13	13	10	13
FAM410.dag	11	11	10	11
FAM510.dag	11	11	10	11
FAM610.dag	14	13	12	13
FAM710.dag	12	12	10	12

Tabela 3.5: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
r1.dag	13	12	10	12
r2.dag	13	13	11	13
ran1.dag	15	13	12	13
ran2.dag	13	12	11	12
ran3.dag	14	13	11	13
ran4.dag	15	14	12	14
ran5.dag	9	9	7	9
ran6.dag	9	8	7	8
ran7.dag	11	10	8	10
ran8.dag	11	11	8	11
ran9.dag	8	7	5	7
ran10.dag	18	17	15	17
ran11.dag	18	17	14	17
ran12.dag	17	17	14	17
ran13.dag	10	10	8	10
ran14.dag	16	15	12	15
ran15.dag	15	14	11	14
ran16.dag	11	10	8	10
ran17.dag	21	21	18	21
ran18.dag	15	14	12	14
ran19.dag	11	10	8	10
ran20.dag	9	9	6	9

Tabela 3.6: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
arv2.dag	6	6	5	6
bin3.dag	5	5	5	5
bin4.dag	7	7	6	7
bin5.dag	9	9	7	9
bin6.dag	11	11	8	11
bin7.dag	13	13	9	13
bin8.dag	15	15	10	15
bin9.dag	17	17	11	17
dag3.dag	8	8	8	8
di16.dag	10	10	10	10
di25.dag	15	15	14	15
di36.dag	16	16	15	16
di64.dag	22	22	21	22
di100.dag	28	28	26	28
di144.dag	34	34	32	34
di225.dag	45	43	41	43
di256.dag	46	46	43	46
di400.dag	58	58	54	58
gauss.dag	10	10	9	10
irr41.dag	16	15	14	15
n7.dag	5	5	5	5
n13.dag	5	5	4	5
tree2.dag	5	5	5	5
tree3.dag	7	6	6	6
tree4.dag	9	8	7	8
tree5.dag	11	10	8	10
tree6.dag	13	12	9	12
tree7.dag	15	14	10	14
tree8.dag	17	16	11	16
celbow.dag	30	28	26	28
cstanford.dag	29	28	25	28
fft2-b-m.dag	16	16	15	16
iterative2-b-m.dag	24	24	23	24

Tabela 3.7: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
FAM12.dag	16	16	15	16
FAM13.dag	16	16	15	16
FAM14.dag	15	15	15	15
FAM15.dag	15	15	15	15
FAM16.dag	15	15	15	15
FAM17.dag	15	15	15	15
FAM18.dag	16	16	16	16
FAM19.dag	15	15	14	15
FAM21.dag	15	15	14	15
FAM22.dag	16	16	16	16
FAM23.dag	10	10	10	10
FAM24.dag	15	15	14	15
FAM25.dag	11	11	10	11
FAM26.dag	15	15	14	15
FAM27.dag	15	15	13	15
FAM28.dag	15	15	15	15
FAM29.dag	15	15	13	15
FAM31.dag	14	14	14	14
FAM32.dag	16	16	15	16
FAM33.dag	14	13	14	14
FAM34.dag	14	13	13	13
FAM35.dag	14	14	13	14
FAM36.dag	17	17	17	17
FAM37.dag	12	10	11	11
FAM38.dag	14	13	14	14
FAM39.dag	13	13	13	13
FAM41.dag	18	17	18	18
FAM42.dag	13	13	13	13
FAM43.dag	13	13	13	13
FAM44.dag	18	18	17	18
FAM45.dag	19	19	18	19
FAM46.dag	18	17	18	18
FAM47.dag	18	18	18	18
FAM48.dag	16	16	15	16
FAM49.dag	18	18	17	18
FAM51.dag	16	14	15	15
FAM52.dag	17	17	16	17
FAM53.dag	17	17	16	17
FAM54.dag	13	12	13	13
FAM55.dag	14	12	13	13
FAM56.dag	14	14	13	14
FAM57.dag	14	12	13	13
FAM58.dag	13	12	12	12
FAM59.dag	14	13	13	13
FAM61.dag	13	12	13	13
FAM62.dag	18	16	17	17
FAM63.dag	14	13	12	13
FAM64.dag	18	17	16	17
FAM65.dag	16	15	15	15
FAM66.dag	16	15	15	15
FAM67.dag	18	17	15	17
FAM68.dag	16	14	15	15
FAM69.dag	14	12	13	13
FAM71.dag	16	15	14	15
FAM72.dag	15	13	13	13
FAM73.dag	14	12	13	13
FAM74.dag	17	15	15	15
FAM75.dag	15	15	14	15
FAM76.dag	14	13	13	13
FAM77.dag	13	12	13	13
FAM78.dag	16	15	15	15
FAM79.dag	14	12	13	13
FAM110.dag	16	16	15	16
FAM210.dag	11	11	11	11
FAM310.dag	16	15	15	15
FAM410.dag	18	17	18	18
FAM510.dag	13	12	13	13
FAM610.dag	16	14	15	15
FAM710.dag	15	13	14	14

Tabela 3.8: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
r1.dag	14	14	13	14
r2.dag	13	13	13	13
ran1.dag	15	14	14	14
ran2.dag	13	12	13	13
ran3.dag	14	13	13	13
ran4.dag	17	16	15	16
ran5.dag	9	9	8	9
ran6.dag	9	9	9	9
ran7.dag	11	11	11	11
ran8.dag	11	11	9	11
ran9.dag	8	8	7	8
ran10.dag	18	17	17	17
ran11.dag	18	17	16	17
ran12.dag	17	17	16	17
ran13.dag	10	10	9	10
ran14.dag	16	15	14	15
ran15.dag	15	14	15	15
ran16.dag	11	10	10	10
ran17.dag	21	21	18	21
ran18.dag	15	14	14	14
ran19.dag	11	10	10	10
ran20.dag	10	10	9	10

Tabela 3.9: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
bin4.dag	9	9	7	9
bin5.dag	10	10	8	10
bin6.dag	12	11	9	11
bin7.dag	13	13	10	13
bin8.dag	15	15	11	15
bin9.dag	17	17	12	17
di64.dag	34	34	34	34
di100.dag	52	52	52	52
di144.dag	52	51	51	51
di225.dag	79	78	78	78
di256.dag	70	67	69	69
di400.dag	88	84	86	86
irr41.dag	23	23	22	23
n13.dag	7	7	5	7
tree3.dag	9	9	7	9
tree4.dag	10	10	8	10
tree5.dag	12	10	9	10
tree6.dag	13	12	10	12
tree7.dag	15	14	11	14
tree8.dag	17	16	12	16
celbow.dag	30	28	26	28
cstanford.dag	29	28	25	28
fft2-b-m.dag	28	28	27	28
iterative2-b-m.dag	47	47	46	47

Tabela 3.10: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
FAM31.dag	26	26	25	26
FAM32.dag	27	27	27	27
FAM33.dag	25	25	25	25
FAM34.dag	25	25	23	25
FAM35.dag	25	25	24	25
FAM37.dag	25	25	24	25
FAM38.dag	25	25	25	25
FAM39.dag	25	25	23	25
FAM42.dag	25	25	23	25
FAM43.dag	25	25	24	25
FAM48.dag	26	26	26	26
FAM51.dag	35	35	34	35
FAM52.dag	36	36	36	36
FAM53.dag	36	36	36	36
FAM54.dag	24	24	23	24
FAM55.dag	24	24	24	24
FAM56.dag	24	24	24	24
FAM57.dag	24	24	24	24
FAM58.dag	25	25	24	25
FAM59.dag	24	24	24	24
FAM61.dag	27	27	26	27
FAM62.dag	40	40	39	40
FAM63.dag	21	21	21	21
FAM64.dag	30	30	29	30
FAM65.dag	28	28	27	28
FAM66.dag	28	27	27	27
FAM67.dag	28	28	28	28
FAM68.dag	28	27	27	27
FAM69.dag	27	27	26	27
FAM71.dag	25	24	24	24
FAM72.dag	23	23	23	23
FAM73.dag	24	23	24	24
FAM74.dag	32	32	32	32
FAM75.dag	30	30	29	30
FAM76.dag	25	24	24	24
FAM77.dag	23	23	23	23
FAM78.dag	32	32	31	32
FAM79.dag	24	23	23	23
FAM310.dag	26	26	26	26
FAM510.dag	24	24	23	24
FAM610.dag	28	27	27	27
FAM710.dag	30	30	30	30

Tabela 3.11: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
r1.dag	27	27	19	27
r2.dag	25	25	19	25
ran1.dag	28	28	22	28
ran2.dag	27	27	18	27
ran3.dag	26	26	19	26
ran4.dag	36	36	27	36
ran5.dag	16	16	9	16
ran6.dag	18	18	10	18
ran7.dag	22	22	14	22
ran8.dag	19	19	12	19
ran9.dag	16	16	8	16
ran10.dag	29	29	23	29
ran11.dag	27	27	21	27
ran12.dag	26	26	20	26
ran13.dag	18	18	11	18
ran14.dag	25	25	19	25
ran15.dag	31	31	24	31
ran16.dag	21	21	12	21
ran17.dag	29	29	23	29
ran18.dag	26	26	19	26
ran19.dag	20	20	11	20
ran20.dag	20	20	12	20

Tabela 3.12: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP	$LB_{i,j}$	$FFB_{i,j}$	$LB_{ij} + FFB_{ij}$
bin5.dag	17	17	9	17
bin6.dag	18	18	10	18
bin7.dag	20	18	11	18
bin8.dag	21	18	12	18
bin9.dag	23	18	13	18
di256.dag	130	130	130	130
di400.dag	202	202	202	202
tree4.dag	17	17	10	17
tree5.dag	18	18	11	18
tree6.dag	20	18	12	18
tree7.dag	21	18	13	18
tree8.dag	23	18	14	18
celbow.dag	35	35	33	35
cstanford.dag	32	32	31	32
fft2-b-m.dag	52	52	51	52
iterative2-b-m.dag	90	90	89	90

Tabela 3.13: Comparação entre os limites inferiores para $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Estruturadas

Analisando os resultados computacionais obtidos, podemos observar que o limite $LB_{i,j}$ é o mais equilibrado, quando percorremos as diversas quantidades de processadores disponíveis. Para a maioria das instâncias utilizadas, o limite $LB_{i,j}$ se aproxima satisfatoriamente do ótimo, independente do número de processadores. Entendemos que isto se deve pela abordagem utilizada no limite, que leva em consideração a estrutura do grafo de tarefas, com $LB_{i,j}^{folha}$, $LB_{i,j}^{branco}$, $LB_{i,j}^{azul}$, $LB_{i,j}^{vermelho}$, sem negligenciar os recursos disponíveis (processadores), com $LB_{i,j}^{preto}$.

Já o limite $FFB_{i,j}$ a partir do caminho crítico se sai melhor quando restringimos fortemente o número de processadores. Observe que, com $m = \lfloor \omega(\prec)/8 \rfloor$, as instâncias do Grupo Aleatórias 1 terão poucos processadores. Neste cenário o limite $FFB_{i,j}$ obtém seu melhor desempenho entre todos os experimentos realizados. Porém, neste caso, $LB_{i,j}$ obtém um excelente resultado, superando inclusive, na maioria das instâncias, o próprio $FFB_{i,j}$. Concluímos, assim, a superioridade de $LB_{i,j}$ em relação a $FFB_{i,j}$.

Quando aplicamos $FFB_{i,j}$ partindo do limite $LB_{i,j}$, percebemos que ocorre melhoria significativa apenas em instâncias com número médio de processadores, $m = \lfloor \omega(\prec)/4 \rfloor$. Para os outros casos, $FFB_{i,j}$ consegue melhorar $LB_{i,j}$ em apenas um instância e, mesmo assim, com um ganho pouco relevante.

3.3.2 Limites Superiores

Nas tabelas dos resultados dos limites superiores temos cinco colunas: a primeira indica a instância do grafo de tarefas(DAG), a seguir temos os valores de cada *makespan* obtido pela resolução exata (IP), limites $CP/MISF$, ISH e PSO.

As instâncias testadas são exatamente as mesmas utilizadas nos experimentos dos limites inferiores.

Para a produção dos gráficos do limite superior utilizamos a seguinte estratégia: tomamos o valor heurístico em questão subtraído do ótimo e este resultado dividimos pelo valor ótimo. O resultado dessas operações, para cada instância, é plotado no gráfico.

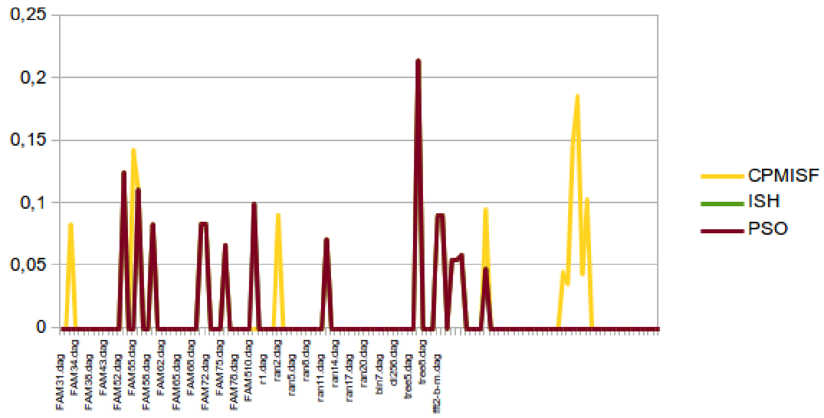


Figura 3.10: Comparação entre os limites superiores para m ilimitado

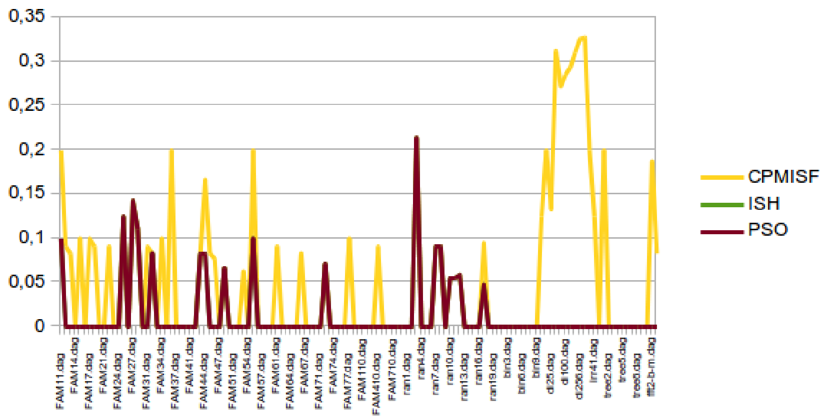


Figura 3.11: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/2 \rfloor$

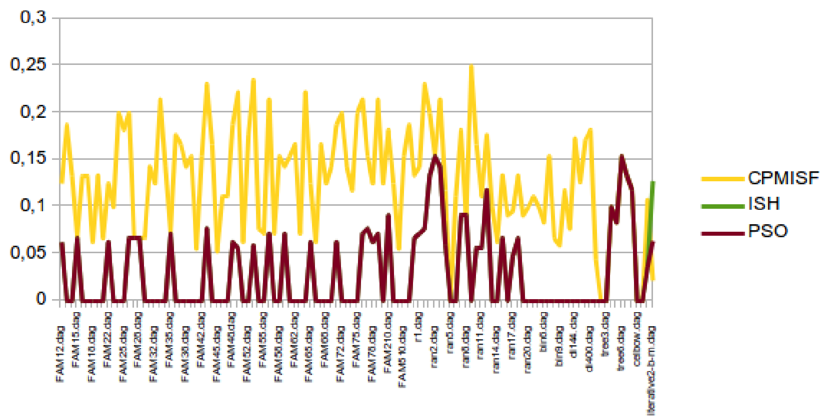


Figura 3.12: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/4 \rfloor$

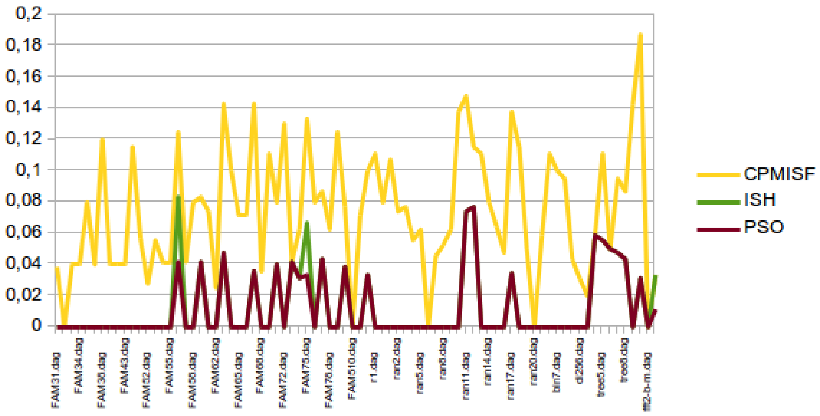


Figura 3.13: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/8 \rfloor$

DAG	IP	CPMISF	ISH	PSO
FAM11.dag	10	10	10	10
FAM12.dag	11	11	11	11
FAM13.dag	12	13	12	12
FAM14.dag	11	11	11	11
FAM15.dag	10	10	10	10
FAM16.dag	10	10	10	10
FAM17.dag	10	10	10	10
FAM18.dag	11	11	11	11
FAM19.dag	10	10	10	10
FAM21.dag	8	8	8	8
FAM22.dag	11	11	11	11
FAM23.dag	7	7	7	7
FAM24.dag	10	10	10	10
FAM25.dag	8	9	9	9
FAM26.dag	10	10	10	10
FAM27.dag	7	8	7	7
FAM28.dag	9	10	10	10
FAM29.dag	8	8	8	8
FAM31.dag	11	11	11	11
FAM32.dag	12	13	13	13
FAM33.dag	12	12	12	12
FAM34.dag	10	10	10	10
FAM35.dag	11	11	11	11
FAM36.dag	10	10	10	10
FAM37.dag	11	11	11	11
FAM38.dag	13	13	13	13
FAM39.dag	9	9	9	9
FAM41.dag	13	13	13	13
FAM42.dag	11	11	11	11
FAM43.dag	12	13	13	13
FAM44.dag	12	13	13	13
FAM45.dag	12	12	12	12
FAM46.dag	13	13	13	13
FAM47.dag	14	14	14	14
FAM48.dag	15	16	16	16
FAM49.dag	14	14	14	14
FAM51.dag	13	13	13	13
FAM52.dag	15	15	15	15
FAM53.dag	16	16	16	16
FAM54.dag	11	11	11	11
FAM55.dag	10	10	11	11
FAM56.dag	14	14	14	14
FAM57.dag	11	11	11	11
FAM58.dag	12	12	12	12
FAM59.dag	12	12	12	12
FAM61.dag	11	12	11	11
FAM62.dag	15	15	15	15
FAM63.dag	14	14	14	14
FAM64.dag	17	17	17	17
FAM65.dag	13	13	13	13
FAM66.dag	12	12	12	12
FAM67.dag	18	18	18	18
FAM68.dag	13	13	13	13
FAM69.dag	13	13	13	13
FAM71.dag	16	16	16	16
FAM72.dag	14	15	15	15
FAM73.dag	13	13	13	13
FAM74.dag	14	14	14	14
FAM75.dag	15	15	15	15
FAM76.dag	14	14	14	14
FAM77.dag	10	10	10	10
FAM78.dag	15	15	15	15
FAM79.dag	13	13	13	13
FAM110.dag	12	12	12	12
FAM210.dag	10	10	10	10
FAM310.dag	13	13	13	13
FAM410.dag	11	11	11	11
FAM510.dag	11	11	11	11
FAM610.dag	14	14	14	14
FAM710.dag	12	12	12	12

Tabela 3.14: Comparação entre os limites superiores para m ilimitado - Grupo de Instâncias Aleatórias 1

DAG	IP	CPMISF	ISH	PSO
r1.dag	13	13	13	13
r2.dag	13	13	13	13
ran1.dag	15	15	15	15
ran2.dag	13	13	13	13
ran3.dag	14	17	17	17
ran4.dag	15	15	15	15
ran5.dag	9	9	9	9
ran6.dag	9	9	9	9
ran7.dag	11	12	12	12
ran8.dag	11	12	12	12
ran9.dag	8	8	8	8
ran10.dag	18	19	19	19
ran11.dag	18	19	19	19
ran12.dag	17	18	18	18
ran13.dag	10	10	10	10
ran14.dag	16	16	16	16
ran15.dag	15	15	15	15
ran16.dag	11	11	11	11
ran17.dag	21	23	22	22
ran18.dag	15	15	15	15
ran19.dag	11	11	11	11
ran20.dag	9	9	9	9

Tabela 3.15: Comparação entre os limites superiores para m ilimitado - Grupo de Instâncias Aleatórias 2

DAG	IP	CPMISF	ISH	PSO
arv2.dag	5	5	5	5
bin3.dag	5	5	5	5
bin4.dag	7	7	7	7
bin5.dag	9	9	9	9
bin6.dag	11	11	11	11
bin7.dag	13	13	13	13
bin8.dag	15	15	15	15
bin9.dag	17	17	17	17
dag3.dag	8	8	8	8
di16.dag	10	10	10	10
di25.dag	13	13	13	13
di36.dag	16	16	16	16
di64.dag	22	23	22	22
di100.dag	28	29	28	28
di144.dag	34	39	34	34
di225.dag	43	51	43	43
di256.dag	46	48	46	46
di400.dag	58	64	58	58
gauss.dag	10	10	10	10
irr41.dag	16	16	16	16
n7.dag	5	5	5	5
n13.dag	5	5	5	5
tree2.dag	5	5	5	5
tree3.dag	7	7	7	7
tree4.dag	9	9	9	9
tree5.dag	11	11	11	11
tree6.dag	13	13	13	13
tree7.dag	15	15	15	15
tree8.dag	17	17	17	17
celbow.dag	30	30	30	30
cstanford.dag	29	29	29	29
fft2-b-m.dag	15	15	15	15
iterative2-b-m.dag	14	14	14	14

Tabela 3.16: Comparação entre os limites superiores para m ilimitado - Grupo de Instâncias Estruturadas

DAG	IP	CPMISF	ISH	PSO
FAM11.dag	10	12	11	11
FAM12.dag	11	12	11	11
FAM13.dag	12	13	12	12
FAM14.dag	11	11	11	11
FAM15.dag	10	11	10	10
FAM16.dag	10	10	10	10
FAM17.dag	10	11	10	10
FAM18.dag	11	12	11	11
FAM19.dag	10	10	10	10
FAM21.dag	8	8	8	8
FAM22.dag	11	12	11	11
FAM23.dag	7	7	7	7
FAM24.dag	10	10	10	10
FAM25.dag	8	9	9	9
FAM26.dag	10	10	10	10
FAM27.dag	7	8	8	8
FAM28.dag	9	10	10	10
FAM29.dag	8	8	8	8
FAM31.dag	11	12	11	11
FAM32.dag	12	13	13	13
FAM33.dag	12	12	12	12
FAM34.dag	10	11	10	10
FAM35.dag	11	11	11	11
FAM36.dag	10	12	10	10
FAM37.dag	11	11	11	11
FAM38.dag	13	13	13	13
FAM39.dag	9	9	9	9
FAM41.dag	13	13	13	13
FAM42.dag	11	11	11	11
FAM43.dag	12	13	13	13
FAM44.dag	12	14	13	13
FAM45.dag	12	13	12	12
FAM46.dag	13	14	13	13
FAM47.dag	14	14	14	14
FAM48.dag	15	16	16	16
FAM49.dag	14	14	14	14
FAM51.dag	13	13	13	13
FAM52.dag	15	15	15	15
FAM53.dag	16	17	16	16
FAM54.dag	11	11	11	11
FAM55.dag	10	12	11	11
FAM56.dag	14	14	14	14
FAM57.dag	11	11	11	11
FAM58.dag	12	12	12	12
FAM59.dag	12	12	12	12
FAM61.dag	11	12	11	11
FAM62.dag	15	15	15	15
FAM63.dag	14	14	14	14
FAM64.dag	17	17	17	17
FAM65.dag	13	13	13	13
FAM66.dag	12	13	12	12
FAM67.dag	18	18	18	18
FAM68.dag	13	13	13	13
FAM69.dag	13	13	13	13
FAM71.dag	16	16	16	16
FAM72.dag	14	15	15	15
FAM73.dag	13	13	13	13
FAM74.dag	14	14	14	14
FAM75.dag	15	15	15	15
FAM76.dag	14	14	14	14
FAM77.dag	10	11	10	10
FAM78.dag	15	15	15	15
FAM79.dag	13	13	13	13
FAM110.dag	12	12	12	12
FAM210.dag	10	10	10	10
FAM310.dag	13	13	13	13
FAM410.dag	11	12	11	11
FAM510.dag	11	11	11	11
FAM610.dag	14	14	14	14
FAM710.dag	12	12	12	12

Tabela 3.17: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP	CPMISF	ISH	PSO
r1.dag	13	13	13	13
r2.dag	13	13	13	13
ran1.dag	15	15	15	15
ran2.dag	13	13	13	13
ran3.dag	14	17	17	17
ran4.dag	15	15	15	15
ran5.dag	9	9	9	9
ran6.dag	9	9	9	9
ran7.dag	11	12	12	12
ran8.dag	11	12	12	12
ran9.dag	8	8	8	8
ran10.dag	18	19	19	19
ran11.dag	18	19	19	19
ran12.dag	17	18	18	18
ran13.dag	10	10	10	10
ran14.dag	16	16	16	16
ran15.dag	15	15	15	15
ran16.dag	11	11	11	11
ran17.dag	21	23	22	22
ran18.dag	15	15	15	15
ran19.dag	11	11	11	11
ran20.dag	9	9	9	9

Tabela 3.18: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP	CPMISF	ISH	PSO
arv2.dag	6	6	6	6
bin3.dag	5	5	5	5
bin4.dag	7	7	7	7
bin5.dag	9	9	9	9
bin6.dag	11	11	11	11
bin7.dag	13	13	13	13
bin8.dag	15	15	15	15
bin9.dag	17	17	17	17
dag3.dag	8	9	8	8
di16.dag	10	12	10	10
di25.dag	15	17	15	15
di36.dag	16	21	16	16
di64.dag	22	28	22	22
di100.dag	28	36	28	28
di144.dag	34	44	34	34
di225.dag	45	59	45	45
di256.dag	46	61	46	46
di400.dag	58	77	58	58
gauss.dag	10	12	10	10
irr41.dag	16	18	16	16
n7.dag	5	5	5	5
n13.dag	5	6	5	5
tree2.dag	5	5	5	5
tree3.dag	7	7	7	7
tree4.dag	9	9	9	9
tree5.dag	11	11	11	11
tree6.dag	13	13	13	13
tree7.dag	15	15	15	15
tree8.dag	17	17	17	17
celbow.dag	30	30	30	30
cstanford.dag	29	29	29	29
fft2-b-m.dag	16	19	16	16
iterative2-b-m.dag	24	26	24	24

Tabela 3.19: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	IP	CPMISF	ISH	PSO
FAM12.dag	16	18	17	17
FAM13.dag	16	19	16	16
FAM14.dag	15	17	15	15
FAM15.dag	15	16	16	16
FAM16.dag	15	17	15	15
FAM17.dag	15	17	15	15
FAM18.dag	16	17	16	16
FAM19.dag	15	17	15	15
FAM21.dag	15	16	15	15
FAM22.dag	16	18	17	17
FAM23.dag	10	11	10	10
FAM24.dag	15	18	15	15
FAM25.dag	11	13	11	11
FAM26.dag	15	18	16	16
FAM27.dag	15	16	16	16
FAM28.dag	15	16	16	16
FAM29.dag	15	16	15	15
FAM31.dag	14	16	14	14
FAM32.dag	16	18	16	16
FAM33.dag	14	17	14	14
FAM34.dag	14	16	14	14
FAM35.dag	14	15	15	15
FAM36.dag	17	20	17	17
FAM37.dag	12	14	12	12
FAM38.dag	14	16	14	14
FAM39.dag	13	15	13	13
FAM41.dag	18	19	18	18
FAM42.dag	13	15	13	13
FAM43.dag	13	16	14	14
FAM44.dag	18	21	18	18
FAM45.dag	19	20	19	19
FAM46.dag	18	20	18	18
FAM47.dag	18	20	18	18
FAM48.dag	16	19	17	17
FAM49.dag	18	22	19	19
FAM51.dag	16	17	16	16
FAM52.dag	17	20	17	17
FAM53.dag	17	21	18	18
FAM54.dag	13	14	13	13
FAM55.dag	14	15	14	14
FAM56.dag	14	17	15	15
FAM57.dag	14	15	14	14
FAM58.dag	13	15	13	13
FAM59.dag	14	16	15	15
FAM61.dag	13	15	13	13
FAM62.dag	18	21	18	18
FAM63.dag	14	15	14	14
FAM64.dag	18	22	18	18
FAM65.dag	16	18	17	17
FAM66.dag	16	17	16	16
FAM67.dag	18	21	18	18
FAM68.dag	16	18	16	16
FAM69.dag	14	16	14	14
FAM71.dag	16	19	17	17
FAM72.dag	15	18	15	15
FAM73.dag	14	16	14	14
FAM74.dag	17	19	17	17
FAM75.dag	15	18	15	15
FAM76.dag	14	17	15	15
FAM77.dag	13	15	14	14
FAM78.dag	16	18	17	17
FAM79.dag	14	17	15	15
FAM110.dag	16	18	16	16
FAM210.dag	11	13	12	12
FAM310.dag	16	18	16	16
FAM410.dag	18	19	18	18
FAM510.dag	13	15	13	13
FAM610.dag	16	19	16	16
FAM710.dag	15	17	16	16

Tabela 3.20: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP	CPMISF	ISH	PSO
r1.dag	14	16	15	15
r2.dag	13	16	14	14
ran1.dag	15	18	17	17
ran2.dag	13	15	15	15
ran3.dag	14	17	16	16
ran4.dag	17	19	18	18
ran5.dag	9	9	9	9
ran6.dag	9	10	9	9
ran7.dag	11	13	12	12
ran8.dag	11	12	12	12
ran9.dag	8	10	8	8
ran10.dag	18	21	19	19
ran11.dag	18	20	19	19
ran12.dag	17	20	19	19
ran13.dag	10	11	10	10
ran14.dag	16	17	16	16
ran15.dag	15	17	16	16
ran16.dag	11	12	11	11
ran17.dag	21	23	22	22
ran18.dag	15	17	16	16
ran19.dag	11	12	11	11
ran20.dag	10	11	10	10

Tabela 3.21: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP	CPMISF	ISH	PSO
bin4.dag	9	10	9	9
bin5.dag	10	11	10	10
bin6.dag	12	13	12	12
bin7.dag	13	15	13	13
bin8.dag	15	16	15	15
bin9.dag	17	18	17	17
di64.dag	34	38	34	34
di100.dag	52	56	52	52
di144.dag	52	61	52	52
di225.dag	79	89	79	79
di256.dag	70	82	70	70
di400.dag	88	104	88	88
irr41.dag	23	24	23	23
n13.dag	7	7	7	7
tree3.dag	9	9	9	9
tree4.dag	10	11	11	11
tree5.dag	12	13	13	13
tree6.dag	13	15	15	15
tree7.dag	15	17	17	17
tree8.dag	17	19	19	19
celbow.dag	30	30	30	30
cstanford.dag	29	29	29	29
fft2-b-m.dag	28	31	29	29
iterative2-b-m.dag	47	48	53	50

Tabela 3.22: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	IP	CPMISF	ISH	PSO
FAM31.dag	26	27	26	26
FAM32.dag	27	27	27	27
FAM33.dag	25	26	25	25
FAM34.dag	25	26	25	25
FAM35.dag	25	27	25	25
FAM37.dag	25	26	25	25
FAM38.dag	25	28	25	25
FAM39.dag	25	26	25	25
FAM42.dag	25	26	25	25
FAM43.dag	25	26	25	25
FAM48.dag	26	29	26	26
FAM51.dag	35	37	35	35
FAM52.dag	36	37	36	36
FAM53.dag	36	38	36	36
FAM54.dag	24	25	24	24
FAM55.dag	24	25	24	24
FAM56.dag	24	27	26	25
FAM57.dag	24	25	24	24
FAM58.dag	25	27	25	25
FAM59.dag	24	26	25	25
FAM61.dag	27	29	27	27
FAM62.dag	40	41	40	40
FAM63.dag	21	24	22	22
FAM64.dag	30	33	30	30
FAM65.dag	28	30	28	28
FAM66.dag	28	30	28	28
FAM67.dag	28	32	29	29
FAM68.dag	28	29	28	28
FAM69.dag	27	30	27	27
FAM71.dag	25	27	26	26
FAM72.dag	23	26	23	23
FAM73.dag	24	25	25	25
FAM74.dag	32	34	33	33
FAM75.dag	30	34	32	31
FAM76.dag	25	27	25	25
FAM77.dag	23	25	24	24
FAM78.dag	32	34	32	32
FAM79.dag	24	27	24	24
FAM310.dag	26	28	27	27
FAM510.dag	24	24	24	24
FAM610.dag	28	30	28	28
FAM710.dag	30	33	31	31

Tabela 3.23: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP	CPMISF	ISH	PSO
r1.dag	27	30	27	27
r2.dag	25	27	25	25
ran1.dag	28	31	28	28
ran2.dag	27	29	27	27
ran3.dag	26	28	26	26
ran4.dag	36	38	36	36
ran5.dag	16	17	16	16
ran6.dag	18	18	18	18
ran7.dag	22	23	22	22
ran8.dag	19	20	19	19
ran9.dag	16	17	16	16
ran10.dag	29	33	29	29
ran11.dag	27	31	29	29
ran12.dag	26	29	28	28
ran13.dag	18	20	18	18
ran14.dag	25	27	25	25
ran15.dag	31	33	31	31
ran16.dag	21	22	21	21
ran17.dag	29	33	30	30
ran18.dag	26	29	26	26
ran19.dag	20	21	20	20
ran20.dag	20	20	20	20

Tabela 3.24: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP	CPMISF	ISH	PSO
bin5.dag	17	18	17	17
bin6.dag	18	20	18	18
bin7.dag	20	22	20	20
bin8.dag	21	23	21	21
bin9.dag	23	24	23	23
di256.dag	130	134	130	130
di400.dag	202	206	202	202
tree4.dag	17	18	18	18
tree5.dag	18	20	19	19
tree6.dag	20	21	21	21
tree7.dag	21	23	22	22
tree8.dag	23	25	24	24
celbow.dag	35	40	35	35
cstanford.dag	32	38	33	33
fft2-b-m.dag	52	52	52	52
iterative2-b-m.dag	90	93	93	91

Tabela 3.25: Comparação entre os limites superiores para $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Estruturadas

A análise dos dados apresentados demonstra que a heurística $CP/MISF$ é satisfatória para o problema com processadores ilimitados, porém, ao restringirmos o número de processadores, ela perde sensivelmente a qualidade do *makespan*. Observamos um comportamento semelhante das heurísticas ISH e PSO: elas produzem bons limites para infinitos processadores e para $m = \lfloor \omega(\prec)/2 \rfloor$, porém para $m = \lfloor \omega(\prec)/4 \rfloor$ é percebida uma dificuldade maior em atingir o ótimo, o que é amenizado quando extrapolamos a restrição sobre os processadores, fazendo $m = \lfloor \omega(\prec)/8 \rfloor$. Este resultado é coerente com o esperado, que sugere tornar-se o problema mais fácil nos dois extremos: com infinitos processadores, temos uma grau de liberdade maior para alocar uma tarefa; com $m = \lfloor \omega(\prec)/8 \rfloor$ acontece o inverso, temos justamente o menor grau de liberdade dos experimentos, porém isto se mostra favorável, pois o tempo de execução de uma tarefa fica praticamente fixado, sem muitas possibilidades de variação. Portanto, a dificuldade dos algoritmos ISH e PSO reside nos casos intermediários.

Analisando o algoritmo PSO percebemos uma clara dependência da solução inicial passada. No caso, fornecemos para PSO a solução de ISH, portanto PSO obtém um limite igual ou melhor ao da ISH. Em poucos casos, a heurística PSO conseguiu superar ISH. Portanto, não há uma vantagem expressiva em usar a meta-heurística, visto que, ela possui um custo computacional muito superior à ISH e não obtém limites melhores em um número significativo de instâncias.

Concluímos que, em geral, a melhor heurística ou a mais equilibrada é ISH, pois possui um custo computacional baixo e obtém bons limites superiores para muitas instâncias, independente da quantidade de processadores disponível.

Capítulo 4

Formulações matemáticas

No Capítulo 2, definimos formalmente o problema de escalonamento de tarefas com restrições de precedência e comunicação em múltiplos processadores, tempos de execução e comunicação unitários. Os conceitos apresentados anteriormente servirão de base para as formulações matemáticas que serão apresentadas neste capítulo. Começaremos apresentando uma formulação existente na literatura, que chamaremos Formulação X, que enxerga um escalonamento viável como uma partição em cadeias de uma extensão da ordem parcial $\prec$. Em seguida, proporemos duas outras formulação, que chamaremos Y e Z. Ambas são baseadas em variáveis indexadas pelo tempo, sendo que a segunda pode ser vista como uma simplificação da primeira.

Neste capítulo não será considerada, como no capítulo anterior, a simplificação em que o grafo de tarefas possui uma única tarefa minimal e única tarefa maximal. Alternatvamente, vamos usar 0 e $n+1$ para marcar o "início"e o "final"do grafo de tarefas. Por exemplo, $G_{0,i}$ e $G_{i,n+1}$ denotam, respectivamente, o subgrafo de G induzido pela tarefa i e suas antecessoras e pela tarefa i e suas sucessoras. Como isso, $LB_{0,i}$ descreve um limite inferior para o tempo de início de execução da tarefa i, considerando suas antecessoras, enquanto $LB_{i,n+1}$ denota um limite inferior para a execução da tarefa i e todas as suas sucessoras. Especialmente, $LB_{0,n+1}$ é um limite inferior para o makespan de $(N, \prec, Q)$.

4.1 Formulação X

Utilizaremos como formulação básica, presente na literatura, a introduzida em [CCMP01] por Campêlo et al., que trata a versão restrita do problema. Para uma melhor intuição sobre esta formulação é útil, inicialmente, recapitular as soluções viáveis mostradas na Figura 2.3. Observe que uma solução define, para cada processador, uma cadeia de tarefas que são executadas nele. Note que o número de cadeias formadas é, logicamente, no máximo o número de processadores; portanto o número de cadeias para o exemplo em questão é 2. São elas:

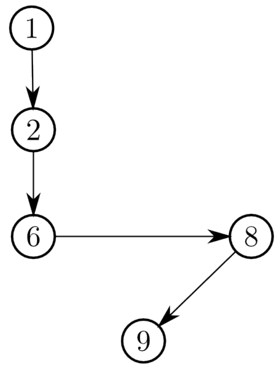


Figura 4.1: Cadeia referente ao processador 1

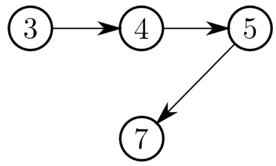


Figura 4.2: Cadeia referente ao processador 2

Nas cadeias formadas, podem estar presentes relações de precedência que não existiam antes no grafo de entrada original, mas que apareceram devido à limitação do número de processadores. No exemplo, são elas: $(6, 8), (3, 4)$ e $(4, 5)$. Observe que $t_8 \geq t_6 + 1$, $t_4 \geq t_3 + 1$ e $t_5 \geq t_4 + 1$, o que equivale ao cumprimento das relações de precedência também entre esses nós. Observado isso, podemos entender um escalonamento viável como sendo uma extensão da relação original que pode ser particionada em até m cadeias, onde m é o número de processadores. Desta forma, cada tarefa pertenceria à uma única cadeia, e cada cadeia poderia ser alocada em um processador diferente.

