



JOSÉ ANTÓNIO PEREIRA GONÇALVES

Mestrado em Engenharia e Gestão Industrial

**ANÁLISE E DESENVOLVIMENTO DE MODELOS
PARA AGENDAMENTO DE EXAMES EM
MEDICINA NUCLEAR: UM ESTUDO DE CASO**

MESTRADO EM ENGENHARIA E GESTÃO INDUSTRIAL

Universidade NOVA de Lisboa

Draft: 25 de setembro de 2025



ANÁLISE E DESENVOLVIMENTO DE MODELOS PARA AGENDAMENTO DE EXAMES EM MEDICINA NUCLEAR: UM ESTUDO DE CASO

JOSÉ ANTÓNIO PEREIRA GONÇALVES

Mestrado em Engenharia e Gestão Industrial

Orientador: Duarte Nuno Caldeira Cordeiro Dinis

Professor Assistente, Universidade NOVA de Lisboa

Coorientadora: Ana Sofia Leonardo Vilela de Matos

Professora Associada, Universidade NOVA de Lisboa

MESTRADO EM ENGENHARIA E GESTÃO INDUSTRIAL

Universidade NOVA de Lisboa

Draft: 25 de setembro de 2025

RESUMO

Os custos associados ao setor de saúde têm vindo a aumentar, consequentemente será necessário encontrar maneiras de atenuar este crescimento. A solução passará pela utilização eficiente dos recursos disponível. Uma das vertentes menos exploradas no setor de saúde para este efeito é o escalonamento eficiente de cada exame ao longo do dia de trabalho.

Desta forma, através da exploração do departamento de medicina nuclear no Hospital Garcia de Orta, foram desenvolvidos vários modelos com o objetivo de ajudar os decisores a responder a duas perguntas: Quanto tempo é necessário para executar todos os exames; quantos exames podem ser feitos com o tempo disponível. O problema em estudo é uma generalização do problema *Job-Shop*, em específico *Flexible Multi-Resource Job-Shop with No-Wait*.

Esta dissertação é repartidas em três passos, no mapeamento dos exames e a respetiva recolha de dados, no formulação dos vários modelos, e na comparação dos modelos e resolução dos problemas práticos apresentados.

O mapeamento dos exames foi realizado através de entrevistas sucessivas com vários profissionais de forma a captar os recursos necessários em cada atividade, de seguida procedeu-se à recolha das duração de cada atividade.

Para cada uma das perguntas a responder foram criados quatro modelos, uma nova formulação *Mixed Integer Linear Programming* e três versões de *Simulated Annealing*, para esta última utilizou-se dois tipos de codificação para a solução.

Os modelos utilizados foram comparados utilizando problemas *benchmark* já existentes na literatura e através de problemas práticos apresentados pelo sistema em estudo. Contudo, não foi possível aplicar na prática a agenda gerada pelos modelos.

Esta dissertação permitiu elaborar um conjuntos de modelos que aumentam a eficiência com a qual decorre o escalonamento de exames, não só permitindo automatizar este processo como também aumentando o volume de exames que ocorrem num dia de trabalho.

Palavras-chave: Saúde, Otimização, Escalonamento, *Job-Shop*, Mapeamento de processos

ABSTRACT

Costs related to the health-care industry have grown, therefor there is the need to slow down this growth. A solution may be through the more efficient utilization of resources. A less explored option in the health-care industry is the efficient scheduling of exams throughout the day.

Utilizing the nuclear medicine department of the Garcia de Orta Hospital, models were developed with the objective of helping decision-makers answer to two questions: How much time is needed to perform all exams; how many exams can be performed in the allowed time. This problem is a generalization of Job-Shop, and can be said to be a Flexible Multi-Resource Job-Shop with No-Wait problem.

This dissertation is separated into three parts, the mapping out of exams and the respective data collection, the formulation of the various models, and the comparison between models and the resolution of practical problem presented.

The mapping of the exams was done though interviews with various workers in the department as to capture the needed resources by each activity, following that, the data collection was performed in order to have an estimate of the duration of each activity.

For each question there were created four models, a Mixed Integer Linear Programming formulation, as well as three version of Simulated Annealing, for which two different solution codifications were used.

The developed models were compared using benchmark instances from existing literature, as well as using practical problems presented by the system in study. However, it was not possible to apply in practice the generated schedules generated by the models.

This dissertation allowed the creation of different models that permit the increased efficiency with which exams are shceduled, not only allowing the process to be automated, but also increasing the amount of exams that are done in a given day.

Keywords: Health-care, Optimization, Scheduling, Job-Shop, Process mapping

ÍNDICE

Índice de Figuras	vi
List of Algorithms	ix
1 Introdução	1
1.1 Motivação e Contexto	1
1.2 Apresentação do Problema e Objetivo	1
1.3 Abordagem Metodológica	2
2 Base Teórica e Revisão de Literatura	3
2.1 Extensões, Restrições, e Critérios	3
2.2 Abordagens para a Resolução	4
2.2.1 Algoritmos Exatos	4
2.2.2 Heurísticas	5
2.2.3 Meta-Heurísticas	5
2.3 Abordagens para a Resolução	7
2.3.1 Algoritmos Exatos	7
2.3.2 Heurísticas	8
2.3.3 Meta-Heurísticas	9
3 Mapeamento dos Exames	10
4 Desenvolvimento dos Modelos	13
4.1 Problema de <i>Makespan</i>	13
4.1.1 Modelo 1	16
4.1.2 Modelo 2	21
4.1.3 Modelo 3	24
4.2 Problema do número de trabalhos	28
4.2.1 Modelo 1	30
4.2.2 Modelo 2	32

4.2.3	Modelo 3	35
5	Resultados e Discussão	38
5.1	Problema de <i>Makespan</i>	38
5.1.1	Modelo 1	39
5.1.2	Modelo 2	40
5.1.3	Modelo 3	43
5.1.4	Comparação entre modelos	44
5.1.5	Comparação com instâncias de <i>benchmark</i>	46
5.2	Problema do número de trabalhos	51
5.2.1	Modelo 1	52
5.2.2	Modelo 2	54
5.2.3	Modelo 3	55
5.2.4	Comparação entre modelos	56
5.2.5	Comparação com instâncias de <i>benchmark</i>	58
5.3	Discussão	62
6	Conclusão	65
Bibliografia		66
Apêndices		
A	Tabelas	71
A.1	Problema de <i>Makespan</i>	71
A.2	Problema do número de trabalhos	84
B	Imagens	96
B.1	Problema de <i>Makespan</i>	97
B.2	Problema do número de trabalhos	102
C	Algoritmos	105
C.1	Problema de <i>Makespan</i>	105
C.2	Problema do número de trabalhos	110

ÍNDICE DE FIGURAS

3.1 Utilização de recursos-tipo de uma agenda	12
3.2 Utilização de recursos-tipo repartidos por exames	12
4.1 Impacto das duas formas de calcular <i>makespan</i>	17
4.2 Impacto das duas forma de gerar t sobre a procura da vizinhança.	18
4.3 Processo de transição da solução s para a solução s' para o Modelo 1.	18
4.4 Evolução do valor da função objetivo ao longo da otimização do problema de <i>makespan</i> com o Modelo 1 ($L_k = 5000, CP = 100, \alpha = 0.8, P = 1000, p_0 = 0.9$).	20
4.5 Impacto do recálculo de <i>ini</i> e <i>fim</i> sobre os instantes considerados na próxima iteração.	21
4.6 Processo de transição da solução s para a solução s' para o Modelo 2.	22
4.7 Evolução do valor da função objetivo ao longo da otimização do problema de <i>makespan</i> com o Modelo 2 ($L_k = 500, CP = 100, \alpha = 0.9, P = 0, p_0 = 0.9$).	23
4.8 Impacto do recálculo de <i>ini</i> e <i>fim</i> sobre a lista de candidatos da próxima iteração.	24
4.9 Exemplo da solução 3-4-1-2	26
4.10 Exemplo da solução 2-3-1-4	26
4.11 Processo de transição da solução s para a solução s' para o Modelo 3.	27
4.12 Evolução do valor da função objetivo ao longo da otimização do problema de <i>makespan</i> com o Modelo 3 ($L_k = 500, CP = 100, \alpha = 0.9, P = 0, p_0 = 0.9$, <i>left-shift timetabling</i> , geração de vizinhos através de troca).	28
4.13 Processo de transição da solução s para a solução s' para o Modelo 1.	31
4.14 Evolução do valor da função objetivo ao longo da otimização do problema do número de trabalhos com o Modelo 1 ($L_k = 5000, CP = 100, \alpha = 0.8, P = 10, p_0 = 0.9$).	32
4.15 Processo de transição da solução s para a solução s' para o modelo 3 do problema 2.	33
4.16 Evolução do valor da função objetivo ao longo da otimização do problema do número de trabalhos com o Modelo 2 ($L_k = 100, CP = 100, \alpha = 0.8, P = 0, p_0 = 0.5$).	34

4.17	As duas possibilidades para o algoritmo de <i>timetabling</i>	35
4.18	As três trocas que podem ocorrer	36
4.19	Evolução do valor da função objetivo ao longo da otimização do problema do número de trabalhos com o Modelo 3 ($L_k = 500$, $CP = 100$, $\alpha = 0.8$, $P = 10$, $p_0 = 0.9$, <i>left-shift timetabling</i> , geração de vizinhos através de troca).	37
5.1	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 1.	41
5.2	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 2.	42
5.3	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com <i>left shifting</i>	43
5.4	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 1.	52
5.5	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 2.	54
5.6	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3.	55
A.1	Gráfico de contorno das cinco variáveis (L_k , CP , α , P , p_0), relativamente ao <i>makespan</i> , do Modelo 1 do problema de <i>makespan</i>	73
A.2	Gráfico de contorno das cinco variáveis (L_k , CP , α , P , p_0), relativamente ao tempo, do Modelo 1 do problema de <i>makespan</i>	75
A.3	Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao <i>makespan</i> , do Modelo 2 do problema de <i>makespan</i>	77
A.4	Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao tempo, do Modelo 2 do problema de <i>makespan</i>	79
A.5	Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao <i>makespan</i> , do Modelo 3 do problema de <i>makespan</i>	81
A.6	Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao tempo, do Modelo 3 do problema de <i>makespan</i>	83
A.7	Gráfico de contorno das cinco variáveis relevantes (L_k , CP , α , P , p_0), relativamente ao número de exames, do Modelo 1 do problema do número de trabalhos	85
A.8	Gráfico de contorno das cinco variáveis relevantes (L_k , CP , α , P , p_0), relativamente ao tempo, do Modelo 1 do problema do número de trabalhos	87
A.9	Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao número de trabalhos, do Modelo 2 do problema do número de trabalhos	89
A.10	Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao tempo, do Modelo 2 do problema do número de trabalhos	91

A.11	Gráfico de contorno das cinco variáveis relevantes (L_k, CP, α, p_0), relativamente ao número de trabalhos, do Modelo 3 do problema do número de trabalhos	93
A.12	Gráfico de contorno das quatro variáveis relevantes (L_k, CP, α, p_0), relativamente ao tempo, do Modelo 3 do problema do número de trabalhos	95
B.1	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 1 do problema de <i>makespan</i>	97
B.2	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 2 do problema de <i>makespan</i>	98
B.3	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com <i>left shifting</i> do problema de <i>makespan</i> . .	98
B.4	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com <i>enhanced left shifting</i> do problema de <i>makespan</i>	99
B.5	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com <i>non-delay</i> do problema de <i>makespan</i> . . .	100
B.6	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com <i>enhanced non-delay</i> do problema de <i>makespan</i>	101
B.7	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 1 do problema do número de trabalhos.	102
B.8	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 2 do problema do número de trabalhos.	103
B.9	Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 do problema do número de trabalhos.	103

LIST OF ALGORITHMS

1	Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição, e retrocesso. Para o problema de <i>makespan</i> com o Modelo 1.	105
2	Pseudo-código que gera os instantes candidatos durante a criação da vizinhança.	106
3	Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição, e retrocesso. Para o problema de <i>makespan</i> com o Modelo 2.	107
4	Pseudo-código do algoritmo para encontrar o instante mais cedo possível.	108
5	Pseudo-código de <i>left-shift timetabling</i>	108
6	Pseudo-código de <i>enhanced left-shift timetabling</i>	108
7	Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição, e retrocesso. Para o problema de <i>makespan</i> com o Modelo 3.	109
8	Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição. Para o problema do número de trabalhos com o Modelo 1.	110
9	Pseudo-código de retrocesso. Para o problema do número de trabalhos com o Modelo 1.	111
10	Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição. Para o problema do número de trabalhos com o Modelo 2.	112
11	Pseudo-código de retrocesso. Para o problema do número de trabalhos com o Modelo 2.	113
12	Pseudo-código de <i>left-shift timetabling</i> para o problema do número de trabalhos	113
13	Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição, e retrocesso. Para o problema do número de trabalhos com o Modelo 3.	114

INTRODUÇÃO

Este capítulo irá introduzir o tema desta dissertação, a sua estrutura, a motivação e contexto, os objetivos a atingir e a abordagem tomada.

