

ABORDAGENS EXATAS PARA O PROBLEMA DE SELEÇÃO E SEQUENCIAMENTO DE ORDENS DE PRODUÇÃO COM TEMPOS DE SETUP

Yuri Laio Teixeira Veras Silva, Arthur Kramer, Anand Subramanian

Departamento de Engenharia de Produção - Universidade Federal da Paraíba
Centro de Tecnologia, Campus I - Bloco G, Cidade Universitária, 58051-970, João Pessoa, PB
{yurilaio, arthurhfrk}@gmail.com, anand@ct.ufpb.br

Artur Alves Pessoa

Departamento de Engenharia de Produção - Universidade Federal Fluminense
Rua Passo da Pátria, 156, Bloco E, 4º andar, sala 440, São Domingos, Niterói, RJ
artur@producao.uff.br

RESUMO

O problema de seleção e sequenciamento de ordens de produção (OAS) consiste em, simultaneamente, decidir quais ordens (tarefas) serão aceitas para serem processadas, assim como sua sequência associada. Esse problema geralmente surge quando uma empresa não tem a capacidade necessária para atender a demanda, sendo assim, forçadas a rejeitar algumas ordens. O objetivo é selecionar e sequenciar um subconjunto de tarefas que maximize o lucro total, que é dado pela receita total menos o total de penalização por atraso. Para resolver esse problema \mathcal{NP} -difícil, foi proposta uma nova formulação matemática com arcos indexados no tempo e um algoritmo exato baseado em tal formulação que se baseia em uma relaxação lagrangeana. Foram realizados experimentos computacionais em 1500 instâncias que variam de 10 até 100 tarefas, e os resultados obtidos indicam que os métodos propostos são capazes de encontrar resultados extremamente competitivos quando comparados com os disponíveis na literatura.

PALAVRAS CHAVE. Formulação com Arcos Indexados no Tempo, Relaxação Lagrangeana, Problema de Seleção e Sequenciamento de Ordens de Produção.

PO na Administração e Gestão da Produção, Otimização Combinatória

ABSTRACT

The Order Acceptance and Scheduling (OAS) problem consists of simultaneously deciding which orders (jobs) are going to be accepted for processing as well as their associated schedule. This problem typically arises when a company does not have the capacity to meet the demand, thus being forced to reject some orders. The objective is to select and schedule a subset of jobs that maximizes the total profit, which is given by the total revenue minus the total weighted tardiness. To solve this \mathcal{NP} -hard problem, we propose a new arc-time-indexed mathematical formulation and one exact algorithm over this formulation based on lagrangian relaxation. Extensive computational experiments were carried out in 1500 benchmark instances ranging from 10 to 100 jobs and the results obtained suggest that the proposed methods are capable of finding extremely competitive results when compared to those available in the literature.

KEYWORDS. Arc-time Indexed Formulation, Lagrangian Relaxation, Order Acceptance and Scheduling Problem

OR in Production Management, Combinatorial Optimization

1. Introdução

O problema de sequenciamento da produção, comumente conhecido como *scheduling*, é atualmente um dos mais estudados dentro da literatura de otimização, tal fato ocorre, principalmente, devido ao alto grau de importância que representa o planejamento e o sequenciamento da produção em uma organização. Tais decisões sobre sequenciamento representam um índice significativo nas variações de custos e receitas associadas ao processo produtivo das empresas.

A acirrada concorrência presente nos diversos mercados na atualidade tem elevado as expectativas dos clientes, tanto em termos de qualidade, como de custos e prazos de entrega (Lopez e Roubellat, 2008). Deste modo, as organizações buscam o melhor aproveitamento dos recursos. Com base nisto, é natural pensar que as empresas desejem entregar os pedidos de seus clientes dentro dos prazos estipulados. Esse é o caso dos problemas de sequenciamento da produção que objetivam minimizar a soma dos atrasos ponderados, como o $\sum_{j \in S} (w_j T_j)$, por exemplo. O $\sum_{j \in S} (w_j T_j)$ é um problema bastante abordado na literatura, tanto por métodos exatos (Tanaka e Araki, 2013), quanto por métodos heurísticos (Kirlik e Oguz, 2012; Xu *et al.*, 2013; Subramanian *et al.*, 2014).

Por outro lado, em virtude das restrições de capacidade, as empresas muitas vezes devem analisar os pedidos com o intuito de decidir quais devem ser aceitos e em qual ordem eles devem ser processados. Tal decisão é feita com o objetivo de proporcionar a maior rentabilidade para a empresa. Esta problemática motivou a proposição do *Order Acceptance and Scheduling Problem* (OAS), em suas diversas formas apresentadas na literatura. O OAS, como definido em Oğuz *et al.* (2010), considera, além de outras características, tempo de *setup* dependente da sequência, *release dates* e *deadlines*. Este problema será denominado no restante do trabalho como problema de seleção e sequenciamento de tarefas em um ambiente de máquina única com tempos de *setup* dependentes da sequência, ou em inglês, *Order Acceptance and Scheduling in a Single Machine environment with sequence-dependent Setup times*, referido no restante do trabalho como OAS-SMS.

Antes da proposição do OAS, como apresentado por Oğuz *et al.* (2010), já era possível perceber o interesse no problema de decidir quais ordem aceitar e como sequenciá-las. Slotnick e Morton (2007) e Rom e Slotnick (2009) abordaram o OAS sem a presença de tempos de *setup* e de *deadlines*. Por sua vez, Wang *et al.* (2013) tratam do OAS num ambiente *flow shop*.

Recentemente, uma generalização desse problema foi proposta por Nobibon e Leus (2011), na qual há um subconjunto de tarefas do problema que devem ser, necessariamente, sequenciadas. O problema não considera a existência de tempos de *setup* nem *deadlines*. O trabalho de Nobibon *et al.* (2009) fornece ainda uma breve visão dos métodos até então utilizados para este problema.

Nos últimos anos, diversos métodos, tanto exatos como heurísticos, foram propostos para resolução do OAS, o que se justifica, principalmente, pela grande dificuldade computacional em encontrar soluções ótimas em instâncias acima de 20 tarefas. Uma visão geral sobre outros problemas e detalhes adicionais sobre a temática OAS podem ser encontrados em (Zhong *et al.*, 2014; Slotnick, 2011; Wei e Ma, 2014).

Para o OAS, na configuração tratada neste trabalho, é possível encontrar os trabalhos de Oğuz *et al.* (2010) e de Cesaret *et al.* (2012). Em Oğuz *et al.* (2010), os autores apresentam uma formulação matemática baseada em programação linear inteira mista, capaz de resolver problemas com até 15 ordens e três algoritmos heurísticos para os problemas de dimensão de médio porte, *Iterative Sequence First-Accept Next* (ISFAN), *Dynamic Release First-Sequence Best* (d-RFSB) e *Modified Apparent Tardiness Cost with Setups* (m-ATCS). Cesaret *et al.* (2012) desenvolveram um algoritmo baseado na meta-heurística Busca Tabu para este problema, o qual foi capaz de obter resultados eficientes para instâncias com até 25 tarefas, com um baixo custo computacional.