Matematicamente, podemos descrever as cadeias através de uma variável binária, $w_{i,j}$, com $i \prec j$ ou $i \parallel j$, tal que $w_{i,j} = 1$ se, e somente se, j suceder i em uma cadeia, ou em outras palavras se, e somente se, a tarefa j for a próxima tarefa executada depois de i em um certo processador. Nesse sentido, para o exemplo anterior: $w_{1,2} = w_{2,6} = w_{6,8} = w_{8,9} = 1$ e $w_{3,4} = w_{4,5} = w_{5,7} = 1$.

Como mencionado, para que um escalonamento seja viável, basta particionarmos em cadeias uma extensão da relação de precedência original e, a partir destas cadeias, determinarmos os tempos de execução de cada tarefa em seus respectivos processadores, respeitando as restrições impostas no Capítulo 2.

Vamos então definir o conjunto de variáveis que será utilizado para formular matematicamente esta versão do problema:

x_{n+1}: *makespan* de $(N, \prec, Q)$.

Para cada tarefa $i \in N$:

x_i: tempo de início da tarefa i.

$$w_{0,i} = \begin{cases} 1, & \text{se } i \text{ é a primeira tarefa executada em algum processador.} \\ 0, & \text{caso contrário.} \end{cases}$$

$$w_{i,n+1} = \begin{cases} 1, & \text{se } i \text{ é a última tarefa executada em algum processador.} \\ 0, & \text{caso contrário.} \end{cases}$$

Para todo $i, j \in N$, tal que $i \prec j$ ou $i \parallel j$:

$$w_{i,j} = \begin{cases} 1, & \text{se } j \text{ for a próxima tarefa a ser executada após } i \text{ em um mesmo processador.} \\ 0, & \text{caso contrário.} \end{cases}$$

Vamos dividir as restrições da formulação em dois grupos para faciliar a compreensão. Primeiramente, abordaremos as restrições responsáveis por formar as cadeias que serão executadas no conjunto de processadores. Depois, apresentaremos as restrições que definirão os tempos de execução de cada tarefa.

Relembrando, uma cadeia de tarefas pode ser caracterizada como um conjunto de tarefas tal que toda tarefa possui exatamente um antecessor e exatamente um sucessor, com exceção da primeira e da última tarefas que, respectivamente, não possuem antecessor e sucessor. Assim as seguintes restrições definem uma partição do grafo em até m cadeias:

$$w_{0,j} + \sum_{i:i \prec j} w_{i,j} + \sum_{i:i \parallel j} w_{i,j} = 1 \quad j \in N$$

$$w_{i,n+1} + \sum_{j:i \prec j} w_{i,j} + \sum_{j:i \parallel j} w_{i,j} = 1 \quad i \in N$$

$$\sum_{i \in N} w_{i,n+1} \leq m$$

A primeira restrição garante que toda tarefa j ou é a primeira a ser executada em algum processador ($w_{0,j} = 1$) ou possui uma tarefa executada antes dela no mesmo processador, seja esta terefa anterior concorrente a ou predecessora de j em $\prec$. Por outro lado, a segunda restrição afirma que toda tarefa i ou é a última tarefa a ser executada em algum processador ($w_{i,n+1} = 1$) ou possui uma tarefa executada depois, e esta pode ser sucessora de ou concorrente a i segundo $\prec$. Finalmente a última restrição garante que o número de cadeias formadas é menor ou igual ao número de processadores dado como entrada, ou seja, o número de cadeias é menor ou igual m.

Com as cadeias formadas, devemos adequar os tempos de início das tarefas para que eles respeitem as restrições de precedência e comunicação. Isto pode ser obtido com as seguintes relações entre as variáveis x e w:

$$x_j - x_i \geq 2 - w_{i,j} \quad i, j \in N \quad i \vdash j \tag{4.1}$$

$$x_j - x_i \geq 1 - \alpha_{i,j}(1 - w_{i,j}) \quad i, j \in N \quad i \parallel j \tag{4.2}$$

$$x_{n+1} \geq x_i + 1 \quad i \in N \tag{4.3}$$

Claramente, as restrições de precedência e comunicação são dependentes da relação que existe entre duas tarefas i e j. Portanto, temos uma restrição para cada caso.

Pela restrição (4.1), se j depende de i, então a diferença entre seus tempos de início deve ser maior ou igual a 1, sendo as duas executadas em seguida em um mesmo processador ($w_{i,j} = 1$), e maior ou igual a 2, caso contrário. Note que só é necessário estabelecer tais restrições para as relações da redução transitiva ($\vdash$), posto que as diferenças de tempo acima mencionadas também são obtidas, por transitividade, para as demais relações do fecho.

As restrições (4.2) estabelem as relações desejadas entre os tempos de execução das tarefas concorrentes. Se i e j são concorrentes devemos garantir que, sendo j executada em seguida a i em um mesmo processador, o tempo de início de j é pelo menos o tempo de início de i mais um. Caso esta situação não aconteça com i e j, a restrição correspondente em (4.2) deve se tornar redundante. Isto é feito através de um Big-M, logo $\alpha_{i,j}$ deve ser grande o suficiente para tornar esta restrição redundante. Em outras palavras, quando $w_{i,j} = 0$, a restrição $x_i - x_j \leq \alpha_{i,j} - 1$ deve ser satisfeita em uma solução ótima do problema. Ou seja, $\alpha_{i,j} - 1$ deve ser um limite superior para $x_i - x_j$ em uma solução ótima do problema.

A constante $\alpha_{i,j}$ pode ser estimada satisfatoriamente utilizando o limite inferior $LB_{i,j}$ apresentado. Note que $LB_{0,i} \leq x_i \leq UB_{makespan} - LB_{i,n+1}$, onde $UB_{makespan}$ é um limite superior para o problema. Sendo assim,

$$\alpha_{i,j} = UB_{makespan} - LB_{0,j} - LB_{i,n+1}$$

Finalmente, as restrições (4.3) definem o tempo total de processamento (*makespan*), que deve ser maior ou igual ao tempo de início de cada tarefa mais 1. Nas verdade, tais restrições só precisam ser impostas para as tarefas maximais. Assim, garantimos que o makespan é maior que o tempo total de execução de todas as tarefas.

Reunindo todas as restrições mostradas e minimizando o *makespan*, podemos escrever a formulação matemática completa:

Formulação X

$$
\begin{aligned}
\min \quad & x_{n+1} && && (4.4)\\
\text{s.a:} \quad & w_{0,j} + \sum_{i:i\prec j} w_{i,j} + \sum_{i:i\| j} w_{i,j} = 1 && j \in N && (4.5)\\
& w_{i,n+1} + \sum_{j:i\prec j} w_{i,j} + \sum_{j:i\| j} w_{i,j} = 1 && i \in N && (4.6)\\
& \sum_{i\in N} w_{i,n+1} \leq m && && (4.7)\\
& w_{i,j} \in \{0,1\} && i,j \in N, i \prec j \quad \text{ou} \quad i \parallel j && (4.8)\\
& w_{0,i}, w_{i,n+1} \in \{0,1\} && i \in N && (4.9)\\
& x_{n+1} \geq x_i + 1 && i \in N && (4.10)\\
& x_j - x_i \geq 2 - w_{i,j} && i,j \in N, \quad i \vdash j && (4.11)\\
& x_j - x_i \geq 1 - \alpha_{ij}(1 - w_{i,j}) && i,j \in N, \quad i \parallel j && (4.12)\\
& x_i \geq 0 && i \in N && (4.13)
\end{aligned}
$$

Esta formulação pode ser simplificada ao tratarmos o problema com infinitos processadores. Neste caso, não teremos tarefas incomparáveis sendo executadas em um mesmo processador. Podemos sempre considerar que se a tarefa j é a próxima a ser executada depois de i em um mesmo processador, então $i \vdash j$. Se $i \nvdash j$, independente de $i \prec j$ ou $i \parallel j$, j e as tarefas seguintes podem ser transferidas para um outro processador sem aumentar o *makespan*. Pontanto, a definição da variável w mudará:

Para todo $i, j \in N$, tal que $i \vdash j$:

$$
w_{i,j} = \begin{cases} 1, & \text{se } j \text{ for a próxima tarefa a ser executada após } i \text{ em um mesmo processador.} \\ 0, & \text{caso contrário.} \end{cases}
$$

E manteremos a definição de x. Esta, na realidade, é a formulação apresentada em [MK97]. Podemos escrever a formulação simplificada para infinitos processadores da seguinte forma:

Formulação X para infinitos processadores

$$
\begin{aligned}
\min \quad & x_{n+1} && && (4.14)\\
\text{s.a:} \quad & \sum_{i:i \vdash j} w_{i,j} \leq 1 && j \in N && (4.15)\\
& \sum_{j:i \vdash j} w_{i,j} \leq 1 && i \in N && (4.16)\\
& w_{i,j} \in \{0,1\} && i,j \in N, i \vdash j && (4.17)\\
& x_{n+1} \geq x_i + 1 && i \in N && (4.18)\\
& x_j - x_i \geq 2 - w_{i,j} && i,j \in N, \quad i \vdash j && (4.19)\\
& x_i \geq 0 && i \in N && (4.20)
\end{aligned}
$$

4.1.1 Incrementos à formulação original

A implementação da Formulação (4.4 – 4.13) demonstra que a resolução inteira para algumas instâncias não é praticável. O tempo de resolução não é razoável, mesmo para instâncias onde sabemos determinar o ótimo a priori, dada uma estrutura especial do grafo, e onde esperávamos uma resolução fácil. Para diminuir esse tempo de resolução, inicialmente, buscamos o fortalecimento da formulação através da inclusão dos limites inferiores descritos na Seção 3, aliado a novas estratégias de ramificação das variáveis da formulação.

Limite inferior $LB_{i,j}$

Podemos utilizar esse limite para gerar restrições válidas para o problema e acrescentá-las à formulação matemática. Estas restrições são do tipo:

$$x_j - x_i \geq LB_{i,j} \quad i,j \in N, i \prec j \tag{4.21}$$

$$x_{n+1} - x_i \geq LB_{i,n+1} \quad i \in N \tag{4.22}$$

$$x_i \geq LB_{0,i} \quad i \in N \cup \{n+1\} \tag{4.23}$$

Para efeito de comparação, a formulação sem as restrições foi resolvida e posteriormente acrescentamos as novas restrições baseadas no limite inferior $LB_{i,j}$ para o problema limitado.

Resultados computacionais

DAG	IP t	IP (X ini.) t	IP c/ rest. t	IP c/ rest. (X ini.) t
ran3	8	7	10	11
ran7	28	50	28	28
ran8	2	23	1	20
ran10	241	239	272	272
ran13	36	36	39	40
ran14	5	28	5	64
bin8	4	3	3	3
bin9	31	32	27	31
di100	4	3	8	9
di144	28	29	34	33
cstanford	1	1	1	2
celbow	2	2	2	1
ssc4.mydag	3	3	3	3
ssc5.mydag	10	16	16	16
ssc6.mydag	22	19	19	19
ssc7.mydag	599	688	695	709
fft2-b-m	7209	1813	1230	1235
divconq-m	7201	291	293	301
prolog-m	7200	-	953	953

Tabela 4.1: Resultados computacionais parciais com a adição das restrições $LB_{i,j}$

A tabela acima possui o nome da instância (DAG) e os tempos das versões analisadas (colunas identificadas com "t"). Essas instâncias representam um subconjunto significativo do conjunto de instâncias utilizado nos experimentos computacionais desta Dissertação e são descritas em detalhes no Apêndice A, onde também se encontra o ambiente computacional utilizado.

Observe que existem duas colunas referentes a execuções da formulação sem as restrições de limites e outras duas onde tais restrições são incluídas (estas identificadas por "c/ rest"). Há também duas colunas relativas a execuções nas quais fornecemos para o solver uma solução inicial heurística dada pela heurística $CP/MISF$, marcadas com "(X ini.)".

Analisando os resultados, percebemos uma melhoria em algumas instâncias quando incluímos as novas restrições. Em alguns casos em que não é possível resolver as instâncias (nos campos marcados com "-") ou a resolução leva um tempo considerável, apenas com as restrições específicas do problema, a inclusão das novas restrições possibilita a resolução dessas instâncias.

Critérios de ramificação

Além da adição de novas restrições, testamos a mudança no critério da ramificação no Branch-and-Bound. Na formulação original possuímos dois conjuntos de variáveis, as variáveis x e as variáveis w. Durante o processo do Branch-and-Bound, podemos priorizar um conjunto de variáveis e, dentro do conjunto de variáveis, podemos ordená-las por algum critério definido antecipadamente. Os seguintes critérios foram definidos a fim de melhorar o tempo de resolução da formulação:

- Critério 1: Priorizar variáveis X sobre variáveis W.
- Critério 2: Priorizar variáveis W sobre variáveis X (em ordem decrescente do valor de $UB_i - LB_i$).
- Critério 3: Priorizar variáveis X (em ordem decrescente do valor de $UB_i - LB_i$) sobre variáveis W.
- Critério 4: Priorizar variáveis X (em ordem crescente do valor de $UB_i - LB_i$) sobre variáveis W.
- Critério 5: Priorizar a variável x_{n+1} sobre as demais.

Os limites superior (UB_i) e inferior (LB_i) que usamos nos critérios são $LB_{0,i} \leq x_i \leq UB_{makespan} - LB_{i,n+1}$, onde o limite superior para o makespan ($UB_{makespan}$) é obtido heuristicamente, por alguma das heurísticas apresentadas no Capítulo 3.

A análise dos resultados que seguem permite observar uma melhora bastante discreta a partir da inclusão das restrições adicionais e utilização dos critérios definidos acima. Como não observamos outros caminhos para seguir melhorando a eficiência desta formulação optamos por trabalhar com novas formulações. Agora apresentaremos três novas formulações para o problema.

A Tabela 4.2 possui o nome da instância (DAG) e os tempos das versões analisadas (colunas identificadas com "t"). As três primeiras versões são, respectivamente, execuções da formulação onde inserimos ou não as restrições de limites e uma solução inicial heurística. Caso sejam inseridas as restrições de limites, marcaremos a coluna com "c/ rest. (X ini.)" e caso seja fornecida uma solução inicial marcaremos a coluna com "(X ini.)". Observe que existem, para cada critério de ramificação, uma coluna referente a execução da formulação sem as restrições de limites e sem uma solução inicial heurística e outra onde tais restrições são incluídas e ainda há o fornecimento para o solver de uma solução inicial heurística dada pela heurística $CP/MISF$, marcamos estas colunas com “c/ rest. (X ini.)”. As instâncias são as mesmas utilizadas nos experimentos acima.

Resultados computacionais

DAG	IPt	IPt (X ini.)	IPt c/ rest.	IPt C1	IPt C1 c/ rest. (X ini.)	IPt C2	IPt C2 c/ rest. (X ini.)	IPt C3	IPt C3 c/ rest. (X ini.)	IPt C4	IPt C4 c/ rest. (X ini.)	IPt C5	IPt C5 c/ rest. (X ini.)
ran3	8	7	10	8	10	8	11	8	11	8	11	8	10
ran7	28	50	28	29	28	28	28	29	28	28	29	27	28
ran8	2	23	1	1	20	2	20	1	20	2	20	2	20
ran10	241	239	272	252	272	240	269	253	285	251	285	235	269
ran13	36	36	39	38	39	35	39	36	40	37	40	35	38
ran14	5	28	5	6	63	6	63	6	62	5	65	5	62
bin8	4	3	3	3	3	3	2	4	3	4	3	3	3
bin9	31	32	27	31	32	30	31	31	32	31	32	31	32
di100	4	3	8	4	9	4	9	4	9	4	9	3	8
di144	28	29	34	29	33	29	33	29	34	29	34	28	33
cstanford	1	1	1	1	2	1	1	1	1	1	2	1	1
celbow	2	2	2	1	1	2	2	2	2	1	2	2	2
ssc4.mydag	3	3	3	3	3	3	3	3	2	4	3	3	2
ssc5.mydag	10	16	16	16	16	16	16	10	17	9	17	10	16
ssc6.mydag	22	19	19	19	19	20	20	21	19	21	155	22	19
ssc7.mydag	599	688	695	703	705	694	829	485	641	545	370	582	693
fft2-b-m	7209	1813	1230	1816	1211	1812	7207	7204	1521	3069	1569	7209	1953
divconq-m	7201	291	293	293	292	293	293	7201	294	7202	7202	7200	293
prolog-m	7200	-	953	-	841	7205	-	7218	56	71	4053	7200	7205

Tabela 4.2: Resultados computacionais parciais dos diversos critérios de ramificação

4.2 Formulação Y

Na formulação anterior, codificamos o tempo de início de uma tarefa como uma variável inteira. Nesta formulação, determinamos o tempo de início t de uma tarefa j através de uma variável binária $y_{j,t}$.

Dado o grafo de precedências $G = (N, \prec)$, sejam $B \subseteq N$ o conjunto de tarefas minimais e $E \subseteq N$ o conjunto maximais. Para $j \in N$, denotamos $\Gamma^+(j)$ o conjunto de sucessores imediatos de j e $\Gamma^-(j)$ o conjunto de predecessores imediatos de j, ou seja, $\Gamma^+(j) = \{i \in N : j \vdash i\}$ e $\Gamma^-(j) = \{i \in N : i \vdash j\}$. Seja T um limite superior para o *makespan*.

Para cada $t \in \{1, 2, \ldots, T\}$, utilizamos as seguintes variáveis:

$$y_{n+1,t} = \begin{cases} 1, & \text{se o makespan é } t. \\ 0, & \text{caso contrário.} \end{cases}$$

Para todo $i \in N$ e todo $t \in \{1, 2, \ldots, T\}$:

$$y_{i,t} = \begin{cases} 1, & \text{se } i \text{ é iniciada no tempo } t. \\ 0, & \text{caso contrário.} \end{cases}$$

Para todo $i, j \in N$, $(i, j) \in \vdash$:

$$w_{i,j} = \begin{cases} 1, & \text{se } j \text{ é a próxima tarefa executada depois de } i \text{ em um mesmo processador} \\ 0, & \text{caso contrário.} \end{cases}$$

Note que as variáveis $w_{i,j}$ tem significado similar ao da Formulação X, porém é definida apenas para os pares $(i, j) \in \vdash$, como no caso de infinitos processadores.

Usando estas variáveis, definimos a seguir uma nova formulação, a qual chamaremos Formulação Y, primeiro para para o problema ilimitado e depois acrescentamos uma restrição para o número de processadores.

Formulação Y

$$
\begin{aligned}
&min \quad \sum_{t=1}^{T} t * y_{n+1,t} && && (4.24)\\
&s.a \quad y_{n+1,t} \leq \sum_{t'=1}^{t-1} y_{j,t'} && \forall j \in E, \forall t \in \{1,...,T\} && (4.25)\\
&\sum_{t=1}^{T} y_{n+1,t} = 1 && && (4.26)\\
&\sum_{t=1}^{T} y_{j,t} = 1 && \forall j \in N && (4.27)\\
&\sum_{j \in \Gamma^{+}(i)} w_{i,j} \leq 1 && \forall i \in N - E && (4.28)\\
&\sum_{i \in \Gamma^{-}(j)} w_{i,j} \leq 1 && \forall j \in N - B && (4.29)\\
&y_{j,t} \leq \sum_{t'=1}^{t-1} y_{i,t'} && \forall (i,j) \in \vdash, \forall t \in \{1,...,T\} && (4.30)\\
&y_{i,t} \leq \sum_{t'=t+1}^{T} y_{j,t'} && \forall (i,j) \in \vdash, \forall t \in \{1,...,T\} && (4.31)\\
&y_{j,t} \leq \sum_{t'=1}^{t-2} y_{i,t'} + w_{i,j} && \forall (i,j) \in \vdash, \forall t \in \{1,...,T\} && (4.32)\\
&y_{i,t} \leq \sum_{t'=t+2}^{T} y_{j,t'} + w_{i,j} && \forall (i,j) \in \vdash, \forall t \in \{1,...,T\} && (4.33)\\
&y_{i,t-1} + y_{j,t} + y_{k,t-1} \leq 2 && \forall j \in N, \forall \{i,k\} \subseteq \Gamma^{-}(j), \forall t \in \{1,...,T\} && (4.34)\\
&y_{i,t} + y_{j,t+1} + y_{k,t+1} \leq 2 && \forall i \in N, \forall \{j,k\} \subseteq \Gamma^{+}(i), \forall t \in \{1,...,T\} && (4.35)\\
&y \in \{0,1\}^{(n+1)T}, w \in \{0,1\}^{|\vdash|} && && (4.36)
\end{aligned}
$$

A restrição (4.25) garante que o makespan só pode ocorrer em um tempo t se todas as tarefas maximais começaram a executar até $t-1$, enquanto a restrição (4.26) estabelece que esse tempo t é único, assegurando a boa definição da variável $y_{n+1,t}$. Como uma tarefa é executada somente uma vez ao longo do escalonamento temos a restrição (4.27). As restrições (4.28) garantem que no máximo uma tarefa sucessora j de i poderá ser executada no instante seguinte no mesmo processador de i. O mesmo vale para tarefas predecessoras, assegurado pelas restrições (4.29).

Relações de precedência e comunicação são garantidas pelas restrições (4.30) a (4.33). Especificamente as restrições (4.30) e (4.31) são responsáveis por garantir a regra de precedência, de forma que uma tarefa só pode executar quando todas as suas predecessoras tiverem executado. As duas últimas desigualdades levam em consideração o tempo de comunicação, além da relação de precedência. Por (4.32) e (4.33), se duas tarefas que possuem uma relação de precedência não forem alocadas em um mesmo processador de forma consecutiva, devem manter uma distância de duas unidades de tempo; caso contrário este será de apenas uma unidade.

As restrições (4.34) e (4.35) são restrições válidas para o problema, mas não necessárias para a corretude da Formulação Y. A restrição (4.34) não permite a alocação simultânea de duas tarefas i e k predecessoras de j no tempo imediatamente seguinte ao tempo de alocação de j. A restrinção (4.35) é similar para tarefas sucessoras.

Observe que não fizemos nenhuma restrição em relação ao número de processadores. Para utilizarmos essa formulação para o problema com um número limitado de processadores, basta adicionarmos as restrições:

$$\sum_{j \in N} y_{j,t} \leq m \quad \forall t \in \{1, ..., T\} \tag{4.37}$$

Por simplicidade, nas demostrações de corretude que se seguem, suporemos que tais restrições estão presentes na formulação para ambos os problemas, sendo $m = n$ no caso ilimitado.

Proposição 5. *A Formulação Y é correta, mesmo sem as restrições (4.31), (4.33), (4.34) e (4.35).*

Demonstração. Sejam $S = (T, P)$ um escalonamento viável e S_q, $\forall\ q \in Q$, o conjunto de tarefas alocado em S no processador q, ordenados de forma crescente em relação ao tempo de início de execução. Denote por $S_q[i]$ a i-ésima tarefa nesse conjunto ordenado, para $i = 1, \ldots, |S_q|$. Construa uma solução (y, w) para a Formulação Y, conforme as definições das variáveis $y_{n+1,t}, y_{i,t}$ e $w_{i,j}$, ou seja, considerando $S = (T, P)$ defina:

$$y_{n+1,t} = \begin{cases} 1, & \text{se o makespan é } t. \\ 0, & \text{caso contrário.} \end{cases} \qquad t \in \{1, 2, \ldots, T\}$$

$$y_{i,t} = \begin{cases} 1, & \text{se } i \text{ é iniciada no tempo } t. \\ 0, & \text{caso contrário.} \end{cases} \qquad i \in N,\ t \in \{1, 2, \ldots, T\}$$

$$w_{i,j} = \begin{cases} 1, & \text{se } j \text{ é a próxima tarefa executada após } i \text{ em um mesmo processador} \\ 0, & \text{caso contrário.} \end{cases} \qquad i \vdash j$$

Em outros termos, as únicas variáveis não-nulas em (y, w) são:

$$w_{S_q[i], S_q[i+1]} = 1, \quad \forall q \in Q, \forall i \in \{1, \ldots, |S_q| - 1\}$$

$$y_{j,t_j} = 1, \quad \forall\, j \in N$$

$$y_{n+1,t^*} = 1, \quad t^* = \max\{t_i : i \in N\} + 1$$

Devemos mostrar que (y, w) verifica as restrições (4.25)-(4.35). Claramente, pela atribuição acima, y satisfaz as restrições (4.25), (4.26) e (4.27). As restrições (4.28)-(4.29) são satisfeitas, pois, para uma tarefa i, apenas um sucessor (predecessor) pode executar após (resp. antes de) i em um mesmo processador no escalonamento S. Também por ser S um escalonamento válido, as restrições (4.30) e (4.31) são satisfeitas: uma tarefa só pode ser executada depois de seus antecessores (restrições (4.30)) e antes de seus sucessores (restrições (4.31)). Para as restrições (4.32) e (4.33), lembre que, caso uma tarefa i e sua sucessora imediata j não estejam executando seguidamente em um mesmo processador ($w_{i,j} = 0$), seus tempos de início serão distanciados em pelo menos duas unidades, em acordo com a definição do problema. Finalmente, as restrições (4.34) (resp. (4.35)) são verificadas porque, se uma tarefa está alocada em S em um certo tempo t, não poderá haver duas antecessoras (resp. sucessoras) suas agendadas para uma unidade de tempo imediatamente anterior (resp. posterior).

Reciprocamente, tomemos w e y binários satisfazendo as restrições (4.25)-(4.30) e (4.32). A partir de (y, w), iremos definir um escalonamento viável (T, P), com makespan t^* tal que $y_{n+1,t^*} = 1$. O vetor $T = [t_i]$ é dado diretamente pela variável y, sendo $t_i = x$ se, e somente se, $y_{i,x} = 1$. A alocação aos processadores será dada em função de w. Primeiro, perceba que podemos admitir, para quaisquer $i \vdash j$, que $w_{i,j} = 0$ quando j não é agendado exatamente uma unidade de tempo depois de i, segundo y. Do contrário, podemos anular $w_{i,j}$ obtendo outra solução viável de mesmo valor. Assim, por (4.28)-(4.29), as variáveis w definem uma partição de G em cadeias, onde as tarefas de cada cadeia são agendadas consecutivamente no tempo. Ordene tais cadeias por ordem não-decrescente de tempo de início de suas tarefas minimais. Nesta ordem, aloque cada cadeia a um processador disponível para executar todas as suas tarefas nos tempos definidos por y. Para ver que existe um tal processador, considere uma cadeia C a ser alocada e sua tarefa minimal i, sendo $y_{i,t} = 1$. Por (4.37), existe um processador onde o tempo t está livre. Nesse mesmo processador, também devem estar livres todos os tempos superiores a t, pois as cadeias já escalonadas começam antes ou no tempo t e todas as suas tarefas são agendadas consecutivamente. Isto mostra a existência do processador desejado. Resta então mostrar que a atribuição dos tempos satisfaz as restrições de precedência e comunicação. Caso $i \vdash j$, as restrições (4.30), juntamente com (4.27), garantem que o tempo de início de j é pelo menos uma unidade de tempo superior ao de início de i. Além disso, se $p_i \neq p_j$, o que implica $w_{i,j} = 0$, essa diferença é de pelo menos 2, devido a (4.32). Por transitividade, a diferença também é de pelo menos 2 quando $i \prec j$. Finalmente, (4.25) e (4.26) implicam que o makespam de (T, P) é t^*. □

Proposição 6. *A Formulação Y é correta, mesmo sem as restrições (4.28), (4.29), (4.31), (4.32)e(4.33).*

Demonstração. Seja $y \in \{0,1\}^{(n+1)T}$ satisfazendo (4.25)-(4.27), (4.30), (4.34) e (4.35). À luz da Proposição 5, devemos mostrar que existe $w \in \{0,1\}^{|\vdash|}$ tal que (y, w) satisfaz (4.28), (4.29) e (4.32). Considere w definido como

$$w_{i,j} = \max\left\{0, y_{j,t^\star} - \sum_{t'=1}^{t^\star-2} y_{i,t'}\right\} \quad \forall (i,j) \in \vdash$$

onde

$$t^\star \in \arg\max_{t\in\{1,\ldots,T\}} \left\{y_{j,t} - \sum_{t'=1}^{t-2} y_{i,t'}\right\}.$$

Por esta definição, temos claramente que $w \in \{0,1\}^{|\vdash|}$ e (y, w) satisfaz (4.32). Além disso,

Afirmação 1. *Se $w_{i,j} = 1$ então $y_{j,t^\star} = y_{i,t^\star-1} = 1$.*

Demonstração. Suponha que $w_{i,j} = 0$. Pela definição acima e sendo $y \in \{0,1\}^{(n+1)T}$, temos que $y_{j,t^\star} = 1$ e $\sum_{t'=1}^{t^\star-2} y_{i,t'} = 0$. Por (4.30) e (4.27) obtemos, respectivamente, que $\sum_{t'=1}^{t^\star-1} y_{i,t'} \geq 1$ e $\sum_{t'=t^\star-1}^{T} y_{i,t'} = 1$. Logo, $y_{i,t^\star-1} = 1$. □

Suponha, por absurdo, que (4.28) não é satisfeita. Então existem $i \in N$ e $j, k \in \Gamma^+(i)$ tais que $w_{i,j} = w_{i,k} =$. Pela afirmação acima teríamos $y_{j,t^\star} = y_{k,t^\star} = y_{i,t^\star-1} = 1$, contradizendo (4.35) para $t = t^* - 1$. Similarmente, caso (4.29) não fosse satisfeita, teríamos, pela afirmação, que $y_{j,t^\star} = y_{k,t^\star-1} = y_{i,t^\star-1} = 1$ para algum $j \in N$ e $i, k \in \Gamma^-(j)$, contradizendo agora (4.35) para t. Sendo assim, concluímos que w verfica (4.28) e (4.29). □

As proposições 5 e 6 mostram que versões simplificadas da Formulação Y também modelam corretamente o problema. Em particular, a segunda proposição demonstra que as variáveis w poderiam ser eliminadas. Todavia essas restrições redundantes podem fortalecer a relaxação linear, motivo pelo qual apresentamos a Formullação Y com todas elas. Nas próximas seções, introduziremos modificações nessa formulação que tornam tais restrições redundantes mesmo na relaxação linear.

4.3 Formulação Y reforçada

Observando a formulação anterior, é possível perceber que, quando fazemos alguma imposição sobre a alocação de uma tarefa i em um certo tempo t, esta imposição não é feita tão somente ao tempo t, mas a um conjunto de instantes de tempo. Este conjunto pode ser formado pelos tempos anteriores ou posteriores a t.

Por exemplo, na restrição (4.30), estabelecemos que a tarefa j não pode ser executada no tempo t caso a tarefa i não tenha sido executada até o tempo $t-1$. Veja que esta condição não atinge apenas o tempo t, mas sim todos os tempos anteriores a t também. Levando em consideração esse aspecto, a restrição (4.30) pode ser fortalecida como

$$\sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t'=1}^{t-1} y_{i,t'}.$$

Na verdade, este mesmo raciocínio se aplica às restrições (4.25), (4.30)-(4.35) da Formulação Y. Teremos, então, desigualdades mais fortes com o lado esquerdo aumentado em cada uma. Além disso, mostraremos nesta seção que a variável w e as restrições das quais ela faz parte podem ser eliminadas, mesmo quando a integralidade é descartada.

Primeiro, utilizando o mesmo conjunto de variáveis, reescrevemos a Formulação Y, reforçando suas desigualdades como indicado acima. Isto nos leva a Formulação Y_{ref}. A seguir, aplicamos outras transformações na Formulação Y_{ref} para obter uma formulação mais forte e compacta (em termos de variáveis), que denominaremos Formulação Y_{exp}.

Estas duas novas formas de modelar o problema são importantes, pois além de gerarem formulações mais fortes que a Y, servem como base para a concepção de outras duas formulações. Através de uma mudança de variáveis, obteremos a Formulação Z e a Formulação Z_{exp}. Posteriormente, faremos um estudo comparativo teórico e através de experimentos computacionais com o objetivo de evidenciar a obtenção de formulações mais eficientes que aquela usada como base, a Formulação X.

A Formulação Y_{ref} é apresentada a seguir. As restrições (4.41), (4.46)-(4.47) são obtidas diretamente com o fortalecimento descrito no início desta seção. Para obter (4.48)-(4.49), primeiro fortalecemos (4.34)-(4.35) de forma similar, obtendo:

$$\sum_{t'=1}^{t} y_{j,t'} \leq 2 - \sum_{t'=t-1}^{T} (y_{i,t'} + y_{k,t'}) \qquad \forall j \in N, \forall\{i,k\} \subseteq \Gamma^{-}(j), \forall t \in \{1,...,T\}$$

$$\sum_{t'=t}^{T} y_{i,t'} \leq 2 - \sum_{t'=1}^{t+1} (y_{j,t'} + y_{k,t'}) \qquad \forall i \in N, \forall\{j,k\} \subseteq \Gamma^{+}(i), \forall t \in \{1,...,T\}$$

Então, usamos (4.43) para chegar a:

$$\sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t'=1}^{t-2} (y_{i,t'} + y_{k,t'}) \qquad \forall j \in N, \forall \{i,k\} \subseteq \Gamma^-(j), \forall t \in \{1,...,T\} \tag{4.38}$$

$$\sum_{t'=t}^{T} y_{i,t'} \leq \sum_{t'=t+2}^{T} (y_{j,t'} + y_{k,t'}) \qquad \forall i \in N, \forall \{j,k\} \subseteq \Gamma^+(i), \forall t \in \{1,...,T\} \tag{4.39}$$

Finalmente, conseguimos fortalecer ainda mais tais restrições, como segue:

$$\sum_{t'=1}^{t-1} y_{j,t'} + \sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t'=1}^{t-2} (y_{i,t'} + y_{k,t'}) \qquad \forall j \in N, \forall \{i,k\} \subseteq \Gamma^-(j), \forall t \in \{1,...,T\}$$

$$\sum_{t'=t+1}^{T} y_{i,t'} + \sum_{t'=t}^{T} y_{i,t'} \leq \sum_{t'=t+2}^{T} (y_{j,t'} + y_{k,t'}) \qquad \forall i \in N, \forall \{j,k\} \subseteq \Gamma^+(i), \forall t \in \{1,...,T\}$$

Para verificar que estas desigualdades continuam válidas, é suficiente analisar o caso onde o lado direito das expressões é menor ou igual a um. Na primeira restrição, isto significa que i ou k começam a partir de $t-1$, de modo que a tarefa sucessora comum j não pode iniciar antes de t, implicando que o somatório adicionado ao lado esquerdo da restrição restrição é nulo. A análise para a segunda restrição é similar.

Reunindo as transformações, chegamos a:

Formulação Y_{ref}

$$\min \sum_{t=1}^{T} t * y_{n+1,t} \tag{4.40}$$

$$\text{s.a:} \sum_{t'=1}^{t} y_{n+1,t'} \leq \sum_{t'=1}^{t-1} y_{j,t'} \qquad \forall j \in E, \forall t \in \{1, ..., T\} \tag{4.41}$$

$$\sum_{t=1}^{T} y_{n+1,t} = 1 \tag{4.42}$$

$$\sum_{t=1}^{T} y_{j,t} = 1 \qquad \forall j \in N \tag{4.43}$$

$$\sum_{j \in \Gamma^+(i)} w_{i,j} \leq 1 \qquad \forall i \in N - E \tag{4.44}$$

$$\sum_{i \in \Gamma^-(j)} w_{i,j} \leq 1 \qquad \forall j \in N - B \tag{4.45}$$

$$\sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t'=1}^{t-1} y_{i,t'} \qquad \forall (i,j) \vdash, \forall t \in \{1, ..., T\} \tag{4.46}$$

$$\sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t'=1}^{t-2} y_{i,t'} + w_{i,j} \qquad \forall (i,j) \vdash, \forall t \in \{1, ..., T\} \tag{4.47}$$

$$\sum_{t'=1}^{t-1} 2y_{j,t'} + y_{j,t} \leq \sum_{t'=1}^{t-2} (y_{i,t'} + y_{k,t'}) \qquad \forall j \in N - B, \forall \{i,k\} \subseteq \Gamma^-(j), \forall t \in \{1, ..., T\} \tag{4.48}$$

$$y_{i,t} + \sum_{t'=t+1}^{T} 2y_{i,t'} \leq \sum_{t'=t+2}^{T} (y_{j,t'} + y_{k,t'}) \qquad \forall i \in N - E, \forall \{j,k\} \subseteq \Gamma^+(i), \forall t \in \{1, ..., T\} \tag{4.49}$$

$$y \in \{0,1\}^{(n+1)T}, w \in \{0,1\}^{|\vdash|} \tag{4.50}$$

Observe que não incluímos em Y_{ref} as restrições

$$\sum_{t'=t}^{T} y_{i,t'} \leq \sum_{t'=t+1}^{T} y_{j,t'} \qquad \forall (i,j) \vdash, \forall t \in \{1, ..., T\} \tag{4.51}$$

$$\sum_{t'=t}^{T} y_{i,t'} \leq \sum_{t'=t+2}^{T} y_{j,t'} + w_{i,j} \qquad \forall (i,j) \vdash, \forall t \in \{1, ..., T\} \tag{4.52}$$

que correspondem aos fortalecimentos de (4.31) e (4.33). Da Proposição 5, deduzimos que elas são redundantes na Formulação Y_{ref}. Na verdade, como mostramos agora, tais restrições são equivalentes, respectivamente, a (4.46) e (4.47), mesmo na relaxação linear e, consequentemente, podem ser descartadas.

Proposição 7. *Sejam $y \in \mathbb{R}^{(n+1)T}$ e $w \in \mathbb{R}^{|\vdash|}$ satisfazendo (4.43). Então (y, w) satisfaz (4.51) e (4.52) se, e somente se, (y, w) satisfaz (4.46) e (4.47).*

Demonstração. De (4.43), temos a seguinte equivalência:

$$\sum_{t'=t+1}^{T} y_{j,t'} = 1 - \sum_{t'=1}^{t} y_{j,t'} \tag{4.53}$$

Sejam $(i, j) \in \vdash$ e $t \in \{1, \ldots, T\}$. Para obtermos a desigualdade (4.51), basta aplicarmos a equivalência acima aos dois lados da desigualdade (4.46):

$$\begin{aligned}
&\sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t'=1}^{t-1} y_{i,t'} \\
\Leftrightarrow &1 - \sum_{t'=t+1}^{T} y_{j,t'} \leq 1 - \sum_{t'=t}^{T} y_{i,t'} \\
\Leftrightarrow &\sum_{t'=t}^{T} y_{i,t'} \leq \sum_{t'=t+1}^{T} y_{j,t'}
\end{aligned}$$

Para a desigualdade (4.52), basta aplicarmos novamente a equivalência aos dois somatórios da desigualdade (4.47) e fazermos $t := t - 1$:

$$\begin{aligned}
&\sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t'=1}^{t-2} y_{i,t'} + w_{i,j} \\
\Leftrightarrow &1 - \sum_{t'=t+1}^{T} y_{j,t'} \leq 1 - \sum_{t'=t-1}^{T} y_{i,t'} + w_{i,j} \\
\Leftrightarrow &\sum_{t'=t}^{T} y_{i,t'} \leq \sum_{t'=t+2}^{T} y_{j,t'} + w_{i,j}
\end{aligned}$$

□

Da Proposição 6, sabemos que as restrições (4.47) são redundantes na definição de Y_{ref}, porém poderiam fortalecer a relaxação linear de tal formulação. Apresentaremos, a seguir, um conjunto desigualdades válidas que podem substituir (4.47), inclusive na relaxação linear, e são definidas apenas sobre as variáveis y. Na verdade, tais desigualdades são uma generalização de (4.48)-(4.49).