1.1 Motivação e Contexto

O custo associado ao setor de saúde tem vindo a aumentar, com os países do G7 a pagar mais de 10% do seu GDP para este fim. O elevado custo deve-se a vários fatores, ao envelhecimento da população, à degradação do estilo de vida, ao aumento da estadia em hospital, entre outros [1].

A realização de exames é um das principais atividades de um hospital, como forma de apoio ao diagnóstico e tratamento. Em Portugal existe por isso tempos máximos de resposta garantidos [2], medicina nuclear (MN) insere-se no meio complementar de diagnóstico e terapêutica, devendo ocorrer um exame desta natureza em menos de 30 dias a partir da indicação clínica. Contudo, frequentemente estes limites não são cumpridos, devido à maior procura destes meios [3] e a ineficiências na realização desta atividade.

1.2 Apresentação do Problema e Objetivo

Esta dissertação teve como motivação principal a melhoria da eficiência do departamento de MN do Hospital Garcia de Orta que, a priori, seria através da melhoria do agendamento de pacientes ao longo do dia de trabalho de forma a realizar um número maior de exames com os recursos humanos e físicos já existentes.

Cada exame é composto por várias tarefas, na qual podem ser necessários vários recursos diferentes, não ocorrendo tempo de espera entre tarefas adjacentes de um mesmo exame. Esta dissertação pretende alcançar três objetivos:

1. O mapeamento dos exames mais frequentemente realizados, com possibilidade de mapear os exames que ocorrem com menor frequência;

2. A criação de ferramentas de apoio à decisão em contexto operacional;
3. A validação de resultados obtidos pelas ferramentas em termos de viabilidade e robustez.

1.3 Abordagem Metodológica

De forma a atingir os objetivos previamente referidos é necessário estruturar a abordagem metodológica. Será realizada uma revisão de literatura que abordará os tipos de problemas de agendamento e os respetivos métodos de resolução. De seguida serão mapeados e validados os exames, e posteriormente recolhidos dados relativos à duração de cada atividade. Seguidamente serão desenvolvidos os modelos de apoio à decisão já mencionados, com aplicação destes com os dados do sistema real com apresentação dos resultados obtidos, mas também a comparação de instâncias *benchmark* existentes na literatura. Finalmente, serão propostos trabalhos futuros, e serão retiradas conclusões.

BASE TEÓRICA E REVISÃO DE LITERATURA

2.1 Extensões, Restrições, e Critérios

O problema aqui apresentado pode ser englobado na família de problemas *Job-Shop*, um dos problemas mais estudados na área de investigação operacional.

Este problema é definido por um conjunto finito J de trabalhos, um conjunto finito R de recursos, e um conjunto finito O de operações. Cada trabalho j é uma sequência de n_j operações consecutivas em O , em que cada operação $i \in O$ é processada no recurso R_i com duração contínua de τ_i e sem antecipação de operações [4].

Contudo, o problema em estudo não é apenas descrito pela definição clássica de *Job-Shop*. Existe uma grande variedade de extensões, restrições, e critérios que permitem modificar o problema base de forma a ser mais fidedigno ao sistema em estudo. Para tal, o problema em estudo pode ser descrito como *Flexible Multi-Resource Job-Shop with No-Wait*.

A extensão *Flexible* foi descrita pela primeira vez por Brucker e Schlie [5] e dado o nome de *Flexible* por Brandimarte [6]. Esta adição é uma das mais importantes na generalização do problema *Job-Shop* para o contexto real, existe no entanto o aumento da complexidade e a impossibilidade de utilizar o método tradicional na sua resolução [4]. Existe para cada operação $i \in O$ um conjunto de recursos $R_i \subseteq R$ que podem ser utilizados, ou seja, existe um conjunto de recursos-tipo que exercem a mesma função, com duração de τ_i^k para o recurso $k \in R_i$.

A extensão *Multi-Resource* foi descrita pela primeira vez por Brucker e Neyer [7] e por Dauzère-Pérès et al. [8], na qual existe para cada operação $i \in O$, $m(i)$ recursos diferentes, tal que o conjunto R_i^k contém o k° recurso-tipo da operação O_i , ou seja, cada operação é processada por $m(i)$ recursos-tipo.

A restrição *Minimum/Maximum Time Lag* limita o tempo decorrido entre operações sequenciais de um trabalho, sendo *No-Wait* o caso extremo desta restrição, garantindo que não existe tempo de espera entre operações sequenciais. Este conceito foi proposto

por Callahan [9], tal que o começo da operação O_{i+1} ocorra no mesmo momento que a operação O_i termina, caso estas façam parte do mesmo trabalho j .

Ao mesmo tempo, também é possível resolver o problema apresentado de forma a maximizar ou minimizar diferentes critérios. Estes podem ser baseado em tempo, número de trabalhos, custo, receita, ou impacto no meio ambiente.

2.2 Abordagens para a Resolução

Existem três categorias gerais para a resolução de problemas *Job-Shop*: algoritmos exatos; heurísticas; e meta-heurísticas [10]. De seguida, serão apresentados alguns métodos de resolução do problema *Job-shop* e uma breve descrição de como funcionam.

2.2.1 Algoritmos Exatos

Historicamente, a utilização de métodos exatos foi a principal abordagem utilizada, com especial ênfase nos artigos iniciais sobre o problema clássico de *Job-Shop*. A utilização de *Branch and Bound(BB)* é a principal abordagem para a resolução deste problema, através de um algoritmo dedicado ou modelado através de *Mixed Integer Linear Programming (MILP)* [4].

A forma canónica de *Linear Programming* pode ser expressa por:

$$\text{Maximizar } z = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

Sujeito a:

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2$$

⋮

$$a_{m1}x_m + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_m$$

$$x_j \geq 0, j = 1, \dots, n$$

Existem vários modelos utilizados para representar um problema *Job-Shop*: o modelo disjuntivo; o modelo indexado no tempo; e o modelo baseado em ordenação [11].

O modelo disjuntivo, ou de Manne [12] utiliza um grafo denominado $G(V, C \cup D)$, em que V é o conjunto de vértices e que representam as operações dos vários trabalhos, C é o conjunto de arcos conjuntivos que ligam as operações i e $i + 1$ de um trabalho, D é o conjunto de arcos disjuntivos que ligam as operações que requerem o mesmo recurso. Desta forma é escolhida a ordem de processamento em cada recurso de forma a cumprir

com as restrições e minimizando/maximizando o critério em estudo. O modelo indexado no tempo, ou de Bowman [13], procura definir o instante de início de cada operação, sendo esta a variável de decisão que o modelo utiliza. O modelo baseado em ordenação, ou de Wagner [14], define explicitamente a sequência de operações que devem ocorrer em cada recurso.

Caso se formule o problema através de *MILP* será necessário utilizar um *solver*, como o Gurobi ou o CPLEX, que utilizam algoritmos como *BB* ou *Cutting Planes* na resolução do problema.

Por sua vez, *Constraint Programming (ConP)* tem vindo a ser mais estudo para a resolução destes problemas [4]. Em problemas de escalonamento, utilizam-se variáveis de intervalo com quatro atributos: se um intervalo se encontra na solução, o tempo de início do intervalo, o tempo de término do intervalo, e o tamanho deste [15].

2.2.2 Heurísticas

Heurísticas são algoritmos que devem permitir a obtenção de uma solução viável de forma rápida, por isso será de esperar que esta abordagem não resulte na obtenção da solução ótima. Frequentemente, a formulação de uma heurística apenas funcionará para o problema para qual foi desenhada, devido à intuição apresentada na sua construção [16].

2.2.3 Meta-Heurísticas

Meta-heurísticas são o meio termo entre a qualidade das soluções apresentadas por algoritmos exatos e a rapidez das heurísticas. Existe uma grande quantidade de abordagens existentes, podem ser baseadas em população, em trajetórias, com memória ou sem, ou baseadas em processos naturais.

Algumas das meta-heurísticas com mais impacto são: *Simulated Annealing (SA)*; *Genetic Algorithm (GA)*; *Differential Evolution (DE)*; *Particle Swarm Optimization (PSO)*; e *Tabu Search (TS)* [16]. Todas estas meta-heurísticas poderão ser utilizadas para resolver o problemas apresentado.

SA é uma meta-heurística evolucionária baseada no processo de arrefecimento de metais, em que se verifica uma menor energia do sistema quando este arrefecimento acontece lentamente. Desta forma, a solução s com uma vizinhança S_s de onde é gerada uma nova solução s' , que será avaliada consoante a função objetivo $f(s')$. A solução s' poderá ser aceite mesmo não apresentando melhor valor, sendo esta a principal diferenciação de

outras meta-heurísticas, de acordo com o algoritmo Metropolis [17]:

$$Pr\{\text{aceitar } s'\} = \begin{cases} 1, & \text{se } f(s') < f(s) \\ e^{\frac{f(s)-f(s')}{c_k}}, & \text{caso contrário} \end{cases}$$

Onde c_k é a temperatura atual. No início do algoritmo, quando a temperatura é alta, existe probabilidade acrescida de aceitar soluções piores, com o arrefecer da temperatura com cada iteração, a probabilidade de aceitar soluções piores que a atual diminui. Também é necessário definir qual a temperatura inicial do algoritmo, como se realiza o arrefecimento de c_k , quantas vizinhanças procurar por iteração L_k , e quando se dá por terminado o algoritmo CP . Cada vizinho visitado pode, ou não, representar uma solução viável, deve-se ter cuidado na escolha da representação da solução e na geração da vizinhança para garantir a viabilidade da solução, ou possivelmente relaxar certas restrições ao incluir a punição sobre a sua violação na função objetivo.

GA é uma meta-heurística populacional baseada no conceito de sobrevivência do mais apto. Inicia-se o algoritmo ao gerar vários membros aleatoriamente, tal que cada um seja representado por um cromossoma, a codificação dessa solução. Para a geração de novas soluções ocorre o cruzamento entre membros, normalmente selecionadas entre os melhores, de forma a obter uma nova descendência. A essa descendência ocorre a mutação do cromossoma para que ocorra maior exploração das soluções [18].

DE é uma meta-heurística populacional também baseada no conceito de sobrevivência do mais apto. Contrariamente a *GA*, nem toda a descendência é aceite e os pais removidos, diminuindo o risco de substituir boas soluções por piores [19]. Inicia-se com uma população inicial, de tamanho NP , de vetores com valores uniformes entre 0 e 1, tal que $X_{m,n,G}$ seja o valor do vetor m na posição n da iteração G . Três vetores r_1 , r_2 e r_3 são gerados de forma a que cada seja composto por uma sequência de tamanho NP , de seguida o vetor mutante é gerado por:

$$V_{m,n,G} = X_{r_1,n,G} + F(X_{r_2,n,G} - X_{r_3,n,G})$$

Por sua vez os vetores a testar são gerados por:

$$U_{m,n,G+1} = \begin{cases} V_{m,n,G}, & \text{se } rand() \leq CR \\ X_{m,n,G}, & \text{caso contrário} \end{cases}$$

Tendo que $rand()$ seja um valor aleatório uniforme entre 0 e 1, e que CR seja a taxa de aceitação de cruzamento. Num passo seguinte é necessário descodificar a solução gerada para a avaliar, o que depende do problema a resolver [20].

PSO é uma meta-heurística de enxame baseada no comportamento natural de animais de rebanho durante a procura de comida. Para tal são gerados P partículas, tal que a

partícula i se encontra na posição $X^i(t)$ na iteração t , bem como uma velocidade $V^i(t)$. Com cada iteração a posição e velocidade de cada partícula é atualizada:

$$X^i(t+1) = X^i(t) + V^i(t+1)$$

$$V^i(t+1) = \omega V^i(t) + c_1 r_1(pbest^i - X^i(t)) + c_2 r_2(gbest - X^i(t))$$

Onde r_1 e r_2 são valores aleatórios uniformes entre 0 e 1, ω , c_1 e c_2 são os parâmetros da meta-heurística e chamam-se de inércia, coeficiente cognitivo e coeficiente social, respectivamente. $pbest^i$ é a melhor solução encontrada pela partícula i , $gbest$ é a melhor solução já encontrada [21].