O presente trabalho busca desenvolver dois métodos exatos para resolução do problema, objetivando obter resultados mais eficientes que os observados na literatura, com um tempo computacional aceitável. O primeiro método exato proposto se trata de uma formulação com arcos indexados no tempo. Por sua vez, o segundo método exato baseia-se em um procedimento *Branch-*

and-Bound que utiliza uma relaxação lagrangeana e uma programação dinâmica para resolver o problema relaxado e obter os limitantes duais.

2. Descrição do Problema

O OAS é uma generalização do problema de sequenciamento de ordens de produção em uma máquina, objetivando a minimização da soma dos atrasos ponderados, também conhecido como $1||\sum(w_j T_j)$, de acordo com a classificação $(\alpha|\beta|\gamma)$ proposta por Graham *et al.* (1979). O $1||\sum(w_j T_j)$ foi provado ser \mathcal{NP} -difícil por Lenstra *et al.* (1977). Desta forma, o OAS também está enquadrado na classe de problemas \mathcal{NP} -difícil. A dificuldade computacional observada no OAS, principalmente em decorrência do tempo de *setup* dependente da sequência, pode ser observada em Oğuz *et al.* (2010).

O problema pode ser definido da seguinte forma: para um ambiente envolvendo uma máquina, é dado um conjunto de ordens de produção O , tal que, para cada ordem $i \in O$, é conhecido seu tempo de processamento p_i , sua *release date* r_i , data de entrega (*due date*) d_i , data limite (*deadline*) \bar{d}_i , tal que $d_i \leq \bar{d}_i$, receita máxima e_i , peso w_i , além dos tempos de *setup* s_{ij} de todas as ordens para todas as outras ordens $j \in O$, de modo que s_{ij} pode ser diferente de s_{ji} . Se uma ordem i for aceita para ser processada, ela deve ser, obrigatoriamente, finalizada antes do seu *deadline*. Caso essa ordem i seja finalizada entre sua *due date* e seu *deadline*, é aplicada, à receita máxima e_i , uma penalidade w_i por unidade de tempo passada da *due date*, de maneira que em \bar{d}_i essa receita é zero. O OAS busca definir quais ordens devem ser processadas e em qual sequência, com o objetivo de maximizar o lucro total. O objetivo é obter a solução que maximize o lucro total.

3. Abordagens Propostas

Para a resolução do problema foram propostas duas abordagens exatas. Inicialmente é apresentada a formulação proposta, com arcos indexados no tempo. Posteriormente, é descrito o segundo método proposto, um procedimento *Branch-and-Bound* (B&B) que utiliza uma relaxação lagrangeana (RL) para obter limitantes duais (superiores, no caso do problema estudado) e soluções viáveis (limitantes inferiores) para o problema. No caso do segundo método, é utilizada uma programação dinâmica para resolver o subproblema da relaxação lagrangeana.

3.1. Formulação com Arcos Indexados no Tempo

A primeira abordagem proposta trata-se de uma formulação com arcos indexados no tempo, com utilização do tempo discretizado, sendo o horizonte de tempo considerado de acordo com o *release date* r_j e o *deadline* \bar{d}_j de cada tarefa. A variável x_{ij}^t indica se a tarefa i precede j e que j inicia seu processamento exatamente no tempo t . Assim, $x_{ij}^t = 1$, para $i \neq j$, quando a tarefa i terminou em algum tempo menor ou igual a $t - s_{ij}$ e a tarefa j começou no tempo t , e, caso $i = j$, quando a tarefa i termina de ser processada em algum tempo menor ou igual a $t - 1$ e o *setup* da próxima tarefa começará a ser contado em algum tempo maior ou igual t .

Dessa forma, as variáveis x_{ii}^t , que indica $i = j$, representam o *idle time* executado após a tarefa i , até que o tempo necessário para executar o *setup* para a tarefa j real ($j \neq i$) seja alcançado. A utilização dessas variáveis são necessárias na formulação descrita pois, por definição do problema, o *setup* entre as tarefas só pode ser realizado após o *release date* da tarefa seguinte ser alcançado. Dessa forma, foi utilizado a tarefa saindo dela e indo para ela mesma, em uma unidade de tempo, como forma de representar o *idle time*, tornando possível a manutenção da informação da tarefa anterior até que o *release date* da tarefa seguinte seja alcançado, permitindo que seja realizada a incrementação do tempo de *setup* corretamente, a partir da informação de qual é a tarefa anterior.

Em função do *idle time* ser representado pela saída da tarefa atual e entrada nela mesma, é necessário utilizar um tempo de processamento diferenciado quando a tarefa seguinte a ser processada é igual a tarefa atual, ou seja, caracterizando um *idle time*. Assim, $p'_{ij} = 1$ quando $i = j$, realizando o acréscimo de uma unidade de tempo referente ao *idle time*, e $p'_{ij} = p_i$, quando $i \neq j$.

O conjunto $J_+ = \{0, 1, \dots, n\}$ representa as tarefas que fazem parte do problema juntamente com a tarefa *dummy* 0, enquanto o conjunto $J = \{1, \dots, n\}$ representa apenas as tarefas

reais do problema de sequenciamento. A necessidade da tarefa 0 ocorre, principalmente, devido a duas razões. A primeira, devido ao *setup* inicial que é necessário na descrição do problema, para a configuração da máquina para processar a primeira tarefa de todas do sequenciamento. A segunda, devido a conservação do fluxo da última tarefa real a ser sequenciada na resolução do problema. Dessa forma, a máquina deve iniciar e finalizar o sequenciamento na tarefa *dummy* representada pelo 0, conforme pode ser observado na formulação apresentada a seguir.

$$z = \max \sum_{i \in J_+} \sum_{\substack{j \in J, \\ j \neq i}} \sum_{\substack{\bar{d}_j - p_j \\ t=r_i+p_i+s_{ij}, \\ t-s_{ij} \geq r_j}} f_j^t x_{ij}^t \quad (1)$$

$$\text{Sujeito a:} \quad \sum_{\substack{j \in J \\ t=r_j+s_{0j}}} x_{0j}^t = 1 \quad (2)$$

$$\sum_{\substack{i \in J_+, \\ i \neq j}} \sum_{\substack{\bar{d}_j - p_j \\ t=r_i+p_i+s_{ij}, \\ t-s_{ij} \geq r_j}} x_{ij}^t \leq 1 \quad (j \in J) \quad (3)$$

$$\sum_{\substack{j \in J_+, \\ t-p_j-s_{ji} \geq r_j, \\ t+p_i \leq \bar{d}_i, \\ t-s_{ji} \geq r_i}} x_{ji}^t - \sum_{\substack{j \in J_+, \\ t+p_i \geq r_j, \\ t+p_i+s_{ij}+p_j \leq \bar{d}_j}} x_{ij}^{t+p'_i+s_{ij}} = 0 \quad (i \in J; t = 0, \dots, T) \quad (4)$$

$$x_{ij}^t \in \{0, 1\} \quad \begin{aligned} &(i \in J_+; j \in J_+; \\ &t = r_i + p_i + s_{ij}, \dots, \bar{d}_j - p_j; \\ &t - s_{ij} \geq r_j) \end{aligned} \quad (5)$$