Lema 1. *As restrições*

$$|S| \sum_{t'=1}^{t-1} y_{j,t'} + (|S| - 1) y_{j,t} \leq \sum_{i \in S} \sum_{t'=1}^{t-2} y_{i,t'} \quad j \in N, S \subseteq \Gamma^{-}(j), \forall t \in \{1, ..., T\} \tag{4.54}$$

$$|S|\sum_{t'=t+1}^{T} y_{i,t'} + (|S|-1)y_{i,t} \leq \sum_{j\in S}\sum_{t'=t+2}^{T} y_{j,t'} \quad i \in N, S \subseteq \Gamma^{+}(i), \forall t \in \{1,...,T\} \tag{4.55}$$

são válidas para a Formulação Y_{ref}.

Demonstração. Seja $y \in \{0,1\}^{(n+1)T}$ satisfazendo (4.43). Para $j \in N$ e $t \in \{1,\ldots,T\}$, defina $K_j(t) = \sum_{t'=1}^{t} y_{j,t'}$. Dados $j \in N$, $S \subseteq \Gamma^{-}(j)$ e $t \in \{1,\ldots,T\}$, a restrição em (4.54) é então reescrita como:

$$|S|K_j(t) - y_{j,t} \leq \sum_{i\in S} K_i(t-2)$$

Se $\sum_{i\in S} K_i(t-2) \geq |S|$, claramente a desigualdade é válida, pois $K_j(t) \leq 1$ e $y_{j,t} \geq 0$. Se $\sum_{i\in S} K_i(t-2) = |S|-1$, então uma tarefa em $S \subseteq \Gamma^{-}(j)$ é executada a partir de $t-1$ e, assim, j é executada a partir de t. Logo, $K_j(t) = y_{j,t} \leq 1$ levando a satisfação da desigualdade. Finalmente, se $\sum_{i\in S} K_i(t-2) < |S|-1$, então pelo menos duas tarefas em $S \subseteq \Gamma^{-}(j)$ são executadas a partir do tempo $(t-1)$. Pelas restrições de precedência e comunicação, $K_j(t) = y_{j,t} = 0$ e assim a desigualdade é trivialmente satisfeita.

A validade de (4.55) pode ser verificada de modo similar. □

Note que (4.48) e (4.49) são casos particulares de (4.54) e (4.55) para $|S| = 2$. Observe também que (4.54) e (4.55) podem ser reescritas respectivamente como:

$$\sum_{i\in S}\left(\sum_{t'=1}^{t} y_{j,t'} - \sum_{t'=1}^{t-2} y_{i,t'}\right) \leq y_{j,t} \quad \forall j, \forall S \in \Gamma^{-}(j), \forall t \in \{1,\ldots,t\} \tag{4.56}$$

$$\sum_{j\in S}\left(\sum_{t'=1}^{t} y_{j,t'} - \sum_{t'=1}^{t-2} y_{i,t'}\right) \leq y_{i,t-1} \quad \forall i, \forall S \in \Gamma^{+}(i), \forall t \in \{1,\ldots,t\} \tag{4.57}$$

onde para obter (4.57) usamos (4.53) e fizemos $t := t+1$.

Analisamos a projeção sobre y do conjunto viável relaxado da Formulação Y_{ref}, objetivando obter uma formulação mais compacta em termos de variáveis.

Proposição 8. *Seja $Y = \{(y,w) \geq 0 : (y,w)$ satisfaz $(4.41)-(4.49)\}$. Então $Proj_y Y \supseteq \{y \geq 0 : y$ satisfaz (4.41)-(4.43), (4.46), (4.54)-(4.55)$\}$.*

Demonstração. Seja $y \in \mathbb{R}_{+}^{(n+1)T}$ satisfazendo (4.41)-(4.43), (4.46), (4.54)-(4.55). É suficiente mostrar que existe $w \geq 0$ tal que, (y,w) satisfaz (4.44), (4.45) e (4.47). Como no Lema 1, para

$j \in N$ e $t \in \{1, \dots, T\}$, defina $K_j(t) = \sum_{t'=1}^{t} y_{j,t'}$. Sejam $i \vdash j$ e $t^*_{i,j} \in \arg\max\{K_j(t) - K_i(t-2) : t = 1, \dots, T\}$. O vetor $w \geq 0$ é tal que:

$$w_{i,j} = \max\{0, K_j(t^*_{i,j}) - K_i(t^*_{i,j} - 2)\} \quad (4.58)$$

Assim, $w_{i,j} \geq K_j(t) - K_i(t-2)\ \forall t \in \{1, \dots, T\}$. Logo, as restrições (4.47) são satisfeitas. Além disso, tome $S_j = \{i : i \vdash j, K_j(t^*_{i,j}) - K_i(t^*_{i,j} - 2) > 0\}$ e $S_i = \{j : i \vdash j, K_j(t^*_{i,j}) - K_i(t^*_{i,j} - 2) > 0\}$. Então, por (4.56) para $S = S_j$ e (4.57) para $S = S_i$

$$\sum_{i \in \Gamma^-(j)} w_{i,j} = \sum_{ji \in S_j} (K_j(t^*_{i,j}) - K_i(t^*_{i,j} - 2)) \leq y_{j,t} \leq 1 \quad (4.59)$$

$$\sum_{j \in \Gamma^+(i)} w_{i,j} = \sum_{j \in S_i} (K_j(t^*_{i,j}) - K_i(t^*_{i,j} - 2)) \leq y_{i,t-1} \leq 1 \quad (4.60)$$

□

A Proposição acima garante que as variáveis do tipo w não são necessárias para a Formulação Y_{ref}, com a adição das restrições (4.54) e (4.55), mesmo sem considerar a integralidade. Assim, obtemos a seguinte simplificação da formulação Y_{ref}:

Formulação Y_{exp}

$$min \quad \sum_{t=1}^{T} t * y_{n+1,t}$$

$$s.a \sum_{t'=1}^{t} y_{n+1,t'} \leq \sum_{t'=1}^{t-1} y_{j,t'} \qquad \forall j \in E, \forall t \in \{1, ..., T\} \quad (4.61)$$

$$\sum_{t=1}^{T} y_{n+1,t} = 1 \quad (4.62)$$

$$\sum_{t=1}^{T} y_{j,t} = 1 \qquad \forall j \in N \quad (4.63)$$

$$\sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t'=1}^{t-1} y_{i,t'} \qquad \forall (i,j) \in \vdash, \forall t \in \{1, ..., T\} \quad (4.64)$$

$$|S| \sum_{t'=1}^{t-1} y_{j,t'} + (|S| - 1) y_{j,t} \leq \sum_{i \in S} \sum_{t'=1}^{t-2} y_{i,t'} \qquad \forall j, S \subseteq \Gamma^-(j), \forall t \in \{1, ..., T\} \quad (4.65)$$

$$|S| \sum_{t'=t+1}^{T} y_{i,t'} + (|S| - 1) y_{i,t} \leq \sum_{j \in S} \sum_{t'=t+2}^{T} y_{j,t'} \qquad \forall j, S \subseteq \Gamma^+(i), \forall t \in \{1, ..., T\} \quad (4.66)$$

$$y \in \{0, 1\}^{(n+1)T} \quad (4.67)$$

Algoritmo de separação para (4.65) e (4.66)

Como mostrado, podemos excluir as variáveis do tipo w, porém, para isso, incluímos uma classe de inequações que pode ter um número exponencial de restrições. Isto posto, mostraremos um algoritmo de separação para que possamos desenvolver uma estratégia de resolução para esta formulação sem as variáveis w e com as classes de restrições (4.65) e (4.66).

Utilizaremos a seguinte estratégia de resolução para o problema inteiro associado à formulação com possíveis restrições em número exponencial:

1. Iniciaremos com os conjuntos Π e Π', respectivamente das restrições (4.65) e (4.66), vazios.

2. Resolveremos o problema inteiro.

3. Descobriremos polinomialmente as restrições mais violadas, sendo uma para cada classe, e as incluiremos no modelo.

4. Voltamos ao passo (2), até que não se tenham mais restrições violadas.

Claramente o ponto mais importante dessa estratégia é a descoberta das restrições violadas. Para esse propósito, apresentamos a proposição, que utiliza a seguinte definição. Dados $i \vdash j$, $t \in \{1, \ldots, T\}$ e $y \in \mathbb{R}^{(n+1)T}$, definimos

$$\delta_{i,j}(t) = \sum_{t'=1}^{t} y_{j,t} - \sum_{t'=1}^{t-2} y_{i,t}$$

Proposição 9. *Um vetor $y \in \mathbb{R}^{(n+1)T}$ viola alguma restrição de (4.65) para $j \in N$ e $t \in \{1, \ldots, T\}$ se, e somente se, $\sum_{i \in S} \delta_{i,j}(t) > y_{j,t}$, onde $S = \{i \in \Gamma^-(j) : \delta_{i,j}(t) > 0\}$.*

Demonstração. Sejam $j \in N$ e $t \in \{1, \ldots, T\}$. Por (4.56), uma restrição de (4.65) é violada se, e somente se,

$$\max_{S \subseteq \Gamma^-(j)} \sum_{i \in S} \delta_{i,j}(t) > y_{j,t}.$$

Claramente, o máximo da expressão acima ocorre em $S = \{i \in \Gamma^-(j) : \delta_{i,j}(t) > 0\}$. □

Desse fato, obtemos que ao somarmos os valores positivos de $(\sum_{t'=1}^{t} y_{j,t} - \sum_{t'=1}^{t-2} y_{i,t})$ estamos procurando uma restrição violada. Efetivamente, quando esta soma supera $y_{j,t}$, obtemos uma restrição violada.

De forma similar podemos identificar uma restrição violada de (4.66).

Proposição 10. *Um vetor $y \in \mathbb{R}^{(n+1)T}$ viola alguma restrição de (4.66) para $i \in N$ e $t \in \{1, \ldots, T\}$ se, e somente se, $\sum_{j \in S} \delta_{i,j}(t) > y_{i,t-1}$, onde $S = \{j \in \Gamma^+(i) : \delta_{i,j}(t) > 0\}$.*

4.4 Formulação Z

Na formulação anterior, com o fortalecimento das restrições, causamos um incremento bastante significativo no número de variáveis em cada restrição. Principalmente quando não possuímos uma boa cota superior para limitar T. A partir disso, podemos verificar a existência de um padrão na forma dos somatórios envolvidos nas restrições. A Formulação Z, definida a seguir, tem como objetivo eliminar esses somatórios, tornando a formulação mais diminuta em número de variáveis envolvidas em cada restrição, porém mantendo os fortalecimentos feitos anteriormente.

Novamente, dado $G = (N, \prec)$ um grafo de tarefas, sejam $B \subseteq N$ o conjunto de tarefas minimais e $E \subseteq N$ o conjunto maximais. Sejam as seguintes variáveis:

$$z_{n+1,t} = \begin{cases} 1, & \text{se o } \textit{makespan} \text{ ocorre até o tempo } t \text{, inclusive.} \\ 0, & \text{caso contrário.} \end{cases}$$

Para todo $i \in N$:

$$z_{i,t} = \begin{cases} 1, & \text{se } i \text{ é executada até o tempo } t \text{, inclusive.} \\ 0, & \text{caso contrário.} \end{cases}$$

Não é difícil perceber que existem equivalências entre a definição das variáveis z e alguns somatórios contidos na formulação anterior. Observe:

$$\sum_{t'=1}^{t} y_{j,t'} \equiv z_{j,t} \quad \forall j \in N \cup \{n+1\}, \forall t \in \{1, ..., T\} \tag{4.68}$$

$$\sum_{t'=t+1}^{T} y_{j,t'} = 1 - \sum_{t'=1}^{t} y_{j,t'} \equiv 1 - z_{j,t} \quad \forall j \in N \cup \{n+1\}, \forall t \in \{1, ..., T\}$$

Desta forma, as restrições da Formulação Y_{ref} podem ser convertidas de forma direta para a variável z, simplesmente aplicando as relações acima. Segue então a formulação:

Formulação Z

$$\begin{aligned}
min \quad & z_{n+1,1} + \sum_{t=2}^{T} t * (z_{n+1,t} - z_{n+1,t-1}) && && (4.69)\\
s.a \quad & z_{j,T} = 1 && \forall j \in N \cup \{n+1\} && (4.70)\\
& z_{n+1,t} \leq z_{i,t-1} && \forall i \in N, \forall t \in \{1, ..., T\} && (4.71)\\
& z_{j,t} \leq z_{i,t-1} && \forall (i,j) \in \vdash, \forall t \in \{1, ..., T\} && (4.72)\\
& z_{j,t} + z_{j,t-1} \leq z_{i,t-2} + z_{k,t-2} && \forall (i,j),(k,j) \in \vdash, \forall t \in \{1, ..., T\} && (4.73)\\
& z_{j,t} + z_{k,t} \leq z_{i,t-2} + z_{i,t-1} && \forall (i,j),(i,k) \in \vdash, \forall t \in \{1, ..., T\} && (4.74)\\
& z \in \{0,1\}^{(n+1)T} && && (4.75)
\end{aligned}$$

Para que esta formulação trate o problema limitado, basta adicionar a restrição:

$$\sum_{j \in N} (z_{j,t} - z_{j,t-1}) \leq m \quad \forall t \in \{1, ..., T\}$$

A Formulação Z é essencialmente uma reescrita direta da Formulação Y_{ref}, usando a transformação de variáveis (4.68), à exceção de que não incluímos na Formulação Z as restrições correspondentes a:

(i) (4.44), (4.45) e (4.47), pois deduzimos da Proposição 6 que elas são redundantes no modelo inteiro Y_{ref};

(ii) $y_{j,t} \geq 0$, ou seja, $z_{j,t} - z_{j,t-1} \geq 0$, pois tais restrições podem ser dispensadas na Formulação Z, conforme mostraremos a seguir.

Lema 2. *Seja $z \in \mathbb{R}^{(n+1)T}$ e defina $f(z) = z_{n+1,1} + \sum_{t=2}^{T} t(z_{n+1,t} - z_{n+1,t-1})$. Se z é viável para a Formulação Z, então existe $\hat{z}$ também viável para a Formulação Z satisfazendo $f(\hat{z}) \leq f(z)$ e $\hat{z}_{j,t} \geq \hat{z}_{j,t-1}$, para todos $j \in N \cup \{n+1\}$ e $t \in \{1, \dots, T\}$.*

Demonstração. Seja z satisfazendo (4.70)-(4.75). Defina $t_j = \min\{t : z_{j,t} = 1\}$ e $\hat{z}$ tal que

$$\hat{z}_{j,t} = \begin{cases} 1, & \text{se } t \geq t_j, \\ 0, & \text{se } t < t_j, \end{cases} \qquad \forall j \in N \cup \{n+1\}, \forall t \in \{1, \dots, T\}$$

Claramente $\hat{z}_{j,t} \geq \hat{z}_{j,t-1}$, para todos $j \in N \cup \{n+1\}$ e $t \in \{1, \dots, T\}$. Note ainda que $\hat{z}_{j,t_j} = z_{j,t_j} = 1$ e, por (4.71)-(4.72), $z_{i,t_j-1} = 1$, para todo $i \in N$ (se $j = n+1$) ou $i \vdash j$ (se $j \neq n+1$). Isto implica que $t_i \leq t_j - 1$.

Para mostrar que $\hat{z}$ é viável para a Formulação Z, é suficiente mostrar que $\hat{z}$ satisfaz (4.71)-(4.74), pois (4.70) e (4.75) são trivialmente verificadas. Suponha, por absurdo, que (4.71) ou (4.72) não são satisfeitas, ou seja, que existem $j \in N \cup \{n+1\}$, $i \in N$ (com $i \vdash j$ se $j \neq n+1$) e $t \in \{1, \ldots, T\}$ tais que $\hat{z}_{j,t} > \hat{z}_{i,t-1}$. Logo, $\hat{z}_{j,t} = 1$ e $\hat{z}_{i,t-1} = 0$. Consequentemente, pela definição de $\hat{z}$, $t_j \leq t < t_i + 1$: uma contradição, pois ocorre, como vimos, a desigualdade contrária $t_i \leq t_j - 1$. Portanto, $\hat{z}$ satisfaz (4.71)-(4.72).

Novamente, por absurdo, suponha que (4.73) não é satisfeita, isto é, que existem $j \in N$, $i, k \in \Gamma^-(j)$ e $t \in \{1, \ldots, T\}$ tais que $\hat{z}_{j,t} + \hat{z}_{j,t-1} > \hat{z}_{i,t-2} + \hat{z}_{k,t-2}$. Dois casos seriam possíveis. No primeiro, $\hat{z}_{j,t} + \hat{z}_{j,t-1} = 1$ (o que implica $\hat{z}_{j,t} = 1$, $\hat{z}_{j,t-1} = 0$) e $\hat{z}_{i,t-2} + \hat{z}_{k,t-2} = 0$. Então, teríamos $t = t_j$, $\hat{z}_{j,t} = z_{j,t}$, $\hat{z}_{j,t-1} = z_{j,t-1}$, $\hat{z}_{i,t-2} = z_{,t-2}$ e $\hat{z}_{k,t-2} = z_{k,t-2}$, contrariando o fato de que z satisfaz (4.73). No segundo caso, $\hat{z}_{j,t} = \hat{z}_{j,t-1} = 1$ e $\hat{z}_{i,t-2} + \hat{z}_{k,t-2} < 2$. Sendo $\hat{z}_{j,t-1} = 1$, concluímos do parágrafo anterior que $\hat{z}_{i,t-2} = \hat{z}_{k,t-2} = 1$: uma contradição. Logo, $\hat{z}$ satisfaz (4.73). De modo similar, podemos mostrar que também satisfaz (4.74).

Finalmente, queremos mostrar que $f(\hat{z}) \leq f(z)$. Temos que

$$\begin{aligned}
f(\hat{z}) - f(z) &= \sum_{t=1}^{T} t(\hat{z}_{n+1,t} - z_{n+1,t}) - \sum_{t=2}^{T} t(\hat{z}_{n+1,t-1} - z_{n+1,t-1}) \\
&= \sum_{t=1}^{T-1} t(\hat{z}_{n+1,t} - z_{n+1,t}) - \sum_{t=1}^{T-1} (t+1)(\hat{z}_{n+1,t} - z_{n+1,t}) \\
&= \sum_{t=1}^{T-1} (z_{n+1,t} - \hat{z}_{n+1,t}) \leq 0
\end{aligned}$$

□

Proposição 11. *A Formulação Z é correta.*

Demonstração. Iremos demonstrar a corretude de Z a partir da corretude de Y_{ref}, mostrando que existe uma correspondência entre as soluções viáveis dos dois modelos.

Seja y viável para Y_{ref} e defina z de acordo com (4.68). Claramente z satisfaz (4.70)-(4.75) e tem o mesmo valor de objetivo que y, pois $f(z) = \sum_{t=1}^{T} t * y_{n+1,t}$.

Por outro lado, seja z viável para Z. Tome $\hat{z}$ viável para Z dado pelo Lema 2. Defina y tal que

$$y_{j,t} = \begin{cases} \hat{z}_{j,t} - \hat{z}_{j,t-1}, & \text{se } t > 1, \\ \hat{z}_{j,t}, & \text{se } t = 1 \end{cases} \qquad \forall j \in N \cup \{n+1\}, \forall t \in \{1, \ldots, T\}.$$

Como $\hat{z} \in \{0,1\}^{(n+1)T}$ e $\hat{z}_{j,t} > \hat{z}_{j,t-1}$, temos que $y \in \{0,1\}^{(n+1)T}$. E $f(z) \geq f(\hat{z}) = \sum_{t=1}^{T} t * y_{n+1,t}$. Além disso, y claramente satisfaz (4.41)-(4.43), (4.46) e (4.48) (4.49). Entao, o resultado segue pela Proposição 6, que assegura que as restrições (4.44), (4.45) e (4.47) são desnecessárias para a correta definição de Y_{ref}. □

Na verdade, como garante a Proposição 8, as restrições (4.44), (4.45) e (4.47) não fazem diferença mesmo na relaxação linear da Formulação Z_{exp} a seguir, obtida diretamente de Y_{exp} com a mudança de variáveis.

Formulação Z_{exp}

$$min \quad z_{n+1,1} + \sum_{t=2}^{T} t * (z_{n+1,t} - z_{n+1,t-1}) \tag{4.76}$$

$$s.a \quad z_{j,T} = 1 \qquad \forall j \in N \cup \{n+1\} \tag{4.77}$$

$$z_{n+1,t} \leq z_{i,t-1} \qquad \forall i \in N, \forall t \in \{1, ..., T\} \tag{4.78}$$

$$z_{j,t} \leq z_{i,t-1} \qquad \forall (i,j) \in \vdash, \forall t \in \{1, ..., T\} \tag{4.79}$$

$$(|S| - 1)z_{j,t} + z_{j,t-1} \leq \sum_{i \in S} z_{i,t-2} \qquad \forall j \in N, \forall S \in\subseteq \Gamma^{-}(j), \forall t \in \{1, ..., T\} \tag{4.80}$$

$$\sum_{j \in S} z_{j,t} \leq (|S| - 1)z_{i,t-2} + z_{i,t-1} \qquad \forall i \in N, \forall S \in\subseteq \Gamma^{+}(i), \forall t \in \{1, ..., T\} \tag{4.81}$$

$$z \in \{0,1\}^{(n+1)T} \tag{4.82}$$

Diteramente das proposições 9 e 10 obtemos a separação das restrições (4.80) e (4.80).

Proposição 12. *Um vetor $y \in \mathbb{R}^{(n+1)T}$ viola alguma restrição de (4.80) para $j \in N$ e $t \in \{1, \ldots, T\}$ se, e somente se, $\sum_{i \in S}(z_{j,t} - z_{i,t-2}) > z_{j,t} - z_{j,t-1}$, onde $S = \{i \in \Gamma^{-}(j) : z_{j,t} > z_{i,t-2}\}$.*

Proposição 13. *Um vetor $y \in \mathbb{R}^{(n+1)T}$ viola alguma restrição de (4.81) para $i \in N$ e $t \in \{1, \ldots, T\}$ se, e somente se, $\sum_{j \in S}(z_{j,t} - z_{i,t-2}) > z_{i,t-1} - z_{i,t-2}$, onde $S = \{j \in \Gamma^{+}(i) : z_{j,t} > z_{i,t-2}\}$.*

Capítulo 5

Estudo comparativo entre as formulações

5.1 Estudo comparativo teórico

Nesta seção abordaremos relações entre as diversas formulações apresentadas anteriormente. É possível estabelecer uma hierarquia entre as formulações apresentadas, explicitanto qual formulação é mais forte e qual é mais fraca, do ponto de vista da relaxação linear. Genericamente, denotaremos por $LP(F)$ o valor da relaxação linear de uma formulação F de programação inteira.

Trivialmente, podemos estabelecer relações entre as formulações Y, Y_{ref}, Y_{exp}, Z e Z_{exp}. A formulação Y_{ref} é claramente mais forte que Y, pois é obtida desta com *lifting* de suas restrições e remoção de restrições redundantes, apontadas pela Proposição 7. Por outro lado, Y_{exp} é um fortalecimento de Y_{ref}, de acordo com a Proposição 8. Por sua vez, Y_{exp} é equivalente a Z_{exp} acrescida das restrições

$$z_{j,t} \geq z_{j,t-1} \quad \forall j \in N \cup \{n+1\}\, \forall t \in \{1, \ldots, T\}.$$

Denotemos por Z'_{exp} essa formulação mais restrita. Note que Z'_{exp} é exatamente Y_{exp} após a mudança de variáveis. Já Z é obtida de Y_{ref} por mudança de variáveis e remoção de algumas restrições (as que envolvem w). Consequentemente, temos que:

$$LP(Y) \leq LP(Y_{ref}) \leq LP(Y_{\text{exp}}) = LP(Z'_{exp}) \quad e \quad LP(Z) \leq LP(Y_{ref}) \leq LP(Z'_{exp}).$$

Em relação à formulação X, não podemos definir uma relação de superioridade com Y. A relaxação linear de uma ou de outra pode fornecer melhor limite, a depender da instância. Porém, podemos comparar X com Z_{exp} ou com uma modificação de Y_{ref} no caso ilimitado. A comparação no caso limitado fica dificultada pela existência do big-M em X.

Começando a comparação, perceba que as seguintes relações podem ser estabelecidas entre as variáveis binárias que definem tais formulações:

$$x_i + 1 = \sum_{t=1}^{T} t y_{i,t} = \sum_{t=1}^{T}\sum_{t'=t}^{T} y_{i,t'} = \sum_{t=1}^{T}(y_{i,t} + \sum_{t'=t+1}^{T} y_{i,t'}) = \sum_{t=1}^{T}(y_{i,t} + 1 - z_{i,t}) = 1 + T - \sum_{t=1}^{T} z_{i,t}$$

Lembre que na Formulação X os tempos começam com 0 e nas formulações Y e Z os tempos são indexados a partir de 1. Usando essas relações, iremos converter uma solução viável relaxada de uma formulação em uma solução viável relaxada de outra, de modo a mostrar que:

Proposição 14. *Para uma mesma instância do problema ilimitado, temos que* $LP(X) \leq LP(Z_{exp})$.

Demonstração. Seja z uma solução viável para a relaxação linear de Z_{exp}. Devemos mostrar que existe solução viável relaxada (x, w) para X com o mesmo valor de objetivo. Tome (x, w) tal que:

$$x_i = T - \sum_{t=1}^{T} z_{i,t} \quad \forall i \in N \cup \{n+1\}$$

e

$$w_{i,j} = \max\left\{0, \sum_{t=1}^{T} z_{j,t} - \sum_{t=1}^{T} z_{i,t-2}\right\} \quad \forall i \vdash j$$

Claramente, $x \geq 0$ e $w \geq 0$. Seja $i \in N$. Por (4.78), temos que $\sum_{t=1}^{T} z_{n+1,t} \leq \sum_{t=1}^{T} z_{i,t-1}$. Como $z_{i,T} = 1$ segundo (4.77), segue-se que $\sum_{t=1}^{T} z_{n+1,t} \leq \sum_{t=1}^{T} z_{i,t} - 1$, implicando em $x_{n+1} \geq x_i + 1$. Seja $i \vdash j$. Por (4.77) e (4.79), $z_{i,T} = z_{i,T-1} = 1$. Logo,

$$w_{i,j} \geq \sum_{t=1}^{T} z_{j,t} - \sum_{t=1}^{T} z_{i,t-2} = \sum_{t=1}^{T} z_{j,t} - \sum_{t=1}^{T} z_{i,t} + z_{i,T-1} + z_{i,T} = x_i - x_j + 2$$

Além disso, definindo $S_j = \{i \in \Gamma^-(j) : \sum_{t=1}^{T} z_{j,t} - \sum_{t=1}^{T} z_{i,t-2} > 0\}$ e $S_i = \{j \in \Gamma^+(i) : \sum_{t=1}^{T} z_{j,t} - \sum_{t=1}^{T} z_{i,t-2} > 0\}$, temos respectivamente por (4.80) e (4.81) que

$$\sum_{i \in \Gamma^-(j)} w_{i,j} = \sum_{i \in S_j}(\sum_{t=1}^{T} z_{j,t} - \sum_{t=1}^{T} z_{i,t-2}) \leq z_{j,t} - z_{j,t-1} \leq 1$$

$$\sum_{j \in \Gamma^+(i)} w_{i,j} = \sum_{j \in S_i}(\sum_{t=1}^{T} z_{j,t} - \sum_{t=1}^{T} z_{i,t-2}) \leq z_{i,t-1} - z_{i,t-2} \leq 1$$

Logo (x, w) é viável para a relaxação de (4.14)-(4.20).

Finalmente, temos a correta correspondência entre os valores das funções objetivo, pois

$$\sum_{t=1}^{T} t(z_{n+1,t} - z_{n+1,t-1}) = \sum_{t=1}^{T} t\, z_{n+1,t} - \sum_{t=1}^{T-1}(t+1)z_{n+1,t} = Tz_{n+1,T} - \sum_{t=1}^{T} z_{n+1,t} + z_{n+1,T}$$
$$= T - \sum_{t=1}^{T} z_{n+1,t} + 1 = x_{n+1} + 1$$

Lembre que o valor da função em Z deve ser uma unidade maior, pois os tempos começam em 1, enquanto em X eles iniciam em 0. □

Mostraremos a seguir que podemos obter uma formulação mais forte que X para o problema ilimitado, mesmo sem a necessidade de incluirmos um conjunto exponencial de restrições como em Z_{exp} ou Y_{exp}. Considere então a seguinte formulação obtida a partir de Y_{ref}, que denominaremos Y'_{ref}. Esta formulação será dada por (4.40)-(4.47) juntamente com as restrições adicionais

$$\sum_{t=1}^{T}\sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t=1}^{T}\sum_{t'=1}^{t-2} y_{i,t'} + w_{i,j} \quad \forall (i,j) \in \vdash \tag{5.1}$$

que podem ser igualmente escritas como

$$\sum_{t=1}^{T}\sum_{t'=1}^{t} y_{j,t'} \leq \sum_{t=1}^{T}\sum_{t'=1}^{t} y_{i,t'} - 2 + w_{i,j} \quad \forall (i,j) \in \vdash$$

pois $\sum_{t=1}^{T} y_{i,t-1} = \sum_{t=1}^{T} y_{i,t} = 1$, já que j é executado depois de i e, portanto, i não pode ser executado no tempo T. Então, usando (4.42)-(4.43), tais expressões tornam-se

$$\sum_{t=1}^{T}\left(\sum_{t'=t}^{T} y_{j,t'} - y_{j,t}\right) \geq \sum_{t=1}^{T}\left(\sum_{t'=t}^{T} y_{i,t'} - y_{i,t}\right) + 2 - w_{i,j} \quad \forall (i,j) \in \vdash$$

ou ainda

$$\sum_{t=1}^{T} t y_{j,t} \geq \sum_{t=1}^{T} t y_{i,t} + 2 - w_{i,j} \quad \forall (i,j) \in \vdash$$

sendo portanto claramente válidas para Y_{ref}.

Proposição 15. *Para uma mesma instância do problema ilimitado, temos que* $LP(X) \leq LP(Y'_{ref})$.

Demonstração. Seja $(y,w) \geq 0$ satisfazendo (4.41)-(4.47) e (5.1). Devemos mostrar que existe x tal que (x,w) é solução viável relaxada para X com o mesmo valor de objetivo. Seja x definido como

$$x_i = \sum_{t=1}^{T}\sum_{t'=t}^{T} y_{i,t'} - 1 = \sum_{t=1}^{T} t y_{i,t} - 1 \quad \forall i \in N \cup \{n+1\}$$

Claramente, $x \geq 0$. Seja $i \in N$. Por (4.41), temos, para todo $t \in \{1, \ldots, T\}$, que $1 - \sum_{t'=t+1}^{T} y_{n+1,t'} \leq 1 - \sum_{t'=t}^{T} y_{i,t'}$, ou seja, $\sum_{t'=t}^{T} y_{n+1,t'} - \sum_{t'=t}^{T} y_{i,t'} \geq y_{n+1,t}$. Somando essas expressões em t e usando (4.42), obtemos que $x_{n+1} - x_i \geq 1$. Seja $i \vdash j$. Dado que y verifica (4.42)-(4.43), podemos usar a última expressão equivalente a (5.1) para obter diretamente $x_j - x_i \geq 2 - w_{i,j}$. Logo, concluímos que (x, w) é viável para a relaxação de X.

Adicionalmente, temos

$$\sum_{t=1}^{T} t y_{n+1,t} = \sum_{t=1}^{T} \sum_{t'=t}^{T} y_{n+1,t'} = x_{n+1} + 1,$$

que é a correta correspondência entre os valores das funções objetivo, posto que em Y_{ref} os tempos começam em 1, enquanto em X eles iniciam em 0. □

5.2 Estudo comparativo computacional

De posse de cinco das sete formulações apresentadas, X, Y, Y_{ref}, Z e Z_{exp}, faremos uma comparação entre elas através de experimentos computacionais. O objetivo dessa comparação é identificar, possivelmente, uma formulação com melhor desempenho computacional dentre todas apresentadas.

Fizemos testes os teste comparativos com as cinco formulações sob quatro variações no número de processadores. Na primeira, consideramos infinitos processadores. Nas seguintes, consideramos que o número de processadores será o valor obtido da divisão da largura do grafo de tarefas respectivamente por 2, 4 e 8. Utilizamos a largura como parâmetro para que tenhamos realmente um limite no número de processadores, esta limitação acontece mais fortemente na terceira variação. Note que se o número de processadores é maior que a largura, o problema equivale ao caso ilimitado. Em cada uma das variações observamos a relaxação linear e o problema inteiro propriamente dito. Como instâncias, utilizamos as descritas no Apêndice A, que são divididas em três grupos (Grupo de Instâncias Aleatórias 1, Grupo de Instâncias Aleatórias 2 e Grupo de Instâncias Estruturadas) e cada um possui características que são detalhadas no Apêndice, onde também apresentamos o ambiente computacional utilizado.

Ainda em relação às formulações, medimos o grau de influência dos limites, inferiores e superiores, apresentados no Capítulo 3 na resolução do problema. Inicialmente, executamos todos testes computacionais, para uma formulação específica, sem utilizar os limites no modelo e repetimos os mesmos testes, mas aplicando no modelo os melhores limites, inferior e superior, encontrados para a instância e quantidade de processadores. A forma como aplicamos os limites é abordada na Seção 5.2.1.

5.2.1 Resultados computacionais

Todas as formulações foram implementadas sobre a mesma arquitetura, sem distinção alguma. Apenas a construção do modelo foi, obviamente, modificada para se adequar exatamente ao que é apresentado no Capítulo 4. Utilizamos os melhores limites encontrados, inferior e superior, para definição do big-M e inclusão das restrições (4.21)-(4.22) na Formulação X e para fixação de variáveis nas diversas variações das formulacões Y e Z. Note que os limites calculados fornecem, para cada tarefa i, um intervalo $[t_i^-, t_i^+]$ onde ela dever ser executada. Sendo assim, podemos fixar

$$y_{i,t} = 0 \text{ e } z_{i,t} = 0 \quad \forall t < t_i^-$$
$$y_{i,t} = 0 \text{ e } z_{i,t} = 1 \quad \forall t > t_i^+$$

Observamos, para cada formulação, os problemas considerando ou não a integralidade. Destacamos que a Formulação Z_{exp} foi implementada da seguinte forma:

- Inicialmente as restrições do modelo são ($4.77 - 4.79$), (4.73) e (4.74). Ou seja, as restrições de pares de tarefas utilizadas em Z estão, inicialmente, presentes em Z_{exp}.
- Na relaxação linear, iterativamente, buscamos restrições violadas segundo as proposições 12 e 13 e as inserimos no modelo.
- Repetimos o passo anterior, até que não haja mais nenhuma restrição conhecidamente violada.
- Tomamos as restrições originais e as restrições violadas, que foram inseridas na relaxação linear, e resolvemos o problema inteiro.

Os resultados computacionais serão apresentados na seguinte ordem: testes com infinitos processadores, testes com m igual a $\lfloor \omega(\prec)/2 \rfloor$, $\lfloor \omega(\prec)/4 \rfloor$ ou $\lfloor \omega(\prec)/8 \rfloor$. As instâncias utilizadas em cada teste estão definidas e detalhadas no Apêndice A. Uma instância só é avaliada se o número de processadores derivado de sua largura é pelo menos 2. Por exemplo, se $\omega(\prec) = 5$, a instância será avaliada apenas para o caso ilimitado e para $m = \lfloor \omega(\prec)/2 \rfloor = 2$. Nas tabelas do Apêndice, onde apresentamos as instâncias, estão presentes as colunas com o nome da instância (DAG), o número de tarefas ($|N|$), a largura do grafo ($\omega(\prec)$), o número de processadores (m), e os melhores limites inferior e superior encontrados, respectivamente, (LB) e (UB).

Nas tabelas (5.1-5.3), (5.7-5.9), (5.13-5.15) e (5.19-5.21) testamos a relaxação linear, em cada formulação, com o número de processadores sendo respectivamente infinitos, $\lfloor \omega(\prec)/2 \rfloor$,

$\lfloor \omega(\prec)/4 \rfloor$ e $\lfloor \omega(\prec)/8 \rfloor$. Apresentamos, para cada instância, as colunas LP que indicam os valores da relaxação obtidos em cada formulação, de forma fracionária, e o tempo de execução de cada instância em cada formulação, são as colunas marcadas com t. Nas tabelas (5.4-5.6), (5.10-5.12), (5.16-5.18) e (5.22-5.24) encontram-se os testes com o problema inteiro. As colunas apresentam, para cada instância, o valor inteiro do problema, são as colunas (IP), e ainda os tempos (t) gastos para resolver a formulação. Estes tempos observados no problema inteiro incluem o tempo de resolução da relaxação linear. Todos os tempos são medidos em segundos. Vale ressaltar que impusemos um limite máximo de tempo de execução do problema em 600 segundos em todas as execuções. Entendemos que, como a Formulação Z obtém tempos bastante inferiores a 600 segundos, a maioria inferior a 60 segundos, este é um limite superior aceitável de tempo total.

Salientamos que, para a construção dos gráficos que avaliam o parâmetro tempo, como possuímos tempos muito discrepantes, tomamos log_{10} do tempo em questão e o plotamos no gráfico. Ou seja, se um experimento de uma instância executou em $50s$, será plotado no gráfico o valor $\log_{10}(50) = 1,6989$ na posição correspondente àquela instância. Por esta razão poderemos ter números negativos.

Para a produção dos gráficos que envolvem o valor do *makespan* obtido pela relaxação linear utilizamos a seguinte estratégia: tomamos o valor ótimo subtraído do valor da relaxação em questão e este resultado dividimos pelo valor ótimo. O resultado dessas operações, para cada instância, é plotado no gráfico. Por esta razão, vale observar que, quanto mais próximo de zero, melhor será o limite obtido, pois estará mais próximo ao ótimo.

Analisando os gráficos a seguir, percebemos que a Formulação X, que serviu de ponto de partida para este trabalho, obtém bons resultados na relaxação linear, tanto em valor quanto em tempo. Todavia, ao observarmos seu desempenho no problema inteiro, esta formulação se mostra muito ineficiente. Em todos os testes com o problema inteiro, X perde para todas as outras analisadas, sendo especialmente deficiente quando temos um número muito reduzido de processadores. Nesse caso, mesmo sendo forte sua relaxação linear, a Formulação X não consegue, para a maioria das instâncias, resolver o problema inteiro dentro do limite de 600 segundos estabelecido. Em particular, quando $m = \lfloor \omega(\prec)/8 \rfloor$, a relaxação da Formulação X está entre as melhores (quase sempre a melhor) tanto na qualidade do limite gerado quanto no tempo gasto para obtê-lo, mas frequentemente o problema inteiro não é resolvido no tempo limite.

Como poderá ser observado nas tabelas e gráficos a seguir, quando analisamos a relaxação, não há uma clara relação de dominância entre X e Y. Para infinitos processadores e para

$m = \lfloor \omega(\prec)/2 \rfloor$, as duas são comparáveis. A relaxação de X supera Y nos casos onde dividimos a largura por 4 e 8, mas, como mencionado, o mesmo não se repete com o problema inteiro. A Formulação Y se coloca como intermediária entre X e Z do ponto de vista inteiro.

Assim como Y, as experiências com Y_{ref} não estabelecem uma dominância sobre X, levando-se em consideração a relaxação. Porém, diferentemente de Y, a Formulação Y_{ref} consegue valores de *makespan* na relaxação mais comparáveis à X, embora se mantenha o comportamento de Y observando o tempo. De fato, Y_{ref} se sobressai em relação a Y, principalmente quando restringimos o número de processadores.

A formulação que obteve os melhores resultados de forma unânime foi a Formulação Z, batendo todas as outras formulações (X, Y e Y_{ref}) na maior parte das instâncias e sendo comparável ao melhor resultado obtido nas instâncias em que não alcançou resultado unânime. O único teste onde houve alguma dominância de outra formulação foi no teste do problema relaxado e $m = \lfloor \omega(\prec)/8 \rfloor$, onde X apresentou uma leve superioridade. Vale observar que o valor da relaxação obtido pelas duas neste teste foi comparável.