TS é uma meta-heurística populacional que utiliza memória para guiar a procura. Através de uma solução x um conjunto de soluções vizinhas são geradas através da sua modificação, o melhor vizinho \tilde{x} é comparada com a melhor solução já encontrada, que se torna no ponto de começo da próxima iteração. A lista tabu inclui um conjunto dos últimos movimentos tomados, o que limita os vizinhos que podem ser visitados em cada iteração, ajudando a guiar a procura. Como a lista é de tamanho limitado, os movimentos mais antigos saem e voltam a estar disponíveis.

2.3 Abordagens para a Resolução

Nesta secção será apresentada uma breve revisão da literatura existente sobre as abordagens para a resolução dos problemas *Job-Shop* e os seus derivados. A literatura aqui apresentada não será exaustiva, mas demonstrará algumas tendências para problemas próximos ao apresentado nesta dissertação.

O critério mais estudado nesta área de investigação é a minimização do *makespan* [22], tido como o tempo necessário para completar todos os trabalhos, mesmo assim existem alguns critérios diferentes que têm aparecido na literatura. Desta forma, pretende-se dar a conhecer alguns destas outras vertentes.

2.3.1 Algoritmos Exatos

Algoritmos exatos foram a principal abordagem utilizada para resolver problemas de *Job-Shop* mas têm vindo a tornar-se menos considerados com o aumento do número de extensões e restrições que os novos problemas apresentam [10].

Como já referido, existem três grandes modelos utilizados neste espaço. O modelo disjuntivo, o modelo indexado no tempo, e o modelo baseado em ordenação. Ku e Beck [11] avaliaram os modelos de acordo com a sua eficiência computacional, onde se observa que o modelo disjuntivo é o mais eficiente a resolver problemas *Job-Shop* de pequenas e

grandes dimensões. Será razoável esperar que este facto seja verdade para outras vertentes do problema. Ao mesmo tempo compararam as formulação em *MILP* com uma de *ConP*, evidenciando a eficiência deste última.

Dauzère-Pérès et al. [4] apresentam uma formulação do modelo disjuntivo utilizando *MILP* para resolver o problema *Flexible Job-Shop* com o objetivo de minimizar o *makespan*, bem como este deve ser modificado de forma a poderem ser adicionadas novas extensões e restrições.

Existem também artigos que geram ou criam problemas que se tornam em *benchmarks*, sendo utilizados em vários artigos de forma a comparar resultados. Um exemplo é o conjunto de problemas *Small Size Flexible Job-Shop Scheduling (SFJS)* e o conjunto *Medium and Large Size Flexible Job-Shop Scheduling (MFJS)* criado por Fattahi et al. [23], onde também se apresenta uma formulação para resolver o problema *Flexible Job-Shop*. Em Ozbaikir et al. [24] é apresentada outra formulação baseada no modelo disjuntivo, ao mesmo tempo é comparada com o modelo de Fattahi et al. [23] em relação à eficiência computacional, verificando que esta formulação é superior na maioria das instâncias. Em Thörnblad et al. [25] é apresentada e comparada uma formulação baseada no modelo indexado no tempo, onde se observa a inferior eficiência deste modelo.

Samarghandi [26] propõe um modelo *MILP* para a resolução do problema *No-wait Job-Shop* com o objetivo de minimizar o *makespan* com a restrição adicional de não ultrapassar a data limite de cada trabalho. Também apresenta dois modelos de *Constraint Programming* que resultam em boas soluções relativamente ao modelo *MILP* em problemas de grande dimensão.

Em Behmanesh et al. [27] é apresentado um modelo *MILP* para resolver o problema *Flexible Multi-Resource Job-Shop*.

2.3.2 Heurísticas

A heurística Nawaz-Enscore-Ham (NEH) [28] foi desenvolvida para o problema *Flow-Shop*, mas que já foi utilizada em vários artigos, especialmente na criação de soluções iniciais em problemas *Job-Shop* [29], [30].

Um possível relaxamento do problema em estudo, *Blocking No-Wait and Controlable Times* ou *Blocking No-Wait and Maximum Time Lag*, já foi estudado. Em Aschauer et al. [31] é apresentada uma heurística para o sequenciamento de trabalhos e um algoritmo de *timetabling*, apresentando boas soluções em pouco tempo.

2.3.3 Meta-Heurísticas

Xia e Wu. [32] utilizam um algoritmo híbrido entre *PSO* e *SA* na resolução do problema *Flexible Job-Shop* com tempos de processamento diferentes dentro de cada recurso-tipo. Onde *PSO* é utilizada na atribuição de operações às máquinas, enquanto *SA* é utilizada para escalar operação a cada recurso.

Sriboonchandr et al. [20] utilizam *DE* na resolução do problema *Flexible Job-Shop*, contudo não apresenta a qualidade esperada das soluções.

Jonathan et al. [33] comparam quatro meta-heurísticas para resolver o problema *Flexible Flow-Shop with No-Wait*. A meta-heurística proposta, *Whale Optimization Algorithm*, gera solução com qualidade semelhante a *SA*, *GA* e *Minimum Deviation Algorithm*. Esta última demonstra tempo computacional muito menor.

Ying e Lin [29] comparam quatro variantes de *SA* na resolução do problema *Job-Shop with No-Wait* com minimização do tempo total de conclusão. Demonstram que a utilização adicional do retorno e a existência de várias soluções iniciais permitem a obtenção de melhores soluções no mesmo espaço de tempo.

Perrachon et al. [34] sugerem a utilização de *SA* para a otimização do problema *Flexible Multi-Resource Job-Shop*. Para tal, são exploradas oito diferentes estruturas da vizinhança baseadas no modelo disjuntivo. A restrição *Blocking* encontra-se relaxada neste problema, ao ser possível que vários recursos sejam necessários durante fases diferentes de cada operação, podendo ser visto como a decomposição da operação em várias partes com a restrição de *No-Wait* entre elas.

Caumond et al. [35] apresentam uma meta-heurística baseada no modelo disjuntivo, utilizando *Memetic Algorithm* para a resolução do problema *Job-Shop with No-Wait* e *Job-Shop with Maximum Time Lag*.

MAPEAMENTO DOS EXAMES

O processo de mapeamento dos exames começou com a decisão sobre quais incluir no estudo. Nem todos os exames foram considerados devido à baixa frequência com que estes ocorrem e a consequente falta de dados em relação à duração das suas atividades. Desta forma, apenas foram mapeados aqueles que ocorrem pelos menos mensalmente. De seguida, foi necessário definir os recursos-tipo existentes no sistema, estes podem ser divididos em recursos humanos e em recursos físicos. A Figura 3.1 apresenta cada recurso-tipo, a correspondente abreviatura e quantidade.

Tabela 3.1: Capacidade dos vários recursos,

Tipo	Nome do Recurso	Abreviatura	Quantidade
Recursos Humano	Técnicos Auxiliares de Saúde	TAS	4
	Enfermeiros	ENF	4
	Técnicos Superiores de Diagnóstico e Terapêutica	TSDT	4
	Médicos Especialistas	ME	2
	Médicos Cardiologistas	MC	1
Recursos Físico	Gabinete Médico	GB	2
	Sala de Espera	SE	4
	Sala de Espera de Crianças	SEC	3
	Sala de Espera para Doentes Acamados	SC	2
	Sala Polivalente	SP	1
	Sala Enfermagem 1	SE1	1
	Sala Enfermagem 2	SE2	3
	Sala Enfermagem 1	SE3	3
	Sala Câmara Gama	SCG	1
	Sala Tomografo	ST	1

Cada recurso-tipo tem uma quantidade de recursos disponíveis. Para os recursos humanos corresponde ao número de profissionais em cada grupo existentes no departamento e para os recursos físicos corresponde à lotação de pacientes que podem suportar. Iremos considerar que existe portanto uma capacidade máxima para cada recurso-tipo. Admite-se que cada recurso atribuído a uma tarefa será utilizado durante a sua duração total.

Considerou-se ainda a possibilidade de ter a capacidade dos recursos variável com o tempo, de forma a modelar os diferentes turnos e os períodos onde existe mudança entre estes. Com mais discussão decidiu-se que as diferenças existente ao longo do dia relativamente à capacidade de cada recurso é mínimo, mas será fácil incluir este aspeto num estudo futuro.

Existem recursos-tipo que podem ser equiparados a outros em certas atividades. A SP e a SE1 podem ser tidas como equivalentes em certas atividades. Por isso foi necessário criar recursos-tipo fictícios. Para o exemplo anterior, foi criado o recurso-tipo SP ou SE1, sempre que uma destas salas é explicitamente necessária, a sua utilização é representadas no recurso-tipo correspondente e no recurso-tipo fictício, caso seja indiferente a sala necessárias a utilização apenas é representada no recurso-tipo fictício.

A SE2 e a SE3 podem também ser tidas como equiparadas, mas quando a SE2 é explicitamente necessária toda ela é ocupada por uma pessoa, efetivamente diminuindo a sua capacidade de três para uma pessoa, ocorrendo em algumas atividades do exame cintigrafia pulmonar de ventilação/inalação + perfusão. O recurso fictício SE2 ou SE3 continua a ter uma capacidade de seis pessoa. Ambos os casos apresentam-se na Tabela 3.2, onde se observa a capacidade efetiva destes recursos.

Tabela 3.2: Capacidade dos recursos-tipo fictícios

Recursos-tipo fictícios	SP	SE1	SP ou SE1	SE2	SE2 ou SE3
Quantidade	1	1	2	1	6

Num passo seguinte, cada exame foi decomposto nas suas atividades. Considerou-se que uma nova atividade deve começar quando existe uma mudança dos recursos necessários. Foram então realizadas entrevistas de forma a realizar a decomposição dos exames. No seguimento das entrevistas as atividades foram validadas pelos vários grupos profissionais. Rapidamente, SE identificou a necessidade de diferenciar o mapeamento entre pacientes com mobilidade autónoma, aqueles com mobilidade condicionada, e crianças existindo diferenças nos recursos necessários e na duração de cada atividade.

Seguidamente, as duração de cada atividade foram recolhidas utilizando folhas de registo provenientes do mapeamento realizado. A duração média de cada atividade foi obtida através da hora de início e de fim de cada atividade registadas pelos profissionais. Será de salientar que mesmo sendo este um problema de *No-Wait*, esta restrição nem sempre foi verificado nas folhas de registo.

Em certas atividades considera-se que a permanência de um recurso durante a sua totalidade como uma simplificação do problema. Podemos utilizar como exemplo a atribuição de um TAS ao tempo de espera decorrido dentro do departamento por cada paciente, contudo não foi possível definir explicitamente quando é que estes recursos seriam necessários, sendo esta uma fonte de variabilidade do sistema, impossibilitando a

CAPÍTULO 3. MAPEAMENTO DOS EXAMES

decomposição da atividade.

Todas as folhas de registo e respetivos dados estão em anexo a esta dissertação.

Através da Figura 3.1 podemos observar qual a utilização dos recursos provenientes de uma agenda, em conjunto com a Figura 3.2 torna-se possível identificar quais exames podem ser inseridos sem ter grande impacto sobre o valor de *makespan*.