A função objetivo (1) maximiza o lucro total. A restrição (2) indica que exatamente uma única tarefa deve iniciar a sequência. As restrições (3) indicam que cada tarefa pode ser processada no máximo uma única vez. As restrições (4) garantem a conservação do fluxo. Por fim, as restrições (5) definem o domínio das variáveis de decisão.

A Figura 1 mostra a rede associada a um exemplo de instância muito pequeno, envolvendo apenas quatro tarefas, cujo os dados de entrada são apresentados na Tabela 1. É importante enfatizar que os dados apresentados na Tabela 1(c) podem ser obtidos através da Tabela 1(a). A Figura 1 também mostra a representação do caminho de uma solução inteira viável de ATIF, na qual é composta pelas seguintes variáveis não-nulas: $x_{04}^2, x_{44}^3, x_{42}^5, x_{21}^8$ e x_{10}^9 . A sequência resultante é (4, 2, 1), onde pode ser visto que as tarefas 1, 2 e 4 são aceitas, enquanto a tarefa 3 foi rejeitada. O lucro da solução neste caso corresponde a $f_4^2 + f_2^5 + f_1^8 = 6 + 4 + 0 = 10$. Note que a máquina teve um tempo de espera de uma unidade de tempo após o processamento da tarefa 4, mais precisamente do tempo 2 para 3. Tal tempo de espera é necessário pois, apesar de a tarefa 4 finalizar o processamento no tempo 3, como $r_2 = 4$, o *setup* para a tarefa 2 não pode ser executado neste tempo.

3.2. Relaxação Lagrangeana

O segundo método desenvolvido para resolução do problema trata-se de um algoritmo B&B que utiliza uma RL para obter limitantes duais e soluções factíveis para o problema.

A RL é uma técnica muito utilizada em problemas de otimização pela sua eficiência na geração de limitantes duais. Essa técnica consiste em relaxar restrições difíceis do problema, adicionando-as na função objetivo de modo a penalizá-la. Tal penalização é efetuada de proporcionalmente as violações causadas nas restrições que foram relaxadas no problema.

Tabela 1: Dados de entrada do exemplo de instância

j	(a) dados por tarefa $j \in J_+$						(b) $s_{ij}, \forall i, j \in J_+$					(c) $f_j^t, \forall j \in J, t = 0, \dots, \max_{j \in J} \{\bar{d}_j\}$											
	r_j	p_j	d_j	\bar{d}_j	e_j	w_j		0	1	2	3	4	0	1	2	3	4	5	6	7	8	9	
0	0	0	-	-	-	-	0	-	1	2	2	1	0	-	-	-	-	-	-	-	-	-	-
1	2	1	6	9	3	1	1	0	0	1	2	3	1	-	-	3	3	3	3	2	1	0	-
2	4	1	6	7	4	4	2	0	2	0	1	1	2	-	-	-	-	4	4	0	-	-	-
3	3	2	6	9	6	2	3	0	1	1	0	1	3	-	-	-	6	6	4	2	0	-	-
4	1	1	5	7	6	3	4	0	1	1	1	0	4	-	6	6	6	6	3	0	-	-	-

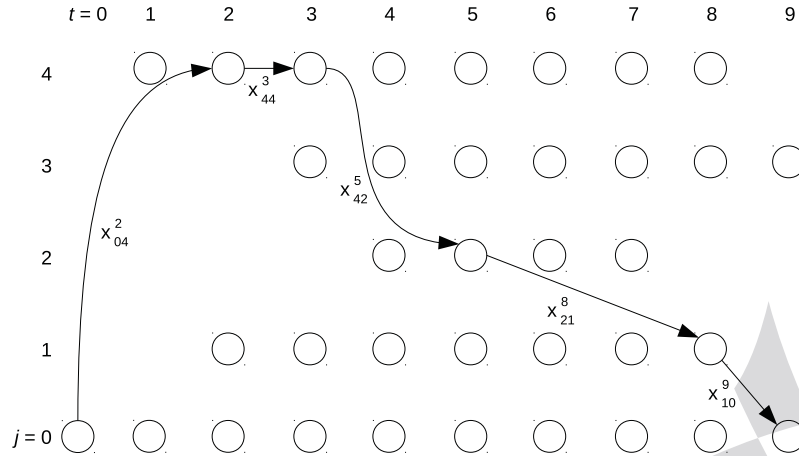


Figura 1: Representação de uma solução associada a instância do exemplo

Dualizando as restrições (3) da formulação com arcos indexados no tempo, nas quais indicam que as tarefas devem ser processadas no máximo uma única vez, com multiplicadores de Lagrange μ_j , foi possível obter o seguinte subproblema Lagrangeano (LS).

$$L(\mu) = \max \sum_{i \in J_+} \sum_{\substack{j \in J, \\ j \neq i}} \sum_{\substack{t=r_i+p_i+s_{ij}, \\ t-s_{ij} \geq r_j}}^{\bar{d}_j-p_j} (f_j^t - \mu_j) x_{ij}^t + \sum_{j \in J} \mu_j \quad (6)$$

Sujeito a: (2), (4) e (5)

Para qualquer vetor de multiplicadores de Lagrange não-negativos μ , $L(\mu)$ fornece um limite superior válido do valor ótimo para (1)-(5). O problema Dual lagrangeano (LD) objetiva minimizar tal limite superior resolvendo $\min_{\mu \in \mathbb{R}^+} \{L(\mu)\}$. Com o intuito de encontrar a solução ótima para LD, é necessário se resolver LS diversas vezes, o que pode resultar em um tempo de execução elevado. Dessa forma, é importante que se tenha um método eficiente para resolução de tal problema para que seja possível obter soluções de boa qualidade em um tempo computacional aceitável.