A Formulação Z_{exp} também obteve resultados expressivos, se comparando à Z em algumas instâncias, nos testes com o problema inteiro. A inclusão dos cortes gerados durante a solução da relaxação foi muitas vezes útil para a resolução do problema inteiro. Entretanto, o esforço demandado para a obtenção das restrições violadas durante a relaxação raramente levou a uma redução no tempo total gasto para resolver o problema. A relaxação de Z_{exp}, em vários casos, consome um tempo bem superior ao da relaxação de Z.

Isto posto, concluímos que a melhor opção de resolução é a Formulação Z, seja com infinitos processadores ou com um número restrito. Além de possuir uma relaxação linear de qualidade, o tempo total gasto para resolver o problema é, em geral, baixo.

Seguem os gráficos e as tabelas detalhados:

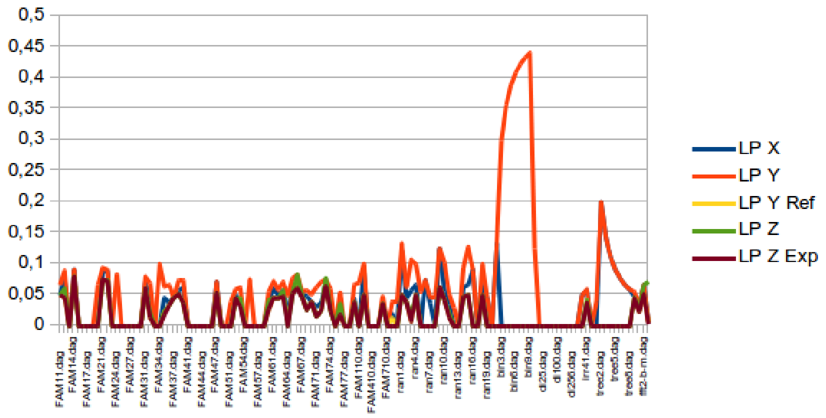


Figura 5.1: Gráfico Instância X Makespan da relaxação linear com infinitos processadores

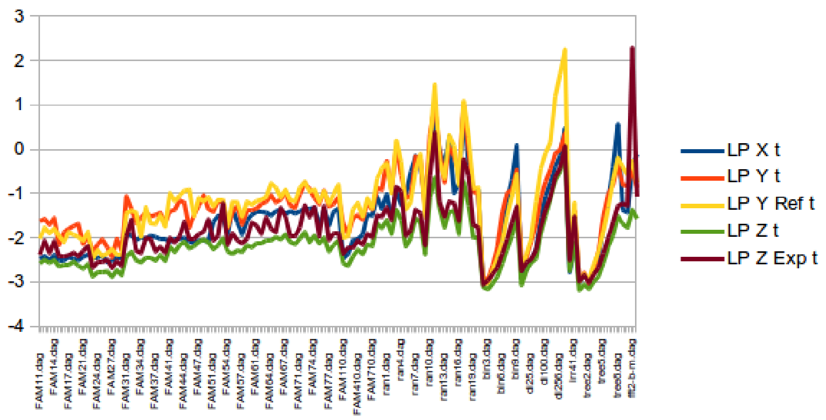


Figura 5.2: Gráfico Instância X Tempo da relaxação linear com infinitos processadores

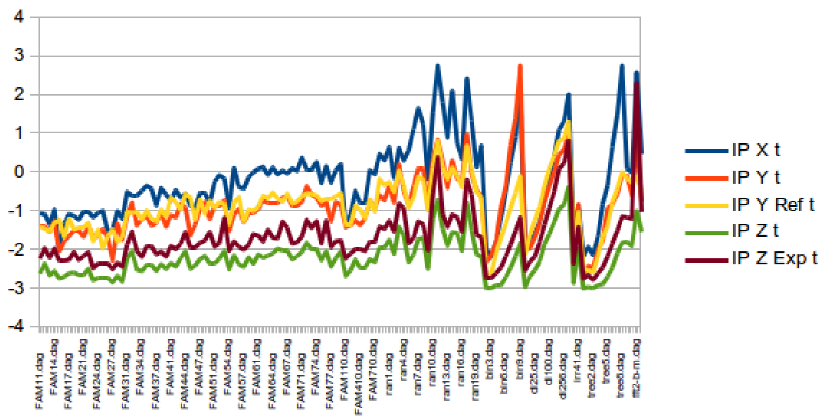


Figura 5.3: Gráfico Instância X Tempo do problema inteiro com infinitos processadores

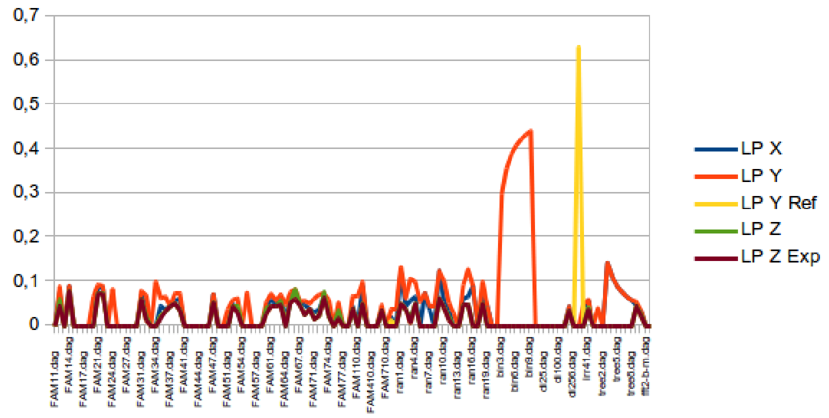


Figura 5.4: Gráfico Instância X Makespan da relaxação linear com $m = \lfloor \omega(\prec)/2 \rfloor$

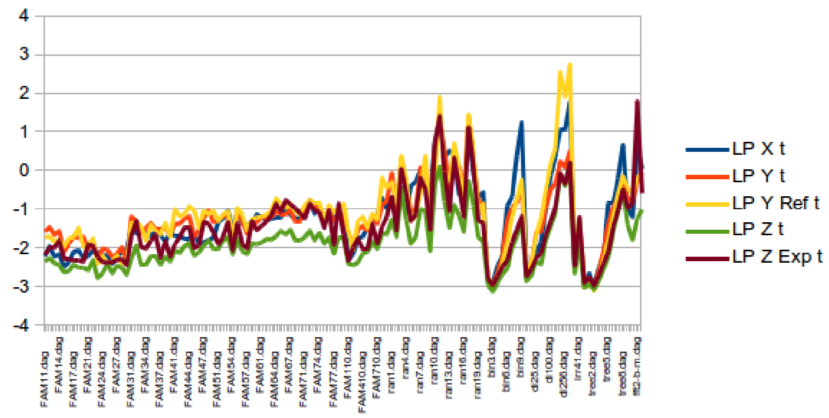
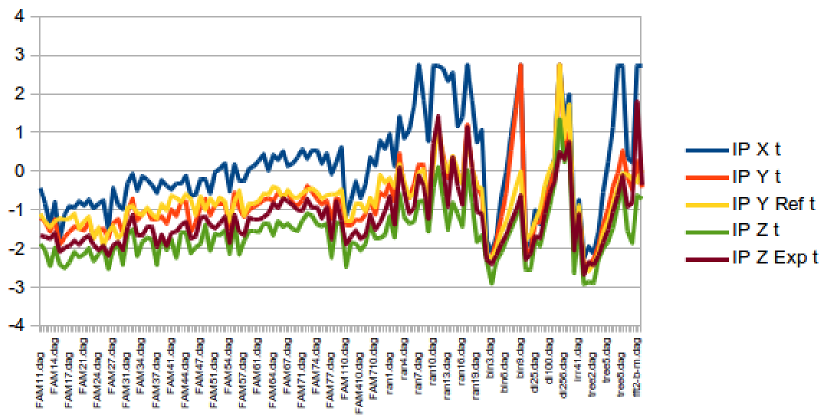


Figura 5.5: Gráfico Instância X Tempo da relaxação linear com $m = \lfloor \omega(\prec)/2 \rfloor$



vspace-0.5cm captionGráfico Instância X Tempo do problema inteiro com $m = \lfloor \omega(\prec)/2 \rfloor$

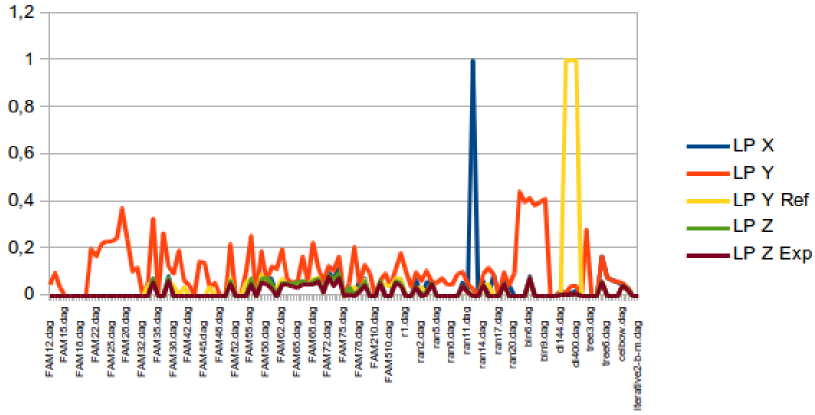


Figura 5.6: Gráfico Instância X Makespan da relaxação linear com $m = \lfloor \omega(\prec)/4 \rfloor$

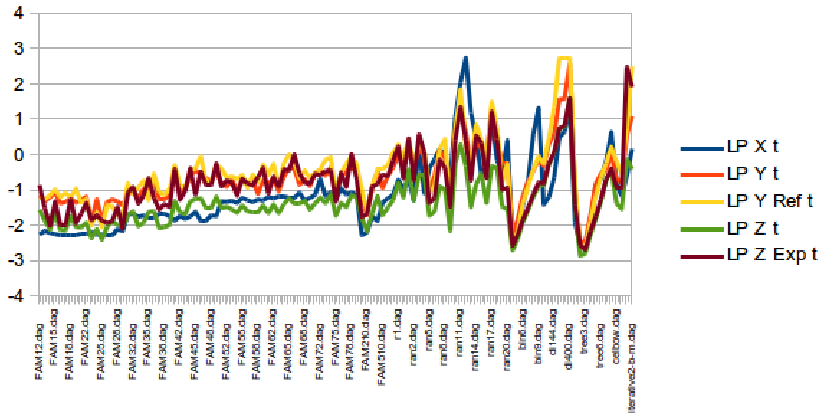


Figura 5.7: Gráfico Instância X Tempo da relaxação linear com $m = \lfloor \omega(\prec)/4 \rfloor$

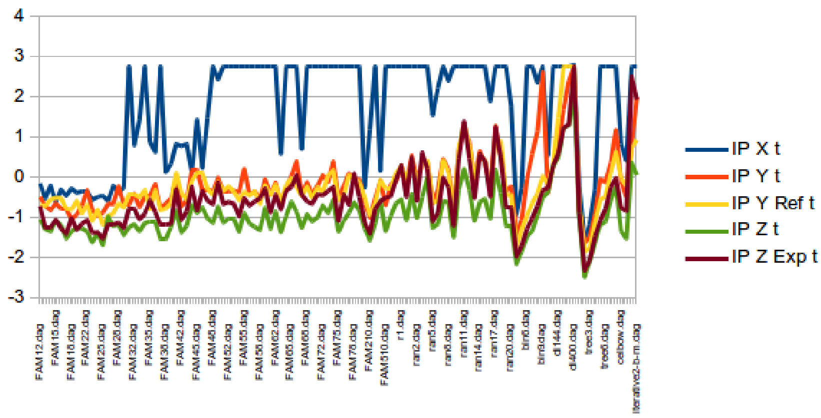


Figura 5.8: Gráfico instância X Tempo do problema inteiro com $m = \lfloor \omega(\prec)/4 \rfloor$

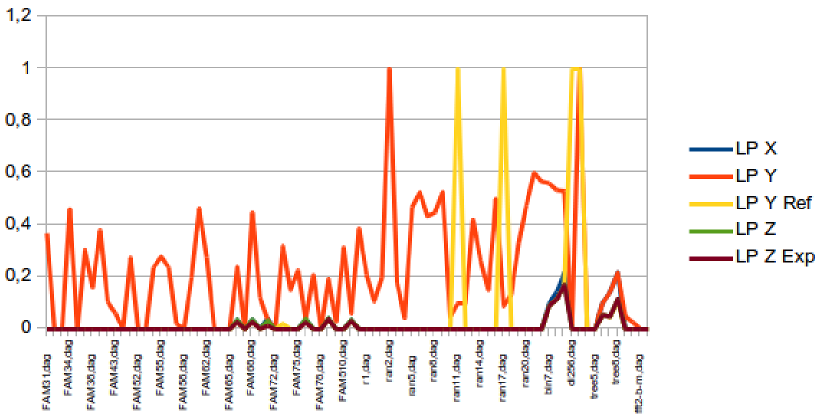
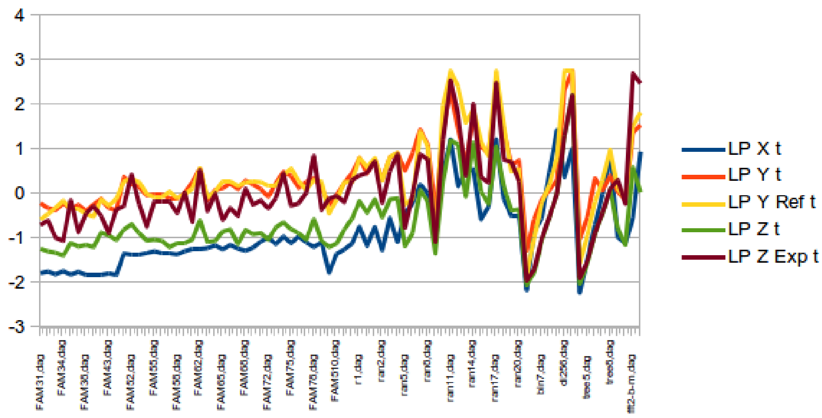


Figura 5.9: Gráfico Instância X Makespan da relaxação linear com $m = \lfloor \omega(\prec)/8 \rfloor$



vspace-0.5cm captionGráfico Instância X Tempo da relaxação linear com $m = \lfloor \omega(\prec)/8 \rfloor$

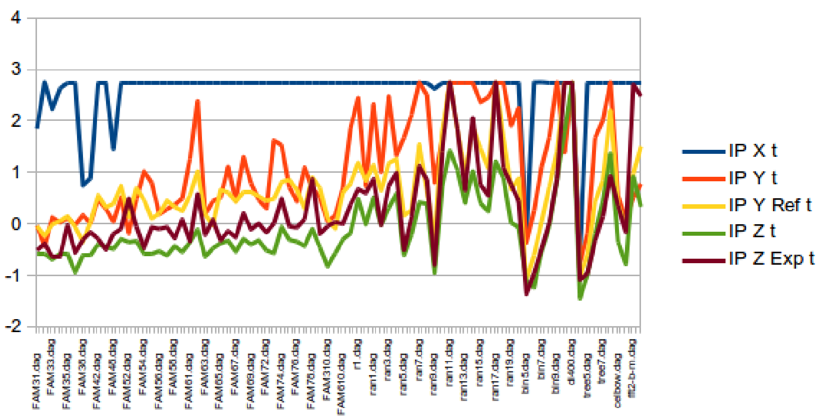


Figura 5.10: Gráfico Instância X Tempo do problema inteiro com $m = \lfloor \omega(\prec)/8 \rfloor$

DAG	IP	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
FAM11.dag	10	9,5	9,33333	9,5	9,5	9,5	0,00358	0,025403	0,010212	0,002821	0,004428
FAM12.dag	11	10,1515	10,0123	10,5	10,3333	10,5	0,004141	0,028406	0,0181	0,00338	0,00916
FAM13.dag	12	12	12	12	12	12	0,003271	0,021348	0,013585	0,002925	0,004845
FAM14.dag	11	10	10	10,125	10,125	10,125	0,004374	0,030023	0,017346	0,003356	0,009082
FAM15.dag	10	10	10	10	10	10	0,003286	0,007466	0,01174	0,002412	0,004025
FAM16.dag	10	10	10	10	10	10	0,003275	0,014278	0,008615	0,002599	0,00403
FAM17.dag	10	10	10	10	10	10	0,003985	0,016987	0,014324	0,002633	0,004269
FAM18.dag	11	11	11	11	11	11	0,00356	0,019802	0,0125	0,003083	0,004917
FAM19.dag	10	10	9,33333	10	10	10	0,003319	0,022156	0,01129	0,002515	0,004061
FAM21.dag	8	7,3	7,25	7,4	7,4	7,4	0,004008	0,007774	0,009181	0,002145	0,005416
FAM22.dag	11	10,0313	10	10,1818	10,1818	10,2	0,004445	0,011946	0,014663	0,002735	0,007013
FAM23.dag	7	7	7	7	7	7	0,003854	0,004994	0,003615	0,001438	0,002281
FAM24.dag	10	10	9,16667	10	10	10	0,003696	0,007671	0,005642	0,001848	0,002988
FAM25.dag	8	8	8	8	8	8	0,004191	0,009831	0,004098	0,001832	0,002942
FAM26.dag	10	10	10	10	10	10	0,003023	0,006539	0,004764	0,001866	0,003403
FAM27.dag	7	7	7	7	7	7	0,003197	0,003827	0,005918	0,001395	0,002221
FAM28.dag	9	9	9	9	9	9	0,003875	0,010066	0,004258	0,001989	0,003175
FAM29.dag	8	8	8	8	8	8	0,003647	0,005546	0,004301	0,001527	0,002473
FAM31.dag	11	10,2143	10,125	10,3333	10,3333	10,3333	0,01316	0,092371	0,038612	0,004079	0,011656
FAM32.dag	12	11,5	11,1667	11,8333	11,75	11,8333	0,012219	0,047664	0,044243	0,00514	0,027162
FAM33.dag	12	12	12	12	12	12	0,009475	0,023185	0,040336	0,003324	0,005451
FAM34.dag	10	10	9	10	10	10	0,011007	0,030396	0,011318	0,00299	0,004862
FAM35.dag	11	10,5	10,3	10,75	10,75	10,8	0,010036	0,040013	0,052737	0,003842	0,010584
FAM36.dag	10	9,625	9,33333	9,66667	9,66667	9,66667	0,011921	0,032624	0,022832	0,003859	0,010179
FAM37.dag	11	10,5	10,5	10,5	10,5	10,5	0,011182	0,034963	0,021774	0,003267	0,005246
FAM38.dag	13	12,25	12,0526	12,3333	12,3333	12,3333	0,010078	0,039264	0,035213	0,004347	0,006695
FAM39.dag	9	8,44444	8,33333	8,66667	8,66667	8,66667	0,01003	0,023893	0,019213	0,003211	0,004922
FAM41.dag	13	13	13	13	13	13	0,008671	0,04249	0,109294	0,006775	0,010675
FAM42.dag	11	11	11	11	11	11	0,009932	0,045793	0,072676	0,005135	0,008389
FAM43.dag	12	12	12	12	12	12	0,009787	0,075345	0,09643	0,0073	0,011434
FAM44.dag	12	12	12	12	12	12	0,010627	0,066743	0,123684	0,008686	0,024542
FAM45.dag	12	12	12	12	12	12	0,008734	0,018217	0,131029	0,006201	0,00993
FAM46.dag	13	13	13	13	13	13	0,008204	0,034874	0,03319	0,006842	0,009907
FAM47.dag	14	13	13	13,25	13,25	13,25	0,009248	0,057721	0,08505	0,008708	0,013021
FAM48.dag	15	15	15	15	15	15	0,009661	0,097829	0,070953	0,009814	0,014683
FAM49.dag	14	14	14	14	14	14	0,008953	0,04795	0,08865	0,008542	0,027748
FAM51.dag	13	13	12,5	13	13	13	0,024557	0,041554	0,054287	0,005923	0,00936
FAM52.dag	15	14,25	14,1111	14,3333	14,3333	14,3333	0,033391	0,071076	0,076979	0,007159	0,011593
FAM53.dag	16	15,3889	15	15,2857	15,28	15,5	0,033425	0,069157	0,074934	0,010402	0,063742
FAM54.dag	11	11	11	11	11	11	0,012603	0,017827	0,019736	0,005233	0,007514
FAM55.dag	10	10	9,25	10	10	10	0,050618	0,040651	0,068768	0,004666	0,013453
FAM56.dag	14	14	14	14	14	14	0,023674	0,056562	0,068103	0,005549	0,00929
FAM57.dag	11	11	11	11	11	11	0,012749	0,021383	0,03209	0,005127	0,008052
FAM58.dag	12	12	12	12	12	12	0,026705	0,043405	0,07014	0,007271	0,010163
FAM59.dag	12	11,5	11,3333	11,6667	11,5	11,6667	0,035395	0,045617	0,060063	0,006513	0,023738
FAM61.dag	11	10,3333	10,2	10,5	10,5	10,5	0,041338	0,049153	0,07256	0,007819	0,020193
FAM62.dag	15	14,25	14,125	14,3333	14,3333	14,3333	0,041211	0,06308	0,078133	0,008217	0,013027
FAM63.dag	14	13,2	13	13,3333	13,2	13,3333	0,040025	0,069874	0,089361	0,009569	0,030734
FAM64.dag	17	16,6667	16,1667	16,8333	17	17	0,034218	0,097951	0,182539	0,009533	0,017007
FAM65.dag	13	12,1667	12	12,3125	12,3125	12,3125	0,042538	0,067747	0,14793	0,011113	0,01435
FAM66.dag	12	11,1503	11	11,2273	11	11,2667	0,046554	0,079058	0,088002	0,009853	0,050764
FAM67.dag	18	17,0769	17	17,1429	17,1429	17,1667	0,036847	0,116612	0,130318	0,012843	0,032417
FAM68.dag	13	12,375	12,25	12,6667	12,6667	12,6667	0,042821	0,05792	0,076021	0,008586	0,011575
FAM69.dag	13	12,5	12,3333	12,5	12,5	12,5	0,037928	0,050581	0,071638	0,007642	0,011777
FAM71.dag	16	15,5	15	15,75	15,75	15,75	0,042446	0,104114	0,143961	0,010223	0,018398
FAM72.dag	14	13,4808	13	13,6667	13,6667	13,6667	0,04722	0,183974	0,198128	0,014383	0,057225
FAM73.dag	13	12,125	12,0303	12,125	12	12,1667	0,047852	0,137988	0,118942	0,008537	0,031326
FAM74.dag	14	13,5	13,1667	13,6667	13,6667	13,6667	0,050747	0,085889	0,131551	0,011923	0,052884
FAM75.dag	15	15	15	15	15	15	0,019603	0,061842	0,069562	0,007997	0,011633
FAM76.dag	14	13,5263	13,25	13,75	13,5	13,75	0,047983	0,123768	0,120645	0,010667	0,055284
FAM77.dag	10	10	10	10	10	10	0,020525	0,05507	0,062483	0,005284	0,009567
FAM78.dag	15	15	15	15	15	15	0,04216	0,062094	0,10476	0,007353	0,01362
FAM79.dag	13	12,4286	12,125	12,5	12,5	12,5	0,048814	0,087675	0,170075	0,009293	0,013541
FAM110.dag	12	12	11,1667	12	12	12	0,003637	0,022879	0,010659	0,002809	0,004689
FAM210.dag	10	9,08333	9	9,5	9,5	9,5	0,004634	0,012896	0,011697	0,002521	0,006483
FAM310.dag	13	13	13	13	13	13	0,009494	0,030172	0,047498	0,004049	0,006446
FAM410.dag	11	11	11	11	11	11	0,009909	0,034676	0,065781	0,005775	0,009209
FAM510.dag	11	11	11	11	11	11	0,01255	0,027637	0,03242	0,004619	0,007519
FAM610.dag	14	13,5	13,3333	13,5	13,5	13,5	0,035505	0,065918	0,08459	0,007566	0,013145
FAM710.dag	12	12	12	12	12	12	0,034023	0,046642	0,049129	0,006931	0,0111

Tabela 5.1: Testes da relaxação linear com infinitos processadores - Grupo de Instâncias Aleatórias 1

DAG	IP	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
r1.dag	13	12,75	12,5	12,8333	13	13	0,092193	0,143966	0,232569	0,022885	0,034181
r2.dag	13	13	12,5	13	13	13	0,048562	0,129737	0,398211	0,017745	0,031848
ran1.dag	15	13,1786	13	14,25	14,25	14,25	0,102783	0,579066	0,522882	0,026715	0,051771
ran2.dag	13	12,4167	12,1818	12,5	12,5	12,5	0,031615	0,121082	0,109596	0,013169	0,026943
ran3.dag	14	13,1944	12,5	13,9	13,75	13,9	0,118169	1,17529	1,669495	0,045647	0,152704
ran4.dag	15	14	13,5	14,25	14,25	14,25	0,058836	0,273426	0,575851	0,026955	0,119335
ran5.dag	9	9	8,5	9	9	9	0,098773	0,084741	0,046968	0,006863	0,012008
ran6.dag	9	8,33333	8,33333	9	9	9	0,331315	0,094421	0,069006	0,010512	0,015912
ran7.dag	11	10,6	10,5	11	11	11	0,750145	0,732242	0,344612	0,029297	0,046209
ran8.dag	11	11	10,5	11	11	11	0,450122	0,624031	0,821984	0,023567	0,038907
ran9.dag	8	7	7	7,5	7,5	7,5	0,077018	0,043754	0,031119	0,00452	0,007212
ran10.dag	18	17	16,1667	17,25	17,25	17,25	0,699853	2,339317	1,465394	0,090041	0,281206
ran11.dag	18	17,3763	17,0667	17,7222	17,625	17,75	6,848749	11,281775	30,856796	0,2447	2,535573
ran12.dag	17	17	16,5	17	17	17	0,998688	0,980343	0,601894	0,05132	0,069282
ran13.dag	10	10	10	10	10	10	0,457924	0,192791	0,237536	0,018325	0,032307
ran14.dag	16	15	14,5	15,25	15,25	15,25	2,057157	1,207788	2,240204	0,040626	0,070565
ran15.dag	15	14	13,0833	14,25	14,25	14,25	0,111132	0,756925	1,295119	0,044208	0,065129
ran16.dag	11	10	10	11	11	11	0,18581	0,321794	0,163163	0,012871	0,025411
ran17.dag	21	21	21	21	21	21	7,906557	10,287749	13,121271	0,196289	0,631764
ran18.dag	15	14	13,5	14,25	14,25	14,25	0,468594	1,011277	2,122386	0,056017	0,267718
ran19.dag	11	10,5	10,5	11	11	11	0,118044	0,11661	0,138539	0,010836	0,021141
ran20.dag	9	9	9	9	9	9	0,126231	0,109445	0,147244	0,010939	0,018332

Tabela 5.2: Testes da relaxação linear com infinitos processadores - Grupo de Instâncias Aleatórias 2

DAG	IP	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
arv2.dag	5	4,33333	4,33333	5	5	5	0,00134	0,001441	0,001141	0,000837	0,000935
bin3.dag	5	5	3,5	5	5	5	0,001051	0,001142	0,000949	0,000733	0,001144
bin4.dag	7	7	4,5	7	7	7	0,001811	0,002568	0,002029	0,000987	0,001647
bin5.dag	9	9	5,5	9	9	9	0,003673	0,006746	0,00387	0,001419	0,002277
bin6.dag	11	11	6,5	11	11	11	0,021866	0,040229	0,00873	0,002822	0,004683
bin7.dag	13	13	7,5	13	13	13	0,067876	0,110441	0,033148	0,005708	0,009915
bin8.dag	15	15	8,5	15	15	15	0,217607	0,128271	0,092668	0,01197	0,023579
bin9.dag	17	17	9,5	17	17	17	1,312532	0,3687	0,295623	0,030986	0,054225
dag3.dag	8	8	7	8	8	8	0,00128	0,003354	0,002252	0,000918	0,001921
di16.dag	10	10	10	10	10	10	0,002084	0,003775	0,004542	0,001983	0,002999
di25.dag	13	13	13	13	13	13	0,003232	0,008679	0,010325	0,002918	0,003447
di36.dag	16	16	16	16	16	16	0,005757	0,022393	0,06906	0,003606	0,005682
di64.dag	22	22	22	22	22	22	0,030194	0,085949	0,395543	0,013233	0,019039
di100.dag	28	28	28	28	28	28	0,094164	0,198308	0,874862	0,034061	0,047207
di144.dag	34	34	34	34	34	34	0,193308	0,385544	1,572539	0,084553	0,091023
di225.dag	43	43	43	43	43	43	0,435329	0,852613	16,500571	0,213233	0,25534
di256.dag	46	46	46	46	46	46	0,755257	1,042509	53,259754	0,336679	0,403723
di400.dag	58	58	58	58	58	58	3,219967	2,465255	189,918383	1,128244	1,261767
gauss.dag	10	9,5	9,5	10	10	10	0,001819	0,004906	0,004277	0,001941	0,003333
irr41.dag	16	15,0833	15,05	15,3333	15,3333	15,4	0,008119	0,043737	0,066132	0,008309	0,032292
n7.dag	5	5	5	5	5	5	0,001036	0,001113	0,001089	0,000697	0,001107
n13.dag	5	5	4,8	5	5	5	0,001496	0,00173	0,001624	0,00094	0,001533
tree2.dag	5	4	4	5	5	5	0,001185	0,000992	0,001189	0,000744	0,001016
tree3.dag	7	6	6	7	7	7	0,001938	0,002008	0,002717	0,0011	0,001625
tree4.dag	9	8	8	9	9	9	0,003737	0,005606	0,005278	0,001423	0,002251
tree5.dag	11	10	10	11	11	11	0,022488	0,032322	0,015192	0,002937	0,004795
tree6.dag	13	12	12	13	13	13	0,078461	0,105599	0,051861	0,006675	0,010958
tree7.dag	15	14	14	15	15	15	0,416439	0,169713	0,212089	0,012047	0,023885
tree8.dag	17	16	16	17	17	17	3,924415	0,500516	0,684607	0,032952	0,056428
celbow.dag	30	28,5833	28,3333	28,625	28,625	28,6429	0,045356	0,168555	0,432301	0,022119	0,063817
cstanford.dag	29	28,0556	28	28,3333	28,25	28,3333	0,040333	0,1594	0,290275	0,018173	0,05732
fft2-b-m.dag	15	14	14	14,2188	14	14,2258	0,462137	0,400352	0,584282	0,046042	211,018157
iterative2-b-m.dag	14	13,9231	13,9231	13,9231	13	13,96	0,830868	0,145211	0,191483	0,028901	0,091804

Tabela 5.3: Testes da relaxação linear com infinitos processadores - Grupo de Instâncias Estruturadas

DAG	IP Z	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
FAM11.dag	10	0,091801	0,041839	0,041731	0,00254	0,006226
FAM12.dag	11	0,088067	0,044548	0,037833	0,004629	0,0115
FAM13.dag	12	0,047897	0,030575	0,030776	0,002306	0,006629
FAM14.dag	11	0,117635	0,058161	0,0523	0,002967	0,011311
FAM15.dag	10	0,010592	0,009781	0,061194	0,001913	0,005306
FAM16.dag	10	0,050434	0,017755	0,022018	0,002115	0,005384
FAM17.dag	10	0,087997	0,023648	0,066907	0,002598	0,005775
FAM18.dag	11	0,07893	0,034474	0,03539	0,002724	0,00924
FAM19.dag	10	0,062587	0,034559	0,039944	0,002379	0,005754
FAM21.dag	8	0,104899	0,022786	0,039391	0,002269	0,007328
FAM22.dag	11	0,099229	0,046297	0,052764	0,003445	0,009263
FAM23.dag	7	0,073196	0,021696	0,017586	0,001708	0,003507
FAM24.dag	10	0,095493	0,02403	0,035321	0,001985	0,004513
FAM25.dag	8	0,10784	0,036367	0,011508	0,002054	0,004516
FAM26.dag	10	0,030469	0,021195	0,023888	0,001979	0,004839
FAM27.dag	7	0,033181	0,005547	0,03572	0,001511	0,003306
FAM28.dag	9	0,105807	0,048602	0,017029	0,002246	0,004867
FAM29.dag	8	0,065884	0,019222	0,022863	0,001562	0,003781
FAM31.dag	11	0,327552	0,081481	0,101248	0,006428	0,014321
FAM32.dag	12	0,26741	0,174661	0,095776	0,009705	0,031077
FAM33.dag	12	0,255505	0,045001	0,099587	0,003247	0,009942
FAM34.dag	10	0,352719	0,064848	0,06341	0,002935	0,006923
FAM35.dag	11	0,484517	0,079997	0,110126	0,004221	0,013328
FAM36.dag	10	0,397246	0,04426	0,068229	0,004318	0,012867
FAM37.dag	11	0,146759	0,052537	0,055114	0,002979	0,008043
FAM38.dag	13	0,409655	0,096163	0,116597	0,004433	0,009439
FAM39.dag	9	0,282101	0,041801	0,074222	0,003363	0,007017
FAM41.dag	13	0,194154	0,08266	0,24302	0,004808	0,013265
FAM42.dag	11	0,366969	0,07284	0,17908	0,003862	0,010988
FAM43.dag	12	0,231816	0,159227	0,113637	0,006378	0,015038
FAM44.dag	12	0,293134	0,132939	0,289109	0,009683	0,030234
FAM45.dag	12	0,146842	0,024911	0,236178	0,003392	0,012318
FAM46.dag	13	0,123497	0,046615	0,061762	0,004084	0,012435
FAM47.dag	14	0,298737	0,125444	0,206992	0,006081	0,016129
FAM48.dag	15	0,307438	0,178848	0,089383	0,007383	0,01816
FAM49.dag	14	0,167609	0,066519	0,195111	0,004691	0,03028
FAM51.dag	13	0,598608	0,14167	0,127242	0,004559	0,012584
FAM52.dag	15	0,896174	0,14118	0,178282	0,006178	0,015682
FAM53.dag	16	0,717546	0,19945	0,160236	0,009833	0,069843
FAM54.dag	11	0,12884	0,031808	0,047994	0,003295	0,00993
FAM55.dag	10	1,351702	0,087955	0,173689	0,007503	0,017211
FAM56.dag	14	0,442363	0,121974	0,246621	0,004441	0,01258
FAM57.dag	11	0,396573	0,054521	0,05615	0,003786	0,010656
FAM58.dag	12	0,794121	0,096499	0,115755	0,006737	0,014168
FAM59.dag	12	1,052506	0,094516	0,104685	0,004627	0,026836
FAM61.dag	11	1,315518	0,120391	0,138259	0,007928	0,024422
FAM62.dag	15	1,463867	0,20189	0,262291	0,006544	0,017007
FAM63.dag	14	0,904788	0,17779	0,223065	0,007425	0,034918
FAM64.dag	17	1,431743	0,168577	0,317545	0,008553	0,021356
FAM65.dag	13	0,933354	0,174693	0,223034	0,011201	0,020716
FAM66.dag	12	1,177914	0,237963	0,183081	0,01009	0,056317
FAM67.dag	18	0,977988	0,254243	0,29717	0,010251	0,038727
FAM68.dag	13	1,338162	0,143819	0,166425	0,005984	0,015162
FAM69.dag	13	1,150547	0,139844	0,183644	0,007162	0,01638
FAM71.dag	16	2,486711	0,184275	0,252299	0,009259	0,023158
FAM72.dag	14	1,255048	0,469067	0,300071	0,016146	0,063093
FAM73.dag	13	1,187754	0,282615	0,29439	0,011366	0,038484
FAM74.dag	14	1,953515	0,234935	0,266612	0,010813	0,057887
FAM75.dag	15	0,55604	0,144754	0,188307	0,005592	0,015268
FAM76.dag	14	1,499333	0,181523	0,217571	0,009545	0,059586
FAM77.dag	10	0,568634	0,056603	0,228228	0,003844	0,012254
FAM78.dag	15	1,181942	0,180331	0,250898	0,006417	0,017525
FAM79.dag	13	1,714564	0,15492	0,304522	0,009619	0,018229
FAM110.dag	12	0,045334	0,040275	0,045682	0,002216	0,006262
FAM210.dag	10	0,081778	0,045469	0,046565	0,003065	0,008444
FAM310.dag	13	0,353486	0,060953	0,146703	0,005973	0,011314
FAM410.dag	11	0,166499	0,045786	0,117302	0,003464	0,011435
FAM510.dag	11	0,16693	0,067784	0,067595	0,003599	0,009778
FAM610.dag	14	1,194761	0,199933	0,20719	0,006014	0,016706
FAM710.dag	12	0,934492	0,126675	0,102004	0,00503	0,016488

Tabela 5.4: Testes do problema inteiro com infinitos processadores - Grupo de Instâncias Aleatórias 1

DAG	IP Z	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
r1.dag	13	3,157527	0,272739	0,713477	0,012285	0,0423
r2.dag	13	2,172162	0,200042	0,486859	0,012013	0,036385
ran1.dag	15	4,803129	0,485188	0,597121	0,01916	0,0595
ran2.dag	13	0,750693	0,173124	0,329744	0,00839	0,031165
ran3.dag	14	4,467663	1,675246	1,16302	0,041033	0,165759
ran4.dag	15	2,114302	0,352573	0,485264	0,024783	0,1261
ran5.dag	9	3,919751	0,152576	0,131872	0,005052	0,015493
ran6.dag	9	13,542721	0,467442	0,237101	0,007792	0,021065
ran7.dag	11	47,293298	1,370535	0,800132	0,019957	0,056949
ran8.dag	11	18,581407	1,279957	0,681487	0,021282	0,047365
ran9.dag	8	0,705978	0,134793	0,109071	0,003412	0,010258
ran10.dag	18	38,820845	2,37165	1,438354	0,067373	0,298769
ran11.dag	18	-	7,430245	6,908174	0,201511	2,573828
ran12.dag	17	69,1811	2,161262	1,449702	0,037519	0,08775
ran13.dag	10	8,301297	0,41584	0,68861	0,013225	0,041121
ran14.dag	16	134,656526	2,141907	1,413367	0,029766	0,085298
ran15.dag	15	5,979144	0,708992	0,851108	0,028363	0,072804
ran16.dag	11	2,355087	0,621952	0,439135	0,01004	0,031158
ran17.dag	21	277,635388	10,17932	5,127379	0,170874	0,67829
ran18.dag	15	22,841573	0,905583	1,245834	0,032709	0,279234
ran19.dag	11	1,438056	0,366117	0,328026	0,00814	0,025703
ran20.dag	9	5,209935	0,241612	0,237016	0,006813	0,022033

Tabela 5.5: Testes do problema inteiro com infinitos processadores - Grupo de Instâncias Aleatórias 2