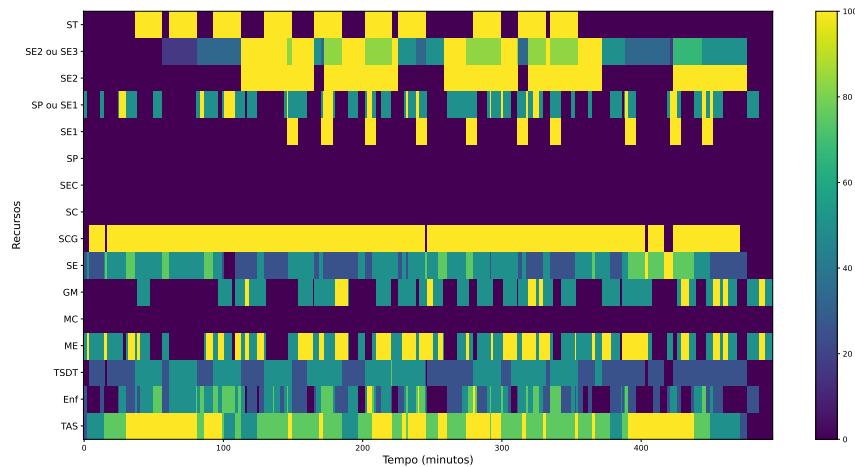


Figura 3.1: Utilização de recursos-tipo de uma agenda

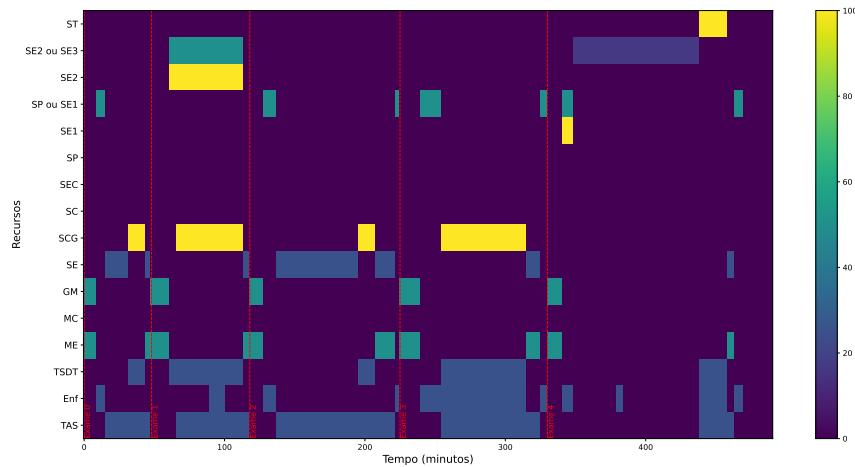


Figura 3.2: Utilização de recursos-tipo repartidos por exames

Visualmente identifica-se que SCG é o principal recurso limitante ao estar quase toda a sua duração a ser utilizado na totalidade. Outros recursos que são também limitantes são ST e SE2, estes são utilizados por menos exames, contudo os exames que os necessitam tendem a ocorrer em grandes números. Recursos como SP, SEC, SC, e MC só são necessários em casos específicos e por isso não tendem a apresentar utilização alta.

DESENVOLVIMENTO DOS MODELOS

Neste capítulo, dois problemas devem ser resolvidos, tendo para isso sido desenvolvidos vários modelos. No primeiro problema existem recursos e exames fixos, sendo o objetivo minimizar o tempo entre o começo e o fim de todos os exames, chamado de *makespan*. O segundo problema tem recursos e tempo fixos, sendo o objetivo maximizar o número de exames a realizar, utilizando um somatório simples ou ponderado.

4.1 Problema de *Makespan*

Este problema utiliza o exemplo de uma segunda-feira típica em termos de exames realizados. Ocorrendo 3 cintigrafia tiroideia, 5 cintigrafia pulmonar de ventilação/inalação + perfusão, 10 cintigrafia miocárdica de perfusão em repouso, 1 cintigrafia das glândulas salivares, 10 PET - estudo corpo inteiro com FDG.

A próxima formulação segue sensivelmente a formulação de Kondili et al. [36], com as modificações necessárias para obedecer às extensões e restrições propostas, continuando a ser uma formulação indexada no tempo, chamá-la-emos de *MILP-trad*:

Conjuntos:

O conjunto de trabalhos $I, i \in I := (1, \dots, n)$

O conjunto de operações $K, k \in K := (1, \dots, K_{\max})$

O conjunto de recursos-tipo $R, p \in R := (1, \dots, R_{\max})$

O conjunto de instantes de tempo $T, t \in T := (1, \dots, T_{\max})$

Parâmetros:

$\rho_{i,k}$ é o tempo de processamento da operação k do trabalho i

$r_{i,p,k}$ é quantidade p necessária para realizar a operação k do trabalho i

C_p é a capacidade do recurso p

Variáveis de Decisão:

$X_{t,i,k}$ é uma variável binária com valor 1 se a operação k do trabalho i ocorrer durante o instante t , caso contrário tem valor 0

$Z_{t,i,k}$ é uma variável binária com valor 1 se a operação k do trabalho i começar no instante t , caso contrário tem valor 0

$S_{i,k}$ é uma variável inteira que representa o valor de começo da operação k do trabalho i

$F_{i,k}$ é uma variável inteira que representa o valor de fim da operação k do trabalho i

C_{\max} é uma variável inteira que representa o *makespan*

Função Objetivo:

$$\min C_{\max} \quad (4.1)$$

Sujeito a:

$$F_{i,K_{\max}} \leq C_{\max} \quad \forall i \quad (4.2)$$

$$\sum_t Z_{t,i,k} = 1 \quad \forall i, k \quad (4.3)$$

$$\sum_t X_{t,i,k} = \rho_{i,k} \quad \forall i, k \quad (4.4)$$

$$\sum_{t'=t}^{t+\rho_{i,k}} X_{t',i,k} \geq \rho_{i,k} Z_{t,i,k} \quad \forall i, k, t = 1, \dots, T - \rho_{i,k} \quad (4.5)$$

$$S_{i,k} = \sum_t t Z_{t,i,k} \quad \forall i, k \quad (4.6)$$

$$S_{i,k+1} = F_{i,k} \quad \forall i, k \quad (4.7)$$

$$F_{i,k} - S_{i,k} = \rho_{i,k} \quad \forall i, k \quad (4.8)$$

$$\sum_i \sum_k r_{i,p,k} X_{t,i,k} \leq C_p \quad \forall t, p \quad (4.9)$$

A função objetivo (4.1) minimiza o *makespan* dos trabalho, ou seja, o tempo entre o início do primeiro exame, e o fim do último.

A restrição (4.2) garante que todos os trabalho acabam antes ou no instante do *makespan*, auxiliando a função objetivo.

A restrição (4.3) garante que só existe um instante em que a operação k do trabalho i é atribuído.

A restrição (4.4) garante, que para a operação k do trabalho i , o somatório de $X_{t,i,k}$ sobre t tem o valor da duração $\rho_{i,k}$.

A restrição (4.5) garante a concordância do posicionamento sobre t entre $X_{t,i,k}$ e $Z_{t,i,k}$.

A restrição (4.6) garante a coerência do momento de início da tarefa k .

A restrição (4.7) garante que, para o trabalho i , a operação $k + 1$ começa quando a operação k acaba, verificando a restrição de *No-Wait*.

A restrição (4.8) garante que a duração da operação k para o trabalho i é de $\rho_{i,k}$.

A restrição (4.9) garante que não existe a sobre-utilização do recursos p durante o instante

t , verificando as extensões *Multi-Resource* e *Flexible*.

Por outro lado, é possível formular modelos mais eficientes do que o clássico. A formulação aqui sugerida, *MILP- δ* , tira partido do facto da restrição *No-Wait* criar um padrão de utilização de recursos, qualquer que seja o instante t onde este se insere. Desta forma, não é necessário explicitamente formular restrições que garantem *No-Wait*.

Conjuntos:

O conjunto de trabalhos $I, i \in I := (1, \dots, n)$

O conjunto de recursos-tipo $R, p \in R := (1, \dots, R_{\max})$

O conjunto de instantes de tempo $T, t \in T := (1, \dots, T_{\max})$

Parâmetros:

ρ_i é duração do trabalho i

$\delta_i(u, p)$ é a quantidade de recursos do tipo p necessários a um offset de u instantes de tempo no trabalho i

C_p é a capacidade do recurso p

Variáveis de Decisão:

$Z_{t,i}$ é uma variável binária com valor 1 se o trabalho i começar no instante t , caso contrário tem valor 0

F_i é uma variável inteira que representa o valor de fim do trabalho i

C_{\max} é uma variável inteira que representa o *makespan*

Função Objetivo:

$$\min C_{\max} \quad (4.10)$$

Sujeito a:

$$F_i \leq C_{\max} \quad \forall i \quad (4.11)$$

$$\sum_{t=0}^{T_{\max}-\rho_i+1} Z_{t,i} = 1 \quad \forall i \quad (4.12)$$

$$\sum_{t=0}^{T_{\max}-\rho_i+1} Z_{t,i} * (t + \rho_i) = F_i \quad \forall i \quad (4.13)$$

$$\sum_i \sum_{\tau=\max(0, t-\rho_i+1)}^{\min(t, T_{\max}-\rho_i)} \delta_i(t - \tau, p) Z_{\tau,i} \leq C_p \quad \forall t, p \quad (4.14)$$

A função objetivo (4.10) minimiza o *makespan* dos trabalhos.

A restrição (4.11) garante que todos os exames acabam antes ou no instante do *makespan*, auxiliando a função objetivo.

A restrição (4.12) garante que só existe um instante de começo do trabalho i .

A restrição (4.13) faz a ligação entre as variáveis $Z_{t,i}$ e F_i .

A restrição (4.14) garante que não há sobre-utilização de recursos, ao verificar, para cada instante t , quais trabalhos estão a ser executados e em que fase este se encontra, garantindo que em cada instante não há sobre-utilização de recursos.

Para ambas das formulações apresentadas, o tamanho do conjunto de instantes de tempo, T , tem impacto no número de variáveis e restrições. Será por isso importante encontrar o menor valor de T_{\max} possível mas que seja maior que *makespan*, a isto tipicamente chama-se de *upper-bound*.

De seguida, foram desenvolvidos três modelos, utilizando *SA*, com duas codificações de solução diferentes. Os dois primeiros modelos utilizam uma codificação que representa o instante de início de cada trabalho. O outro modelo utiliza a sequência de escalonamento de cada trabalho como solução.

SA foi a meta-heurística escolhida devido a alguns fatores. Devido ao algoritmo de Metropolis, esta meta-heurística é capaz de escapar a mínimos locais através da aceitação de soluções que pioram a função objetivo. Ao mesmo tempo, com a escolha correta dos níveis das variáveis do algoritmo existe garantia de se alcançar a solução ótima [37].

4.1.1 Modelo 1

Este modelo é definido pelo o facto de não existir a garantia da viabilidade de cada solução. Por isso é necessário criar uma função objetivo diferente das restantes, esta será dada por:

$$f(s) = C_{\max} + P \sum_{t=start}^{end-1} \sum_p v(t, p)$$

Onde C_{\max} é *makespan*, P a punição relativamente à sobre-utilização de recursos (sendo esta uma variável a estudar). Esta sobre-utilização é dada pelo somatório dos recursos e instantes de tempo, $v(t, p)$ que por sua vez é dado por:

$$v(t, p) = \begin{cases} 1 & \text{se } u(t, p) > C_p \\ 0 & \text{caso contrário} \end{cases}$$

Temos que $u(t, p)$ é a utilização de cada recurso p em cada instante t , C_p será a capacidade máxima de cada recurso, tornando-se assim uma variável binária para cada recurso e instante. Poderia ter-se considerado uma definição diferente de $v(t, p)$:

$$v(t, p) = \begin{cases} u(t, p) - C_p & \text{se } u(t, p) > C_p \\ 0 & \text{caso contrário} \end{cases}$$

Apesar de corresponder melhor à verdadeira sobre-utilização de recursos, ao diferenciar a quantidade, esta formulação apresenta piores resultados.

Será de salientar como C_{\max} é calculado, dado que existem duas formas semelhantes para tal. A Figura 4.1a apresenta a primeira forma, temos que *makespan* é dado pelo máximo do término de cada trabalho, fim . Por sua vez, a Figura 4.1b apresenta a alternativa para o cálculo de *makespan*, ao considerar que este valor é obtido pela diferença entre o valor máximo de término de cada trabalho e o valor mínimo de início de cada trabalho, $fim - ini$. Ambas as figuras representam três soluções distintas, com o segmento a vermelho a demonstrar o aparente valor de *makespan*. Utilizou-se a segunda forma para calcular *makespan* porque esta não pune uma solução só por esta não começar no instante 0.