Seja $f_j^{it} = f_j^t - \mu_j$ os coeficientes de lucro modificados pelos custos lagrangeanos. Foi definido $R^*(i, t)$ como o maior (modificado) lucro da seqüência que satisfaz (2), (4) and (5) em que a última tarefa sequenciada i finaliza no tempo t ou antes. Observe que o problema definido por (6),

(2), (4) and (5) pode ser resolvido pela recursão da programação dinâmica apresentada a seguir.

$$R^*(i, t) = \begin{cases} \max\{R^*(i, t-1), \max_{\substack{j \in J_+ \\ j \neq i, r_i + s_{j_i} + p_i \leq t \leq \bar{d}_i}} \{f_j^{tt} + R^*(j, t - (p_i + s_{j_i}))\}\}, & i \neq 0 \\ 0, & i = 0 \text{ e } t = 0 \\ -\infty, & \text{caso contrário.} \end{cases}$$

Com a relaxação das restrições (3), torna-se permitido o processamento de uma mesma tarefa múltiplas vezes. Entretanto, a cada iteração da RL, os multiplicadores de Lagrange da iteração corrente são atualizados com base nas violações às restrições relaxadas que estejam presentes na melhor solução obtida pela programação dinâmica na iteração anterior da RL. Tal atualização é realizada através do método do subgradiente proposto por Held *et al.* (1974).

3.3. Branch-and-Bound

De forma geral, um algoritmo B&B é um procedimento que realiza uma enumeração de todas as possíveis soluções para um determinado problema de otimização combinatória. O principal objetivo do método é analisar apenas um determinado conjunto de partes do espaço de soluções existente de modo que possibilite determinar a solução ótima para o problema tratado.

O método implementado ocorre da seguinte maneira: Primeiramente, é executada a RL para resolver o problema. Caso a solução ótima seja obtida, o problema é finalizado. Caso contrário, o B&B é inicializado com uma lista de nós abertos, na qual terá como nó raiz a solução primária obtida, que contém a informação sobre qual tarefa e em qual instante de tempo deverá ser realizado o *branching*. Inicialmente, é escolhido um nó da lista de acordo com a estratégia de *branching* adotada (caso seja a primeira iteração, haverá apenas o nó raiz na lista). Após isso, é realizado *branching* no nó escolhido, utilizando a regra de *branching* (informação sobre qual tarefa e em qual tempo deverá ser feito).

Os nós gerados a partir do nó escolhido são passados, juntamente com os multiplicadores de Lagrange μ_j do nó originário, para serem resolvidos pela RL, de modo que assegure que a solução gerada pela RL apresentará um limitante superior melhor ou igual ao do nó originário. Caso a melhor solução obtida da RL s' seja uma solução viável, e seu lucro real maior que o lucro da melhor solução viável corrente \bar{s} , ela é atualizada. Caso a solução s' obtida seja viável e seu lucro real inferior ao da melhor solução corrente \bar{s} , o nó é descartado. Caso a solução s' seja inviável e seu lucro relaxado (associado ao dual lagrangeano) seja superior ao da melhor solução viável corrente \bar{s} , o nó é adicionado na lista. Por fim, caso a solução seja inviável s' e seu lucro relaxado seja menor que o lucro da melhor solução viável corrente \bar{s} , o nó é descartado. O procedimento é repetido iterativamente até que a lista de nós abertos seja esvaziada ou o tempo de execução atinja o tempo limite determinado. É importante citar que o limitante inferior é inicializado com um valor de acordo com os resultados obtidos pelo método de Busca Tabu proposto por Cesaret *et al.* (2012) e publicados em http://home.ku.edu.tr/~coguz/OAS_TS_Solutions_UBs.txt.

A regra de *branching* é o procedimento responsável por indicar qual tarefa e em qual tempo deverá ser realizado o *branching*. Dado que s' representa a solução final da RL na resolução de um determinado nó, a tarefa j a ser escolhida para realização futura do *branching* é a que mais se repete na sequência s' . Caso haja empate, a tarefa de maior índice será a escolhida. Com relação ao tempo, o *branching* deverá ser realizado no instante de término t da primeira aparição da tarefa escolhida. Dessa forma, em um dos subproblemas será proibido que a tarefa j finalize seu processamento em $t' \leq t$, e no outro subproblema será proibido que a tarefa j finalize seu processamento em $t' > t$.

4. Resultados das Abordagens Exatas

Para avaliar o desempenho dos métodos propostos, foram utilizadas as instâncias propostas por Cesaret *et al.* (2012). Os resultados obtidos tanto pela relaxação lagrangeana (RL),

como pela formulação *arc-time-indexed* (ATIF) proposta, foram comparados com os obtidos pela formulação *Mixed Integer Linear Programming* (MILP) publicada por Cesaret *et al.* (2012).

Devido ao grande número de instâncias, os resultados foram reportados em grupos de instâncias para uma melhor comparação do desempenho dos métodos, caracterizados por seus respectivos tamanho, e parâmetros τ e R , conforme foram apresentadas no método heurístico. Cada grupo de instância é composto por 10 instâncias com as mesmas características. Posteriormente, os resultados são apresentados no apêndice A de maneira individual para uma melhor verificação do desempenho dos métodos nos testes realizados. Não foi possível realizar uma comparação com a formulação MILP nos resultados individuais pois não foram reportados dessa forma em Cesaret *et al.* (2012), apenas em relação aos grupos de instâncias.

A comparação do desempenho dos métodos nos grupos de instâncias foi realizada através do *Gap* entre a melhor solução factível obtida e o limitante superior (UB) obtido com a execução de cada método, considerando um tempo limite de 3600 segundos. São apresentados, além do *Gap* médio de cada grupo de instância, o menor e o maior *Gap* obtido em cada grupo. Foram reportadas ainda a quantidade de soluções ótimas obtidas em cada grupo de instâncias por cada um dos métodos, assim como o tempo médio obtido com a execução de todas as instâncias de cada grupo. As Tabelas 2-7 apresentam a comparação dos resultados obtidos.

 Tabela 2: Resultados ATIF-RL para $n = 10$

$n = 10$		<i>Gap</i> (%)									# soluções ótimas			$t(s)$		
τ	R	MILP			ATIF			RL			MILP	ATIF	RL	MILP	ATIF	RL
		<i>Min</i>	<i>Avg</i>	<i>Max</i>	<i>Min</i>	<i>Avg</i>	<i>Max</i>	<i>Min</i>	<i>Avg</i>	<i>Max</i>						
0,1	0,1	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1123,9	0,8	1,1
	0,3	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1119,9	1,5	0,9
	0,5	0	1	6	0,00	0,00	0,00	0,00	0,00	0,00	9	10	10	1037,2	4,1	2,0
	0,7	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	435,0	2,2	0,3
	0,9	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	289,6	1,0	1,1
0,3	0,1	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	466,2	0,4	0,3
	0,3	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	372,4	0,5	0,3
	0,5	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	257,3	0,6	0,2
	0,7	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	310,3	1,4	0,4
	0,9	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	118,2	0,3	0,1
0,5	0,1	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	22,1	0,1	0,1
	0,3	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	52,6	0,3	0,1
	0,5	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	13,9	0,2	0,0
	0,7	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	24,3	0,1	0,0
	0,9	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	26,7	0,3	0,1
0,7	0,1	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,3	0,0	0,0
	0,3	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,5	0,0	0,0
	0,5	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,6	0,1	0,0
	0,7	0	0	1	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,7	0,0	0,0
	0,9	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,9	0,1	0,0
0,9	0,1	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,0	0,0	0,0
	0,3	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,0	0,0	0,0
	0,5	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,0	0,0	0,0
	0,7	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,0	0,0	0,0
	0,9	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,0	0,0	0,0
<i>Avg.</i>		0,00	0,04	0,28	0,00	0,00	0,00	0,00	0,00	0,00	-	-	-	227,3	0,6	0,3