DAG	IP Z	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
arv2.dag	5	0,011008	0,005044	0,001999	0,001081	0,002011
bin3.dag	5	0,007048	0,00735	0,002497	0,001051	0,002031
bin4.dag	7	0,017097	0,017731	0,008656	0,001253	0,00259
bin5.dag	9	0,070918	0,125698	0,021145	0,001345	0,003628
bin6.dag	11	0,316067	0,461938	0,049498	0,002048	0,007063
bin7.dag	13	1,888234	5,125019	0,12631	0,003587	0,01422
bin8.dag	15	8,755587	26,30281	0,31132	0,007238	0,032419
bin9.dag	17	131,138595	600,062012	0,841071	0,015621	0,071545
dag3.dag	8	0,012163	0,011305	0,008098	0,001163	0,003103
di16.dag	10	0,014262	0,011464	0,025715	0,002171	0,005172
di25.dag	13	0,027429	0,021996	0,059953	0,003104	0,007094
di36.dag	16	0,055858	0,052269	0,140673	0,005283	0,016507
di64.dag	22	0,19701	0,191557	0,463791	0,014719	0,046576
di100.dag	28	0,78204	0,476513	1,329965	0,027183	0,127361
di144.dag	34	2,744347	1,096648	2,898533	0,054256	0,330957
di225.dag	43	13,530381	3,021878	6,885857	0,12135	1,33334
di256.dag	46	21,83101	4,142563	7,795472	0,154204	1,778554
di400.dag	58	108,233592	10,987562	20,745302	0,439668	6,888414
gauss.dag	10	0,015662	0,011401	0,014737	0,001448	0,004649
irr41.dag	16	0,147072	0,154559	0,101112	0,010902	0,040111
n7.dag	5	0,007249	0,002529	0,00352	0,001065	0,001901
n13.dag	5	0,012606	0,004029	0,003015	0,001145	0,002369
tree2.dag	5	0,007797	0,003606	0,002747	0,001078	0,001769
tree3.dag	7	0,023087	0,010753	0,00723	0,001293	0,002583
tree4.dag	9	0,159537	0,017837	0,022148	0,00135	0,003626
tree5.dag	11	0,488801	0,124857	0,043194	0,002117	0,007157
tree6.dag	13	4,739997	0,158957	0,172979	0,00363	0,015177
tree7.dag	15	32,819634	0,29632	0,396489	0,008669	0,032393
tree8.dag	17	-	0,903585	1,060615	0,016041	0,076142
celbow.dag	30	1,32079	0,853497	0,732067	0,016788	0,073033
cstanford.dag	29	1,046224	0,269957	0,567983	0,01305	0,065821
fft2-b-m.dag	15	394,511007	7,200583	0,832436	0,105718	211,051724
iterative2-b-m.dag	14	3,144411	0,188248	0,33671	0,030766	0,099682

Tabela 5.6: Testes do problema inteiro com infinitos processadores - Grupo de Instâncias Estruturadas

DAG	IP	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
FAM11.dag	10	10	10	10	10	10	0,006472	0,028249	0,02036	0,004794	0,007068
FAM12.dag	11	10,1515	10,0123	10,5	10,3333	10,5	0,0117	0,037299	0,02135	0,005843	0,01122
FAM13.dag	12	12	12	12	12	12	0,006413	0,023463	0,015613	0,0043	0,010961
FAM14.dag	11	10	10	10,125	10,125	10,125	0,007416	0,029772	0,0173	0,003868	0,016748
FAM15.dag	10	10	10	10	10	10	0,003451	0,008613	0,011626	0,002533	0,005734
FAM16.dag	10	10	10	10	10	10	0,004664	0,013765	0,019208	0,00271	0,005745
FAM17.dag	10	10	10	10	10	10	0,008512	0,020954	0,02024	0,003916	0,005171
FAM18.dag	11	11	11	11	11	11	0,009287	0,019456	0,035797	0,003444	0,005231
FAM19.dag	10	10	9,33333	10	10	10	0,00585	0,020919	0,011574	0,003364	0,004713
FAM21.dag	8	7,3	7,25	7,4	7,4	7,41176	0,006527	0,010161	0,012857	0,002842	0,013212
FAM22.dag	11	10,0313	10	10,1818	10,1818	10,2	0,009493	0,016225	0,019093	0,005162	0,012127
FAM23.dag	7	7	7	7	7	7	0,008432	0,005735	0,004231	0,00176	0,006172
FAM24.dag	10	10	9,16667	10	10	10	0,006719	0,010288	0,006786	0,002418	0,004627
FAM25.dag	8	8	8	8	8	8	0,007106	0,009646	0,004162	0,004094	0,004464
FAM26.dag	10	10	10	10	10	10	0,004249	0,006248	0,004827	0,002388	0,004668
FAM27.dag	7	7	7	7	7	7	0,006083	0,007659	0,00693	0,003749	0,005227
FAM28.dag	9	9	9	9	9	9	0,008617	0,011047	0,004308	0,003312	0,005517
FAM29.dag	8	8	8	8	8	8	0,009687	0,005148	0,004911	0,002171	0,003853
FAM31.dag	11	10,2143	10,125	10,3333	10,3333	10,3333	0,041462	0,069743	0,047171	0,005843	0,022511
FAM32.dag	12	11,5	11,1667	11,8333	11,75	11,8333	0,025514	0,051662	0,049453	0,012133	0,053641
FAM33.dag	12	12	12	12	12	12	0,030348	0,025516	0,039673	0,004037	0,011444
FAM34.dag	10	10	9	10	10	10	0,031533	0,035679	0,016806	0,003878	0,010338
FAM35.dag	11	10,5	10,3	10,75	10,75	10,8	0,021385	0,047255	0,043296	0,00687	0,016424
FAM36.dag	10	9,625	9,33333	9,66667	9,66667	9,66667	0,029038	0,033923	0,029879	0,006393	0,023336
FAM37.dag	11	10,5	10,5	10,5	10,5	10,5	0,022976	0,032921	0,02411	0,003982	0,006039
FAM38.dag	13	12,25	12,0526	12,3333	12,3333	12,3333	0,0157	0,036493	0,047196	0,006262	0,016162
FAM39.dag	9	8,44444	8,33333	8,66667	8,66667	8,66667	0,025424	0,023181	0,021059	0,004809	0,006523
FAM41.dag	13	13	13	13	13	13	0,020955	0,044225	0,10818	0,009215	0,013357
FAM42.dag	11	11	11	11	11	11	0,021325	0,045403	0,07061	0,008045	0,020528
FAM43.dag	12	12	12	12	12	12	0,019024	0,06182	0,081512	0,012132	0,0355
FAM44.dag	12	12	12	12	12	12	0,018593	0,071262	0,128117	0,014511	0,035777
FAM45.dag	12	12	12	12	12	12	0,024852	0,02047	0,099652	0,006781	0,01076
FAM46.dag	13	13	13	13	13	13	0,01381	0,032397	0,034854	0,008237	0,012367
FAM47.dag	14	13	13	13,25	13,25	13,25	0,014891	0,065609	0,098923	0,01223	0,0509
FAM48.dag	15	15	15	15	15	15	0,017193	0,100593	0,079613	0,017037	0,044936
FAM49.dag	14	14	14	14	14	14	0,019493	0,050238	0,097646	0,010147	0,027492
FAM51.dag	13	13	12,5	13	13	13	0,050408	0,055381	0,052914	0,009737	0,01338
FAM52.dag	15	14,25	14,1111	14,3333	14,3333	14,3333	0,060271	0,072333	0,078438	0,016548	0,020609
FAM53.dag	16	15,3889	15	15,2857	15,28	15,5	0,098462	0,073596	0,094522	0,020314	0,049102
FAM54.dag	11	11	11	11	11	11	0,031452	0,022482	0,020085	0,007299	0,009366
FAM55.dag	10	10	9,25	10	10	10	0,097815	0,086066	0,114313	0,018089	0,046202
FAM56.dag	14	14	14	14	14	14	0,045131	0,050936	0,078419	0,008764	0,013208
FAM57.dag	11	11	11	11	11	11	0,040643	0,025227	0,031011	0,007472	0,010524
FAM58.dag	12	12	12	12	12	12	0,048682	0,038642	0,059834	0,014178	0,052785
FAM59.dag	12	11,5	11,3333	11,6667	11,5	11,6667	0,079453	0,049719	0,065348	0,013432	0,030807
FAM61.dag	11	10,3333	10,2	10,5	10,5	10,5	0,067422	0,067466	0,070163	0,014778	0,040326
FAM62.dag	15	14,25	14,125	14,3333	14,3333	14,3333	0,057478	0,065297	0,078811	0,018319	0,05128
FAM63.dag	14	13,2	13	13,3333	13,2	13,3333	0,06001	0,069253	0,096061	0,017567	0,073862
FAM64.dag	17	16,6667	16,1667	16,8333	17	17	0,064177	0,09561	0,202336	0,021347	0,150758
FAM65.dag	13	12,1667	12	12,3125	12,3125	12,3125	0,065576	0,076829	0,152874	0,029761	0,094069
FAM66.dag	12	11,1503	11	11,2273	11	11,2667	0,105045	0,081349	0,11617	0,024096	0,183127
FAM67.dag	18	17,0769	17	17,1429	17,1429	17,1667	0,076104	0,104989	0,150451	0,032058	0,146149
FAM68.dag	13	12,375	12,25	12,6667	12,6667	12,6667	0,063739	0,052331	0,094428	0,016775	0,114562
FAM69.dag	13	12,5	12,3333	12,5	12,5	12,5	0,06174	0,049374	0,080424	0,016507	0,083641
FAM71.dag	16	15,5	15	15,75	15,75	15,75	0,071298	0,114245	0,138142	0,020367	0,063353
FAM72.dag	14	13,4808	13	13,6667	13,6667	13,6667	0,180173	0,167968	0,183623	0,030054	0,150303
FAM73.dag	13	12,125	12,0303	12,125	12	12,1667	0,082434	0,134194	0,161381	0,017041	0,101295
FAM74.dag	14	13,5	13,1667	13,6667	13,6667	13,6667	0,127746	0,111471	0,151218	0,026094	0,089526
FAM75.dag	15	15	15	15	15	15	0,052313	0,050457	0,058991	0,013842	0,035621
FAM76.dag	14	13,5263	13,25	13,75	13,5	13,75	0,105779	0,127969	0,135753	0,019692	0,101835
FAM77.dag	10	10	10	10	10	10	0,068175	0,052269	0,088219	0,007719	0,013197
FAM78.dag	15	15	15	15	15	15	0,073504	0,063929	0,117459	0,019604	0,152897
FAM79.dag	13	12,4286	12,125	12,5	12,5	12,5	0,083186	0,079346	0,150095	0,020794	0,024232
FAM110.dag	12	12	11,1667	12	12	12	0,004925	0,026275	0,0288	0,00417	0,00504
FAM210.dag	10	9,08333	9	9,5	9,5	9,5	0,007573	0,011974	0,009591	0,003813	0,010992
FAM310.dag	13	13	13	13	13	13	0,019181	0,031721	0,053386	0,004544	0,018932
FAM410.dag	11	11	11	11	11	11	0,020724	0,040236	0,06917	0,007646	0,010189
FAM510.dag	11	11	11	11	11	11	0,03739	0,030102	0,034991	0,008708	0,010459
FAM610.dag	14	13,5	13,3333	13,5	13,5	13,5	0,061201	0,075218	0,082301	0,016048	0,044614
FAM710.dag	12	12	12	12	12	12	0,073789	0,045009	0,057795	0,010201	0,014101

Tabela 5.7: Testes da relaxação linear com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
r1.dag	13	12,75	12,5	12,8333	13	13	0,21014	0,141327	0,707117	0,026052	0,037797
r2.dag	13	13	12,5	13	13	13	0,114352	0,173739	0,387377	0,024365	0,068936
ran1.dag	15	13,1786	13	14,25	14,25	14,25	0,202576	0,926224	0,54276	0,056606	0,22513
ran2.dag	13	12,4167	12,1818	12,5	12,5	12,5	0,054665	0,105979	0,156388	0,02067	0,03046
ran3.dag	14	13,1944	12,5	13,9	13,75	13,9	0,373551	2,051527	2,533475	0,352284	1,168632
ran4.dag	15	14	13,5	14,25	14,25	14,25	0,084621	0,546461	0,690045	0,056209	0,298534
ran5.dag	9	9	8,5	9	9	9	0,440476	0,091415	0,055203	0,013996	0,053487
ran6.dag	9	8,33333	8,33333	9	9	9	0,552196	0,112347	0,070584	0,02077	0,079649
ran7.dag	11	10,6	10,5	11	11	11	1,176184	1,289709	0,295586	0,107376	0,691934
ran8.dag	11	11	10,5	11	11	11	0,761324	0,858696	2,587605	0,10092	0,35023
ran9.dag	8	7	7	7,5	7,5	7,5	0,197623	0,046597	0,033709	0,009005	0,031932
ran10.dag	18	17	16,1667	17,25	17,25	17,25	5,984773	5,203031	0,991406	0,515006	5,513443
ran11.dag	18	17,3763	17,0667	17,7222	17,625	17,75	26,132651	23,016495	87,15024	1,349354	28,221585
ran12.dag	17	17	16,5	17	17	17	2,38772	3,694098	0,525199	0,173431	1,020059
ran13.dag	10	10	10	10	10	10	3,584274	0,271172	0,635627	0,036026	0,096572
ran14.dag	16	15	14,5	15,25	15,25	15,25	2,596047	2,652117	5,525827	0,138256	2,332949
ran15.dag	15	14	13,0833	14,25	14,25	14,25	0,264256	1,395815	1,233104	0,075431	0,419446
ran16.dag	11	10	10	11	11	11	0,343007	0,372179	0,193278	0,027931	0,068749
ran17.dag	21	21	21	21	21	21	10,171253	24,546768	30,474115	0,566816	14,186885
ran18.dag	15	14	13,5	14,25	14,25	14,25	0,821659	2,229791	3,225759	0,135286	1,27924
ran19.dag	11	10,5	10,5	11	11	11	0,211437	0,194768	0,086755	0,020405	0,055626
ran20.dag	9	9	9	9	9	9	0,296007	0,127117	0,151116	0,015952	0,047502

Tabela 5.8: Testes da relaxação linear com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
arv2.dag	6	6	6	6	6	6	0,00153	0,001466	0,001749	0,001177	0,001706
bin3.dag	5	5	3,5	5	5	5	0,001355	0,001092	0,001302	0,000794	0,001195
bin4.dag	7	7	4,5	7	7	7	0,003776	0,002536	0,001748	0,001355	0,001991
bin5.dag	9	9	5,5	9	9	9	0,006785	0,006464	0,004188	0,002257	0,003537
bin6.dag	11	11	6,5	11	11	11	0,132422	0,057474	0,0147	0,003177	0,005114
bin7.dag	13	13	7,5	13	13	13	0,226714	0,118638	0,067536	0,010381	0,015101
bin8.dag	15	15	8,5	15	15	15	2,734692	0,16146	0,198655	0,022212	0,032316
bin9.dag	17	17	9,5	17	17	17	19,060061	0,368938	0,617652	0,05048	0,071089
dag3.dag	8	8	8	8	8	8	0,002763	0,002598	0,002985	0,001491	0,001994
di16.dag	10	10	10	10	10	10	0,003539	0,004191	0,004232	0,002065	0,003089
di25.dag	15	15	15	15	15	15	0,007837	0,026792	0,022808	0,004875	0,005812
di36.dag	16	16	16	16	16	16	0,015216	0,02817	0,062122	0,004034	0,007968
di64.dag	22	22	22	22	22	22	0,108165	0,113342	0,329893	0,016518	0,022552
di100.dag	28	28	28	28	28	28	0,803546	0,346494	1,552071	0,039096	0,051873
di144.dag	34	34	34	34	34	34	2,533046	0,512348	4,453031	0,091265	0,124668
di225.dag	45	43	43	43,4072	43,4072	43,4072	12,54887	1,87243	376,923213	0,886632	0,919533
di256.dag	46	46	46	46	46	46	12,578372	1,31151	88,922095	0,436269	0,490915
di400.dag	58	58	58	21,407983	58	58	65,801433	3,454256	603,166599	1,62753	1,724035
gauss.dag	10	10	9,5	10	10	10	0,003704	0,005208	0,004273	0,002373	0,003747
irr41.dag	16	15,0833	15,05	15,3333	15,3333	15,4	0,025597	0,047647	0,057465	0,012597	0,067503
n7.dag	5	5	5	5	5	5	0,001161	0,001301	0,00102	0,000999	0,001342
n13.dag	5	5	4,8	5	5	5	0,002352	0,00181	0,001369	0,00124	0,001828
tree2.dag	5	5	5	5	5	5	0,001165	0,001061	0,000849	0,000887	0,001142
tree3.dag	7	6	6	7	7	7	0,003141	0,002825	0,001885	0,001555	0,002116
tree4.dag	9	8	8	9	9	9	0,007161	0,009535	0,005487	0,002979	0,004293
tree5.dag	11	10	10	11	11	11	0,15172	0,047216	0,016685	0,006591	0,008575
tree6.dag	13	12	12	13	13	13	0,155705	0,126802	0,075542	0,029233	0,038744
tree7.dag	15	14	14	15	15	15	0,674318	0,224388	0,265897	0,090375	0,101188
tree8.dag	17	16	16	17	17	17	4,910372	0,459949	0,754988	0,32119	0,341209
celbow.dag	30	28,5833	28,3333	28,625	28,625	28,6429	0,112846	0,204168	0,380534	0,042373	0,113857
cstanford.dag	29	28,0556	28	28,3333	28,25	28,3333	0,068706	0,171376	0,272063	0,017615	0,156544
fft2-b-m.dag	16	16	16	16	16	16	4,299853	0,59909	0,739812	0,063042	67,762582
iterative2-b-m.dag	24	24	24	24	24	24	1,207406	0,95908	0,845734	0,105005	0,283107

Tabela 5.9: Testes da relaxação linear com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	IP Z	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
FAM11.dag	10	0,39476	0,06776	0,085902	0,014311	0,023278
FAM12.dag	11	0,158704	0,052801	0,053401	0,009644	0,021835
FAM13.dag	12	0,04511	0,029878	0,03807	0,00384	0,019299
FAM14.dag	11	0,172024	0,061335	0,054713	0,01127	0,0277
FAM15.dag	10	0,019297	0,013579	0,065197	0,004191	0,008996
FAM16.dag	10	0,081817	0,021244	0,0603	0,003327	0,011345
FAM17.dag	10	0,135333	0,029378	0,06397	0,00474	0,012503
FAM18.dag	11	0,12436	0,03996	0,085233	0,008831	0,017481
FAM19.dag	10	0,181022	0,036343	0,035153	0,006436	0,013509
FAM21.dag	8	0,141499	0,032086	0,054001	0,007677	0,019687
FAM22.dag	11	0,19922	0,054665	0,072102	0,011021	0,023329
FAM23.dag	7	0,119847	0,022529	0,019987	0,005148	0,013217
FAM24.dag	10	0,158197	0,03372	0,038236	0,007772	0,009786
FAM25.dag	8	0,188528	0,035199	0,012539	0,011847	0,012613
FAM26.dag	10	0,033169	0,020448	0,029104	0,003209	0,007023
FAM27.dag	7	0,395925	0,047889	0,042157	0,009655	0,013142
FAM28.dag	9	0,149027	0,062276	0,020162	0,014638	0,016051
FAM29.dag	8	0,111876	0,026155	0,026931	0,00416	0,009848
FAM31.dag	11	0,532481	0,099305	0,124557	0,024777	0,038483
FAM32.dag	12	0,931241	0,208132	0,139696	0,033656	0,078853
FAM33.dag	12	0,344508	0,057654	0,089861	0,006873	0,023129
FAM34.dag	10	0,790278	0,088854	0,079274	0,016213	0,024008
FAM35.dag	11	0,651895	0,107636	0,125512	0,020707	0,037999
FAM36.dag	10	0,480797	0,059582	0,083477	0,018632	0,038699
FAM37.dag	11	0,300355	0,062561	0,065095	0,004147	0,011964
FAM38.dag	13	0,631166	0,073765	0,152575	0,02213	0,02453
FAM39.dag	9	0,453654	0,047658	0,104516	0,010017	0,012678
FAM41.dag	13	0,362856	0,111268	0,255221	0,019222	0,027487
FAM42.dag	11	0,527577	0,075247	0,221856	0,006211	0,02955
FAM43.dag	12	0,527629	0,179231	0,171275	0,028488	0,043606
FAM44.dag	12	0,809576	0,175728	0,276487	0,02784	0,052981
FAM45.dag	12	0,247528	0,02947	0,201074	0,008295	0,019825
FAM46.dag	13	0,261962	0,049477	0,079755	0,011386	0,021819
FAM47.dag	14	0,683404	0,13263	0,239479	0,014294	0,068377
FAM48.dag	15	0,665834	0,207225	0,10699	0,04442	0,072291
FAM49.dag	14	0,292909	0,079253	0,212487	0,009677	0,04336
FAM51.dag	13	1,045705	0,148142	0,137016	0,025973	0,035927
FAM52.dag	15	1,234683	0,154905	0,226272	0,023694	0,048347
FAM53.dag	16	1,720844	0,158161	0,1757	0,032114	0,071656
FAM54.dag	11	0,32992	0,036009	0,053228	0,007829	0,015104
FAM55.dag	10	1,593298	0,301619	0,174671	0,047602	0,078654
FAM56.dag	14	0,618279	0,114462	0,342405	0,007474	0,02906
FAM57.dag	11	0,609007	0,073172	0,077811	0,017326	0,024056
FAM58.dag	12	1,264995	0,106826	0,154816	0,029975	0,068462
FAM59.dag	12	1,44591	0,129127	0,158262	0,029513	0,059447
FAM61.dag	11	2,037812	0,156988	0,180617	0,028777	0,059876
FAM62.dag	15	2,970053	0,193983	0,284423	0,047976	0,069884
FAM63.dag	14	1,147037	0,213739	0,269247	0,045879	0,087538
FAM64.dag	17	2,862713	0,183678	0,432573	0,023527	0,165887
FAM65.dag	13	2,135611	0,297076	0,376011	0,055062	0,121282
FAM66.dag	12	3,547954	0,237961	0,205612	0,039492	0,213897
FAM67.dag	18	1,468593	0,309414	0,34952	0,047119	0,18208
FAM68.dag	13	1,718514	0,160589	0,23069	0,034657	0,145001
FAM69.dag	13	2,468648	0,137949	0,221248	0,030742	0,106947
FAM71.dag	16	3,969768	0,191254	0,292553	0,057085	0,102881
FAM72.dag	14	2,285718	0,455594	0,358072	0,087192	0,206492
FAM73.dag	13	3,665315	0,324763	0,386874	0,077293	0,124277
FAM74.dag	14	3,573749	0,205581	0,310483	0,045449	0,126037
FAM75.dag	15	1,690716	0,150158	0,207435	0,04063	0,065323
FAM76.dag	14	3,069078	0,185327	0,257679	0,046954	0,127472
FAM77.dag	10	0,897318	0,064705	0,274528	0,006228	0,018882
FAM78.dag	15	1,815011	0,215089	0,268193	0,050967	0,177677
FAM79.dag	13	4,47043	0,182247	0,356786	0,046949	0,05716
FAM110.dag	12	0,073908	0,042539	0,056639	0,003611	0,013699
FAM210.dag	10	0,196366	0,042755	0,056929	0,015284	0,02184
FAM310.dag	13	0,598222	0,060382	0,156354	0,014588	0,032223
FAM410.dag	11	0,220037	0,056897	0,160671	0,009731	0,019379
FAM510.dag	11	0,500516	0,083899	0,096854	0,014276	0,02413
FAM610.dag	14	2,43977	0,193692	0,221994	0,04353	0,080155
FAM710.dag	12	1,519712	0,082393	0,128483	0,019417	0,03391

Tabela 5.10: Testes do problema inteiro com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP Z	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
r1.dag	13	6,739535	0,302026	0,865481	0,02022	0,049966
r2.dag	13	4,152873	0,243674	0,550613	0,023978	0,09905
ran1.dag	15	9,910611	0,568702	0,717374	0,075282	0,243629
ran2.dag	13	1,469947	0,18265	0,415862	0,021228	0,045344
ran3.dag	14	28,008894	3,062808	1,632011	0,278714	1,326922
ran4.dag	15	7,650089	0,408657	0,585532	0,070765	0,329319
ran5.dag	9	12,50746	0,219717	0,222901	0,048127	0,088563
ran6.dag	9	50,933572	0,511696	0,295304	0,052966	0,120767
ran7.dag	11	-	1,64488	0,994788	0,176888	0,826345
ran8.dag	11	76,931947	1,559577	1,198483	0,185923	0,448176
ran9.dag	8	6,513521	0,134451	0,113023	0,029896	0,064598
ran10.dag	18	600,081226	3,494185	1,608956	0,413528	5,558724
ran11.dag	18	600,495581	11,584192	17,141459	1,358473	28,811129
ran12.dag	17	482,517286	2,757941	2,113065	0,231883	1,163651
ran13.dag	10	229,768982	0,710768	1,036239	0,033033	0,131755
ran14.dag	16	376,801294	2,522785	2,60637	0,165646	2,431797
ran15.dag	15	15,715879	0,765177	1,118955	0,084595	0,440682
ran16.dag	11	26,700634	0,533687	0,501728	0,040085	0,150438
ran17.dag	21	600,45585	17,207189	8,392581	1,144711	14,880345
ran18.dag	15	71,99724	1,215163	1,346166	0,164837	1,364711
ran19.dag	11	6,092879	0,448318	0,430102	0,016152	0,09478
ran20.dag	9	12,472415	0,230727	0,382218	0,023632	0,085152

Tabela 5.11: Testes do problema inteiro com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP Z	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
arv2.dag	6	0,018018	0,006453	0,00765	0,005448	0,005563
bin3.dag	5	0,008558	0,006743	0,006075	0,001358	0,004299
bin4.dag	7	0,023391	0,018117	0,011142	0,005085	0,007355
bin5.dag	9	0,22986	0,072057	0,029203	0,008709	0,014344
bin6.dag	11	1,051851	0,296832	0,054807	0,017116	0,024247
bin7.dag	13	8,839866	3,742628	0,134445	0,034416	0,053005
bin8.dag	15	65,166976	44,791021	0,425136	0,069636	0,104395
bin9.dag	17	600,480939	600,049903	1,060046	0,167133	0,256051
dag3.dag	8	0,011961	0,008272	0,018342	0,003058	0,005658
di16.dag	10	0,01492	0,009837	0,024912	0,003055	0,007701
di25.dag	15	0,10598	0,051115	0,05557	0,018277	0,022151
di36.dag	16	0,046588	0,050272	0,115864	0,012222	0,020158
di64.dag	22	0,19626	0,175149	0,462279	0,046092	0,086591
di100.dag	28	0,785869	0,476904	1,209133	0,146417	0,301467
di144.dag	34	2,540638	1,218334	2,198602	0,506152	0,595109
di225.dag	45	600,098019	600,153501	600,435748	23,410749	3,373825
di256.dag	46	21,029919	4,627172	7,532285	1,851894	2,224078
di400.dag	58	103,843418	10,975311	58	4,897076	6,066537
gauss.dag	10	0,026589	0,014424	0,016805	0,002525	0,009737
irr41.dag	16	0,19076	0,101125	0,139745	0,035206	0,083231
n7.dag	5	0,006206	0,004471	0,003837	0,001309	0,00233
n13.dag	5	0,0121	0,004222	0,002803	0,001473	0,004782
tree2.dag	5	0,008091	0,005599	0,004296	0,001385	0,004046
tree3.dag	7	0,029358	0,012998	0,008552	0,005398	0,007035
tree4.dag	9	0,313501	0,045104	0,02766	0,011044	0,015404
tree5.dag	11	1,894304	0,116142	0,057204	0,015888	0,032867
tree6.dag	13	14,421025	0,423503	0,156767	0,036132	0,09493
tree7.dag	15	600,192583	0,947877	0,403037	0,142287	0,260167
tree8.dag	17	600,317809	3,717022	0,973843	0,479568	0,846622
celbow.dag	30	2,464958	0,758648	0,890324	0,030989	0,133023
cstanford.dag	29	1,791314	0,348603	0,663223	0,014698	0,165531
fft2-b-m.dag	16	600,043315	1,98988	0,560726	0,256095	68,061916
iterative2-b-m.dag	24	600,212526	0,403635	1,171759	0,207708	0,474091

Tabela 5.12: Testes do problema inteiro com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	IP	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
FAM12.dag	16	16	15,2222	16	16	16	0,005978	0,079087	0,06663	0,030569	0,15134
FAM13.dag	16	16	14,4286	16	16	16	0,007544	0,052083	0,060245	0,015027	0,033776
FAM14.dag	15	15	14,5	15	15	15	0,00663	0,068331	0,078479	0,008221	0,010699
FAM15.dag	15	15	15	15	15	15	0,006105	0,09354	0,112744	0,023549	0,052003
FAM16.dag	15	15	15	15	15	15	0,005876	0,045893	0,069288	0,008024	0,011313
FAM17.dag	15	15	15	15	15	15	0,005757	0,050282	0,091225	0,007862	0,011124
FAM18.dag	16	16	16	16	16	16	0,005944	0,064523	0,067071	0,019192	0,05916
FAM19.dag	15	15	15	15	15	15	0,005914	0,047436	0,119839	0,009703	0,012527
FAM21.dag	15	15	12	15	15	15	0,006543	0,058087	0,048445	0,009402	0,02317
FAM22.dag	16	16	13,3	16	16	16	0,006289	0,070339	0,060811	0,012855	0,047058
FAM23.dag	10	10	7,8	10	10	10	0,007062	0,014192	0,010994	0,004713	0,015314
FAM24.dag	15	15	11,5	15	15	15	0,005926	0,059273	0,025147	0,008346	0,021119
FAM25.dag	11	11	8,45455	11	11	11	0,006087	0,015452	0,009732	0,004303	0,014914
FAM26.dag	15	15	11,3333	15	15	15	0,005915	0,05011	0,040649	0,009932	0,012704
FAM27.dag	15	15	9,41379	15	15	15	0,005924	0,059162	0,037947	0,013925	0,014288
FAM28.dag	15	15	11,3478	15	15	15	0,00849	0,052478	0,020433	0,011434	0,03604
FAM29.dag	15	15	13,4286	15	15	15	0,007496	0,039749	0,022587	0,009388	0,00878
FAM31.dag	14	14	12,3333	14	14	14	0,022005	0,084367	0,168939	0,021785	0,093207
FAM32.dag	16	16	16	16	16	16	0,023277	0,100305	0,085807	0,027467	0,133186
FAM33.dag	14	14	13,1818	13,3529	14	14	0,019078	0,067073	0,130848	0,010249	0,044975
FAM34.dag	14	13	9,42857	13,2	13	13,2	0,020865	0,117651	0,198255	0,01294	0,07628
FAM35.dag	14	14	14	14	14	14	0,0181	0,11806	0,055826	0,026357	0,237762
FAM36.dag	17	17	12,5	17	17	17	0,017155	0,198432	0,317178	0,027467	0,064156
FAM37.dag	12	11	10,5	11,2222	11,0938	11,2353	0,023327	0,059607	0,095735	0,00901	0,030426
FAM38.dag	14	14	12,6667	13,4	14	14	0,023385	0,062532	0,079369	0,009669	0,043134
FAM39.dag	13	13	10,5	13	13	13	0,019619	0,079939	0,116533	0,011008	0,036454
FAM41.dag	18	18	16,8	17,3333	18	18	0,014867	0,361413	0,547574	0,053629	0,422651
FAM42.dag	13	13	12,4	13	13	13	0,019891	0,093728	0,154283	0,025405	0,057833
FAM43.dag	13	13	13	13	13	13	0,016807	0,116718	0,12621	0,022875	0,101667
FAM44.dag	18	18	15,3714	18	18	18	0,019356	0,276949	0,397216	0,0527	0,451463
FAM45.dag	19	19	16,3333	19	19	19	0,026705	0,396861	0,441942	0,060368	0,080883
FAM46.dag	18	18	17,3333	17,3372	18	18	0,014769	0,300089	0,975594	0,063355	0,358318
FAM47.dag	18	18	17	18	18	18	0,014775	0,243987	0,188753	0,033266	0,154972
FAM48.dag	16	16	16	16	16	16	0,020846	0,181885	0,199459	0,035551	0,157699
FAM49.dag	18	18	18	18	18	18	0,020011	0,388859	0,564551	0,07109	0,620776
FAM51.dag	16	15	12,5	14,8621	15	15,1905	0,052428	0,164925	0,442483	0,033843	0,137938
FAM52.dag	17	17	17	17	17	17	0,051248	0,133723	0,242421	0,036555	0,203326
FAM53.dag	17	17	17	17,1185	17	17	0,054322	0,214893	0,272105	0,031144	0,176579
FAM54.dag	13	13	11,7	12,2474	13	13	0,047032	0,104224	0,2209	0,024512	0,075493
FAM55.dag	14	13	10,4286	13,0105	13	13,3158	0,065348	0,213446	0,295068	0,039739	0,227323
FAM56.dag	14	14	14	14	14	14	0,056155	0,161065	0,126135	0,02728	0,168218
FAM57.dag	14	13	11,3333	12,7317	13	13,2105	0,050038	0,207247	0,29856	0,025154	0,147546
FAM58.dag	13	12	12,1176	12,2162	12	12,3333	0,05916	0,085855	0,181664	0,026269	0,25224
FAM59.dag	14	13	12,2632	13,5344	13,274	13,6018	0,05518	0,198864	0,574311	0,040176	0,460842
FAM61.dag	13	13	11,5	12,5	13	13	0,069675	0,158809	0,291389	0,024146	0,087843
FAM62.dag	18	17	14,4493	16,7111	17	17,1212	0,063732	0,243511	0,611952	0,043207	0,264077
FAM63.dag	14	13,2	13	13,3333	13,2	13,3333	0,073479	0,099913	0,129582	0,024159	0,131975
FAM64.dag	18	17	17	17,1851	17,0769	17,2727	0,075483	0,257465	0,638639	0,043621	0,417688
FAM65.dag	16	15	15,0698	15,3695	15,0863	15,4494	0,066134	0,458682	1,091019	0,063464	0,397147
FAM66.dag	16	15	13,3565	15,1644	15	15,1967	0,067916	0,435952	0,740899	0,043194	1,094485
FAM67.dag	18	17,0769	17	17,1429	17,1429	17,1667	0,091469	0,154306	0,235501	0,044649	0,327272
FAM68.dag	16	15	12,4095	14,8889	15	15,1818	0,054416	0,21834	0,391295	0,054349	0,177812
FAM69.dag	14	13	12,3333	12,9342	13	13,1345	0,065384	0,142378	0,24637	0,030954	0,163548
FAM71.dag	16	15,5	15	15,75	15,75	15,75	0,080123	0,326817	0,341201	0,045472	0,311679
FAM72.dag	15	13,4808	13,1077	13,807	13,6852	13,8081	0,226573	0,400233	0,412714	0,067309	0,290406
FAM73.dag	14	13	12,5	12,9943	13,0417	13,4286	0,074092	0,282128	0,796339	0,046382	0,371082
FAM74.dag	17	15	14,1875	15,6364	15,3659	15,7273	0,093055	0,473648	0,907176	0,084059	0,386557
FAM75.dag	15	15	15	15	15	15	0,059656	0,088626	0,097894	0,020563	0,052132
FAM76.dag	14	13,5263	13,25	13,9414	13,5	13,9273	0,120688	0,252784	0,323016	0,046445	0,341882
FAM77.dag	13	13	10,3061	12,5005	13	13	0,082457	0,382283	0,598818	0,035992	0,134691
FAM78.dag	16	15	15,12	15,7163	15,574	15,7354	0,09234	0,453488	0,995315	0,078217	1,111368
FAM79.dag	14	13	12,1618	13,0729	13,1569	13,3774	0,079462	0,398707	0,601406	0,069225	0,212768
FAM110.dag	16	16	14,4286	16	16	16	0,005722	0,063841	0,105145	0,017479	0,018862
FAM210.dag	11	11	11	11	11	11	0,006903	0,021317	0,026511	0,007413	0,022934
FAM310.dag	16	15	15	15,0615	15	15,1538	0,019407	0,128245	0,134397	0,024468	0,138431
FAM410.dag	18	18	16,2857	17,1538	18	18	0,014419	0,367897	0,448496	0,073763	0,161074
FAM510.dag	13	13	12,5	12,4783	13	13	0,048019	0,126545	0,43326	0,021458	0,296144
FAM610.dag	16	15	14,2059	14,8276	15	15,129	0,064283	0,277898	0,559678	0,034831	0,272795
FAM710.dag	15	14	12,2692	13,871	14,1928	14,367	0,07795	0,47738	1,25035	0,056897	0,732636

Tabela 5.13: Testes da relaxação linear com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
r1.dag	14	14	12,5	14	14	14	0,210305	1,034042	2,093709	0,169521	1,854843
r2.dag	13	13	12,5	13	13	13	0,1169	0,524495	0,929851	0,066904	0,232702
ran1.dag	15	14	13,5	14,4538	14,4392	14,4734	0,224389	1,514387	2,988433	0,41458	3,119209
ran2.dag	13	13	12,1667	12,5664	13	13	0,056328	0,495265	0,587916	0,055415	0,208055
ran3.dag	14	13,1944	12,5	13,9	13,75	13,9	2,003089	1,665164	2,266846	0,292582	4,062317
ran4.dag	17	16	16	16,125	16,098	16,15	0,089731	0,91964	1,470089	0,297126	1,297469
ran5.dag	9	9	8,5	9	9	9	0,452519	0,100118	0,089278	0,020451	0,047219
ran6.dag	9	9	8,33333	9	9	9	0,823963	0,286141	0,139732	0,026835	0,065289
ran7.dag	11	11	10,5	11	11	11	1,596132	1,474356	1,610718	0,139406	0,797542
ran8.dag	11	11	10,5	11	11	11	1,05489	0,894765	3,008584	0,110142	0,445745
ran9.dag	8	8	7,25532	8	8	8	0,208063	0,067139	0,031844	0,007613	0,03709
ran10.dag	18	17	16,1667	17,25	17,25	17,25	12,978294	4,928721	6,086949	0,627745	2,851626
ran11.dag	18	17,3763	17,0667	17,7222	17,625	17,75	121,53905	24,656617	77,218968	2,116073	24,287962
ran12.dag	17	-	16,5	17	17	17	-	4,763698	1,97471	0,520717	2,723909
ran13.dag	10	10	10	10	10	10	16,789477	0,519442	0,683783	0,035899	0,203354
ran14.dag	16	15	14,5	15,25	15,25	15,25	3,167382	2,661599	7,849543	0,159294	3,766164
ran15.dag	15	15	13,2195	14,25	15	15	0,277395	1,671829	3,018327	0,320103	2,365037
ran16.dag	11	10	10	11	11	11	0,541236	1,014384	0,78141	0,047837	0,238654
ran17.dag	21	21	21	21	21	21	11,306051	24,015976	34,570638	0,522339	17,954993
ran18.dag	15	14	13,5	14,25	14,25	14,25	0,786185	2,680367	6,583088	0,45207	3,048776
ran19.dag	11	10,5	10,5	11	11	11	0,342276	0,604974	0,183058	0,036795	0,109786
ran20.dag	10	10	9,02857	10	10	10	2,792091	0,615207	0,612474	0,031502	0,12658

Tabela 5.14: Testes da relaxação linear com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
bin4.dag	9	9	5	9	9	9	0,003677	0,006048	0,003409	0,002143	0,002935
bin5.dag	10	10	6	10	10	10	0,007171	0,016022	0,007425	0,004519	0,005897
bin6.dag	12	11	7	11,0938	11,0938	11,0938	0,041791	0,075968	0,046866	0,013992	0,016132
bin7.dag	13	13	8	13	13	13	0,219348	0,194	0,147305	0,030106	0,035573
bin8.dag	15	15	9	15	15	15	4,259804	0,395033	0,411347	0,07156	0,083862
bin9.dag	17	17	10	17	17	17	22,855771	0,817121	0,988519	0,140207	0,180697
di64.dag	34	34	34	34	34	34	0,041924	0,519723	0,6121	0,151792	0,168243
di100.dag	52	52	52	52	52	52	0,070119	2,27337	3,304468	0,601183	0,632374
di144.dag	52	51	51,0625	51,3762	51,6667	51,6667	0,249415	4,346795	25,094351	1,252743	1,255301
di225.dag	79	78	78,0625	TLE	78,6667	78,6667	3,381211	40,189052	603,644094	6,212029	6,10119
di256.dag	70	69	67,1481	TLE	69,451	69451	4,847811	41,849408	603,32694	6,5952	7,1024
di400.dag	88	86	84,0889	-	87,0769	87,0769	21,165406	403,249445	-	42,964749	43,01082
irr41.dag	23	23	23	23	23	23	0,013181	0,138977	0,18624	0,036226	0,039724
n13.dag	7	7	5,02857	7	7	7	0,002801	0,003272	0,002227	0,001499	0,003009
tree3.dag	9	9	9	9	9	9	0,002101	0,004082	0,002685	0,001743	0,002239
tree4.dag	10	10	10	10	10	10	0,008807	0,021087	0,014583	0,005833	0,007449
tree5.dag	12	10	10,0135	11,3	11,3	11,3	0,150311	0,164107	0,075937	0,017215	0,019542
tree6.dag	13	12	12	13	13	13	0,220896	0,32442	0,245089	0,052409	0,059731
tree7.dag	15	14	14	15	15	15	0,710308	0,392314	0,63023	0,210129	0,219219
tree8.dag	17	16	16	17	17	17	4,706318	1,182404	1,812505	0,402789	0,434326
celbow.dag	30	28,5833	28,3333	28,625	28,625	28,6429	0,154847	0,218542	0,692083	0,044802	0,124163
cstanford.dag	29	28,0556	28	28,3333	28,25	28,3333	0,073973	0,178308	0,318947	0,031814	0,130158
fft2-b-m.dag	28	28	28	28	28	28	0,285211	4,053047	6,773737	0,801475	335,650542
iterative2-b-m.dag	47	47	47	47	47	47	1,609539	13,296153	350,890621	0,421344	90,135246