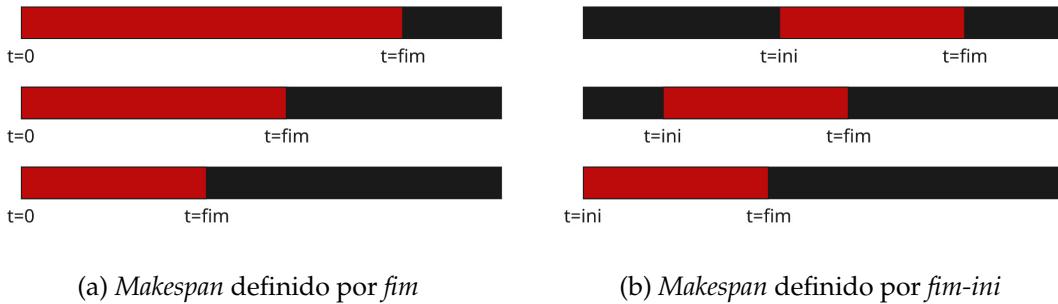
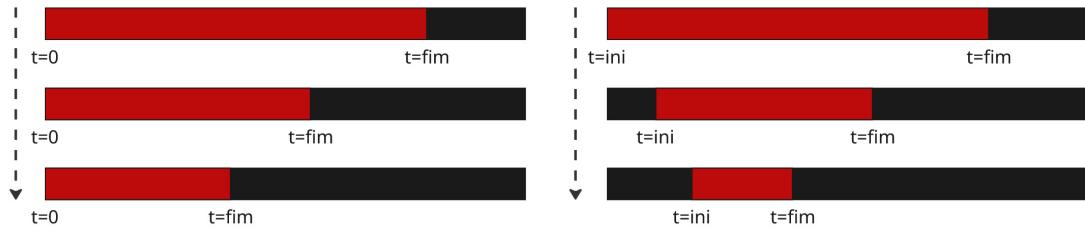


Figura 4.1: Impacto das duas formas de calcular *makespan*.

Para este modelo, a solução s é dada pelo instante t em que começa cada trabalho i . Por sua vez, a vizinhança de s , S_s , é definida pelo trabalho a reagendar e quando este deve começar. Ambos estes valores são obtidos aleatoriamente, contudo t poderá ser obtido através de uma distribuição uniforme $U(0, makespan\text{-duração})$, ou $U(ini, fim\text{-duração})$, sendo a duração o tempo necessário para completar o trabalho i . Esta diferença aparentemente pequena gera soluções finais diferentes.

A Figura 4.2 salienta a diferença entre estes dois métodos. A Figura 4.2a e a Figura 4.2b representam uma sequência de soluções vizinhas, com o segmento a vermelho a demonstrar o valor de *makespan* calculado como já discutido. Enquanto que para o método apresentado na Figura 4.2b é possível reduzir *makespan* através do atraso do trabalho de início e do adiantamento do trabalho de término. Para o método apresentado na Figura 4.2a apenas é possível reduzir *makespan* através do adiantamento do trabalho de término. Utilizou-se o método apresentado na Figura 4.2b.



(a) Instante gerado por $U(0, \text{makespan-duração})$ (b) Instante gerado por $U(\text{ini}, \text{fim-duração})$

Figura 4.2: Impacto das duas forma de gerar t sobre a procura da vizinhança.

O processo de modificar a solução atual s de forma a obter uma nova solução s' da vizinhança é resumido à figura 4.3. Sendo depois necessário atualizar a utilização dos recursos em cada instante, $u(t, p)$, recalcular $f(s')$, decidir se a nova solução é aceite ou não, atualizar a melhor solução já encontrada se for o caso, e o processo de retrocedimento. O Algoritmo 1, que se encontra no anexo C, descreve este processo.

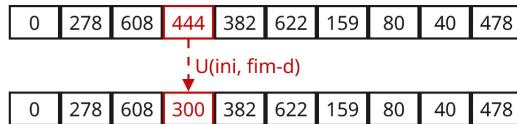


Figura 4.3: Processo de transição da solução s para a solução s' para o Modelo 1.

A temperatura inicial segue o esquema IT6 [38]. Mais especificamente, gerou-se uma boa solução inicial utilizando a heurística NEH, e de seguida procedeu-se a uma caminhada aleatório da vizinhança desta solução utilizando $c = \infty$ e $L_k = 100000$, de forma a aceitar todas as novas soluções encontradas. Definindo-se então que a temperatura inicial c_0 é dada por:

$$c_0 = |\Delta_{avg} / \log(p_0)|$$

Onde Δ_{avg} é a diferença média entre a solução atual s e o vizinho s' , e p_0 é a probabilidade inicial de aceitar cada solução, este último valor será uma das variáveis a estudar. O critério de paragem, CP , utilizado foi uma modificação de SC9 [38], onde se termina o algoritmo quando um número fixo de reduções de temperatura não geram novas soluções melhores, este valor será uma das variáveis a estudar.

O critério de aceitação utilizado foi o tradicional descrito por Metropolis [17], ou seja, AC1 [38].

O esquema de arrefecimento utilizado foi CS2 [38], dado por $c_{k+1} = \alpha c_k$, α tipicamente é um valor perto de 1, de forma a que a redução da temperatura se dê lentamente, este valor será uma das variáveis a estudar.

O comprimento da temperatura, ou seja, L_k , será uma variável a estudar, adotou-se o critério

TL1 [38], considerando um número fixo de iterações durante cada redução de temperatura.

Existem três diferentes algoritmos considerados para a geração da solução inicial: um algoritmo *greedy*; um algoritmo aleatório; e a heurística NEH. O primeiro agenda cada trabalho aleatoriamente de forma a que o fim do trabalho i corresponde ao início do trabalho $i + 1$, garantindo que esta solução não apresenta sobre-utilização de recursos. O segundo agenda cada trabalho de forma aleatória, sem atender à possível sobre-utilização de recursos. Verificou-se que o segundo algoritmo apresenta, na sua generalidade, melhores soluções em relação ao primeiro algoritmo.

Podemos observar a Figura 4.4 de acordo com o comportamento de cada solução e o porquê de se rejeitar a heurística NEH como solução inicial. Nas Figuras 4.4a, 4.4c, 4.4e observa-se um pico da função objetivo devido à elevada temperatura inicial, que por sua vez causa a aceitação da maioria das soluções candidatas, algo necessário para haver alguma independência entre a solução inicial e a final. Com o decair da temperatura a aceitação contrai, havendo aqui grande parte da melhoria observada. Quando a temperatura é muito reduzida, SA transforma-se numa simples procura local, ao tornar-se quase impossível aceitar soluções piores que a atual.

Nas Figuras 4.4b, 4.4d, 4.4f observa-se este comportamento quando a função objetivo está limitada a 600. As Figuras 4.4b, 4.4d apresentam comportamento semelhante, contudo a Figura 4.4f apresenta uma primeira solução muito boa, este facto torna-se prejudicial no decorrer do SA, porque dificilmente haverá uma solução melhor que a inicial nas primeiras reduções de temperatura, podendo nunca melhorar até que o critério de paragem seja acionado, o que limita a qualidade das soluções encontradas.

As temperaturas iniciais geradas são muito elevadas, Δ_{avg} é elevado devido à grande probabilidade que cada solução gerada ter vários instantes com sobre-utilização de recursos, e devido ao elevado valor de P utilizado.

CAPÍTULO 4. DESENVOLVIMENTO DOS MODELOS

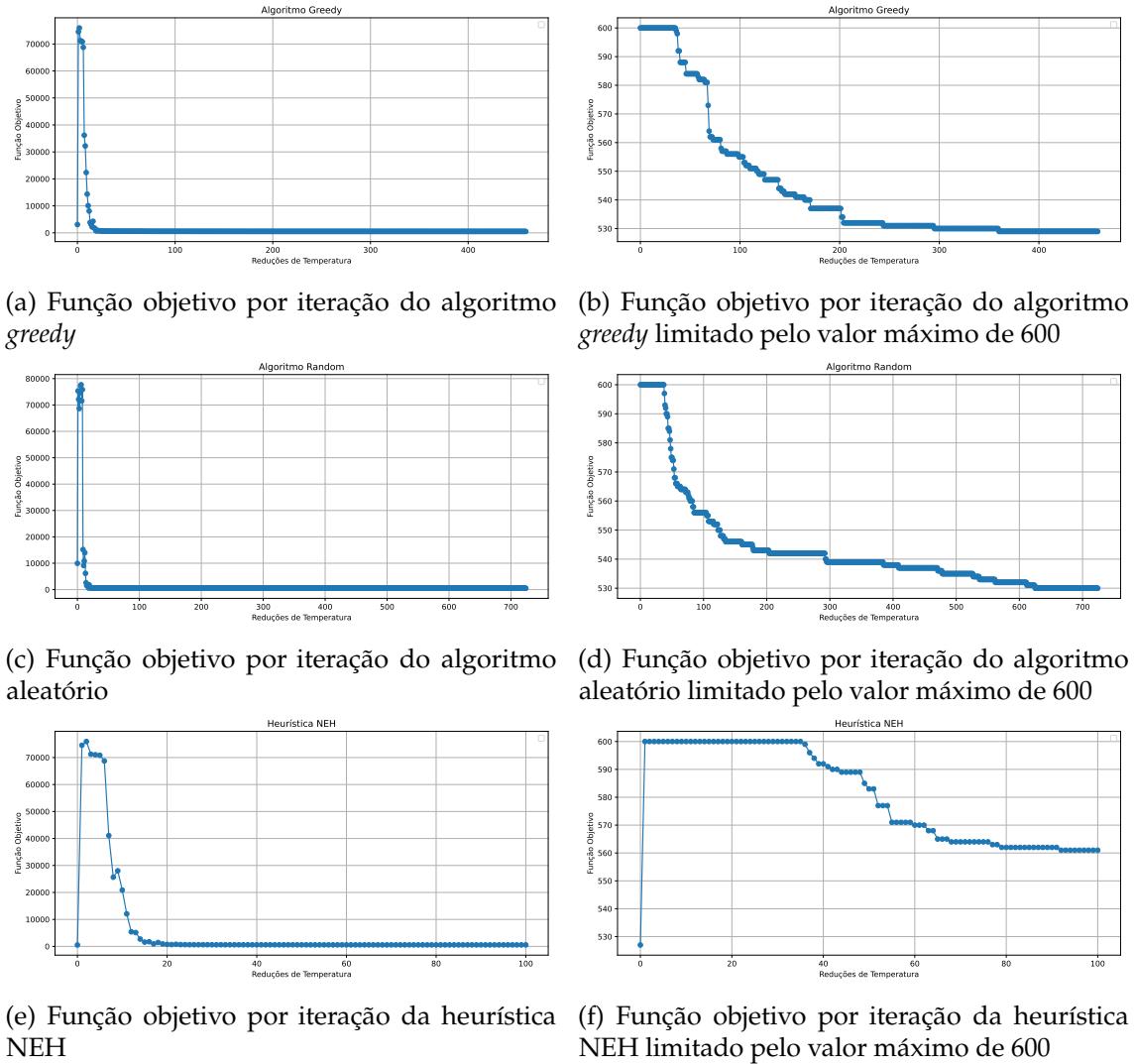


Figura 4.4: Evolução do valor da função objetivo ao longo da otimização do problema de makespan com o Modelo 1 ($L_k = 5000$, $CP = 100$, $\alpha = 0.8$, $P = 1000$, $p_0 = 0.9$).

Utilizar-se-à o algoritmos aleatório para gerar a solução inicial deste modelo.

Será de realçar ainda duas peculiaridades deste modelo. Depois da redução da temperatura, o primeiro vizinho terá de ter o seu instante t gerado utilizando $U(0, T_{\max} - \text{duração})$, sendo T_{\max} o somatório das durações de todos os trabalhos, caso contrário o algoritmo parece não funcionar, isto será algo a estudar no futuro.

Na rejeição de uma solução dever-se-ia recalcular o início e o fim da agenda, de forma a que, na próxima iteração, o novo instante gerado seja gerado com os valores de *ini* e *fin* relativos à solução atual. Contudo isto resulta em piores soluções finais. A Figura 4.5 pretende realçar esta diferença.

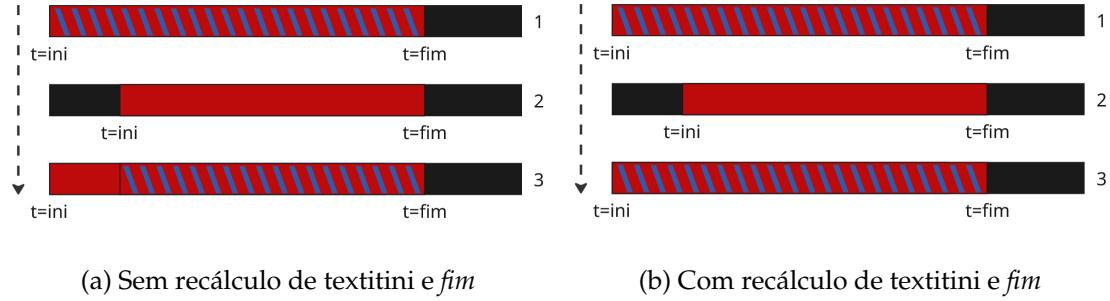


Figura 4.5: Impacto do recálculo de *ini* e *fim* sobre os instantes considerados na próxima iteração.