Através dos resultados apresentados é possível verificar que o desempenho, tanto da RL, como da ATIF foi significativamente superior ao apresentado por MILP em todos os grupos de instâncias, considerando-se os *Gaps*, a quantidade de soluções ótimas obtidas e o tempo médio de execução. Tal superioridade dos métodos RL e ATIF torna-se mais acentuada ainda nos grupos de instâncias de tamanhos maiores. Com relação aos resultados de RL e ATIF, é possível observar que ambos apresentam um desempenho semelhante em todos os indicadores. Os *Gaps* obtidos com a RL mostram uma ligeira superioridade em relação aos alcançados pela ATIF, enquanto a ATIF apresentou um desempenho levemente superior a RL na quantidade de soluções ótimas obtidas.

Para uma melhor comparação acerca dos índices de desempenho médio obtidos pelos

Tabela 3: Resultados ATIF-RL para $n = 15$

$n=15$		Gap (%)									# soluções ótimas			$t(s)$		
τ	R	MILP			ATIF			RL			MILP	ATIF	RL	MILP	ATIF	RL
		Min	Avg	Max	Min	Avg	Max	Min	Avg	Max						
0,1	0,1	1	8	13	0,00	0,00	0,00	0,00	0,00	0,02	0	10	10	3600,0	308,2	125,2
	0,3	1	7	17	0,00	0,13	1,31	0,00	0,01	0,14	0	9	9	3600,0	874,7	774,8
	0,5	1	6	10	0,00	0,12	0,62	0,00	0,03	0,26	0	8	9	3600,0	761,8	467,2
	0,7	1	7	21	0,00	0,05	0,27	0,00	0,00	0,00	0	8	10	3600,0	759,7	732,8
	0,9	0	6	15	0,00	0,00	0,00	0,00	0,00	0,00	1	10	10	3241,7	6,2	287,9
0,3	0,1	4	10	13	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	6,5	20,7
	0,3	4	11	22	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	25,9	25,3
	0,5	4	11	20	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	92,9	45,9
	0,7	3	11	16	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	24,2	13,2
	0,9	0	9	18	0,00	0,00	0,00	0,00	0,00	0,00	1	10	10	3444,5	16,8	26,4
0,5	0,1	2	12	15	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	2,5	6,0
	0,3	7	11	26	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	2,8	2,4
	0,5	7	15	12	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	1,1	1,2
	0,7	0	7	18	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	2,9	3,7
	0,9	0	7	0	0,00	0,00	0,00	0,00	0,00	0,00	1	10	10	3369,8	1,3	1,3
0,7	0,1	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	83,9	0,3	0,4
	0,3	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	88,6	0,3	0,1
	0,5	0	0	8	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	101,3	0,3	0,3
	0,7	0	1	0	0,00	0,00	0,00	0,00	0,00	0,00	9	10	10	883,9	0,3	0,1
	0,9	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	288,5	0,3	0,4
0,9	0,1	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,0	0,1	0,0
	0,3	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,0	0,1	0,0
	0,5	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	6,1	0,1	0,0
	0,7	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	4,4	0,1	0,0
	0,9	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	12,4	0,2	0,1
Avg.		1,40	5,56	9,76	0,00	0,01	0,09	0,00	0,00	0,02	-	-	-	2189,1	115,6	101,4

 Tabela 4: Resultados ATIF-RL para $n = 20$

$n=20$		Gap (%)									# soluções ótimas			$t(s)$		
τ	R	MILP			ATIF			RL			MILP	ATIF	RL	MILP	ATIF	RL
		Min	Avg	Max	Min	Avg	Max	Min	Avg	Max						
0,1	0,1	6	11	19	0,00	0,19	1,07	0,00	0,10	0,64	0	7	6	3600,0	1286,5	1805,3
	0,3	2	12	19	0,00	0,29	0,94	0,00	0,09	0,32	0	5	6	3600,0	2120,2	1880,9
	0,5	3	10	22	0,00	0,37	1,10	0,00	0,18	0,50	0	5	5	3600,0	1959,3	1943,5
	0,7	0	11	23	0,00	0,12	0,52	0,00	0,08	0,20	1	7	3	3415,5	1128,7	2829,7
	0,9	1	7	15	0,00	0,00	0,00	0,00	0,00	0,01	0	10	9	3600,0	65,6	724,3
0,3	0,1	7	18	29	0,00	0,06	0,62	0,00	0,03	0,26	0	9	9	3600,0	567,5	531,5
	0,3	4	14	21	0,00	0,13	0,75	0,00	0,00	0,00	0	7	10	3600,0	1379,4	1029,0
	0,5	7	16	25	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	184,9	134,8
	0,7	5	12	25	0,00	0,18	0,64	0,00	0,04	0,21	0	6	8	3600,0	1918,6	1210,1
	0,9	2	10	20	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3444,5	48,9	389,6
0,5	0,1	6	13	30	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	9,6	49,8
	0,3	7	17	26	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	52,8	51,7
	0,5	9	16	23	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	21,0	30,4
	0,7	8	14	27	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	50,4	177,7
	0,9	5	12	21	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	6,0	7,5
0,7	0,1	6	12	26	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	8,5	14,9
	0,3	6	10	14	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	2,1	3,8
	0,5	0	10	27	0,00	0,00	0,00	0,00	0,00	0,00	2	10	10	3048,2	1,1	1,1
	0,7	0	8	20	0,00	0,00	0,00	0,00	0,00	0,00	3	10	10	2954,4	1,5	2,5
	0,9	0	10	17	0,00	0,00	0,00	0,00	0,00	0,00	1	10	10	3242,8	0,7	0,4
0,9	0,1	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	1,7	0,1	0,0
	0,3	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	8,7	0,2	0,1
	0,5	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	80,1	0,3	0,2
	0,7	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	114,0	0,3	0,6
	0,9	0	1	13	0,00	0,00	0,00	0,00	0,00	0,00	9	10	10	467,0	0,5	0,1
Avg.		3,36	9,76	18,48	0,00	0,05	0,23	0,00	0,02	0,09	-	-	-	2831,1	432,6	512,8