Tabela 5.15: Testes da relaxação linear com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	IP Z	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
FAM12.dag	16	0,752825	0,346236	0,218325	0,095695	0,204302
FAM13.dag	16	0,314901	0,204448	0,204709	0,055183	0,062654
FAM14.dag	15	0,656004	0,161519	0,297297	0,047737	0,061921
FAM15.dag	15	0,262132	0,240648	0,329607	0,089865	0,093329
FAM16.dag	15	0,517284	0,181858	0,322084	0,067717	0,057134
FAM17.dag	15	0,36846	0,181796	0,176044	0,031894	0,04125
FAM18.dag	16	0,553372	0,09859	0,167821	0,048162	0,101784
FAM19.dag	15	0,439277	0,123975	0,275661	0,060052	0,052447
FAM21.dag	15	0,458238	0,209662	0,136184	0,057611	0,073952
FAM22.dag	16	0,475833	0,502483	0,218378	0,047615	0,092044
FAM23.dag	10	0,292512	0,159355	0,087963	0,025703	0,044428
FAM24.dag	15	0,345712	0,116588	0,166098	0,04382	0,045742
FAM25.dag	11	0,368927	0,139409	0,071363	0,021924	0,03201
FAM26.dag	15	0,274169	0,235118	0,1177	0,111707	0,070509
FAM27.dag	15	0,647846	0,232085	0,132529	0,066109	0,073799
FAM28.dag	15	0,448511	0,634559	0,236749	0,070326	0,078975
FAM29.dag	15	0,586073	0,22877	0,187888	0,038676	0,058119
FAM31.dag	14	600,062819	0,232955	0,403489	0,061543	0,174482
FAM32.dag	16	6,634764	0,415196	0,398918	0,072541	0,176071
FAM33.dag	14	27,795178	0,193406	0,338554	0,048475	0,095773
FAM34.dag	14	600,243534	0,353765	0,507048	0,077903	0,13245
FAM35.dag	14	8,097918	0,36412	0,173144	0,082118	0,283418
FAM36.dag	17	4,483873	0,715444	0,505352	0,086894	0,143328
FAM37.dag	12	600,178863	0,176156	0,16301	0,031719	0,070942
FAM38.dag	14	1,436479	0,236814	0,174492	0,030071	0,071834
FAM39.dag	13	2,243757	0,31454	0,244947	0,060826	0,074456
FAM41.dag	18	7,099778	0,927795	1,352849	0,15704	0,523211
FAM42.dag	13	6,356178	0,201884	0,436852	0,044617	0,092375
FAM43.dag	13	7,131851	0,279702	0,29708	0,065807	0,148773
FAM44.dag	18	1,355353	1,638249	0,872265	0,215117	0,613918
FAM45.dag	19	28,231979	1,61106	1,329265	0,138967	0,161515
FAM46.dag	18	1,838231	0,489415	1,363308	0,189193	0,480483
FAM47.dag	18	31,759034	0,527733	0,547308	0,100752	0,28005
FAM48.dag	16	600,220437	0,46618	0,385302	0,078374	0,229295
FAM49.dag	18	287,082175	1,03254	0,811083	0,198885	0,781288
FAM51.dag	16	600,351941	0,45292	0,488729	0,079783	0,231773
FAM52.dag	17	600,257852	0,463023	0,631159	0,100649	0,259698
FAM53.dag	17	-	0,492274	0,468878	0,09626	0,235625
FAM54.dag	13	600,114483	0,449284	0,282372	0,046775	0,11448
FAM55.dag	14	600,332325	1,693377	0,456956	0,134837	0,316379
FAM56.dag	14	600,191588	0,394898	0,373308	0,074842	0,210489
FAM57.dag	14	-	0,470962	0,475374	0,062523	0,265469
FAM58.dag	13	600,32529	0,308707	0,23478	0,051829	0,318042
FAM59.dag	14	600,307259	0,932362	0,670097	0,185349	0,552945
FAM61.dag	13	600,108622	0,40012	0,386136	0,056216	0,14274
FAM62.dag	18	-	0,774965	0,700821	0,147563	0,392709
FAM63.dag	14	4,11125	0,220989	0,267602	0,045924	0,168636
FAM64.dag	18	600,766801	0,522186	0,690384	0,114503	0,492865
FAM65.dag	16	-	1,126741	0,987466	0,261354	0,600831
FAM66.dag	16	-	2,666635	0,603905	0,146922	1,182675
FAM67.dag	18	5,385112	0,388028	0,371245	0,059883	0,38576
FAM68.dag	16	600,14526	0,803066	0,621301	0,128224	0,264925
FAM69.dag	14	600,444731	0,388814	0,322005	0,084851	0,235427
FAM71.dag	16	600,115998	0,668151	0,596267	0,108251	0,400923
FAM72.dag	15	601,99208	1,187878	0,445363	0,211471	0,376475
FAM73.dag	14	-	0,825339	1,18247	0,138371	0,485392
FAM74.dag	17	600,380138	2,627735	1,597594	0,293074	0,608258
FAM75.dag	15	600,098919	0,318812	0,284461	0,047586	0,088492
FAM76.dag	14	602,593154	0,661689	0,373084	0,087686	0,402863
FAM77.dag	13	600,471596	1,313196	1,176546	0,139221	0,221626
FAM78.dag	16	600,628208	0,993571	1,042516	0,250913	1,318956
FAM79.dag	14	600,698352	0,93121	0,781264	0,159293	0,344696
FAM110.dag	16	0,596885	0,228143	0,323888	0,05644	0,07983
FAM210.dag	11	15,800784	0,108049	0,124307	0,029043	0,042225
FAM310.dag	16	600,229084	0,316652	0,232028	0,076319	0,178919
FAM410.dag	18	1,562858	0,869883	0,894986	0,277848	0,265257
FAM510.dag	13	600,101839	0,218337	0,570998	0,048846	0,33302
FAM610.dag	16	600,399773	0,720276	0,542465	0,116377	0,35512
FAM710.dag	15	600,758186	0,856613	1,348157	0,241923	0,935056

Tabela 5.16: Testes do problema inteiro com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP Z	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
r1.dag	14	600,511138	2,084522	2,020178	0,296195	2,087295
r2.dag	13	-	0,737544	1,021858	0,091731	0,343619
ran1.dag	15	600,029194	3,654524	2,314902	0,406367	3,315362
ran2.dag	13	600,575041	0,933079	0,910132	0,105146	0,281689
ran3.dag	14	-	2,418877	2,199975	0,340769	4,337658
ran4.dag	17	600,105134	2,56928	2,726896	1,018681	1,72816
ran5.dag	9	37,190329	0,247087	0,247449	0,059204	0,085512
ran6.dag	9	174,494585	0,535477	0,275583	0,073099	0,140138
ran7.dag	11	600,155214	2,987756	2,792021	0,272588	1,025711
ran8.dag	11	263,345649	1,708289	1,452208	0,253141	0,600439
ran9.dag	8	593,103464	0,153771	0,114718	0,034736	0,067114
ran10.dag	18	600,096517	5,720344	6,143279	0,517028	3,045218
ran11.dag	18	600,837505	14,6094	21,766909	1,683578	25,797443
ran12.dag	17	-	8,422379	7,033581	0,526979	3,090532
ran13.dag	10	-	0,944154	1,069627	0,089281	0,326424
ran14.dag	16	601,257673	4,719227	3,627864	0,261836	3,959163
ran15.dag	15	600,024894	2,462701	2,839649	0,303533	2,733403
ran16.dag	11	83,82547	0,622788	0,590302	0,100223	0,350136
ran17.dag	21	600,377021	20,344555	7,493523	1,629053	18,852168
ran18.dag	15	600,086308	3,607858	8,214383	0,432413	3,273564
ran19.dag	11	600,168253	0,489022	0,380392	0,066619	0,189138
ran20.dag	10	62,337228	0,618783	0,38538	0,063806	0,191653

Tabela 5.17: Testes do problema inteiro com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP Z	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
bin4.dag	9	0,127629	0,041167	0,020035	0,007386	0,011364
bin5.dag	10	0,575118	0,090419	0,053429	0,015535	0,020855
bin6.dag	12	600,222994	0,952475	0,116926	0,035823	0,055657
bin7.dag	13	601,244067	3,969817	0,238079	0,048909	0,100784
bin8.dag	15	239,682329	14,558459	0,420486	0,123142	0,20698
bin9.dag	17	600,808968	429,868545	1,131718	0,323692	0,499096
di64.dag	34	3,785197	0,449221	0,720338	0,459162	0,579116
di100.dag	52	600,090901	1,955426	2,322323	1,943047	2,066091
di144.dag	52	620,297691	6,716644	31,631903	3,40097	4,138226
di225.dag	79	600,038938	52,122953	-	17,628576	17,936855
di256.dag	70	600,036387	256,172014	-	20,676178	21,775041
di400.dag	88	656,128269	601,117695	-	95,327225	-
irr41.dag	23	0,636805	0,153635	0,215075	0,112833	0,135706
n13.dag	7	0,028959	0,024732	0,015396	0,003516	0,005131
tree3.dag	9	0,06949	0,034847	0,01792	0,008592	0,009072
tree4.dag	10	0,527512	0,128792	0,080396	0,026054	0,036319
tree5.dag	12	600,090152	0,95073	0,288337	0,069598	0,088379
tree6.dag	13	602,446266	0,801503	0,35352	0,083401	0,171948
tree7.dag	15	600,068356	2,426823	1,356079	0,278769	0,593758
tree8.dag	17	600,159229	15,514024	4,395014	0,629616	1,003174
celbow.dag	30	7,202245	0,83193	0,997334	0,048071	0,182426
cstanford.dag	29	2,778273	0,348793	0,642025	0,032069	0,155831
fft2-b-m.dag	28	600,016951	5,893057	5,72963	2,437217	339,014204
iterative2-b-m.dag	47	600,92127	103,416875	8,924609	1,1995	91,042356

Tabela 5.18: Testes do problema inteiro com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	IP Z	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
FAM31,dag	26	26	16,4348	26	26	26	0,017652	0,654498	0,281565	0,061448	0,211236
FAM32,dag	27	27	27	27	27	27	0,019041	0,509419	0,374132	0,054412	0,266151
FAM33,dag	25	25	25	25	25	25	0,016585	0,453344	0,515823	0,050514	0,109382
FAM34,dag	25	25	13,5	25	25	25	0,019464	0,651362	0,738331	0,043349	0,093607
FAM35,dag	25	25	25	25	25	25	0,016406	0,491535	0,478847	0,083246	0,750035
FAM37,dag	25	25	17,4	25	25	25	0,018816	0,599673	0,50556	0,069166	0,145513
FAM38,dag	25	25	21	25	25	25	0,0162	0,441054	0,374699	0,075586	0,382754
FAM39,dag	25	25	15,5	25	25	25	0,016001	0,608685	0,328749	0,067421	0,555761
FAM42,dag	25	25	22,4	25	25	25	0,016335	0,812463	0,793448	0,144053	0,339906
FAM43,dag	25	25	23,5	25	25	25	0,017355	0,507008	0,577709	0,123313	0,140159
FAM48,dag	26	26	26	26	26	26	0,016204	0,454247	0,80567	0,098182	0,471255
FAM51,dag	35	35	25,4	35	35	35	0,048455	2,524894	2,029931	0,167672	0,56214
FAM52,dag	36	36	36	36	36	36	0,044499	1,765569	2,274328	0,219422	2,847695
FAM53,dag	36	36	36	36	36	36	0,047116	1,520572	1,949958	0,138801	0,633549
FAM54,dag	24	24	18,3333	24	24	24	0,049643	0,976143	1,047019	0,094473	0,193856
FAM55,dag	24	24	17,3333	24	24	24	0,054195	1,016659	0,866531	0,099387	0,710584
FAM56,dag	24	24	18,3	24	24	24	0,048617	1,010068	0,887069	0,093524	0,705447
FAM57,dag	24	24	23,5	24	24	24	0,049623	0,850782	1,194499	0,066793	0,740106
FAM58,dag	25	25	25	25	25	25	0,046273	0,820259	0,850115	0,08413	0,394122
FAM59,dag	24	24	19,2632	24	24	24	0,054596	1,094181	1,039481	0,082782	1,119898
FAM61,dag	27	27	14,4872	27	27	27	0,062448	1,966848	1,391843	0,100516	0,247657
FAM62,dag	40	40	29	40	40	40	0,060583	3,900206	3,962611	0,266841	3,569327
FAM63,dag	21	21	21	21	21	21	0,063838	0,757446	0,879787	0,088209	0,433855
FAM64,dag	30	30	30	30	30	30	0,073728	1,293377	1,197978	0,0926	1,086178
FAM65,dag	28	28	28	28	28	28	0,060397	1,424095	1,950432	0,15119	0,275279
FAM66,dag	28	27	21,3	27,25	27	27,25	0,075288	1,825419	1,945421	0,167729	0,510018
FAM67,dag	28	28	28	28	28	28	0,062401	1,349341	1,594436	0,078566	0,338337
FAM68,dag	28	27	15,4805	27,25	27	27,25	0,056156	2,095488	1,906016	0,162701	1,371707
FAM69,dag	27	27	23,7143	27	27	27	0,065233	1,71261	2,01678	0,131395	0,606099
FAM71,dag	25	24	24,1176	24,64	24,1905	24,6957	0,090057	1,357324	1,942817	0,139565	0,749524
FAM72,dag	23	23	23	23	23	23	0,110179	0,913166	1,616554	0,100002	0,50073
FAM73,dag	24	24	16,3537	23,5	24	24	0,078489	1,82652	1,606042	0,187719	0,845356
FAM74,dag	32	32	27,1912	32	32	32	0,116836	3,324104	2,794448	0,240654	3,038227
FAM75,dag	30	30	23,2625	30	30	30	0,082926	2,683	3,907192	0,170504	0,578737
FAM76,dag	25	24	24,1463	24,2849	24	24,3429	0,118497	1,435911	2,068601	0,138685	0,64785
FAM77,dag	23	23	18,2338	23	23	23	0,086109	1,976499	1,285669	0,099448	1,122036
FAM78,dag	32	32	32	32	32	32	0,067759	2,390945	2,175119	0,285298	7,494936
FAM79,dag	24	23	19,375	23,08	23	23,087	0,087308	1,472494	1,999723	0,093489	0,457758
FAM310,dag	26	26	25,25	26	26	26	0,018127	0,607317	0,38738	0,067595	0,83267
FAM510,dag	24	24	16,5	24	24	24	0,047794	0,995598	0,774411	0,081222	0,952553
FAM610,dag	28	27	26,3333	27,1143	27	27,129	0,059776	1,852855	1,962498	0,168621	0,687823
FAM710,dag	30	30	18,4235	30	30	30	0,080408	3,044985	1,943266	0,29025	2,158489

Tabela 5.19: Testes da relaxação linear com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP Z	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
r1,dag	27	27	21,3456	27	27	27	0,19316	6,717666	6,840244	0,722756	2,73918
r2,dag	25	25	22,3623	25	25	25	0,071393	3,029381	4,113826	0,396213	3,135435
ran1,dag	28	28	22,5	28	28	28	0,188998	6,247007	6,704979	0,788854	5,656237
ran2,dag	27	27	22359	27	27	27	0,057784	2,226142	2,138079	0,317754	0,668129
ran3,dag	26	26	21,3475	26	26	26	0,306075	6,952342	7,221789	0,793936	3,410045
ran4,dag	36	36	34,4545	36	36	36	0,088016	8,57677	8,704638	0,838425	8,063992
ran5,dag	16	16	8,5	16	16	16	0,386481	3,555129	0,614503	0,07096	0,185104
ran6,dag	18	18	8,5426	18	18	18	0,653271	8,750707	0,802216	0,140534	1,062865
ran7,dag	22	22	12,4362	22	22	22	1,700384	28,842508	26,358922	1,273294	8,119159
ran8,dag	19	19	10,5	19	19	19	1,102504	13,580395	12,504149	0,706054	6,322394
ran9,dag	16	16	7,58065	16	16	16	0,198535	0,425906	0,087005	0,048373	0,088065
ran10,dag	29	29	27,6146	29	29	29	3,246891	28,160828	101,425995	2,093202	17,773632
ran11,dag	27	27	24,3607	TLE	27	27	17,540669	220,432358	604,691972	16,63825	372,097427
ran12,dag	26	26	23,4028	26	26	26	1,562319	28,67566	268,164633	13,535674	63,714969
ran13,dag	18	18	10,4432	18	18	18	3,432061	5,22712	42,028204	0,916833	2,729615
ran14,dag	25	25	18,4445	25	25	25	3,754219	52,131907	85,364168	14,811567	109,961847
ran15,dag	31	31	26,3392	31	31	31	0,281173	12,12497	15,85662	1,194263	2,524036
ran16,dag	21	21	10,5161	21	21	21	0,541094	8,602637	7,559789	0,633456	1,899821
ran17,dag	29	29	26,4529	TLE	29	29	17,75917	227,415501	603,171755	12,209306	327,555713
ran18,dag	26	26	22,5301	26	26	26	0,79986	27,904425	34,919778	1,527327	6,114073
ran19,dag	20	20	13,3403	20	20	20	0,34358	4,634612	3,372242	0,451004	5,544106
ran20,dag	20	20	10,4805	20	20	20	0,349765	6,043453	3,764289	0,476365	2,023209

Tabela 5.20: Testes da relaxação linear com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP Z	LP X	LP Y	LP Y Ref	LP Z	LP Z Exp	LP X t	LP Y t	LP Y Ref t	LP Z t	LP Z Exp t
bin5,dag	17	17	6,79167	17	17	17	0,007228	0,056917	0,014773	0,009545	0,01191
bin6,dag	18	18	7,79167	18	18	18	0,160343	0,282957	0,108957	0,018221	0,021337
bin7,dag	20	18	8,79167	18,2188	18,2188	18,2188	0,292033	0,800075	0,663142	0,095885	0,104864
bin8,dag	21	18	9,79167	18,5938	18,5938	18,5938	3,21629	1,309255	1,711053	0,319055	0,323394
bin9,dag	23	18	10,7917	19,1055	19,1055	19,1055	28,163868	2,118234	6,89703	1,064751	1,067459
di256,dag	130	130	130	-	130	130	2,505762	231,029729	-	21,062459	21,341547
di400,dag	202	202	-	-	202	202	10,831989	-	-	170,098355	169,416069
tree4,dag	17	17	17	17	17	17	0,006417	0,103706	0,029268	0,010231	0,013713
tree5,dag	18	18	18	18	18	18	0,036773	0,323173	0,12441	0,029287	0,032502
tree6,dag	20	18	18,013	18,9231	18,9231	18,9231	0,212202	2,342058	0,704748	0,13716	0,141553
tree7,dag	21	18	18,027	20,0426	20	20	1,05363	1,272767	2,146244	0,377821	0,396123
tree8,dag	23	18	18,0156	20,4889	20,3333	20,3333	7,710862	2,682222	10,398299	1,326176	1,359074
celbow,dag	35	35	33,3333	35	35	35	0,112796	1,185782	1,392792	0,176314	2,160652
cstanford,dag	32	32	31,1444	32	32	32	0,081963	0,755377	0,799568	0,075712	0,642832
fft2-b-m,dag	52	52	52	52	52	52	0,305317	25,051322	34,987189	4,154984	533,348618
iterative2-b-m,dag	90	90	90	90	90	90	9,337599	37,107131	71,438506	1,150628	319,648124

Tabela 5.21: Testes da relaxação linear com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	IP	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
FAM31.dag	26	75,612456	0,991174	0,971826	0,285761	0,328109
FAM32.dag	27	600,255455	0,397165	0,621626	0,275203	0,437214
FAM33.dag	25	180,759139	1,425633	1,041375	0,216825	0,243321
FAM34.dag	25	454,48274	1,130393	1,2201	0,276791	0,244125
FAM35.dag	25	600,274367	1,391534	1,493764	0,285275	1,003898
FAM37.dag	25	600,060529	1,016016	0,929847	0,121827	0,294022
FAM38.dag	25	5,942879	1,57887	0,48121	0,25844	0,524634
FAM39.dag	25	8,160913	1,113694	1,15064	0,258068	0,730766
FAM42.dag	25	600,058329	3,181251	3,843547	0,431199	0,548161
FAM43.dag	25	600,049366	2,101451	2,236877	0,383602	0,339111
FAM48.dag	26	30,682378	1,184561	2,614766	0,348697	0,68207
FAM51.dag	35	600,104743	3,468553	5,825146	0,540285	0,856069
FAM52.dag	36	602,159728	0,697782	1,451248	0,471021	3,293639
FAM53.dag	36	600,093383	3,251359	5,281202	0,493186	0,945272
FAM54.dag	24	-	11,205576	3,107084	0,283332	0,362208
FAM55.dag	24	600,257572	6,674582	1,362261	0,274582	0,917553
FAM56.dag	24	600,311817	1,652523	1,694036	0,315217	0,853632
FAM57.dag	24	600,103875	2,051186	3,016083	0,258019	0,925184
FAM58.dag	25	600,107578	2,532075	2,27805	0,395307	0,562562
FAM59.dag	24	-	3,305978	1,896682	0,304866	1,29354
FAM61.dag	27	600,100822	19,515686	3,83627	0,467563	0,487144
FAM62.dag	40	600,102767	256,185252	11,055683	0,822849	3,983044
FAM63.dag	21	600,102374	1,654442	1,639917	0,246969	0,655802
FAM64.dag	30	602,807143	2,949099	1,000011	0,358044	1,296358
FAM65.dag	28	600,109931	3,478382	4,863531	0,448388	0,511465
FAM66.dag	28	600,108332	13,562932	4,366314	0,501707	0,769895
FAM67.dag	28	600,324765	3,005332	2,867139	0,305805	0,581727
FAM68.dag	28	-	21,067278	4,434876	0,544516	1,697835
FAM69.dag	27	600,110402	6,593355	4,491995	0,423892	0,834437
FAM71.dag	25	600,434565	3,098954	3,667469	0,511929	1,092003
FAM72.dag	23	600,358915	2,148891	3,001403	0,325114	0,727604
FAM73.dag	24	-	44,821326	3,291825	0,289287	1,089564
FAM74.dag	32	600,103837	35,625124	6,951902	0,937857	3,329489
FAM75.dag	30	600,449718	5,467908	7,61965	0,520229	0,94959
FAM76.dag	25	600,783701	2,935016	4,944524	0,482827	0,907185
FAM77.dag	23	-	13,279751	2,099466	0,39385	1,326638
FAM78.dag	32	600,111424	5,250637	8,38882	0,843666	7,816305
FAM79.dag	24	600,460434	3,488773	5,346144	0,372684	0,680364
FAM310.dag	26	600,237984	1,185667	1,360452	0,158367	0,967677
FAM510.dag	24	600,100308	1,546111	0,865293	0,287462	1,155802
FAM610.dag	28	600,45958	5,466984	4,320525	0,530924	1,067272
FAM710.dag	30	-	77,348789	6,495715	0,709076	2,544197

Tabela 5.22: Testes do problema inteiro com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	IP	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
r1.dag	27	600,063433	293,314152	16,356435	3,236727	5,009428
r2.dag	25	600,522518	5,778815	7,397511	1,050407	4,208628
ran1.dag	28	600,077081	225,507684	14,865037	3,452571	7,859178
ran2.dag	27	600,191372	11,017053	4,768286	1,032928	1,016854
ran3.dag	26	600,03706	319,069886	16,052089	2,153632	5,832614
ran4.dag	36	600,114422	22,514104	19,459936	4,107531	10,412721
ran5.dag	16	600,035052	50,825251	1,556928	0,26368	0,333877
ran6.dag	18	600,068184	141,770061	1,992761	0,660126	1,449287
ran7.dag	22	600,861757	600,095501	37,09338	2,792419	14,004484
ran8.dag	19	600,088415	340,453907	4,792128	2,636953	7,646332
ran9.dag	16	449,32216	6,823945	0,504328	0,119755	0,171803
ran10.dag	29	600,082247	48,265492	68,755006	5,603908	27,045573
ran11.dag	27	600,474611	600,36425	-	28,222354	-
ran12.dag	26	601,020806	600,131007	69,393767	11,725771	73,314618
ran13.dag	18	602,077395	600,063842	9,913952	2,761172	4,819297
ran14.dag	25	600,279959	600,120436	81,234203	11,089474	120,610643
ran15.dag	31	600,031372	247,943307	31,82695	2,705327	6,230285
ran16.dag	21	600,042423	303,47505	12,897648	1,873724	3,805976
ran17.dag	29	600,55086	600,340967	-	17,286972	-
ran18.dag	26	600,056558	585,1316	52,544593	8,216291	13,059044
ran19.dag	20	600,054974	87,731419	5,022907	1,12769	6,229379
ran20.dag	20	600,042138	187,005768	8,047018	0,913157	2,937665

Tabela 5.23: Testes do problema inteiro com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	IP	IP X t	IP Y t	IP Y Ref t	IP Z t	IP Z Exp t
bin5.dag	17	0,577645	0,460328	0,085525	0,07875	0,045883
bin6.dag	18	600,080616	1,915368	0,263052	0,062926	0,116111
bin7.dag	20	604,183063	13,639731	1,189108	0,319011	0,352409
bin8.dag	21	600,091792	51,630585	5,58757	0,962765	1,128165
bin9.dag	23	601,039609	600,089903	26,866574	7,707222	7,964075
di256.dag	130	600,067506	26,304187	-	64,50238	-
di400.dag	202	600,223574	-	-	593,293926	-
tree4.dag	17	0,265272	0,188043	0,100701	0,03822	0,086898
tree5.dag	18	600,097775	0,675407	0,270352	0,111304	0,117158
tree6.dag	20	600,240314	50,468141	2,91383	0,682533	0,510971
tree7.dag	21	600,050036	109,377929	7,351997	2,893864	1,4949
tree8.dag	23	600,592629	600,124948	167,795063	24,692261	8,959185
celbow.dag	35	600,797031	3,870648	2,55444	0,475886	2,414093
cstanford.dag	32	600,102339	1,268119	1,000163	0,178275	0,739558
fft2-b-m.dag	52	600,013645	3,492525	10,234395	8,712027	539,625353
iterative2-b-m.dag	90	600,327286	6,388216	34,367733	2,294928	321,348768

Tabela 5.24: Testes do problema inteiro com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Estruturadas

5.2.2 Influência dos limites sobre a Formulação Z

Nesta seção, desejamos observar qual a influências dos limites, inferior e superior, apresentados no Capítulo 3, na Formulação Z, a que apresentou os melhores resultados em comparação com as outras. Os testes consistiram em resolver a formulação considerando ou não os limites. Quando utilizamos os limites inferior e superior, o fazemos segundo a estratégia apresentada na Seção 5.2.1, por fixação de variáveis.

Diferentemente dos testes anteriores, não consideraremos o ótimo nesta análise, ou seja, nossa intenção não é medir a proximidade ao valor ótimo e sim comparar unicamente as duas formas de implementação, com ou sem os limites. Adotamos os mesmo grupos de instâncias que foram utilizados ao longo de todo o texto e fizemos os testes para as seguintes quantidades de processadores: infinitos, $\lfloor \omega(\prec)/2 \rfloor$, $\lfloor \omega(\prec)/4 \rfloor$ e $\lfloor \omega(\prec)/8 \rfloor$.

Como nossa intenção é observar a influência dos limites puramente entre as duas verões, com e sem limites, utilizamos a seguinte estratégia para produzir os gráficos: tomamos o valor do parâmetro em questão (tempo de resolução da relaxação linear, valor da relaxação linear ou tempo de resolução do problema inteiro) na formulação com os limites subtraído do valor do parâmetro na formulação sem os limites e este resultado dividimos pelo valor do parâmetro na formulação sem os limites. O resultado dessas operações, para cada instância, é plotado no gráfico. Denominamos os parâmetros tempo de resolução da relaxação linear, valor da relaxação linear e tempo de resolução do problema inteiro, respectivamente, para a versão sem os limites, como LP Z s t, LP Z s e IP Z s t, já para a versão com os limites eles serão denominados, respectivamente, como LP Z c t, LP Z c e IP Z c t. No caso do parâmetro tempo (t) não foi necessário utilizar a função $\log_{10}(t)$ sobre os tempos obtidos, pois estes são mais uniformes, ou seja, não são discrepantes como os tempos dos testes anteriores. Apesar de não tomarmos o $\log_{10}$ do tempo em questão, continuamos podendo obter valores negativos nos gráficos, visto que o cálculo utilizado é o mencionado acima. Portanto, quando o tempo da formulação com o limites é menor que o tempo da formulação sem limites, teremos um número negativo.

Analisando os resultados dos experimentos observamos que o tempo de resolução da relaxação linear é pior sem os limites, na maioria das instâncias, e isto se agrava com a diminuição no número de processadores. Comportamento semelhante acontece com o parâmetro valor da relaxação linear. Para a maioria das instâncias, com os limites, obtemos valores melhores e com a diminuição do número de processadores as formulações se distanciam na qualidade da relaxação. Ao analisar o tempo de resolução do problema inteiro, o comportamento é exatamente igual ao dos parâmetros anteriores. Dessa forma, percebemos uma influência considerável dos limites nos parâmetros analisados, principalmente, com número limitado de processadores. Fica evidenciada, portanto, a necessidade de se obter bons limites para este problema. Vale ressaltar que esta influência não se restringe à Formulação Z, todas as outras formulações são dependentes, de certa forma, dos limites, inferior e superior, utilizados diretamente no modelo.

A seguir, estão detalhados os gráficos e posteriormente as tabelas que possibilitam esta análise.

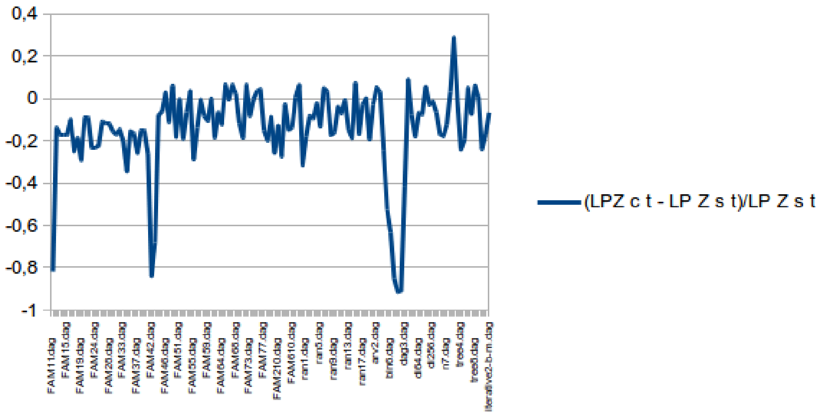


Figura 5.11: Gráfico Instância X Tempo da relaxação linear com infinitos processadores

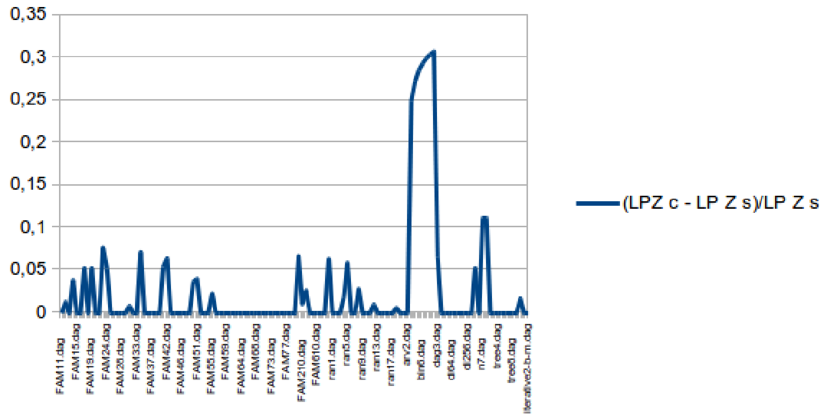


Figura 5.12: Gráfico Instância X Makespan da relaxação linear com infinitos processadores

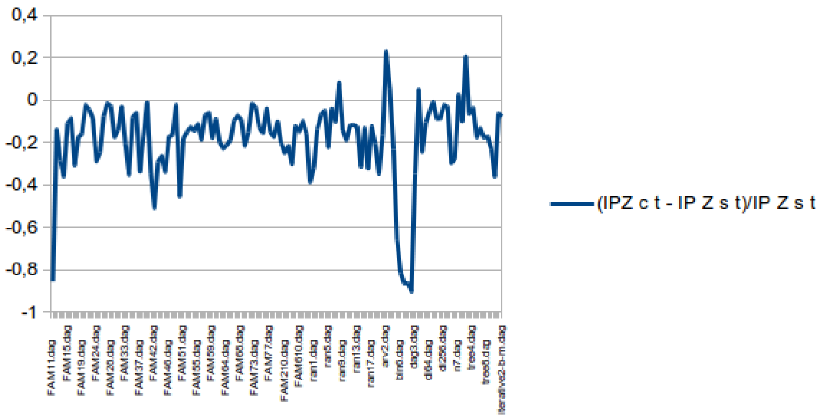


Figura 5.13: Gráfico Instância X Tempo do problema inteiro com infinitos processadores

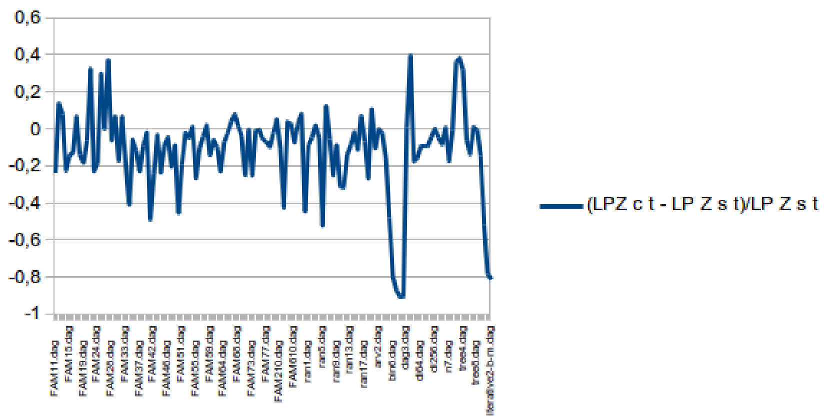


Figura 5.14: Gráfico Instância X Tempo da relaxação linear com $m = \lfloor \omega(\prec)/2 \rfloor$

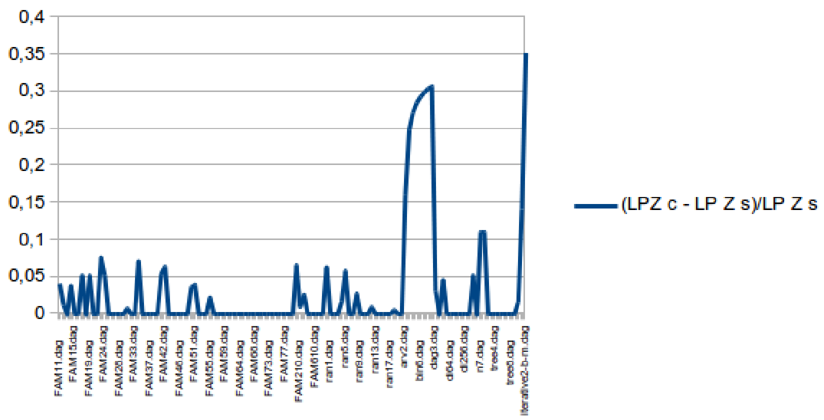


Figura 5.15: Gráfico Instância X Makespan da relaxação linear com $m = \lfloor \omega(\prec)/2 \rfloor$

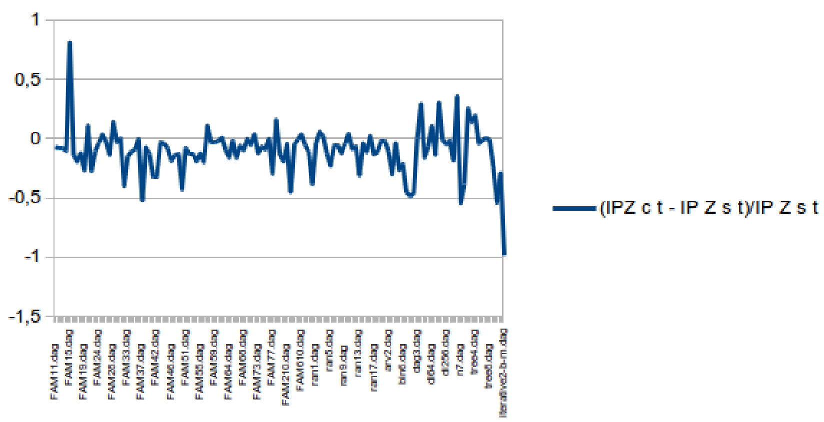


Figura 5.16: Gráfico Instância X Tempo do problema inteiro com $m = \lfloor \omega(\prec)/2 \rfloor$

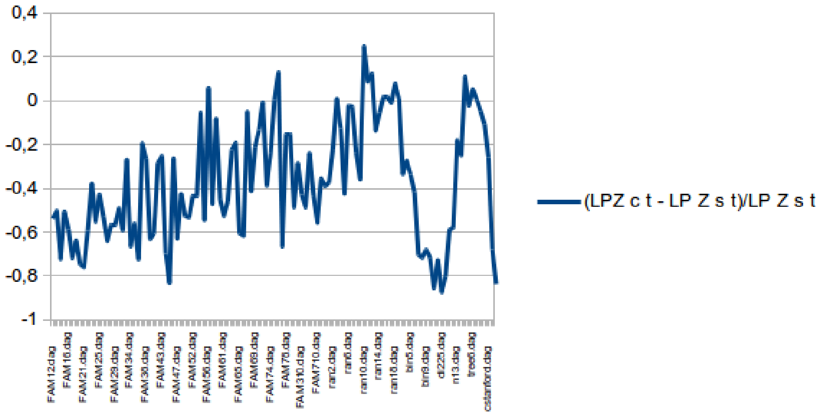


Figura 5.17: Gráfico Instância X Tempo da relaxação linear com $m = \lfloor \omega(\prec)/4 \rfloor$

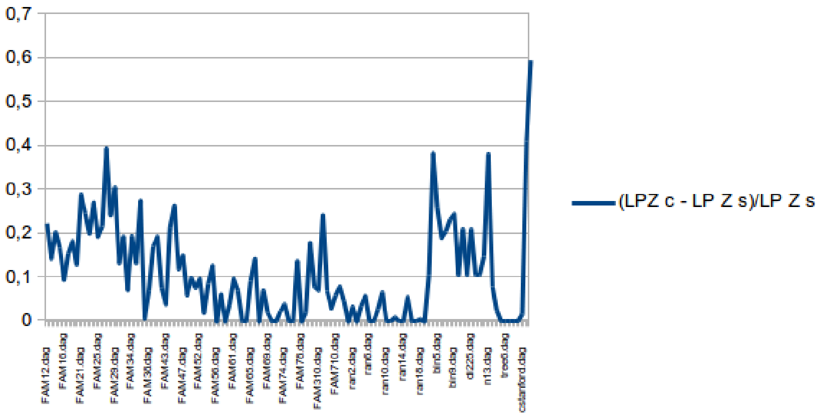


Figura 5.18: Gráfico Instância X Makespan da relaxação linear com $m = \lfloor \omega(\prec)/4 \rfloor$