Nas Figuras 4.5a, 4.5b são apresentadas três agendas. A primeira é a solução atual com os possíveis instantes gerados pela distribuição uniforme a azul. A segunda apresenta a solução depois de mover o trabalho i para o novo instante, admite-se que é rejeitada. A terceira apresenta uma nova solução com os possíveis instantes gerados.

Na Figura 4.5a não se recalcula os instantes de início e de fim, desta forma os possíveis instantes gerados para a nova solução vão ser menos, o que irá dirigir a procura para a redução mais rápida de *makespan*.

Na Figura 4.5b já existe o recálculo de *ini* e de *fim*, ou seja, retrocedemos completamente à solução anterior.

4.1.2 Modelo 2

Este modelo é semelhante ao anterior, contudo existe garantia da viabilidade de cada solução durante a geração da vizinhança. Desta forma, a função objetivo é mais simples, e dada por:

$$f(s) = C_{\max}$$

Sendo C_{\max} o valor de *makespan* calculado como anteriormente, a diferença entre o máximo de término de cada trabalho e o valor mínimo de início de cada trabalho.

A codificação utilizada para este modelo é igual à do anterior. Contudo a forma de gerar a vizinhança, especificamente o novo instante t que um dado trabalho deve começar, difere substancialmente. Como existe garantia da viabilidade da solução, ou seja, todas as restrições são obedecidas, não podemos selecionar qualquer instante t para iniciar o trabalho i . Será preciso por isso um algoritmo que encontre todos os instantes onde se pode inserir o trabalho i sem ocorrer sobre-utilização dos recursos. O Algoritmo 2 descreve este processo.

Verifica-se se para cada instante t se é possível inserir o exame i sem ocorrer sobre-utilização de recursos. Caso não haja, então o instante t é adicionado à lista de candidatos. A partir dessa lista, é selecionado aleatoriamente um instante t que será o momento no qual o trabalho i começa. Foi necessário incluir na lista de candidatos um instante

que se encontra-se fora da solução atual, antes do início com o valor $\max(ini - d, 0)$ ou depois do fim, com o valor $\min(fim + 1, T_{max} - d)$. Caso contrário não se verificava nenhuma exploração de soluções, devido ao facto que os candidatos frequentemente eram apenas o mesmo instante da solução atual, causando uma temperatura inicial próxima de 0.

O processo de modificar a solução s para uma solução s' resume-se à figura 4.6. Encontra-se os candidatos com o Algoritmo 2 e depois escolhe-se aleatoriamente um para o instante de começo. De seguida actualiza-se a utilização dos recursos, decide-se se a nova solução é aceite, se é a melhor já encontrada, ou se eventual é necessário retroceder. O Algoritmo 3 descreve este processo.



Figura 4.6: Processo de transição da solução s para a solução s' para o Modelo 2.

Foram utilizados os mesmo critérios do modelo anterior, descritos em Franzin et al. [38], ou seja, **NE1**, **UT6**, uma modificação de **SC9**, **AC1**, **CS2**, e **TL1**.

Foram considerados novamente as três diferentes soluções iniciais, o algoritmo *greedy*, o algoritmo aleatório, e a heurística NEH.

A Figura 4.7 apresenta o comportamento destas três diferentes soluções iniciais sobre a função objetivo de acordo com o número de iterações. Na Figura 4.7a observa-se que a primeira solução é má, mas não deve apresentar sobre-utilização de recursos devido à forma como é construída, existe depois o arrefecimento e a melhoria da solução. A Figura 4.7c também apresenta uma solução inicial má, mas com sobre-utilização de recursos, com o decorrer do algoritmo observa-se a melhoria esperada da solução. A Figura 4.7e apresenta uma boa solução inicial, que devido à temperatura inicial piora, mas como esperado existe a melhoria da solução.

Nas Figuras 4.7b, 4.7d observa-se um comportamento semelhante da redução da função objetivo, com ambas a terminarem em soluções com qualidade comparável. Na Figura 4.7f o comportamento é semelhante, mas devido à boa solução inicial o algoritmo pode terminar prematuramente.

A temperatura inicial geradas neste modelo é muito inferior ao anterior, devido à inexistência de punição da sobre-utilização de recursos. A maior diferença entre duas solução vizinhas será reagendar um trabalho que se encontra inserido no meio da agenda para um dos extremos, tal que, Δ_{avg} será sempre inferior à duração do maior trabalho.

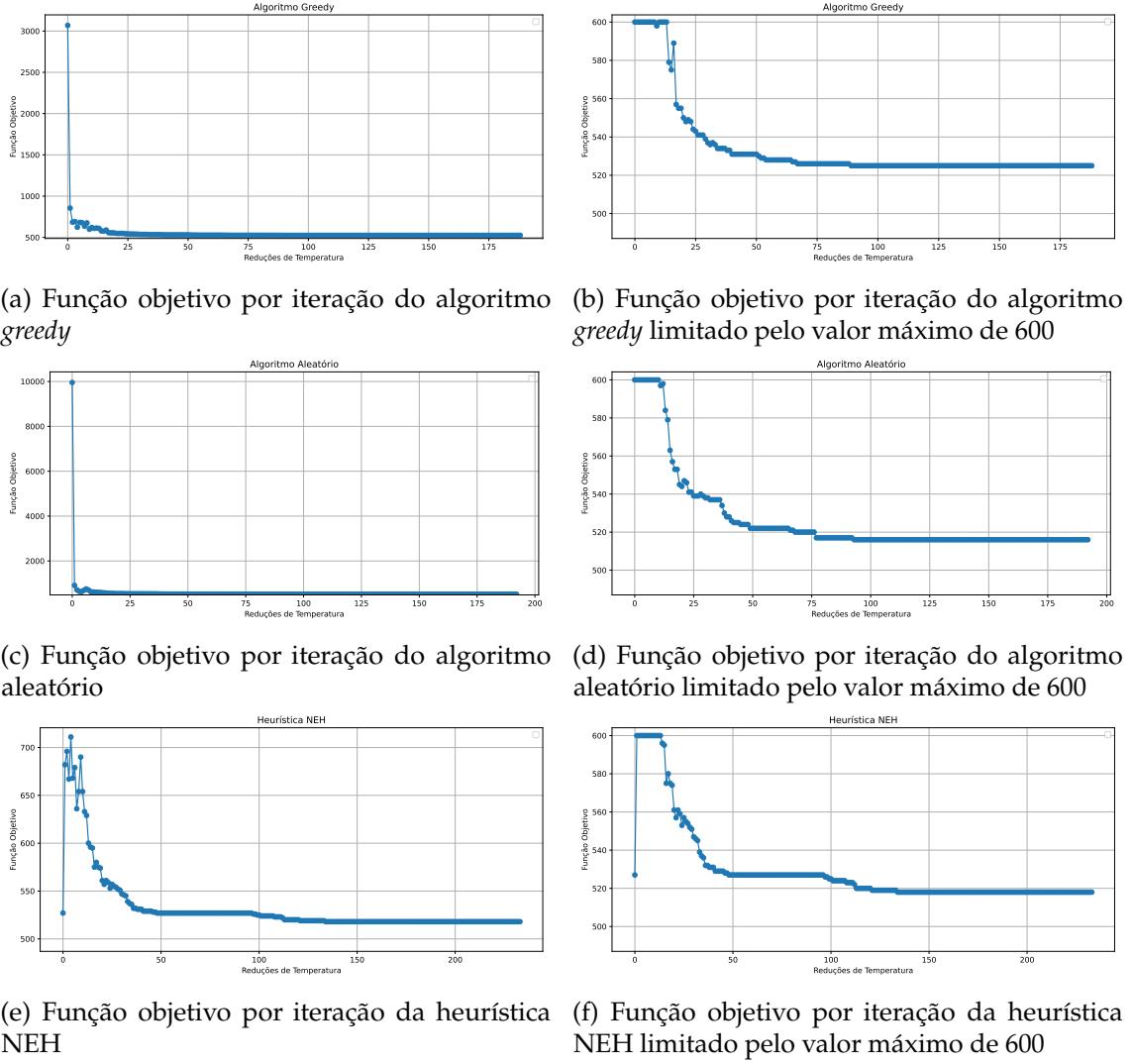


Figura 4.7: Evolução do valor da função objetivo ao longo da otimização do problema de makespan com o Modelo 2 ($L_k = 500$, $CP = 100$, $\alpha = 0.9$, $P = 0$, $p_0 = 0.9$).

De forma a apresentar modelos uniformes entre si, manteve-se o algoritmo aleatório na geração de soluções iniciais.

Este modelo tem uma peculiaridade já referida: se a lista de candidatos contiver apenas um instante entre ini e $fim-d$ que não causam sobre-utilização, acabamos por verificar pouco impacto com a variação de α e p_0 . Este facto é agravado quando, ao rejeitar uma solução, recalcula-se os valores de ini e fim , o que irá causar que poucos instantes além da agenda inicial sejam considerados. Ao não recalcular o início e o fim da agenda existem mais instantes disponíveis para serem selecionados. Este facto é realçado na Figura 4.8.

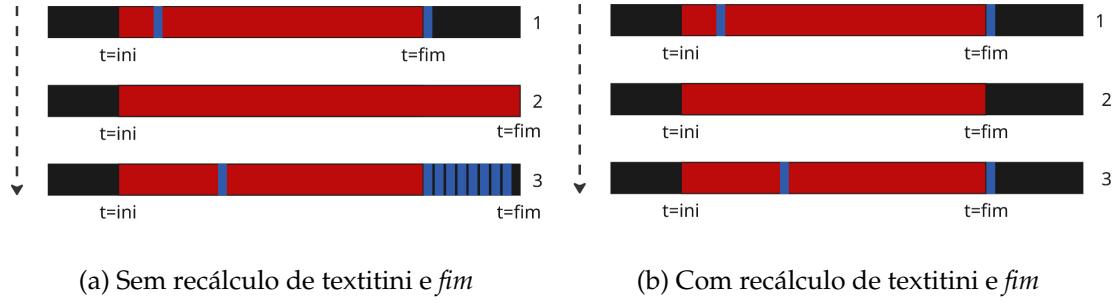


Figura 4.8: Impacto do recálculo de *ini* e *fim* sobre a lista de candidatos da próxima iteração.

Nas Figuras 4.8a, 4.8b são apresentadas três agendas. A primeira é a solução atual com os dois candidatos a novo instante a azul, admite-se que o segundo candidato é escolhido. A segunda apresenta a os valores de *ini* e de *fim* depois de se rejeitar esta agenda. A terceira apresenta os novos candidatos.

Na Figura 4.8a não se recalcula *ini* e de *fim*, existe por isso um maior número de candidatos que, apesar de piorarem a solução, permitem que haja maior exploração da vizinhança. Na Figura 4.8b já se recalcula *ini* de *fim*, o que limita o número de candidatos possíveis e diminui a exploração da vizinhança que ocorre.

4.1.3 Modelo 3

Este modelo continua a garantir a viabilidade de cada solução considerada, por isso a função objetivo não será mais que o valor de *makespan*.

Contudo, a codificação utilizada difere fortemente dos modelos anteriores. Em vez de representar uma solução pelo tempo de começo de cada trabalho, a solução é representada pela sequência pela qual se agenda cada trabalho, iremos representar esta sequência por π .

Esta sequência pouca informação nos fornece por si só, nomeadamente está ausente o instante em que cada trabalho deverá começar. Para tal é necessário um algoritmo auxiliar que descodifica a sequência de forma a obter a agenda, a este algoritmo chamamos de *timetabling*. Existem vários algoritmos utilizados para este fim, contudo apenas dois vão ser descritos com mais profundidade, *left-shift timetabling* e *enhanced left-shift timetabling* [39].

O Algoritmo 4 permite encontrar o instante de tempo t mais cedo onde é possível agendar o trabalho i que se está a agendar. A forma de como isto é feito é algo bruta, pois percorre-se cada instante t e insere-se o trabalho i verificando se existe, ou não, sobre-utilização de recursos, caso não haja retorna-se t . Este algoritmo consome a maioria do tempo computacional deste modelo.

Uma adaptação pode ser utilizada, caso se conheça quais os recursos limitantes, ou seja, aqueles que mais provavelmente originaram sobre-utilização, pode-se primeiro verificar apenas esses recursos, e só depois verificar os restantes. Esta adaptação foi implementada,

utilizando como recursos limitantes SCG e TM.