Tabela 5: Resultados ATIF-RL para $n = 25$

$n=25$		Gap (%)									# soluções ótimas			$t(s)$		
τ	R	MILP			ATIF			RL			MILP	ATIF	RL	MILP	ATIF	RL
		Min	Avg	Max	Min	Avg	Max	Min	Avg	Max						
0,1	0,1	8	12	19	0,00	0,58	1,52	0,00	0,39	1,18	0	3	3	3600,0	2738,7	2596,3
	0,3	6	13	23	0,00	0,37	0,88	0,00	0,26	0,66	0	4	4	3600,0	2775,6	2292,1
	0,5	5	7	9	0,00	0,23	0,96	0,00	0,20	0,64	0	6	2	3600,0	2219,5	2889,6
	0,7	1	5	11	0,00	0,00	0,00	0,00	0,12	0,48	0	10	4	3600,0	358,8	2172,0
	0,9	1	4	10	0,00	0,00	0,00	0,00	0,06	0,24	0	10	6	3600,0	612,5	1553,9
0,3	0,1	9	15	24	0,00	0,23	0,90	0,00	0,13	0,44	0	6	6	3600,0	1674,2	1964,0
	0,3	7	11	20	0,00	0,42	0,82	0,00	0,12	0,48	0	2	6	3600,0	2933,4	1686,8
	0,5	2	5	8	0,00	0,24	0,56	0,00	0,09	0,31	0	3	3	3600,0	2746,5	2680,0
	0,7	1	5	9	0,00	0,14	0,62	0,00	0,05	0,39	0	6	8	3600,0	1909,9	1488,1
	0,9	11	18	27	0,00	0,14	0,79	0,00	0,05	0,24	0	7	8	3600,0	1381,9	1383,6
0,5	0,1	6	10	17	0,00	0,00	0,00	0,00	0,02	0,14	0	10	9	3600,0	472,7	556,3
	0,3	4	9	13	0,00	0,12	0,63	0,00	0,02	0,16	0	8	9	3600,0	1377,7	709,4
	0,5	3	10	17	0,00	0,13	0,94	0,00	0,04	0,33	0	8	8	3600,0	852,7	810,4
	0,7	2	8	13	0,00	0,06	0,54	0,00	0,02	0,19	0	8	9	3600,0	879,4	817,1
	0,9	6	20	42	0,00	0,04	0,37	0,00	0,01	0,10	0	9	9	3600,0	650,8	438,8
0,7	0,1	2	10	19	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	7,3	25,9
	0,3	8	12	17	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	14,1	25,2
	0,5	7	13	21	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	7,0	7,0
	0,7	2	9	15	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	8,5	31,0
	0,9	0	10	17	0,00	0,00	0,00	0,00	0,00	0,00	0	10	10	3600,0	9,6	87,5
0,9	0,1	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	5,4	0,3	0,1
	0,3	0	0	0	0,00	0,00	0,00	0,00	0,00	0,00	10	10	10	8,3	0,4	0,2
	0,5	0	3	12	0,00	0,00	0,00	0,00	0,00	0,00	6	10	10	1452,6	0,7	0,4
	0,7	0	7	22	0,00	0,00	0,00	0,00	0,00	0,00	6	10	10	1446,4	1,0	0,9
	0,9	0	6	21	0,00	0,00	0,00	0,00	0,00	0,00	4	10	10	2176,2	1,1	2,0
Avg.		3,64	8,88	16,24	0,00	0,11	0,38	0,00	0,06	0,24	-	-	-	3083,6	945,4	968,7

 Tabela 6: Resultados ATIF-RL para $n = 50$

$n=50$		Gap (%)									# soluções ótimas			$t(s)$		
τ	R	MILP			ATIF			RL			MILP	ATIF	RL	MILP	ATIF	RL
		Min	Avg	Max	Min	Avg	Max	Min	Avg	Max						
0,1	0,1	9	24	81	1,04	2,11	2,86	1,17	2,17	2,98	0	0	0	3600,0	3600,0	3240,5
	0,3	7	16	22	0,91	2,10	3,42	1,00	2,19	3,47	0	0	0	3600,0	3600,0	3600,0
	0,5	11	17	25	1,12	1,60	2,03	1,48	1,88	2,39	0	0	0	3600,0	3600,0	3600,0
	0,7	9	13	18	0,00	0,81	2,04	0,44	1,22	2,61	0	2	0	3600,0	3600,0	3600,0
	0,9	2	10	27	0,00	0,54	1,90	0,00	0,55	1,92	0	2	0	3600,0	3600,0	3600,0
0,3	0,1	13	24	33	0,10	1,59	2,59	0,07	1,62	2,57	0	0	0	3600,0	3600,0	3600,0
	0,3	18	28	48	1,56	2,56	4,12	1,54	2,60	4,12	0	0	0	3600,0	3600,0	3240,5
	0,5	18	24	33	1,01	2,67	11,46	1,00	2,71	11,49	0	0	0	3600,0	3600,0	3600,0
	0,7	13	20	26	0,00	1,44	3,44	0,29	1,69	3,84	0	1	0	3600,0	3600,0	3600,0
	0,9	13	21	34	0,40	1,31	2,53	0,42	1,42	2,52	0	0	0	3600,0	3600,0	3600,0
0,5	0,1	19	28	31	0,12	3,24	13,25	0,05	3,16	13,28	0	0	0	3600,0	3600,0	3600,0
	0,3	25	42	86	0,11	2,59	4,20	0,00	2,46	4,09	0	0	2	3600,0	3600,0	3089,7
	0,5	17	31	44	0,16	1,83	4,55	0,06	1,74	4,29	0	0	0	3600,0	3600,0	3600,0
	0,7	10	21	38	0,17	1,81	2,84	0,09	1,76	2,74	0	0	0	3600,0	3600,0	3600,0
	0,9	8	18	22	0,08	1,25	3,33	0,02	1,16	3,24	0	0	0	3600,0	3600,0	3600,0
0,7	0,1	13	21	25	0,38	1,07	1,85	0,10	0,80	1,53	0	0	0	3600,0	3600,0	3600,0
	0,3	11	18	23	0,27	0,57	0,94	0,00	0,25	0,71	0	0	2	3600,0	3600,0	2948,0
	0,5	13	29	52	0,00	0,67	1,70	0,00	0,42	1,46	0	1	1	3600,0	3414,8	3389,4
	0,7	9	21	28	0,00	0,28	1,95	0,00	0,19	1,35	0	6	5	3600,0	2138,7	2279,4
	0,9	14	19	26	0,00	0,06	0,28	0,00	0,16	0,48	0	7	3	3600,0	2055,5	2640,4
0,9	0,1	8	14	18	0,00	0,00	0,00	0,00	0,01	0,13	0	10	9	3600,0	33,3	673,4
	0,3	11	16	22	0,00	0,00	0,00	0,00	0,02	0,18	0	10	9	3600,0	14,0	527,0
	0,5	5	15	21	0,00	0,00	0,00	0,00	0,00	0,01	0	10	10	3600,0	47,0	529,0
	0,7	9	17	25	0,00	0,00	0,00	0,00	0,04	0,33	0	10	8	3600,0	46,8	1178,7
	0,9	14	21	30	0,00	0,00	0,00	0,00	0,01	0,08	0	10	9	3600,0	45,9	771,4
Avg.		11,96	21,12	33,52	0,30	1,20	2,85	0,31	1,21	2,87	-	-	-	3600,0	2759,8	2852,3