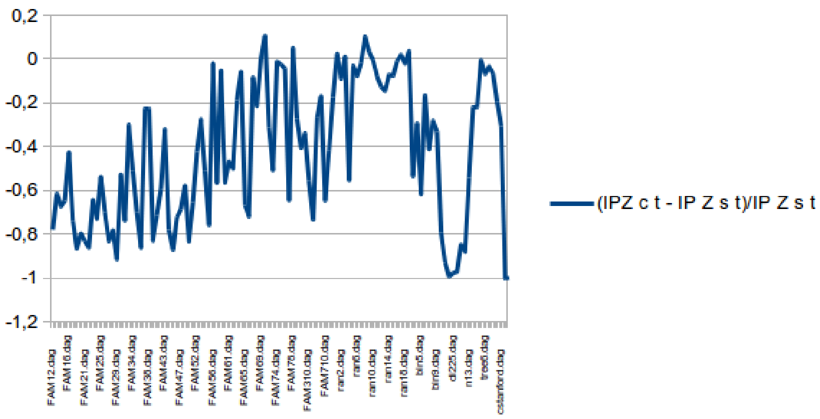


Figura 5.19: Gráfico Instância X Tempo do problema inteiro com $m = \lfloor \omega(\prec)/4 \rfloor$

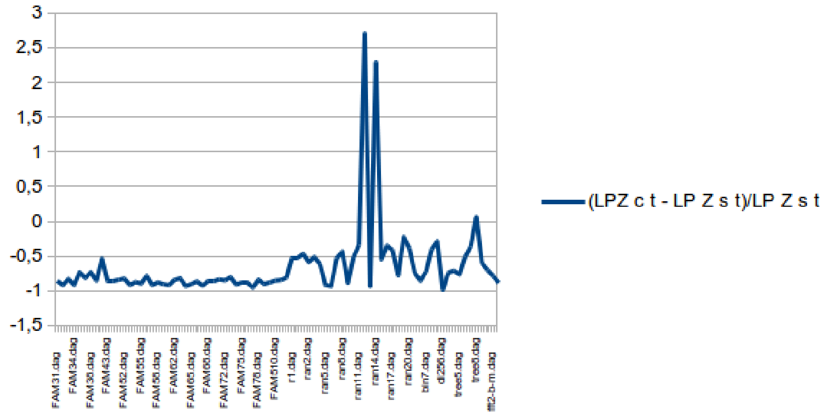


Figura 5.20: Gráfico Instância X Tempo da relaxação linear com $m = \lfloor \omega(\prec)/8 \rfloor$

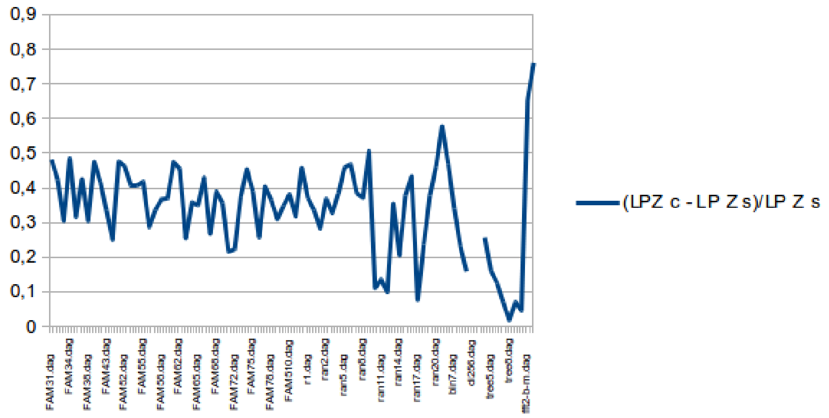


Figura 5.21: Gráfico Instância X Makespan da relaxação linear com $m = \lfloor \omega(\prec)/8 \rfloor$

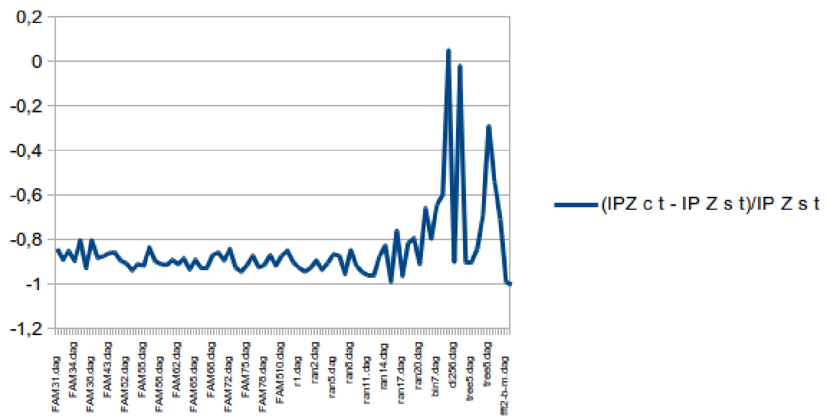


Figura 5.22: Gráfico Instância X Tempo do problema inteiro com $m = \lfloor \omega(\prec)/8 \rfloor$

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
FAM11.dag	0,014899	0,002821	9,5	9,5	0,016748	0,00254
FAM12.dag	0,003895	0,00338	10,2	10,3333	0,00534	0,004629
FAM13.dag	0,003515	0,002925	12	12	0,003197	0,002306
FAM14.dag	0,004012	0,003356	9,75	10,125	0,004586	0,002967
FAM15.dag	0,002894	0,002412	10	10	0,002133	0,001913
FAM16.dag	0,002865	0,002599	10	10	0,002298	0,002115
FAM17.dag	0,003474	0,002633	9,5	10	0,003712	0,002598
FAM18.dag	0,003758	0,003083	11	11	0,003275	0,002724
FAM19.dag	0,003515	0,002515	9,5	10	0,002804	0,002379
FAM21.dag	0,002337	0,002145	7,4	7,4	0,002308	0,002269
FAM22.dag	0,002976	0,002735	10,1818	10,1818	0,003579	0,003445
FAM23.dag	0,001863	0,001438	6,5	7	0,001856	0,001708
FAM24.dag	0,002384	0,001848	9,5	10	0,002764	0,001985
FAM25.dag	0,002337	0,001832	8	8	0,002702	0,002054
FAM26.dag	0,002079	0,001866	10	10	0,002123	0,001979
FAM27.dag	0,00157	0,001395	7	7	0,001524	0,001511
FAM28.dag	0,002236	0,001989	9	9	0,002295	0,002246
FAM29.dag	0,001792	0,001527	8	8	0,001877	0,001562
FAM31.dag	0,004871	0,004079	10,25	10,3333	0,007422	0,006428
FAM32.dag	0,00597	0,00514	11,75	11,75	0,009954	0,009705
FAM33.dag	0,004131	0,003324	12	12	0,004061	0,003247
FAM34.dag	0,004509	0,00299	9,33333	10	0,004484	0,002935
FAM35.dag	0,004514	0,003842	10,75	10,75	0,004568	0,004221
FAM36.dag	0,00458	0,003859	9,66667	9,66667	0,004564	0,004318
FAM37.dag	0,004354	0,003267	10,5	10,5	0,004441	0,002979
FAM38.dag	0,005076	0,004347	12,3333	12,3333	0,005284	0,004433
FAM39.dag	0,003766	0,003211	8,66667	8,66667	0,003383	0,003363
FAM41.dag	0,009108	0,006775	12,3333	13	0,00726	0,004808
FAM42.dag	0,030627	0,005135	10,3333	11	0,007768	0,003862
FAM43.dag	0,022484	0,0073	12	12	0,008933	0,006378
FAM44.dag	0,009365	0,008686	12	12	0,013039	0,009683
FAM45.dag	0,006555	0,006201	12	12	0,005078	0,003392
FAM46.dag	0,006613	0,006842	13	13	0,004902	0,004084
FAM47.dag	0,009706	0,008708	13,25	13,25	0,007212	0,006081
FAM48.dag	0,0092	0,009814	15	15	0,007504	0,007383
FAM49.dag	0,010349	0,008542	13,5	14	0,008489	0,004691
FAM51.dag	0,00591	0,005923	12,5	13	0,005522	0,004559
FAM52.dag	0,008762	0,007159	14,3333	14,3333	0,007201	0,006178
FAM53.dag	0,011036	0,010402	15,28	15,28	0,011177	0,009833
FAM54.dag	0,00502	0,005233	11	11	0,003818	0,003295
FAM55.dag	0,006487	0,004666	9,77778	10	0,008397	0,007503
FAM56.dag	0,006377	0,005549	14	14	0,00541	0,004441
FAM57.dag	0,005123	0,005127	11	11	0,004051	0,003786
FAM58.dag	0,007868	0,007271	12	12	0,007116	0,006737
FAM59.dag	0,007216	0,006513	11,5	11,5	0,005592	0,004627
FAM61.dag	0,007774	0,007819	10,5	10,5	0,008644	0,007928
FAM62.dag	0,009994	0,008217	14,3333	14,3333	0,008086	0,006544
FAM63.dag	0,01017	0,009569	13,2	13,2	0,009538	0,007425
FAM64.dag	0,010785	0,009533	17	17	0,010777	0,008553
FAM65.dag	0,010361	0,011113	12,3125	12,3125	0,013679	0,011201
FAM66.dag	0,009816	0,009853	11	11	0,011073	0,01009
FAM67.dag	0,011984	0,012843	17,1429	17,1429	0,010981	0,010251
FAM68.dag	0,008345	0,008586	12,6667	12,6667	0,006553	0,005984
FAM69.dag	0,008567	0,007642	12,5	12,5	0,009053	0,007162
FAM71.dag	0,012454	0,010223	15,75	15,75	0,010832	0,009259
FAM72.dag	0,01342	0,014383	13,6667	13,6667	0,016334	0,016146
FAM73.dag	0,009234	0,008537	12	12	0,011695	0,011366
FAM74.dag	0,011879	0,011923	13,6667	13,6667	0,012354	0,010813
FAM75.dag	0,00767	0,007997	15	15	0,006554	0,005592
FAM76.dag	0,010152	0,010667	13,5	13,5	0,009879	0,009545
FAM77.dag	0,006165	0,005284	10	10	0,004485	0,003844
FAM78.dag	0,009092	0,007353	15	15	0,007696	0,006417
FAM79.dag	0,010116	0,009293	12,5	12,5	0,010632	0,009619
FAM110.dag	0,003739	0,002809	11,25	12	0,002754	0,002216
FAM210.dag	0,002872	0,002521	9,4	9,5	0,004045	0,003065
FAM310.dag	0,005519	0,004049	12,6667	13	0,007567	0,005973
FAM410.dag	0,005905	0,005775	11	11	0,004913	0,003464
FAM510.dag	0,005372	0,004619	11	11	0,004074	0,003599
FAM610.dag	0,008702	0,007566	13,5	13,5	0,007	0,006014
FAM710.dag	0,006847	0,006931	12	12	0,005539	0,00503

Tabela 5.25: Testes da influência dos limites com infinitos processadores - Grupo de Instâncias Aleatórias 1

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
r1.dag	0,021389	0,022885	13	13	0,014514	0,012285
r2.dag	0,025652	0,017745	12,2222	13	0,019386	0,012013
ran1.dag	0,032282	0,026715	14,25	14,25	0,027823	0,01916
ran2.dag	0,014257	0,013169	12,5	12,5	0,00963	0,00839
ran3.dag	0,049954	0,045647	13,75	13,75	0,043727	0,041033
ran4.dag	0,027366	0,026955	14	14,25	0,025898	0,024783
ran5.dag	0,007843	0,006863	8,5	9	0,006441	0,005052
ran6.dag	0,009961	0,010512	9	9	0,00807	0,007792
ran7.dag	0,028178	0,029297	11	11	0,022059	0,019957
ran8.dag	0,028244	0,023567	10,7	11	0,019557	0,021282
ran9.dag	0,005335	0,00452	7,5	7,5	0,003939	0,003412
ran10.dag	0,093125	0,090041	17,25	17,25	0,082313	0,067373
ran11.dag	0,26116	0,2447	17,625	17,625	0,22729	0,201511
ran12.dag	0,051395	0,05132	16,8333	17	0,042128	0,037519
ran13.dag	0,021313	0,018325	10	10	0,015041	0,013225
ran14.dag	0,049587	0,040626	15,25	15,25	0,042948	0,029766
ran15.dag	0,040964	0,044208	14,25	14,25	0,032416	0,028363
ran16.dag	0,015305	0,012871	11	11	0,014664	0,01004
ran17.dag	0,200375	0,196289	21	21	0,193493	0,170874
ran18.dag	0,055648	0,056017	14,1667	14,25	0,040868	0,032709
ran19.dag	0,013308	0,010836	11	11	0,012348	0,00814
ran20.dag	0,011197	0,010939	9	9	0,008088	0,006813

Tabela 5.26: Testes da influência dos limites com infinitos processadores - Grupo de Instâncias Aleatórias 2

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
arv2.dag	0,00079	0,000837	5	5	0,000874	0,001081
bin3.dag	0,000707	0,000733	4	5	0,000989	0,001051
bin4.dag	0,001294	0,000987	5,5	7	0,001619	0,001253
bin5.dag	0,002948	0,001419	7	9	0,003857	0,001345
bin6.dag	0,007544	0,002822	8,5	11	0,010702	0,002048
bin7.dag	0,037039	0,005708	10	13	0,02552	0,003587
bin8.dag	0,129151	0,01197	11,5	15	0,051188	0,007238
bin9.dag	0,315043	0,030986	13	17	0,151375	0,015621
dag3.dag	0,001561	0,000918	7,5	8	0,001763	0,001163
di16.dag	0,001807	0,001983	10	10	0,002054	0,002171
di25.dag	0,003142	0,002918	13	13	0,004064	0,003104
di36.dag	0,004355	0,003606	16	16	0,00585	0,005283
di64.dag	0,014086	0,013233	22	22	0,015411	0,014719
di100.dag	0,036562	0,034061	28	28	0,027219	0,027183
di144.dag	0,07967	0,084553	34	34	0,058961	0,054256
di225.dag	0,218044	0,213233	43	43	0,131712	0,12135
di256.dag	0,338845	0,336679	46	46	0,156756	0,154204
di400.dag	1,195061	1,128244	58	58	0,45098	0,439668
gauss.dag	0,002312	0,001941	9,5	10	0,002039	0,001448
irr41.dag	0,010032	0,008309	15,3333	15,3333	0,014911	0,010902
n7.dag	0,000789	0,000697	4,5	5	0,001032	0,001065
n13.dag	0,000906	0,00094	4,5	5	0,001261	0,001145
tree2.dag	0,000575	0,000744	5	5	0,00089	0,001078
tree3.dag	0,001098	0,0011	7	7	0,001371	0,001293
tree4.dag	0,001858	0,001423	9	9	0,001393	0,00135
tree5.dag	0,003611	0,002937	11	11	0,002549	0,002117
tree6.dag	0,00631	0,006675	13	13	0,004158	0,00363
tree7.dag	0,012891	0,012047	15	15	0,010453	0,008669
tree8.dag	0,030878	0,032952	17	17	0,019197	0,016041
celbow.dag	0,021786	0,022119	28,625	28,625	0,021609	0,016788
cstanford.dag	0,023676	0,018173	27,7778	28,25	0,020185	0,01305
fft2-b-m.dag	0,054913	0,046042	14	14	0,112018	0,105718
iterative2-b-m.dag	0,030708	0,028901	13	13	0,033101	0,030766

Tabela 5.27: Testes da influência dos limites com infinitos processadores - Grupo de Instâncias Estruturadas

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
FAM11.dag	0,006215	0,004794	9,6	10	0,01524	0,014311
FAM12.dag	0,005103	0,005843	10,2	10,3333	0,010412	0,009644
FAM13.dag	0,003959	0,0043	12	12	0,004104	0,00384
FAM14.dag	0,004938	0,003868	9,75	10,125	0,012426	0,01127
FAM15.dag	0,002935	0,002533	10	10	0,002304	0,004191
FAM16.dag	0,003083	0,00271	10	10	0,003802	0,003327
FAM17.dag	0,00364	0,003916	9,5	10	0,005808	0,00474
FAM18.dag	0,003969	0,003444	11	11	0,009956	0,008831
FAM19.dag	0,004076	0,003364	9,5	10	0,008662	0,006436
FAM21.dag	0,003005	0,002842	7,4	7,4	0,006851	0,007677
FAM22.dag	0,003876	0,005162	10,1818	10,1818	0,014945	0,011021
FAM23.dag	0,002253	0,00176	6,5	7	0,005675	0,005148
FAM24.dag	0,002932	0,002418	9,5	10	0,007962	0,007772
FAM25.dag	0,003138	0,004094	8	8	0,011324	0,011847
FAM26.dag	0,002367	0,002388	10	10	0,003261	0,003209
FAM27.dag	0,002721	0,003749	7	7	0,011046	0,009655
FAM28.dag	0,003507	0,003312	9	9	0,012716	0,014638
FAM29.dag	0,002021	0,002171	8	8	0,00424	0,00416
FAM31.dag	0,006976	0,005843	10,25	10,3333	0,024524	0,024777
FAM32.dag	0,011289	0,012133	11,75	11,75	0,054831	0,033656
FAM33.dag	0,004891	0,004037	12	12	0,007983	0,006873
FAM34.dag	0,006459	0,003878	9,33333	10	0,017956	0,016213
FAM35.dag	0,007237	0,00687	10,75	10,75	0,022515	0,020707
FAM36.dag	0,007188	0,006393	9,66667	9,66667	0,01854	0,018632
FAM37.dag	0,005101	0,003982	10,5	10,5	0,008418	0,004147
FAM38.dag	0,006831	0,006262	12,3333	12,3333	0,023651	0,02213
FAM39.dag	0,004871	0,004809	8,66667	8,66667	0,011322	0,010017
FAM41.dag	0,017733	0,009215	12,3333	13	0,028114	0,019222
FAM42.dag	0,010526	0,008045	10,3333	11	0,009033	0,006211
FAM43.dag	0,01245	0,012132	12	12	0,029085	0,028488
FAM44.dag	0,018799	0,014511	12	12	0,028644	0,02784
FAM45.dag	0,007385	0,006781	12	12	0,008926	0,008295
FAM46.dag	0,008567	0,008237	13	13	0,013843	0,011386
FAM47.dag	0,01522	0,01223	13,25	13,25	0,016402	0,014294
FAM48.dag	0,018554	0,017037	15	15	0,050385	0,04442
FAM49.dag	0,018322	0,010147	13,5	14	0,016654	0,009677
FAM51.dag	0,011935	0,009737	12,5	13	0,02793	0,025973
FAM52.dag	0,016793	0,016548	14,3333	14,3333	0,026867	0,023694
FAM53.dag	0,021126	0,020314	15,28	15,28	0,036161	0,032114
FAM54.dag	0,007159	0,007299	11	11	0,009541	0,007829
FAM55.dag	0,024357	0,018089	9,77778	10	0,053752	0,047602
FAM56.dag	0,009702	0,008764	14	14	0,009182	0,007474
FAM57.dag	0,007734	0,007472	11	11	0,01551	0,017326
FAM58.dag	0,013797	0,014178	12	12	0,030662	0,029975
FAM59.dag	0,01549	0,013432	11,5	11,5	0,030136	0,029513
FAM61.dag	0,015606	0,014778	10,5	10,5	0,029043	0,028777
FAM62.dag	0,020119	0,018319	14,3333	14,3333	0,047117	0,047976
FAM63.dag	0,022522	0,017567	13,2	13,2	0,050137	0,045879
FAM64.dag	0,022785	0,021347	17	17	0,027668	0,023527
FAM65.dag	0,030246	0,029761	12,3125	12,3125	0,055501	0,055062
FAM66.dag	0,023054	0,024096	11	11	0,046508	0,039492
FAM67.dag	0,029519	0,032058	17,1429	17,1429	0,049555	0,047119
FAM68.dag	0,016379	0,016775	12,6667	12,6667	0,037963	0,034657
FAM69.dag	0,017087	0,016507	12,5	12,5	0,030564	0,030742
FAM71.dag	0,026854	0,020367	15,75	15,75	0,059685	0,057085
FAM72.dag	0,029985	0,030054	13,6667	13,6667	0,083247	0,087192
FAM73.dag	0,022553	0,017041	12	12	0,086916	0,077293
FAM74.dag	0,026195	0,026094	13,6667	13,6667	0,04827	0,045449
FAM75.dag	0,01382	0,013842	15	15	0,044106	0,04063
FAM76.dag	0,02068	0,019692	13,5	13,5	0,046647	0,046954
FAM77.dag	0,008225	0,007719	10	10	0,008697	0,006228
FAM78.dag	0,02153	0,019604	15	15	0,043584	0,050967
FAM79.dag	0,021086	0,020794	12,5	12,5	0,053403	0,046949
FAM110.dag	0,003936	0,00417	11,25	12	0,004405	0,003611
FAM210.dag	0,004223	0,003813	9,4	9,5	0,015799	0,015284
FAM310.dag	0,007791	0,004544	12,6667	13	0,025971	0,014588
FAM410.dag	0,00732	0,007646	11	11	0,01013	0,009731
FAM510.dag	0,008418	0,008708	11	11	0,0142	0,014276
FAM610.dag	0,01715	0,016048	13,5	13,5	0,041615	0,04353
FAM710.dag	0,009865	0,010201	12	12	0,020342	0,019417

Tabela 5.28: Testes da influência dos limites com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
r1.dag	0,023959	0,026052	13	13	0,022446	0,02022
r2.dag	0,043275	0,024365	12,2222	13	0,038134	0,023978
ran1.dag	0,061485	0,056606	14,25	14,25	0,07739	0,075282
ran2.dag	0,02152	0,02067	12,5	12,5	0,019852	0,021228
ran3.dag	0,343384	0,352284	13,75	13,75	0,270133	0,278714
ran4.dag	0,058675	0,056209	14	14,25	0,079791	0,070765
ran5.dag	0,028869	0,013996	8,5	9	0,061609	0,048127
ran6.dag	0,018364	0,02077	9	9	0,055768	0,052966
ran7.dag	0,112439	0,107376	11	11	0,184092	0,176888
ran8.dag	0,133471	0,10092	10,7	11	0,209095	0,185923
ran9.dag	0,009801	0,009005	7,5	7,5	0,03088	0,029896
ran10.dag	0,738563	0,515006	17,25	17,25	0,393535	0,413528
ran11.dag	1,957592	1,349354	17,625	17,625	1,468852	1,358473
ran12.dag	0,200697	0,173431	16,8333	17	0,245719	0,231883
ran13.dag	0,039215	0,036026	10	10	0,046906	0,033033
ran14.dag	0,139736	0,138256	15,25	15,25	0,170535	0,165646
ran15.dag	0,084316	0,075431	14,25	14,25	0,094064	0,084595
ran16.dag	0,025925	0,027931	11	11	0,038894	0,040085
ran17.dag	0,602977	0,566816	21	21	1,302875	1,144711
ran18.dag	0,182488	0,135286	14,1667	14,25	0,183041	0,164837
ran19.dag	0,018347	0,020405	11	11	0,016214	0,016152
ran20.dag	0,017628	0,015952	9	9	0,023744	0,023632

Tabela 5.29: Testes da influência dos limites com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
arv2.dag	0,00117	0,001177	5,16667	6	0,006034	0,005448
bin3.dag	0,000806	0,000794	4	5	0,001908	0,001358
bin4.dag	0,001597	0,001355	5,5	7	0,005234	0,005085
bin5.dag	0,00428	0,002257	7	9	0,011674	0,008709
bin6.dag	0,015067	0,003177	8,5	11	0,021443	0,017116
bin7.dag	0,076366	0,010381	10	13	0,06128	0,034416
bin8.dag	0,228446	0,022212	11,5	15	0,132163	0,069636
bin9.dag	0,487554	0,05048	13	17	0,3054	0,167133
dag3.dag	0,001465	0,001491	7,75	8	0,003019	0,003058
di16.dag	0,001472	0,002065	10	10	0,002348	0,003055
di25.dag	0,005851	0,004875	14,3333	15	0,021526	0,018277
di36.dag	0,004743	0,004034	16	16	0,013001	0,012222
di64.dag	0,018002	0,016518	22	22	0,04135	0,046092
di100.dag	0,042798	0,039096	28	28	0,167457	0,146417
di144.dag	0,099513	0,091265	34	34	0,385724	0,506152
di225.dag	0,918031	0,886632	43,4072	43,4072	23,409494	23,410749
di256.dag	0,432889	0,436269	46	46	1,924073	1,851894
di400.dag	1,691958	1,62753	58	58	4,930229	4,897076
gauss.dag	0,002566	0,002373	9,5	10	0,003043	0,002525
irr41.dag	0,012421	0,012597	15,3333	15,3333	0,025745	0,035206
n7.dag	0,001196	0,000999	4,5	5	0,002804	0,001309
n13.dag	0,001242	0,00124	4,5	5	0,002347	0,001473
tree2.dag	0,000649	0,000887	5	5	0,001093	0,001385
tree3.dag	0,001118	0,001555	7	7	0,004694	0,005398
tree4.dag	0,002243	0,002979	9	9	0,009164	0,011044
tree5.dag	0,006965	0,006591	11	11	0,016368	0,015888
tree6.dag	0,033596	0,029233	13	13	0,036164	0,036132
tree7.dag	0,0889	0,090375	15	15	0,140161	0,142287
tree8.dag	0,320949	0,32119	17	17	0,479838	0,479568
celbow.dag	0,049194	0,042373	28,625	28,625	0,040389	0,030989
cstanford.dag	0,036254	0,017615	27,7778	28,25	0,031252	0,014698
fft2-b-m.dag	0,279962	0,063042	14	16	0,357212	0,256095
iterative2-b-m.dag	0,543969	0,105005	17,7368	24	8,709764	0,207708

Tabela 5.30: Testes da influência dos limites com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
FAM12.dag	0,065424	0,030569	13,0694	16	0,420295	0,095695
FAM13.dag	0,029889	0,015027	13,9877	16	0,141099	0,055183
FAM14.dag	0,029031	0,008221	12,4548	15	0,143395	0,047737
FAM15.dag	0,047207	0,023549	12,8235	15	0,250532	0,089865
FAM16.dag	0,019129	0,008024	13,6889	15	0,116922	0,067717
FAM17.dag	0,02731	0,007862	12,9797	15	0,119782	0,031894
FAM18.dag	0,052153	0,019192	13,5107	16	0,341742	0,048162
FAM19.dag	0,037347	0,009703	13,273	15	0,287109	0,060052
FAM21.dag	0,038275	0,009402	11,6229	15	0,327202	0,057611
FAM22.dag	0,031225	0,012855	12,8278	16	0,328778	0,047615
FAM23.dag	0,007521	0,004713	8,325	10	0,071192	0,025703
FAM24.dag	0,018406	0,008346	11,7935	15	0,158425	0,04382
FAM25.dag	0,007451	0,004303	9,21154	11	0,04701	0,021924
FAM26.dag	0,020869	0,009932	12,3077	15	0,362792	0,111707
FAM27.dag	0,038123	0,013925	10,7413	15	0,378171	0,066109
FAM28.dag	0,0261	0,011434	12,0617	15	0,315069	0,070326
FAM29.dag	0,021286	0,009388	11,482	15	0,425891	0,038676
FAM31.dag	0,042244	0,021785	12,3543	14	0,128629	0,061543
FAM32.dag	0,066284	0,027467	13,4008	16	0,270356	0,072541
FAM33.dag	0,013918	0,010249	13,0566	14	0,068567	0,048475
FAM34.dag	0,037979	0,01294	10,8773	13	0,15629	0,077903
FAM35.dag	0,059303	0,026357	12,3509	14	0,269816	0,082118
FAM36.dag	0,097149	0,027467	13,324	17	0,60204	0,086894
FAM37.dag	0,011093	0,00901	11,0127	11,0938	0,040749	0,031719
FAM38.dag	0,013075	0,009669	13,0623	14	0,038454	0,030071
FAM39.dag	0,029354	0,011008	11,0924	13	0,343534	0,060826
FAM41.dag	0,133656	0,053629	15,0735	18	0,533389	0,15704
FAM42.dag	0,035345	0,025405	12,0667	13	0,107836	0,044617
FAM43.dag	0,030371	0,022875	12,513	13	0,095946	0,065807
FAM44.dag	0,171111	0,0527	14,7986	18	0,947121	0,215117
FAM45.dag	0,344653	0,060368	15,0192	19	1,026099	0,138967
FAM46.dag	0,085338	0,063355	16,0781	18	0,666522	0,189193
FAM47.dag	0,088193	0,033266	15,6366	18	0,31482	0,100752
FAM48.dag	0,061331	0,035551	15,0909	16	0,183265	0,078374
FAM49.dag	0,148287	0,07109	16,3631	18	1,147251	0,198885
FAM51.dag	0,071634	0,033843	13,925	15	0,22722	0,079783
FAM52.dag	0,063878	0,036555	15,4776	17	0,172665	0,100649
FAM53.dag	0,054543	0,031144	16,6619	17	0,13175	0,09626
FAM54.dag	0,025751	0,024512	11,9524	13	0,094476	0,046775
FAM55.dag	0,086093	0,039739	11,5202	13	0,543948	0,134837
FAM56.dag	0,025642	0,02728	14	14	0,075907	0,074842
FAM57.dag	0,046972	0,025154	12,2404	13	0,141522	0,062523
FAM58.dag	0,028464	0,026269	12	12	0,054443	0,051829
FAM59.dag	0,0724	0,040176	12,772	13,274	0,418691	0,185349
FAM61.dag	0,050338	0,024146	11,8338	13	0,104472	0,056216
FAM62.dag	0,078863	0,043207	15,8723	17	0,291128	0,147563
FAM63.dag	0,030922	0,024159	13,2	13,2	0,056081	0,045924
FAM64.dag	0,053568	0,043621	17,0314	17,0769	0,120985	0,114503
FAM65.dag	0,15802	0,063464	13,803	15,0863	0,76936	0,261354
FAM66.dag	0,110752	0,043194	13,1196	15	0,511331	0,146922
FAM67.dag	0,046699	0,044649	17,1429	17,1429	0,064927	0,059883
FAM68.dag	0,091425	0,054349	14,0088	15	0,16211	0,128224
FAM69.dag	0,03893	0,030954	12,7381	13	0,084329	0,084851
FAM71.dag	0,052123	0,045472	15,75	15,75	0,097344	0,108251
FAM72.dag	0,067489	0,067309	13,6852	13,6852	0,30433	0,211471
FAM73.dag	0,074966	0,046382	12,744	13,0417	0,277146	0,138371
FAM74.dag	0,109855	0,084059	14,7778	15,3659	0,295302	0,293074
FAM75.dag	0,020366	0,020563	15	15	0,048436	0,047586
FAM76.dag	0,04092	0,046445	13,5	13,5	0,091027	0,087686
FAM77.dag	0,105686	0,035992	11,4207	13	0,385165	0,139221
FAM78.dag	0,091659	0,078217	15,574	15,574	0,237516	0,250913
FAM79.dag	0,081258	0,069225	12,8955	13,1569	0,216831	0,159293
FAM110.dag	0,033684	0,017479	13,5626	16	0,093865	0,05644
FAM210.dag	0,010301	0,007413	10,19	11	0,043509	0,029043
FAM310.dag	0,042423	0,024468	14	15	0,168612	0,076319
FAM410.dag	0,142316	0,073763	14,4823	18	1,01832	0,277848
FAM510.dag	0,028026	0,021458	12,1402	13	0,065835	0,048846
FAM610.dag	0,060709	0,034831	14,5556	15	0,139307	0,116377
FAM710.dag	0,126611	0,056897	13,3827	14,1928	0,671665	0,241923

Tabela 5.31: Testes da influência dos limites com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
r1.dag	0,260518	0,169521	12,9551	14	0,504602	0,296195
r2.dag	0,108443	0,066904	12,4272	13	0,11029	0,091731
ran1.dag	0,656254	0,41458	14,4392	14,4392	0,394636	0,406367
ran2.dag	0,070658	0,055415	12,5664	13	0,11469	0,105146
ran3.dag	0,288495	0,292582	13,75	13,75	0,335138	0,340769
ran4.dag	0,337334	0,297126	15,5451	16,098	2,246106	1,018681
ran5.dag	0,035142	0,020451	8,5	9	0,060629	0,059204
ran6.dag	0,027329	0,026835	9	9	0,07865	0,073099
ran7.dag	0,142368	0,139406	11	11	0,275867	0,272588
ran8.dag	0,142913	0,110142	10,7	11	0,228398	0,253141
ran9.dag	0,011749	0,007613	7,5	8	0,033418	0,034736
ran10.dag	0,500664	0,627745	17,25	17,25	0,516798	0,517028
ran11.dag	1,933881	2,116073	17,625	17,625	1,814407	1,683578
ran12.dag	0,460985	0,520717	16,8333	17	0,598463	0,526979
ran13.dag	0,041245	0,035899	10	10	0,103714	0,089281
ran14.dag	0,167606	0,159294	15,25	15,25	0,280157	0,261836
ran15.dag	0,31348	0,320103	14,2	15	0,32635	0,303533
ran16.dag	0,046687	0,047837	11	11	0,100537	0,100223
ran17.dag	0,52405	0,522339	21	21	1,588017	1,629053
ran18.dag	0,417185	0,45207	14,1667	14,25	0,438352	0,432413
ran19.dag	0,03629	0,036795	11	11	0,063869	0,066619
ran20.dag	0,046919	0,031502	9,02857	10	0,135405	0,063806

Tabela 5.32: Testes da influência dos limites com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
bin4.dag	0,00293	0,002143	6,5	9	0,010372	0,007386
bin5.dag	0,006757	0,004519	7,9375	10	0,039917	0,015535
bin6.dag	0,023736	0,013992	9,3125	11,0938	0,042688	0,035823
bin7.dag	0,099412	0,030106	10,7812	13	0,082332	0,048909
bin8.dag	0,247188	0,07156	12,1641	15	0,169972	0,123142
bin9.dag	0,430288	0,140207	13,6406	17	0,479755	0,323692
di64.dag	0,523779	0,151792	30,7274	34	2,146691	0,459162
di100.dag	4,019083	0,601183	42,9236	52	24,911845	1,943047
di144.dag	4,517628	1,252743	46,7001	51,6667	256,139968	3,40097
di225.dag	46,959469	6,212029	64,9486	78,6667	600,07322	17,628576
di256.dag	32,157091	6,5952	62,7775	69,451	600,06046	20,676178
di400.dag	102,992267	42,964749	78,6234	87,0769	600,274451	95,327225
irr41.dag	0,084891	0,036226	20,0098	23	0,888893	0,112833
n13.dag	0,001819	0,001499	5,0625	7	0,007585	0,003516
tree3.dag	0,002301	0,001743	8,33333	9	0,010973	0,008592
tree4.dag	0,005224	0,005833	9,75	10	0,033088	0,026054
tree5.dag	0,017499	0,017215	11,3	11,3	0,069603	0,069598
tree6.dag	0,049541	0,052409	13	13	0,08888	0,083401
tree7.dag	0,207661	0,210129	15	15	0,286666	0,278769
tree8.dag	0,419675	0,402789	17	17	0,670298	0,629616
celbow.dag	0,049802	0,044802	28,625	28,625	0,059284	0,048071
cstanford.dag	0,042592	0,031814	27,7778	28,25	0,04582	0,032069
fft2-b-m.dag	2,441831	0,801475	19,8161	28	600,042695	2,437217
iterative2-b-m.dag	2,48252	0,421344	29,4263	47	160,137934	1,1995

Tabela 5.33: Testes da influência dos limites com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
FAM31.dag	0,375126	0,061448	17,4986	26	1,734435	0,285761
FAM32.dag	0,532543	0,054412	18,9063	27	2,376803	0,275203
FAM33.dag	0,265482	0,050514	19,1005	25	1,396285	0,216825
FAM34.dag	0,420044	0,043349	16,7895	25	2,50488	0,276791
FAM35.dag	0,289994	0,083246	18,9458	25	1,409055	0,285275
FAM37.dag	0,336946	0,069166	17,5013	25	1,564354	0,121827
FAM38.dag	0,261429	0,075586	19,1148	25	1,276595	0,25844
FAM39.dag	0,399279	0,067421	16,9122	25	2,079617	0,258068
FAM42.dag	0,298006	0,144053	17,5971	25	3,295505	0,431199
FAM43.dag	0,72909	0,123313	18,6759	25	2,634062	0,383602
FAM48.dag	0,622007	0,098182	20,7333	26	2,343194	0,348697
FAM51.dag	0,931538	0,167672	23,6565	35	4,865399	0,540285
FAM52.dag	1,111488	0,219422	24,5472	36	4,681067	0,471021
FAM53.dag	1,3352	0,138801	25,5247	36	7,313994	0,493186
FAM54.dag	0,639538	0,094473	16,998	24	2,959754	0,283332
FAM55.dag	0,793437	0,099387	16,8789	24	3,09386	0,274582
FAM56.dag	0,394192	0,093524	18,589	24	1,85132	0,315217
FAM57.dag	0,647134	0,066793	17,9043	24	2,35911	0,258019
FAM58.dag	0,588967	0,08413	18,2374	25	4,114179	0,395307
FAM59.dag	0,718644	0,082782	17,4732	24	3,31084	0,304866
FAM61.dag	0,98528	0,100516	18,2673	27	4,074889	0,467563
FAM62.dag	1,487027	0,266841	27,4128	40	8,566067	0,822849
FAM63.dag	0,439657	0,088209	16,6737	21	2,031939	0,246969
FAM64.dag	1,070263	0,0926	22,0299	30	4,962527	0,358044
FAM65.dag	1,264562	0,15119	20,678	28	3,886174	0,448388
FAM66.dag	1,077852	0,167729	18,8312	27	6,313012	0,501707
FAM67.dag	0,842905	0,078566	22,0158	28	4,070869	0,305805
FAM68.dag	0,990986	0,162701	19,3694	27	3,979995	0,544516
FAM69.dag	0,816576	0,131395	19,8445	27	2,855851	0,423892
FAM71.dag	0,758557	0,139565	19,8405	24,1905	4,54106	0,511929
FAM72.dag	0,585569	0,100002	18,7422	23	1,986873	0,325114
FAM73.dag	0,858332	0,187719	17,4039	24	3,639331	0,289287
FAM74.dag	2,149378	0,240654	21,9588	32	15,149236	0,937857
FAM75.dag	1,192696	0,170504	21,5412	30	5,591102	0,520229
FAM76.dag	1,037694	0,138685	19,0225	24	3,594081	0,482827
FAM77.dag	1,525447	0,099448	16,3444	23	4,820269	0,39385
FAM78.dag	1,579302	0,285298	23,3882	32	9,219371	0,843666
FAM79.dag	0,79317	0,093489	17,5096	23	2,743542	0,372684
FAM310.dag	0,482398	0,067595	19,226	26	1,780898	0,158367
FAM510.dag	0,484637	0,081222	17,3253	24	2,149565	0,287462
FAM610.dag	0,960973	0,168621	20,4222	27	3,382744	0,530924
FAM710.dag	1,39979	0,29025	20,5291	30	6,779021	0,709076

Tabela 5.34: Testes da influência dos limites com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
r1.dag	1,447876	0,722756	19,6327	27	40,44373	3,236727
r2.dag	0,795609	0,396213	18,6557	25	16,733605	1,050407
ran1.dag	1,428098	0,788854	21,7709	28	44,796058	3,452571
ran2.dag	0,737485	0,317754	19,6799	27	9,184361	1,032928
ran3.dag	1,571807	0,793936	19,5327	26	30,984398	2,153632
ran4.dag	2,077453	0,838425	25,9319	36	40,455988	4,107531
ran5.dag	0,700885	0,07096	10,9358	16	1,87362	0,26368
ran6.dag	1,629045	0,140534	12,2342	18	4,948524	0,660126
ran7.dag	2,594155	1,273294	15,8431	22	53,974601	2,792419
ran8.dag	1,205667	0,706054	13,8127	19	16,694861	2,636953
ran9.dag	0,355751	0,048373	10,5957	16	1,353246	0,119755
ran10.dag	4,169624	2,093202	26,0144	29	93,816758	5,603908
ran11.dag	24,346487	16,63825	23,6938	27	600,130325	28,222354
ran12.dag	3,638636	13,535674	23,5612	26	262,125811	11,725771
ran13.dag	10,446639	0,916833	13,2567	18	21,092512	2,761172
ran14.dag	4,474346	14,811567	20,6731	25	62,122985	11,089474
ran15.dag	2,520458	1,194263	22,4286	31	172,458784	2,705327
ran16.dag	0,945643	0,633456	14,6143	21	7,639397	1,873724
ran17.dag	20,152526	12,209306	26,8411	29	410,564596	17,286972
ran18.dag	6,265786	1,527327	20,9516	26	43,163708	8,216291
ran19.dag	0,567197	0,451004	14,4687	20	5,304364	1,12769
ran20.dag	0,75254	0,476365	13,6476	20	9,347411	0,913157

Tabela 5.35: Testes da influência dos limites com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	LP Z s t	LP Z c t	LP Z s	LP Z c	IP Z s t	IP Z c t
bin5.dag	0,034829	0,009545	10,75	17	0,226412	0,07875
bin6.dag	0,109733	0,018221	12,2136	18	0,298272	0,062926
bin7.dag	0,322512	0,095885	13,5667	18,2188	0,880569	0,319011
bin8.dag	0,515105	0,319055	15,0563	18,5938	2,372302	0,962765
bin9.dag	1,450749	1,064751	16,4272	19,1055	7,30609	7,707222
di256.dag	600,593428	21,062459	-	130	600,524503	64,50238
di400.dag	601,747406	170,098355	-	202	601,577783	593,293926
tree4.dag	0,033177	0,010231	13,4826	17	0,371384	0,03822
tree5.dag	0,115048	0,029287	15,4589	18	1,042426	0,111304
tree6.dag	0,269104	0,13716	16,7602	18,9231	4,183558	0,682533
tree7.dag	0,575266	0,377821	18,6364	20	9,200493	2,893864
tree8.dag	1,224977	1,326176	19,9022	20,3333	34,516896	24,692261
celbow.dag	0,417315	0,176314	32,5877	35	1,018829	0,475886
cstanford.dag	0,236698	0,075712	30,4801	32	0,597257	0,178275
fft2-b-m.dag	17,712311	4,154984	31,384	52	600,120455	8,712027
iterative2-b-m.dag	8,596053	1,150628	50,9879	90	600,915678	2,294928

Tabela 5.36: Testes da influência dos limites com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Estruturadas

Capítulo 6

Conclusão e Trabalhos Futuros

Neste trabalho estudamos o Problema de escalonamento de tarefas UET-UCT com custos unitários de execução e comunicação. Inicialmente, foi apresentada uma definição formal do problema. Junto a essa definição foi feita uma revisão bibliográfica da literatura, que para este tipo de problema é bem vasta e diversificada. Localizamos o problema estudado no universo de problemas de escalonamento. Apresentamos as complexidades computacionais das versões, limitada e ilimitada, do problema abordado nesta Dissertação, bem como algoritmos aproximativos existentes para cada versão. Mostramos ainda algumas complexidades computacionais de problemas semelhantes ao abordado neste trabalho.