O Algoritmo 5 encontra, para cada elemento da solução, o instante de tempo mais cedo onde este pode ser inserido, modificando a matriz de utilização de recursos. Na literatura admite-se que assegurar que $t_{\pi[i]} \geq t_{\pi[i-1]}$, ou seja que cada elemento da sequência deve começar no mesmo instante ou depois do elemento anterior, tende a reduzir a qualidade da solução, por isso esta característica não é assegurada.

O Algoritmo 6 é semelhante, para cada elemento da sequência $\pi = \pi[0], \dots, \pi[n - 1]$, encontra o instante mais cedo onde cada elemento pode ser inserido. Ao mesmo tempo faz este processo para a sequência inversa, $\pi = \pi[n - 1], \dots, \pi[0]$, em que cada trabalho também é o inverso, ao inverter δ . Não há a garantia que um ou outro algoritmo é melhor, mas existe a possibilidade de serem geradas soluções melhores com este passo adicional. Em contrapartida, o tempo computacional associado à descodificação da solução dupla.

Para demonstrar a diferença entre estes dois algoritmos irá utilizar-se o exemplo apresentado por Ying e Lin [40], onde é apresentado um pequeno problema *No-Wait Job-Shop*. Para cada um dos quatro trabalhos deste problema, a Tabela 4.1 apresenta a sequência das operações que devem ser realizadas, e o respetivo tempo de processamento. Ou seja, o trabalho I_1 primeiro passa pelo recurso 4 com tempo de processamento de 7 unidades de tempo, a segunda operação será no recurso 3 durante 9 unidades de tempo, a terceira será no recurso 1 durante 8 unidades de tempo, a quarta operação será no recurso 2 durante 4 unidades de tempo.

Tabela 4.1: Problema definido pela operações de cada trabalho e os recursos necessários e tempo de processamento, apresentado por Ying e Lin [40].

Trabalho	Sequência de recursos (tempo)			
I_1	4(7)→	3(9)→	1(8)→	2(4)
I_2	1(4)→	4(8)→	2(2)→	3(4)
I_3	3(5)→	2(7)→	1(1)→	4(4)
I_4	4(3)→	2(1)→	1(2)→	3(3)

O exemplo dado utiliza a sequência 3-4-1-2, ou seja, agenda-se o mais cedo possível o trabalho 3, depois o trabalho 4, de forma consecutiva sem garantir que $t_3 \leq t_4 \leq t_1 \leq t_2$. Pode ver-se, na Figura 4.9 as duas fases do Algoritmo 6, com a agenda *forward* a utilizar a sequência original, e a agenda *backward* com a sequência inversa e com cada trabalho também invertido, é evidente que das duas agendas apresentadas iremos escolher aquela que possua o menor *makespan*, sendo neste caso a agenda *forward*.

CAPÍTULO 4. DESENVOLVIMENTO DOS MODELOS

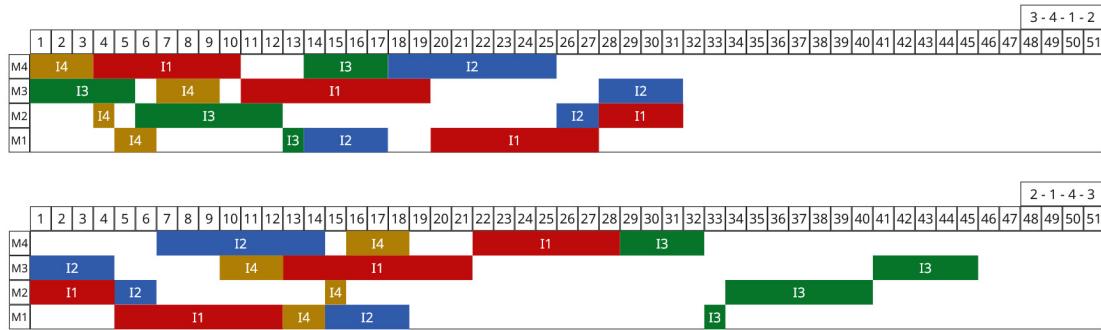


Figura 4.9: Exemplo da solução 3-4-1-2

Novamente, observamos outro exemplo de uma solução na figura 4.10, agora com a sequência 2-3-1-4, com a agenda *forward* e a *backward*, respetivamente. Este exemplo serve para demonstrar a utilidade de representar a mesma solução com mais de um algoritmo de *timetabling*. Podendo transformar uma solução aparente má numa boa solução.

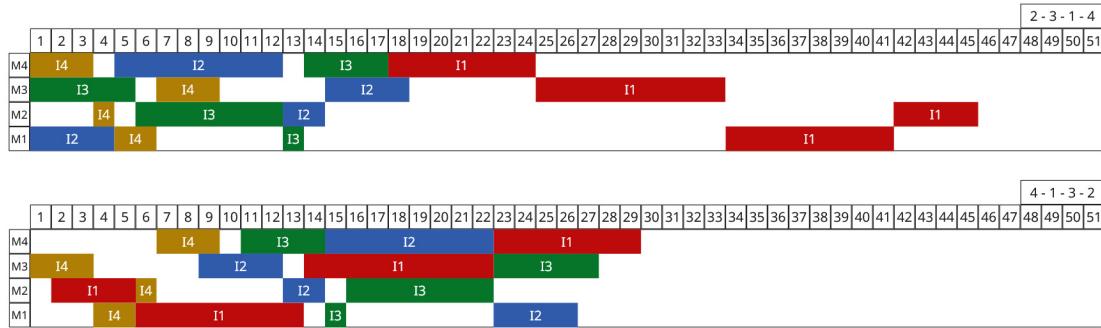


Figura 4.10: Exemplo da solução 2-3-1-4

Por sua vez, a vizinhança de uma solução pode ser gerada de três formas, com várias variantes. Através de uma troca de índice entre trabalhos, Figura 4.11a. Através da retirada de um trabalho e a sua inserção noutro índice da sequência, Figura 4.11b. Através da inversão de uma sub-sequência de trabalhos, Figura 4.11c. Não existe, contudo, nenhuma evidência que um dos métodos é superior ao outro, na continuação iremos utilizar a troca de trabalhos. O Algoritmo 7 apresenta o processo de gerar a vizinhança, de avaliação, rejeição ou aceitação, atribuição da melhor solução, e possível retrocesso.

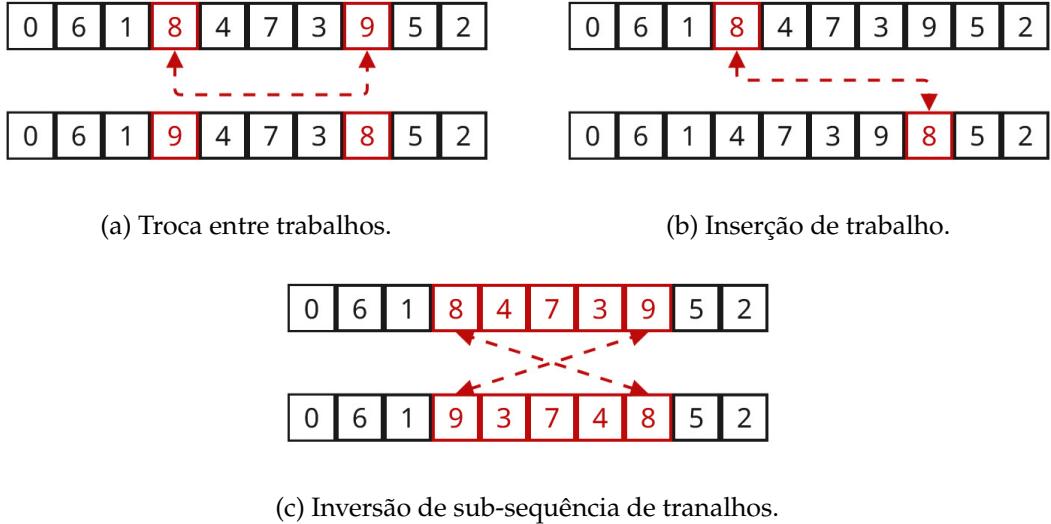


Figura 4.11: Processo de transição da solução s para a solução s' para o Modelo 3.

Como o problema aqui definido contem vários trabalhos do mesmo tipo, ou seja, existem vários exames que são iguais, é essencial verificar que ao trocar trabalhos não cometemos o erro destes serem do mesmo tipo. Caso contrário estaremos a desperdiçar tempo computacional com soluções que são iguais.

Foram utilizados os mesmo critérios do modelo anterior, descritos em Franzin et al. [38], ou seja, **NE1**, **UT6**, uma modificação de **SC9**, **AC1**, **CS2**, e **TL1**.

A geração da solução inicial é diferente dos modelos anterior, devido à codificação diferente. A forma mais simples de gerar uma solução será pela escolha aleatória de uma sequência, existindo sempre garantia da viabilidade da solução graças ao algoritmo de *timetabling*. Também será possível utilizar a heurística de NEH para este fim.

A Figura 4.12 apresenta o comportamento destas duas formas de gerar a solução inicial relativamente à função objetivo com o decorrer das iterações. Nas Figuras 4.12a, 4.12c observa-se um comportamento semelhante entre si, contudo este difere do observado nos modelos anteriores. Continua-se a verificar a tendência de obter soluções más no início do algoritmo que tendem a diminuir com o arrefecimento, contudo não existe a tendência de ocorrer uma diminuição estrita da função objetivo.

O informação retirada das Figuras 4.12b, 4.12d é menos relevante do que nos problemas anteriores devido à menor amplitude na qualidade das soluções observadas.

A temperatura inicial gerada neste modelo é superior à do Modelo 2 mas ainda muito inferior à do Modelo 1. Este facto deve-se ao grande impacto sobre o *makespan* que um trabalho pode ter ao ser colocado no lugar errado da sequência.

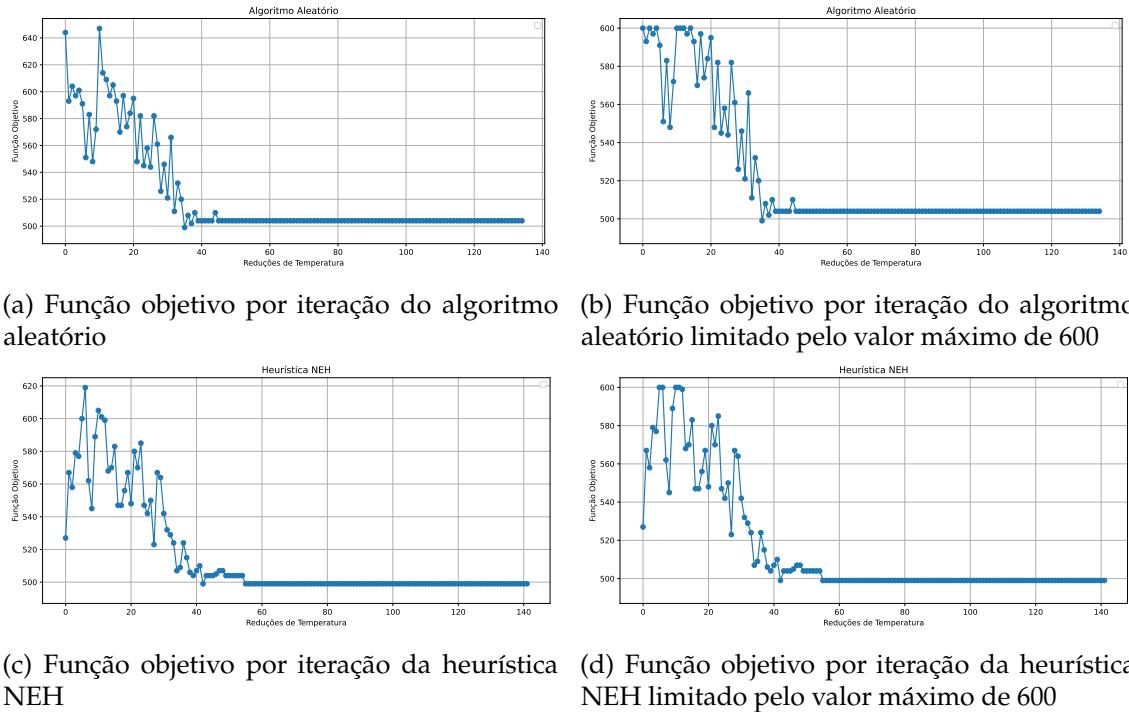


Figura 4.12: Evolução do valor da função objetivo ao longo da otimização do problema de *makespan* com o Modelo 3 ($L_k = 500$, $CP = 100$, $\alpha = 0.9$, $P = 0$, $p_0 = 0.9$, *left-shift timetabling*, geração de vizinhos através de troca).