Tabela 7: Resultados RL para $n = 100$

$n = 100$		Gap (%)						# soluções ótimas		$t(s)$	
τ	R	MILP			RL			MILP	RL	MILP	RL
		Min	Avg	Max	Min	Avg	Max				
0,1	0,1	19,00	44,00	56,00	2,43	3,28	3,82	0	0	3600	3600
	0,3	20,00	51,00	88,00	2,14	2,80	4,24	0	0	3600	3600
	0,5	19,00	51,00	60,00	1,71	2,53	4,36	0	0	3600	3600
	0,7	16,00	54,00	60,00	1,23	1,76	2,17	0	0	3600	3600
	0,9	11,00	60,00	70,00	0,11	0,83	1,45	0	0	3600	3600
0,3	0,1	24,00	62,00	68,00	1,76	2,59	3,73	0	0	3600	3600
	0,3	23,00	64,00	75,00	1,38	2,61	3,94	0	0	3600	3600
	0,5	21,00	66,00	75,00	1,14	2,10	3,24	0	0	3600	3600
	0,7	32,00	73,00	84,00	1,40	2,29	2,96	0	0	3600	3600
0,5	0,1	18,00	66,00	82,00	0,20	1,27	2,97	0	0	3600	3600
	0,3	28,00	61,00	86,00	1,56	2,81	4,39	0	0	3600	3600
	0,5	31,00	70,00	88,00	1,43	2,32	3,34	0	0	3600	3600
	0,7	34,00	71,00	92,00	1,61	2,43	3,24	0	0	3600	3600
0,7	0,1	36,00	66,00	100,00	1,11	2,15	4,08	0	0	3600	3600
	0,3	28,00	67,00	86,00	1,77	3,60	5,04	0	0	3600	3600
	0,5	34,00	57,00	66,00	2,81	4,26	8,62	0	0	3600	3600
	0,7	37,00	55,00	65,00	2,56	3,32	4,53	0	0	3600	3600
0,9	0,1	38,00	60,00	76,00	2,41	3,99	8,24	0	0	3600	3600
	0,3	37,00	53,00	64,00	2,86	5,01	8,65	0	0	3600	3600
	0,5	25,00	37,00	48,00	0,04	4,11	5,78	0	0	3600	3600
	0,7	36,00	40,00	50,00	1,69	5,12	10,73	0	0	3600	3600
0,9	0,5	39,00	38,00	54,00	1,82	5,78	8,62	0	0	3600	3600
	0,7	40,00	38,00	48,00	2,87	5,60	9,79	0	0	3600	3600
	0,9	40,00	35,00	39,00	1,51	5,38	8,95	0	0	3600	3600
Avg.		28,40	56,64	70,68	1,66	3,24	5,24	-	-	3600	3600

métodos, foi realizado uma análise de seus resultados de acordo com grupos formados por tamanhos das instâncias. Os resultados obtidos com a análise reafirmam as melhorias já destacadas dos métodos ATIF e RL em relação ao método MILP. Tomando como base as instâncias de tamanho até 50, o *Gap* médio geral do método MILP foi de 9,07%, enquanto a ATIF e a RL obtiveram índices muito próximos, 0,28% e 0,26%, respectivamente, uma redução de aproximadamente 96,9% em comparação ao obtido pelo MILP. Considerando todas as instâncias, incluindo-se as de tamanho 100, os resultados obtidos pelo método MILP apresentaram um *Gap* médio geral de aproximadamente 17%, contra 0,76% da RL, ocasionando em uma redução de 95,5%. Devido a dificuldade computacional em solucionar instâncias de tamanhos grandes, o método ATIF foi limitado para testes nas instâncias de no máximo 50 tarefas, não tendo portanto resultados reportados para instâncias de 100 tarefas, razão pela qual não o método não foi incluído na segunda comparação.

Com relação aos tempos médios de execução, foi possível constatar que os métodos ATIF e RL apresentaram resultados bastante superiores aos obtidos pelo MILP, tendo a RL apresentado resultados melhores nas instâncias de tamanho 10 e 15, e a ATIF nas de tamanho 20, 25 e 50. Nas instâncias de 100 tarefas, os tempos obtidos por MILP e RL foram os mesmos, pois não foram encontradas nenhuma solução ótima em nenhum dos métodos nessa dimensão. O tempo médio geral alcançado na ATIF e na RL também apresentaram resultados superiores em comparação com o MILP. Considerando as instâncias de tamanho até 50, o tempo médio geral do MILP foi 2386,2 segundos, contra 850,8 segundos da ATIF e 887,1 segundos da RL, uma redução de 64,3% e 62,8%, respectivamente, quando comparados ao MILP.

Além disso, a ATIF e a RL foram capazes de determinar uma quantidade de soluções ótimas superior as obtidas pelo MILP. Considerando as instâncias de até 50 tarefas, a formulação MILP foi capaz de determinar a solução ótima em 35,4% das instâncias, enquanto ATIF e RL obtiveram a solução ótimas em 79,2% e 78,1% das instâncias, respectivamente. Assim como visto nos demais indicadores, houve equilíbrio no desempenho dos métodos ATIF e RL em todos os tamanhos de instâncias.

Tabela 8: Resumo dos Resultados ATIF-RL - Tamanhos das Instâncias

n	MILP			ATIF			RL		
	<i>Gap</i> (%)	<i>t</i> (s)	<i>Opt</i> (%)	<i>Gap</i> (%)	<i>t</i> (s)	<i>Opt</i> (%)	<i>Gap</i> (%)	<i>t</i> (s)	<i>Opt</i> (%)
10	0,04	227,3	99,6	0,00	0,6	100,0	0,00	0,3	100,0
15	5,56	2189,1	40,8	0,01	115,6	98,0	0,00	101,4	99,2
20	9,76	2831,1	22,4	0,05	432,6	90,4	0,02	512,8	90,4
25	8,88	3083,6	14,4	0,11	945,4	80,0	0,06	968,7	77,6
50	21,12	3600,0	0,0	1,20	2759,8	27,6	1,21	2852,3	23,2
100	56,64	3600,0	0,0	-	-	-	3,24	3600,0	0,0
<i>Avg.</i> 50	9,07	2386,2	35,4	0,28	850,8	79,2	0,26	887,1	78,1
<i>Avg.</i> 100	17,00	2588,5	29,5	-	-	-	0,76	1339,3	65,1

Através da Tabela 8 é possível perceber que há uma melhoria significativa nos resultados gerados tanto pela ATIF como pela RL, em relação ao MILP. Houve uma redução significativa do *Gap* médio em todos os tamanhos, um grande acréscimo na quantidade de soluções ótimas obtidas, além de uma diminuição no tempo médio de execução para se alcançar a solução ótima nas instâncias em que foram possíveis de serem determinadas até o tempo limite.