Na busca da resolução do problema de forma eficiente, partimos dos limites inferior e superior, pois os melhores limites aplicados neste problema não eram satisfatórios. Esta observação é confirmada ao compararmos os limites existentes com o ótimo.

Para o limite inferior, fizemos alterações no limite $LB_{i,j}$ proposto na literatura. Estas alterações fizeram com que fosse levada em consideração a quantidade de recurso disponível (processadores). Estas alterações tornaram o limite consideravelmente melhor nos testes onde restringimos o número de processadores. Mesmo com este ganho significativo, para algumas instâncias, $LB_{i,j}$ teve maior dificuldade em chegar ao ótimo. Devido a isso, aplicamos os limites $IFB_{i,j}$ e $FFB_{i,j}$. A implementação deste último, partindo de $LB_{i,j}$ demonstrou um ganho significativo, justamente no ponto em que desejávamos, quando o número de processadores impõe restrições reais ao problema.

Estando o limite inferior satisfatórios na maioria dos testes, partimos para o limite superior. Claramente, a heurística $CP/MISF$ possui deficiências na alocação de tarefas. Em alguns casos, processadores ficam ociosos e não executam tarefas que poderiam ser executadas. Este problema foi resolvido com a heurística gulosa ISH. Este algoritmo se mostrou bastante eficiente, nos dois pontos principais, valor de *makespan* e custo computacional. Mesmo o algoritmo anterior sendo satisfatório, buscamos uma meta-heurística de base evolucionária para confrontarmos com ISH. Em alguns casos, a meta-heurística PSO conseguiu obter um limite de maior qualidade em relação à ISH, porém com um custo computacional razoavelmente mais elevado. Portanto, atestamos a superioridade da heurística ISH.

De posse de bons limites, inferior e superior, buscamos a resolução do problema pela formulação matemática base deste problema na literatura, a Formulação X. Claramente, a Formulação X é dependente de bons limites para que tenha um bom desempenho, porém, mesmo com os limites fortalecidos e a aplicação de outras estratégias, a Formulação X não se mostrou eficiente na resolução do problema.

A busca por novas formulações resultou na concepção da Formulação Y, esta semelhante à utilizada em [MK97]. O modelo gerado por Y é bastante grande, visto que possuímos muitas variáveis e muitas restrições. Uma transição natural para Y_{ref} acontece ao observarmos a influência de uma variável $y_{i,t}$, de uma tarefa i e tempo t, nas variáveis $y_{i,t'}$, com $t' < t$. Esta observação indica que Y pode ser reforçada. Neste momento, temos uma formulação que pode ser definida sem as variáveis w, porém estas se mostram importantes na relaxação linear de Y_{ref}. Surgem então as restrições (4.54) e (4.55), que são exponenciais. Elas conseguem eliminar as variáveis w mesmo na relaxação linear. Uma outra formulação é criada com estas restrições, a Y_{exp}. Propomos, então, um algoritmo de separação para estas restrições exponenciais.

Com o intuito de dominuir o modelo Y_{ref}, construímos, através de uma mudança de variáveis, a Formulação Z, que possui menos variáveis e restrições. De forma análoga, obtemos Z_{exp}, a partir de Y_{exp}.

Fizemos um estudo comparativo das formulações apresentadas sob dois aspectos, teórico e computacional. Teoricamente, obtivemos algumas relações entre as formulações apresentadas e modificações delas. Computacionalmente, o estudo permitiu observar que a Formulação Z é a mais adequada para a resolução deste problema, independente do número de processadores. Sua relaxação linear possui boa qualidade e seu custo computacional é baixo.

Do ponto de vista de resolução matemática, acreditamos ter chegado a um ponto satisfatório. Portanto, como trabalho futuro, gostaríamos de investigar a influência da melhoria da qualidade da relaxação linear no fator de aproximação deste problema, visto que o melhor algoritmo aproximativo existente para o caso ilimitado é baseado na relaxação linear da Formulação X. Como obtivemos uma formulação com relaxação melhor, é provável que consigamos melhorar o fator de aproximação do problema ilimitado. Feito isto, melhoraríamos também o fator de aproximação da versão limitada, pois este é baseado no fator de aproximação do caso ilimitado.

Apêndice A

Instâncias e Ambiente Computacional

A.1 Instâncias utilizadas

As instâncias utilizadas neste trabalho foram classificadas em três grupos diferentes. A primeira classe de instâncias, denominada Grupo de Instâncias Aleatórias 1, é constituída por instâncias geradas pelo gerador de instâncias *ProGen* [RKA99].

A segunda classe de instâncias compreende as mesmas instâncias utilizadas em [CCMP02]. Também são instâncias aleatórias geradas da seguintes forma. Primeiro, escolha o número de tarefas. Então, para cada tarefa i, escolha o número de sucessores imediatos e eleja os candidatos a serem sucessores de i, ambos aleatoriamente e utilizando uma distribuição uniforme. Não permita que uma tarefa eleita j seja uma predecessora de i e nem dos predecessores de i.

A terceira classe de instâncias é constituída por instâncias que possuem uma estrutura conhecida, por exemplo:

- $binh$: árvores binárias completas de altura h;
- $treeh$: são as instâncias $binh$, invertendo o sentido dos arcos;
- din: grafos diamantes com n tarefas;
- $irr41$: grafo irregular com 41 tarefas
- $fft2$: Fast Fourier transformation;

A seguir, alguns exemplos dos grafos citados e detalhes das informações de cada instância utilizada, de acordo com o número de processadores.

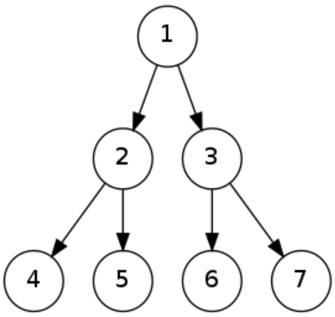


Figura A.1: Instância bin3.dag

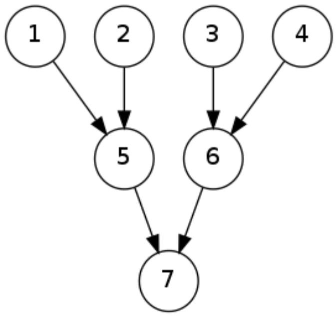


Figura A.2: Instância tree2.dag

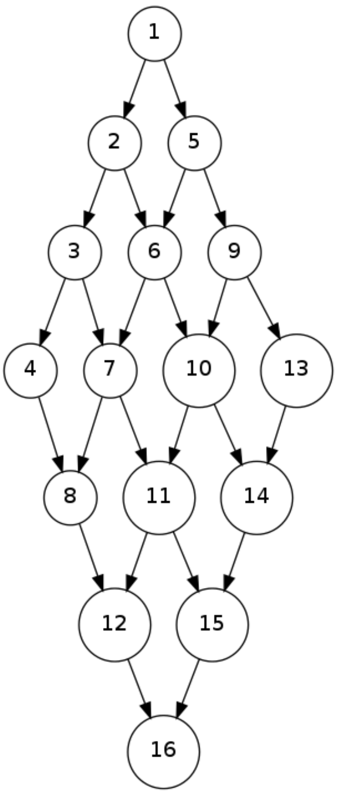


Figura A.3: Instância di16.dag

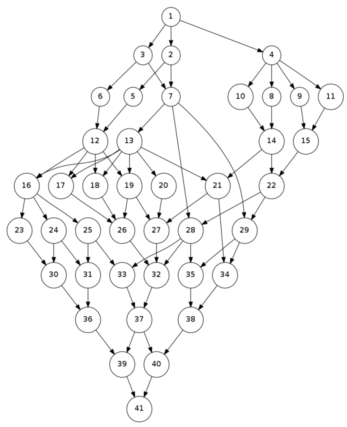


Figura A.4: Instância irr41.dag

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
FAM11.dag	30	7	31	9	10
FAM12.dag	30	9	31	10	11
FAM13.dag	30	9	31	12	12
FAM14.dag	30	10	31	10	11
FAM15.dag	30	8	31	10	10
FAM16.dag	30	8	31	10	10
FAM17.dag	30	8	31	10	10
FAM18.dag	30	9	31	11	11
FAM19.dag	30	10	31	10	10
FAM21.dag	30	10	31	7	8
FAM22.dag	30	9	31	10	11
FAM23.dag	30	12	31	7	7
FAM24.dag	30	11	31	10	10
FAM25.dag	30	13	31	8	9
FAM26.dag	30	11	31	10	10
FAM27.dag	30	11	31	7	7
FAM28.dag	30	11	31	9	10
FAM29.dag	30	11	31	8	8
FAM31.dag	50	19	51	10	11
FAM32.dag	50	16	51	11	13
FAM33.dag	50	16	51	12	12
FAM34.dag	50	19	51	10	10
FAM35.dag	50	16	51	10	11
FAM36.dag	50	14	51	9	10
FAM37.dag	50	21	51	10	11
FAM38.dag	50	17	51	12	13
FAM39.dag	50	19	51	8	9
FAM41.dag	50	15	51	13	13
FAM42.dag	50	17	51	11	11
FAM43.dag	50	16	51	12	13
FAM44.dag	50	14	51	12	13
FAM45.dag	50	14	51	12	12
FAM46.dag	50	13	51	13	13
FAM47.dag	50	15	51	13	14
FAM48.dag	50	16	51	15	16
FAM49.dag	50	13	51	14	14
FAM51.dag	70	21	71	13	13
FAM52.dag	70	21	71	14	15
FAM53.dag	70	21	71	15	16
FAM54.dag	70	27	71	11	11
FAM55.dag	70	25	71	10	10
FAM56.dag	70	26	71	14	14
FAM57.dag	70	24	71	11	11
FAM58.dag	70	28	71	12	12
FAM59.dag	70	25	71	11	12
FAM61.dag	80	29	81	10	11
FAM62.dag	80	23	81	14	15
FAM63.dag	80	32	81	13	14
FAM64.dag	80	26	81	16	17
FAM65.dag	80	27	81	12	13
FAM66.dag	80	26	81	11	12
FAM67.dag	80	28	81	17	18
FAM68.dag	80	26	81	12	13
FAM69.dag	80	28	81	12	13
FAM71.dag	90	34	91	15	16
FAM72.dag	90	33	91	13	15
FAM73.dag	90	34	91	12	13
FAM74.dag	90	29	91	13	14
FAM75.dag	90	29	91	15	15
FAM76.dag	90	37	91	13	14
FAM77.dag	90	32	91	10	10
FAM78.dag	90	29	91	15	15
FAM79.dag	90	32	91	12	13
FAM110.dag	30	9	31	12	12
FAM210.dag	30	13	31	9	10
FAM310.dag	50	19	51	13	13
FAM410.dag	50	14	51	11	11
FAM510.dag	70	25	71	11	11
FAM610.dag	80	25	81	13	14
FAM710.dag	90	29	91	12	12

Tabela A.1: Instâncias para testes com infinitos processadores - Grupo de Instâncias Aleatórias 1

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
r1.dag	135	41	136	12	13
r2.dag	98	33	99	13	13
ran1.dag	140	42	141	13	15
ran2.dag	80	29	81	12	13
ran3.dag	153	48	154	13	17
ran4.dag	108	30	109	14	15
ran5.dag	170	94	171	9	9
ran6.dag	223	110	224	8	9
ran7.dag	298	119	299	10	12
ran8.dag	256	118	257	11	12
ran9.dag	124	65	125	7	8
ran10.dag	286	83	287	17	19
ran11.dag	510	153	511	17	19
ran12.dag	310	99	311	17	18
ran13.dag	357	167	358	10	10
ran14.dag	364	121	365	15	16
ran15.dag	152	47	153	14	15
ran16.dag	186	76	187	10	11
ran17.dag	546	158	547	21	22
ran18.dag	234	74	235	14	15
ran19.dag	154	64	155	10	11
ran20.dag	154	68	155	9	9

Tabela A.2: Instâncias para testes com infinitos processadores - Grupo de Instâncias Aleatórias 2

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
arv2.dag	9	5	10	4	5
bin3.dag	7	4	8	5	5
bin4.dag	15	8	16	7	7
bin5.dag	31	16	32	9	9
bin6.dag	63	32	64	11	11
bin7.dag	127	64	128	13	13
bin8.dag	255	128	256	15	15
bin9.dag	511	256	512	17	17
dag3.dag	13	5	14	8	8
di16.dag	16	4	17	10	10
di25.dag	25	5	26	13	13
di36.dag	36	6	37	16	16
di64.dag	64	8	65	22	22
di100.dag	100	10	101	28	28
di144.dag	144	12	145	34	34
di225.dag	225	15	226	43	43
di256.dag	256	16	257	46	46
di400.dag	400	20	401	58	58
gauss.dag	18	5	19	9	10
irr41.dag	41	11	42	15	16
n7.dag	7	4	8	5	5
n13.dag	13	8	14	5	5
tree2.dag	7	4	8	4	5
tree3.dag	15	8	16	6	7
tree4.dag	31	16	32	8	9
tree5.dag	63	32	64	10	11
tree6.dag	127	64	128	12	13
tree7.dag	255	128	256	14	15
tree8.dag	511	256	512	16	17
celbow.dag	103	31	104	28	30
cstanford.dag	90	30	91	28	29
fft2-b-m.dag	194	32	195	14	15
iterative2-b-m.dag	262	26	263	13	14

Tabela A.3: Instâncias para testes com infinitos processadores - Grupo de Instâncias Estruturadas

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
FAM11.dag	30	7	3	10	11
FAM12.dag	30	9	4	10	11
FAM13.dag	30	9	4	12	12
FAM14.dag	30	10	5	10	11
FAM15.dag	30	8	4	10	10
FAM16.dag	30	8	4	10	10
FAM17.dag	30	8	4	10	10
FAM18.dag	30	9	4	11	11
FAM19.dag	30	10	5	10	10
FAM21.dag	30	10	5	7	8
FAM22.dag	30	9	4	10	11
FAM23.dag	30	12	6	7	7
FAM24.dag	30	11	5	10	10
FAM25.dag	30	13	6	8	9
FAM26.dag	30	11	5	10	10
FAM27.dag	30	11	5	7	8
FAM28.dag	30	11	5	9	10
FAM29.dag	30	11	5	8	8
FAM31.dag	50	19	9	10	11
FAM32.dag	50	16	8	11	13
FAM33.dag	50	16	8	12	12
FAM34.dag	50	19	9	10	10
FAM35.dag	50	16	8	10	11
FAM36.dag	50	14	7	9	10
FAM37.dag	50	21	10	10	11
FAM38.dag	50	17	8	12	13
FAM39.dag	50	19	9	8	9
FAM41.dag	50	15	7	13	13
FAM42.dag	50	17	8	11	11
FAM43.dag	50	16	8	12	13
FAM44.dag	50	14	7	12	13
FAM45.dag	50	14	7	12	12
FAM46.dag	50	13	6	13	13
FAM47.dag	50	15	7	13	14
FAM48.dag	50	16	8	15	16
FAM49.dag	50	13	6	14	14
FAM51.dag	70	21	10	13	13
FAM52.dag	70	21	10	14	15
FAM53.dag	70	21	10	15	16
FAM54.dag	70	27	13	11	11
FAM55.dag	70	25	12	10	11
FAM56.dag	70	26	13	14	14
FAM57.dag	70	24	12	11	11
FAM58.dag	70	28	14	12	12
FAM59.dag	70	25	12	11	12
FAM61.dag	80	29	14	10	11
FAM62.dag	80	23	11	14	15
FAM63.dag	80	32	16	13	14
FAM64.dag	80	26	13	16	17
FAM65.dag	80	27	13	12	13
FAM66.dag	80	26	13	11	12
FAM67.dag	80	28	14	17	18
FAM68.dag	80	26	13	12	13
FAM69.dag	80	28	14	12	13
FAM71.dag	90	34	17	15	16
FAM72.dag	90	33	16	13	15
FAM73.dag	90	34	17	12	13
FAM74.dag	90	29	14	13	14
FAM75.dag	90	29	14	15	15
FAM76.dag	90	37	18	13	14
FAM77.dag	90	32	16	10	10
FAM78.dag	90	29	14	15	15
FAM79.dag	90	32	16	12	13
FAM110.dag	30	9	4	12	12
FAM210.dag	30	13	6	9	10
FAM310.dag	50	19	9	13	13
FAM410.dag	50	14	7	11	11
FAM510.dag	70	25	12	11	11
FAM610.dag	80	25	12	13	14
FAM710.dag	90	29	14	12	12

Tabela A.4: Instâncias para testes com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
r1.dag	135	41	20	12	13
r2.dag	98	33	16	13	13
ran1.dag	140	42	21	13	15
ran2.dag	80	29	14	12	13
ran3.dag	153	48	24	13	17
ran4.dag	108	30	15	14	15
ran5.dag	170	94	47	9	9
ran6.dag	223	110	55	8	9
ran7.dag	298	119	59	10	12
ran8.dag	256	118	59	11	12
ran9.dag	124	65	32	7	8
ran10.dag	286	83	41	17	19
ran11.dag	510	153	76	17	19
ran12.dag	310	99	49	17	18
ran13.dag	357	167	83	10	10
ran14.dag	364	121	60	15	16
ran15.dag	152	47	23	14	15
ran16.dag	186	76	38	10	11
ran17.dag	546	158	79	21	22
ran18.dag	234	74	37	14	15
ran19.dag	154	64	32	10	11
ran20.dag	154	68	34	9	9

Tabela A.5: Instâncias para testes com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
arv2.dag	9	5	2	6	6
bin3.dag	7	4	2	5	5
bin4.dag	15	8	4	7	7
bin5.dag	31	16	8	9	9
bin6.dag	63	32	16	11	11
bin7.dag	127	64	32	13	13
bin8.dag	255	128	64	15	15
bin9.dag	511	256	128	17	17
dag3.dag	13	5	2	8	8
di16.dag	16	4	2	10	10
di25.dag	25	5	2	15	15
di36.dag	36	6	3	16	16
di64.dag	64	8	4	22	22
di100.dag	100	10	5	28	28
di144.dag	144	12	6	34	34
di225.dag	225	15	7	43	45
di256.dag	256	16	8	46	46
di400.dag	400	20	10	58	58
gauss.dag	18	5	2	10	10
irr41.dag	41	11	5	15	16
n7.dag	7	4	2	5	5
n13.dag	13	8	4	5	5
tree2.dag	7	4	2	5	5
tree3.dag	15	8	4	6	7
tree4.dag	31	16	8	8	9
tree5.dag	63	32	16	10	11
tree6.dag	127	64	32	12	13
tree7.dag	255	128	64	14	15
tree8.dag	511	256	128	16	17
celbow.dag	103	31	15	28	30
cstanford.dag	90	30	15	28	29
fft2-b-m.dag	194	32	16	16	16
iterative2-b-m.dag	262	26	13	24	24

Tabela A.6: Instâncias para testes com $m = \lfloor \omega(\prec)/2 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
FAM12.dag	30	9	2	16	17
FAM13.dag	30	9	2	16	16
FAM14.dag	30	10	2	15	15
FAM15.dag	30	8	2	15	16
FAM16.dag	30	8	2	15	15
FAM17.dag	30	8	2	15	15
FAM18.dag	30	9	2	16	16
FAM19.dag	30	10	2	15	15
FAM21.dag	30	10	2	15	15
FAM22.dag	30	9	2	16	17
FAM23.dag	30	12	3	10	10
FAM24.dag	30	11	2	15	15
FAM25.dag	30	13	3	11	11
FAM26.dag	30	11	2	15	16
FAM27.dag	30	11	2	15	16
FAM28.dag	30	11	2	15	16
FAM29.dag	30	11	2	15	15
FAM31.dag	50	19	4	14	14
FAM32.dag	50	16	4	16	16
FAM33.dag	50	16	4	14	14
FAM34.dag	50	19	4	13	14
FAM35.dag	50	16	4	14	15
FAM36.dag	50	14	3	17	17
FAM37.dag	50	21	5	11	12
FAM38.dag	50	17	4	14	14
FAM39.dag	50	19	4	13	13
FAM41.dag	50	15	3	18	18
FAM42.dag	50	17	4	13	13
FAM43.dag	50	16	4	13	14
FAM44.dag	50	14	3	18	18
FAM45.dag	50	14	3	19	19
FAM46.dag	50	13	3	18	18
FAM47.dag	50	15	3	18	18
FAM48.dag	50	16	4	16	17
FAM49.dag	50	13	3	18	19
FAM51.dag	70	21	5	15	16
FAM52.dag	70	21	5	17	17
FAM53.dag	70	21	5	17	18
FAM54.dag	70	27	6	13	13
FAM55.dag	70	25	6	13	14
FAM56.dag	70	26	6	14	15
FAM57.dag	70	24	6	13	14
FAM58.dag	70	28	7	12	13
FAM59.dag	70	25	6	13	15
FAM61.dag	80	29	7	13	13
FAM62.dag	80	23	5	17	18
FAM63.dag	80	32	8	13	14
FAM64.dag	80	26	6	17	18
FAM65.dag	80	27	6	15	17
FAM66.dag	80	26	6	15	16
FAM67.dag	80	28	7	17	18
FAM68.dag	80	26	6	15	16
FAM69.dag	80	28	7	13	14
FAM71.dag	90	34	8	15	17
FAM72.dag	90	33	8	13	15
FAM73.dag	90	34	8	13	14
FAM74.dag	90	29	7	15	17
FAM75.dag	90	29	7	15	15
FAM76.dag	90	37	9	13	15
FAM77.dag	90	32	8	13	14
FAM78.dag	90	29	7	15	17
FAM79.dag	90	32	8	13	15
FAM110.dag	30	9	2	16	16
FAM210.dag	30	13	3	11	12
FAM310.dag	50	19	4	15	16
FAM410.dag	50	14	3	18	18
FAM510.dag	70	25	6	13	13
FAM610.dag	80	25	6	15	16
FAM710.dag	90	29	7	14	16

Tabela A.7: Instâncias para testes com com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	$\lvert N\rvert$	$\omega(\prec)$	m	LB	UB
r1.dag	135	41	10	14	15
r2.dag	98	33	8	13	14
ran1.dag	140	42	10	14	17
ran2.dag	80	29	7	13	15
ran3.dag	153	48	12	13	16
ran4.dag	108	30	7	16	18
ran5.dag	170	94	23	9	9
ran6.dag	223	110	27	9	9
ran7.dag	298	119	29	11	12
ran8.dag	256	118	29	11	12
ran9.dag	124	65	16	8	8
ran10.dag	286	83	20	17	19
ran11.dag	510	153	38	17	19
ran12.dag	310	99	24	17	19
ran13.dag	357	167	41	10	10
ran14.dag	364	121	30	15	16
ran15.dag	152	47	11	15	16
ran16.dag	186	76	19	10	11
ran17.dag	546	158	39	21	22
ran18.dag	234	74	18	14	16
ran19.dag	154	64	16	10	11
ran20.dag	154	68	17	10	10

Tabela A.8: Instâncias para testes com com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	$\lvert N\rvert$	$\omega(\prec)$	m	LB	UB
bin4.dag	15	8	2	9	9
bin5.dag	31	16	4	10	10
bin6.dag	63	32	8	11	12
bin7.dag	127	64	16	13	13
bin8.dag	255	128	32	15	15
bin9.dag	511	256	64	17	17
di64.dag	64	8	2	34	34
di100.dag	100	10	2	52	52
di144.dag	144	12	3	51	52
di225.dag	225	15	3	78	79
di256.dag	256	16	4	69	70
di400.dag	400	20	5	86	88
irr41.dag	41	11	2	23	23
n13.dag	13	8	2	7	7
tree3.dag	15	8	2	9	9
tree4.dag	31	16	4	10	11
tree5.dag	63	32	8	10	13
tree6.dag	127	64	16	12	15
tree7.dag	255	128	32	14	17
tree8.dag	511	256	64	16	19
celbow.dag	103	31	7	28	30
cstanford.dag	90	30	7	28	29
fft2-b-m.dag	194	32	8	28	29
iterative2-b-m.dag	262	26	6	47	48

Tabela A.9: Instâncias para testes com com $m = \lfloor \omega(\prec)/4 \rfloor$ - Grupo de Instâncias Estruturadas

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
FAM31.dag	50	19	2	26	26
FAM32.dag	50	16	2	27	27
FAM33.dag	50	16	2	25	25
FAM34.dag	50	19	2	25	25
FAM35.dag	50	16	2	25	25
FAM37.dag	50	21	2	25	25
FAM38.dag	50	17	2	25	25
FAM39.dag	50	19	2	25	25
FAM42.dag	50	17	2	25	25
FAM43.dag	50	16	2	25	25
FAM48.dag	50	16	2	26	26
FAM51.dag	70	21	2	35	35
FAM52.dag	70	21	2	36	36
FAM53.dag	70	21	2	36	36
FAM54.dag	70	27	3	24	24
FAM55.dag	70	25	3	24	24
FAM56.dag	70	26	3	24	25
FAM57.dag	70	24	3	24	24
FAM58.dag	70	28	3	25	25
FAM59.dag	70	25	3	24	25
FAM61.dag	80	29	3	27	27
FAM62.dag	80	23	2	40	40
FAM63.dag	80	32	4	21	22
FAM64.dag	80	26	3	30	30
FAM65.dag	80	27	3	28	28
FAM66.dag	80	26	3	27	28
FAM67.dag	80	28	3	28	29
FAM68.dag	80	26	3	27	28
FAM69.dag	80	28	3	27	27
FAM71.dag	90	34	4	24	26
FAM72.dag	90	33	4	23	23
FAM73.dag	90	34	4	24	25
FAM74.dag	90	29	3	32	33
FAM75.dag	90	29	3	30	31
FAM76.dag	90	37	4	24	25
FAM77.dag	90	32	4	23	24
FAM78.dag	90	29	3	32	32
FAM79.dag	90	32	4	23	24
FAM310.dag	50	19	2	26	27
FAM510.dag	70	25	3	24	24
FAM610.dag	80	25	3	27	28
FAM710.dag	90	29	3	30	31

Tabela A.10: Instâncias para testes com com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 1

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
r1.dag	135	41	5	27	27
r2.dag	98	33	4	25	25
ran1.dag	140	42	5	28	28
ran2.dag	80	29	3	27	27
ran3.dag	153	48	6	26	26
ran4.dag	108	30	3	36	36
ran5.dag	170	94	11	16	16
ran6.dag	223	110	13	18	18
ran7.dag	298	119	14	22	22
ran8.dag	256	118	14	19	19
ran9.dag	124	65	8	16	16
ran10.dag	286	83	10	29	29
ran11.dag	510	153	19	27	29
ran12.dag	310	99	12	26	28
ran13.dag	357	167	20	18	18
ran14.dag	364	121	15	25	25
ran15.dag	152	47	5	31	31
ran16.dag	186	76	9	21	21
ran17.dag	546	158	19	29	30
ran18.dag	234	74	9	26	26
ran19.dag	154	64	8	20	20
ran20.dag	154	68	8	20	20

Tabela A.11: Instâncias para testes com com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Aleatórias 2

DAG	$\|N\|$	$\omega(\prec)$	m	LB	UB
bin5.dag	31	16	2	17	17
bin6.dag	63	32	4	18	18
bin7.dag	127	64	8	18	20
bin8.dag	255	128	16	18	21
bin9.dag	511	256	32	18	23
di256.dag	256	16	2	130	130
di400.dag	400	20	2	202	202
tree4.dag	31	16	2	17	18
tree5.dag	63	32	4	18	19
tree6.dag	127	64	8	18	21
tree7.dag	255	128	16	18	22
tree8.dag	511	256	32	18	24
celbow.dag	103	31	3	35	35
cstanford.dag	90	30	3	32	33
fft2-b-m.dag	194	32	4	52	52
iterative2-b-m.dag	262	26	3	90	91

Tabela A.12: Instâncias para testes com com $m = \lfloor \omega(\prec)/8 \rfloor$ - Grupo de Instâncias Estruturadas

A.2 Ambiente Computacional

Utilizamos em todos os testes a linguagem C++ para implementar os algoritmos. A implementação dos algoritmos exatos deste trabalho faz uso do pacote de otimização ILOG CPLEX©Versão 12.4. Os experimentos realizados foram executados utilizando uma máquina com processador de dois núcleos de 2.4GHz, 2.0GB de memória RAM e sistema operacional Linux de 32 bits.

Referências Bibliográficas

[APMS+99] G. Ausiello, M. Protasi, A. Marchetti-Spaccamela, G. Gambosi, P. Crescenzi, and V. Kann. *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1st edition, 1999.

[BS13] Ahmad Al Badawi and Ali Shatnawi. Static scheduling of directed acyclic data flow graphs onto multiprocessors using particle swarm optimization. *Computers and Operations Research*, pages 2322–2328, 2013.

[CCMP01] Manoel Campêlo, Ricardo Corrêa, Nelson Maculan, and Fábio Protti. Ilp formulations for scheduling ordered tasks on a bounded number of processors. *Electronic Notes in Discrete Mathematics*, 7:166 – 169, 2001. Brazilian Symposium on Graphs, Algorithms and Combinatorics.

[CCMP02] Manoel Campêlo, Ricardo Corrêa, Nelson Maculan, and Fábio Protti. Improved lower bounds for scheduling ordered tasks on a bounded number of processors. *XI Congresso Latino-Americano de Investigación Operativa - CLAIO, 2002, Concepcion. Acta de Trabajos*, page 10, 2002.

[Chr89] Philippe Chretienne. A polynomial algorithm to optimally schedule tasks on a virtual distributed system under tree-like precedence constraints. *European Journal of Operational Research*, 43(2):225 – 230, 1989.

[CPW98] B. Chen, C.N. Potts, and G.J. Woeginger. A review of machine scheduling: complexity, algorithms and approximability. *Technical Report Woe-29*, 1998.

[DW85] Danny Dolev and Manfred K. Warmuth. Profile scheduling of opposing forests and level orders. *SIAM. J. on Algebraic and Discrete Methods*, pages 665–687, 1985.

[FB73] Eduardo B. Fernandez and Bertram Bussell. Bounds on the number of processors and time for multiprocessor optimal schedules. *IEEE Trans. Comput.*, 22(8):745–751, August 1973.

[FHa] Noriyuki Fujimoto and Kenichi Hagihara. Near-optimal task scheduling of a complete k-ary tree with communication delays.

[FHb] Noriyuki Fujimoto and Kenichi Hagihara. Optimal Task Scheduling of a Complete K-Ary Tree with Communication Delays. In Roman Wyrzykowski, Jack Dongarra, Marcin Paprzycki, and Jerzy Waśniewski, editors, *Parallel Processing* *and Applied Mathematics*, volume 2328 of *Lecture Notes in Computer Science*, page 71–78. Springer Berlin Heidelberg.

[FKN69] M. Fujii, T. Kasami, and K. Ninomiya. Optimal sequencing of two equivalent processors. *Journal on Applied Mathematics*, 17(4):784–789, 1969.

[FLMB96] Lucian Finta, Zhen Liu, Ioannis Mills, and Evripidis Bampis. Scheduling UET-UCT series-parallel graphs on two processors. *Theoretical Computer Science*, 162(2):323 – 340, 1996.

[Fuj11] S. Fujita. A branch-and-bound algorithm for solving the multiprocessor scheduling problem with improved lower bounding techniques. *IEEE Transactions on Computers*, 60(7):1006–1016, July 2011.

[GLLK79] R.L. Graham, E.L. Lawler, J.K. Lenstra, and A.H.G.Rinnooy Kan. Optimization and Approximation in Deterministic Sequencing and Scheduling: a Survey. In E.L. Johnson P.L. Hammer and B.H. Korte, editors, *Discrete Optimization II Proceedings of the Advanced Research Institute on Discrete Optimization and Systems Applications of the Systems Science Panel of NATO and of the Discrete Optimization Symposium co-sponsored by IBM Canada and SIAM Banff, Aha. and Vancouver*, volume 5 of *Annals of Discrete Mathematics*, page 287–326. Elsevier, 1979.

[HLV94] J. A. Hoogeveen, J. K. Lenstra, and B. Veltman. Three, four, five, six, or the complexity of scheduling with communication delays. *Oper. Res. Lett.*, 16(3):129–137, oct 1994.

[Hu61] T C Hu. Parallel sequencing and assembly line problems. *Operations Research*, pages 841–848, 1961.

[JKS89] H. Jung, L. Kirousis, and P. Spirakis. Lower bounds and efficient algorithms for multiprocessor scheduling of dags with communication delays. In *Proceedings of the first annual ACM symposium on Parallel algorithms and architectures*, SPAA '89, pages 254–264, New York, NY, USA, 1989.

[JR92] Andreas Jakoby and Rüdiger Reischuk. The complexity of scheduling problems with communication delays for trees. In Otto Nurmi and Esko Ukkonen, editors, *Algorithm Theory — SWAT '92*, volume 621 of *Lecture Notes in Computer Science*, pages 165–177. Springer Berlin Heidelberg, 1992.

[JST08] Shiyuan Jin, Guy Schiavone, and Damla Turgut. A performance study of multiprocessor task scheduling algorithms. *The Journal of Supercomputing*, 43(1):77–97, 2008.

[KB87] Lewis T Kruatrachue B. *Duplication scheduling heuristic DSH: A new precedence task scheduler for parallel processing systems.* PhD thesis, Oregon State University, 1987.

[KJY01] Eberhart RC Kennedy J and Shi Y. *Swarm Intelligence.* Morgan Kaufmann Publishers, San Francisco, USA, 1st edition, 2001.

[KN84] H. Kasahara and S. Narita. Practical multiprocessor scheduling algorithms for efficient parallel processing. *IEEE Transactions on Computers*, C-33(11):1023–1029, 1984.

[LVV96] Jan Karel Lenstra, Marinus Veldhorst, and Bart Veltman. The complexity of scheduling trees with communication delays. *Journal of Algorithms*, 20(1):157 – 173, 1996.

[MH97] A. Munier and C. Hanen. Using duplication for scheduling unitary tasks on m processors with unit communication delays. *Theoretical Computer Science*, 178(1–2):119 – 127, 1997.

[MH01] A. Munier and C. Hanen. An approximation algorithm for scheduling dependent tasks on m processors with small communication delays. *Discrete Applied Mathematics*, 108(3):239 – 257, 2001.

[MK97] A. Munier and J-C König. A heuristic for a scheduling problem with communication delays. *Operations Research*, pages 145–147, 1997.

[MRGY83] R. E. Tarjan M. R. Garey, D. S. Johnson and M. Yannakakis. Scheduling opposing forests. *Journal on Algebraic and Discrete Methods*, pages 72–93, 1983.

[PC95] C. Picouleau P. Chrétienne. Scheduling with communication delays: A survey. In *Scheduling Theory and Its Applications*, page 65–89. Wiley, 1995.

[Pic92] Christophe Picouleau. *Etude de problemes d'optimisation dans les systemes distribues.* PhD thesis, Universite Paris, 1992.

[PY88] Christos Papadimitriou and Mihalis Yannakakis. Towards an architecture-independent analysis of parallel algorithms. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, STOC '88, pages 510–513, New York, NY, USA, 1988. ACM.

[RKA99] C. Schwindt R. Kolisch and A.Sprecher. *Benchmark instances for project scheduling problems.* Kluwer; Weglarz, J. (Hrsg.): Handbook on recent advances in project scheduling, 1999.

[RS87] V.J. Rayward-Smith. UET scheduling with unit interprocessor communication delays. *Discrete Applied Mathematics*, 18(1):55 – 71, 1987.

[Sta04] Hans Stadtherr. Scheduling interval orders with communication delays in parallel. *Journal of Parallel and Distributed Computing*, 64(1):1 – 15, 2004.

[Ull75] J. D. Ullman. NP-complete scheduling problems. *J. Comput. Syst. Sci.*, 10(3):384–393, June 1975.

[VRKL96] Theodora A. Varvarigou, Vwani P. Roychowdhury, Thomas Kailath, and Eugene Lawler. Scheduling in and out forests in the presence of communication delays. *IEEE Trans. Parallel Distrib. Syst.*, 7(10):1065–1074, oct 1996.