5. Considerações Finais

Este trabalho apresentou a aplicação das duas abordagens exatas para a resolução do problema de seleção e sequenciamento de ordens de produção. A primeira, uma formulação com arcos indexados no tempo, e a segunda, um procedimento *Branch and Bound* que utiliza uma relaxação lagrangeana para obter limitantes duais. Os métodos de resolução propostos foram testados nas instâncias propostas por Cesaret *et al.* (2012) com o intuito de se comparar os resultados obtidos com os principais resultados encontrados na literatura.

Através dos resultados obtidos foi constatado uma superioridade significativa das abordagens propostas em comparação com o MILP em todos os indicadores analisados. Com relação aos resultados obtidos pela ATIF e RL, houve equilíbrio na maioria dos grupos de instâncias. Considerando instâncias de até 50 tarefas, o *Gap* médio geral dos métodos ATIF e RL foram próximos, 0,28% e 0,26%, respectivamente, e bastante inferior aos 9,07% obtido pelo MILP. Com relação ao número de soluções ótimas obtidas, o equilíbrio entre os métodos propostos permaneceu, com o ATIF obtendo 79,2% de soluções ótimas e a RL 78,1%, assim como a superioridade de desempenho em comparação ao MILP, que obteve soluções ótimas em 35,4% das instâncias.

Por fim, com relação ao tempo médio de execução, ATIF obteve uma média de 850,8 segundos por execução, próximo ao desempenho da RL, que teve tempo médio de execução de 887,1 segundos, tendo ambos apresentado superioridade em relação ao desempenho obtido com o MILP, no qual apresentou uma média de 2386,2 segundos por execução. Além disso, é importante destacar que, tanto o modelo ATIF como a RL foram capazes de resolver problemas de tamanho de até 50 tarefas, enquanto a formulação MILP foi capaz de resolver instâncias de até 25 tarefas.

Como trabalhos futuros, pretende-se implementar um método baseado em geração de colunas, com o objetivo de encontrar soluções de qualidade superior e melhores limitantes duais.

Referências

- Cesaret, B., Oğuz, C. e Salman, F. S. (2012), A tabu search algorithm for order acceptance and scheduling. *Computers & Operations Research*, v. 39, n. 6, p. 1197 – 1205. Special Issue on Scheduling in Manufacturing Systems.
- Graham, R., Lawler, E., Lenstra, J. e Kan, A. R. Optimization and approximation in deterministic sequencing and scheduling: a survey. *Discrete Optimization*, volume 5 of *Annals of Discrete Mathematics*, p. 287 – 326. Elsevier, 1979.

- Held, M., Wolfe, P. e Crowder, H. P.** (1974), Validation of subgradient optimization. *Mathematical Programming*, v. 6, n. 1, p. 62–88.
- Kirlik, G. e Oguz, C.** (2012), A variable neighborhood search for minimizing total weighted tardiness with sequence dependent setup times on a single machine. *Computers & Operations Research*, v. 39, n. 7, p. 1506 – 1520.
- Lenstra, J., Kan, A. R. e Brucker, P.** Complexity of machine scheduling problems. P.L. Hammer, E.L. Johnson, B. K. e Nemhauser, G. (Eds.), *Studies in Integer Programming*, volume 1 of *Annals of Discrete Mathematics*, p. 343 – 362. Elsevier, 1977.
- Lopez, P. e Roubellat, F.** *Production Scheduling*. ISTE. Wiley, 2008.
- Nobibon, F. T., Herbots, J. e Leus, R.** Order acceptance and scheduling in a single-machine environment: exact and heuristic algorithms. *Proceedings of the 4th Multidisciplinary International Scheduling Conference: Theory and Applications, 2009, Dublin, Ireland*, p. 772–774, 2009.
- Nobibon, F. T. e Leus, R.** (2011), Exact algorithms for a generalization of the order acceptance and scheduling problem in a single-machine environment. *Computers & Operations Research*, v. 38, n. 1, p. 367 – 378. Project Management and Scheduling.
- Oğuz, C., Salman, F. S. e Yalçın, Z. B.** (2010), Order acceptance and scheduling decisions in make-to-order systems. *International Journal of Production Economics*, v. 125, p. 200–211.
- Rom, W. O. e Slotnick, S. A.** (2009), Order acceptance using genetic algorithms. *Computers & Operations Research*, v. 36, n. 6, p. 1758 – 1767.
- Slotnick, S. A.** (2011), Order acceptance and scheduling: A taxonomy and review. *European Journal of Operational Research*, v. 212, n. 1, p. 1 – 11.
- Slotnick, S. A. e Morton, T. E.** (2007), Order acceptance with weighted tardiness. *Computers & Operations Research*, v. 34, n. 10, p. 3029 – 3042.
- Subramanian, A., Battarra, M. e Potts, C. N.** (2014), An iterated local search heuristic for the single machine total weighted tardiness scheduling problem with sequence-dependent setup times. *International Journal of Production Research*, v. 52, n. 9, p. 2729–2742.
- Tanaka, S. e Araki, M.** (2013), An exact algorithm for the single-machine total weighted tardiness problem with sequence-dependent setup times. *Computers & Operations Research*, v. 40, n. 1, p. 344 – 352.
- Wang, X., Xie, X. e Cheng, T.** (2013), A modified artificial bee colony algorithm for order acceptance in two-machine flow shops. *International Journal of Production Economics*, v. 141, n. 1, p. 14 – 23. Meta-heuristics for manufacturing scheduling and logistics problems.
- Wei, J. e Ma, Y.-S.** (2014), Design of a feature-based order acceptance and scheduling module in an ERP system. *Computers in Industry*, v. 65, n. 1, p. 64 – 78.
- Xu, H., Lü, Z. e Cheng, T.** (2013), Iterated local search for single-machine scheduling with sequence-dependent setup times to minimize total weighted tardiness. *Journal of Scheduling*, p. 1–17.
- Zhong, X., Ou, J. e Wang, G.** (2014), Order acceptance and scheduling with machine availability constraints. *European Journal of Operational Research*, v. 232, n. 3, p. 435 – 441.