Este modelo acaba por ser dividido em dois problemas distintos, o problema de sequenciamento e o de *timetabling* já discutido. Sem um bom algoritmo de *timetabling* a melhor solução possível não será a ótima, mas a sequência é o que realmente define a solução.

4.2 Problema do número de trabalhos

Este problema é definido por apresentar um período, ou *makespan*, fixo, com capacidades máximas de recursos pré-definidas, mas com a quantidade de trabalhos a maximizar. Esta otimização poderá ser uma soma dos trabalhos que se inserem no período pré-definido ou utilizar-se um peso para cada trabalho de forma a representar as escolhas do decisor. Este pode ser definido antes de começar o algoritmo ou ser um valor que se adapta ao longo deste.

Por exemplo, se existir a possibilidade de agendar 20 trabalhos do tipo *A*, ou em contrapartida agendar 19 trabalhos do tipo *A* e 1 do tipo *B*, pode ser benéfico assegurar que a segunda opção seja valorizada sobre a primeira. Semelhantemente, o somatório sem consideração de pesos, beneficia a escolha de exames de curta duração sobre exames de longa duração.

A próxima formulação é derivada da *MILP – δ* apresentada na secção anterior:

Conjuntos:

O conjunto de trabalhos $I, i \in I := (1, \dots, n)$

O conjunto de recursos-tipo $R, p \in R := (1, \dots, R_{\max})$

O conjunto de instantes de tempo $T, t \in T := (1, \dots, T_{\max})$

Parâmetros:

ρ_i é duração do trabalho i

$\delta_i(u, p)$ é a quantidade de recursos do tipo p necessários a um offset de u instantes de tempo no trabalho i

C_p é a capacidade do recurso p

Variáveis de Decisão:

Y_i é uma variável binária com valor 1 se o trabalho i estiver agendado, caso contrário tem valor 0

$Z_{t,i}$ é uma variável binária com valor 1 se o trabalho i começar no instante t , caso contrário tem valor 0

Função Objetivo:

$$\min \sum -Y \quad (4.15)$$

Sujeito a:

$$\sum_{t=0}^{T_{\max}-\rho_i+1} Z_{t,i} = Y \quad \forall i \quad (4.16)$$

$$\sum_i \sum_{\tau=\max(0, t-\rho_i+1)}^{\min(t, T_{\max}-\rho_i)} \delta_i(t-\tau, p) Z_{\tau,i} \leq C_p \quad \forall t, p \quad (4.17)$$

A função objetivo (4.15) minimiza o *makespan* dos trabalhos.

A restrição (4.16) garante que só existe um instante de começo do trabalho i quando este exame estiver agendado, caso contrário não existe instante de começo.

A restrição (4.17) garante que não há sobre utilização de recursos, ao verificar para cada instante t quais trabalhos estão a ser executados e em que fase este se encontra, garantindo que em cada instante não há sobre-utilização de recursos.

Iremos minimizar o problema de forma a manter a coerência entre os valores obtidos com esta formulação e os valores obtidos em *SA*. Para a função objetivo consideramos apenas o somatório dos exames sem nenhum peso adicional. Assegura-se que todos os exames agendados se encontram dentro do período através da definição de T_{\max} .

De seguida foram desenvolvidos três modelos análogos ao problema anterior, dois deles com a codificação dos instantes de começo de cada trabalho, e o terceiro modelo com a solução representada pela sequência de escalonamento.

4.2.1 Modelo 1

Para começar, será necessário definir a vizinhança. Ao contrário do problema anterior, onde todos os trabalhos estavam agendados e apenas era necessário definir o instante de começo de cada um. Este problema requer definir quais exames são agendados e em que instante devem começar, para tal existem três movimentos diferentes que podem ocorrer. O mais comum será a definição de um novo instante de começo t para um dado trabalho i já pertencente à agenda, sem assegurar a viabilidade da solução.

Por outro lado, tem de existir alguma probabilidade de remover algum exame i pertencente à agenda, como também adicionar um novo exame, esta adição iremos considerar que ocorre sem garantia de viabilidade.

A função objetivo que pretendemos minimizar é dada por:

$$f(s) = \sum_{t=0}^{T_{\max}-1} \sum_p v(t, p)$$

Onde Y é uma variável binária com os trabalhos agendados, P é a punição relativamente à sobre-utilização de recursos, e com $v(t, p)$ a representar se existe ou não sobre-utilização do recurso p no instante t .

Para a definição do novo vizinho são necessários três componentes: m, i, t . Tal que m represente que movimento iremos realizar, i o trabalho a alterar, e t o instante. Iremos considerar que o movimento 0 seja da mudança do instante de começo t para um trabalho i . O movimento 1 implica a remoção do trabalho i da agenda. O movimento 2 adiciona-se o trabalho i à agenda no instante de começo t .

Para o movimento 0, deve-se ter o cuidado de apenas selecionar um trabalho que já se encontre na agenda atual, por sua vez o instante t pode ser gerado por $U(0, T_{\max}-duração)$, não existindo necessidade de utilizar $U(\text{ini}, \text{fim}-duração)$. Para o movimento 1, também é necessário ter o cuidado de apenas selecionar um trabalho i que se encontre na agenda. Para o movimento 2, o trabalho i a adicionar não deve pertencer à agenda, e o instante de começo t é gerado por $U(0, T_{\max}-duração)$. Esta dinâmica pode ser representada pela figura 4.13. O Algoritmo 8 apresenta como se deve proceder na escolha do movimento e que passos são necessários para gerar a vizinhança, avalia-la, aceita-la ou rejeita-la. O Algoritmo 9 apresenta como se deve fazer o retrocesso para cada movimento.

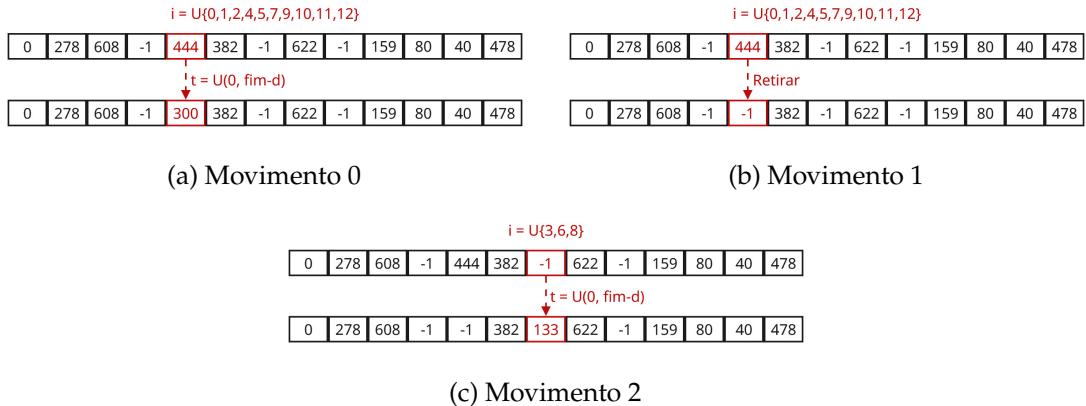


Figura 4.13: Processo de transição da solução s para a solução s' para o Modelo 1.

Cada um destes movimentos ocorrem de forma aleatória com uma probabilidade pré-definida. De forma a diminuir o número de variáveis a testar, consideramos que a probabilidade de ocorrer o movimento 1 e o movimento 2 é de 5% cada.

Relativamente à temperatura inicial, gerou-se uma boa solução através da heurística NEH, de forma a tentar agendar todos os trabalhos dentro do tempo limite de T_{max} , aqueles que de facto terminam antes de T_{max} são contabilizados. Procedesse a uma caminhada aleatória com $c = \infty$ e $L_k = 100000$ para que se aceite todos as soluções visitadas. Finalmente a temperatura inicial é dada por $c_0 = |\Delta_{avg}/\log(p_0)|$.

Para a solução inicial iremos considerar dois algoritmos aleatórios. A Figura 4.14 apresenta a função objetivo ao longo do número de iteração para cada uma das soluções iniciais.

A solução inicial que origina a Figura 4.14a agenda todos os trabalhos de forma aleatória, enquanto a solução inicial que origina a Figura 4.14c agenda um número aleatório de trabalhos. Em ambos os casos observamos um comportamento semelhante ao já descrito. Ocorre o aumento inicial da função objetivo, mesmo que mais acentuado na Figura 4.14a, seguido do declínio com o arrefecimento da temperatura.

As Figuras 4.14b, 4.14d não diferem desta análise. Ocorre um rápido declínio da função objetivo com o arrefecimento da temperatura até que se congele e CP seja ativado, dando o algoritmo como terminado.

Analogamente ao Modelo 1 do problema de *makespan*, a temperatura inicial é bastante alto. Contudo é menor devido às menores diferenças entre soluções vizinhas e ao menor valor de P utilizado. Apesar do algoritmo aleatório alternativo apresentar melhor solução neste caso, de forma a manter a uniformidade utiliza-se o algoritmo aleatório para gerar as soluções iniciais.

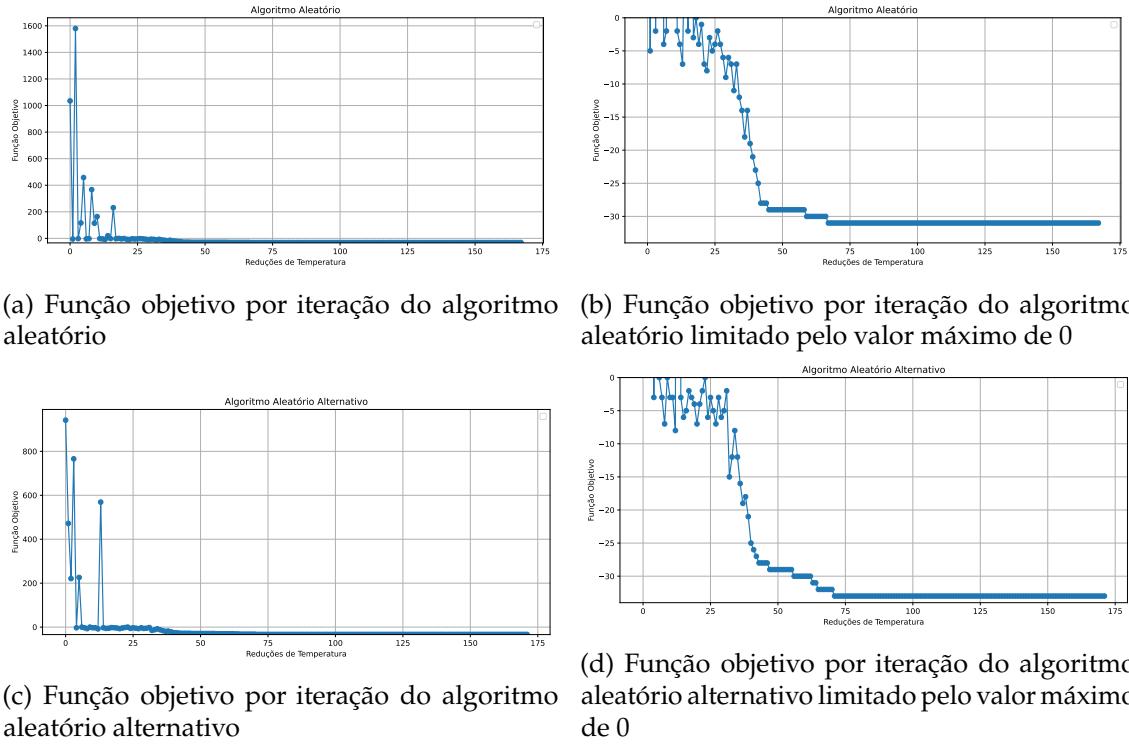


Figura 4.14: Evolução do valor da função objetivo ao longo da otimização do problema do número de trabalhos com o Modelo 1 ($L_k = 5000$, $CP = 100$, $\alpha = 0.8$, $P = 10$, $p_0 = 0.9$).

Continuaremos a utilizar os critérios utilizados no problema anterior, ou seja, **NE1**, **UT6**, uma modificação de **SC9**, **AC1**, **CS2**, e **TL1**.

4.2.2 Modelo 2

Por sua vez, este modelo assegura que todos os movimentos devem assegurar a viabilidade da solução. Desta forma, a função objetivo é diferente dos modelos anteriores, ao admitirmos que a sobre-utilização de recursos não ocorre, será então dada por:

$$f(s) = \sum -Y$$

Durante a geração da vizinhança com o movimento 0 e com o movimento 2 será escolhido de forma aleatória um instante de tempo de uma lista de candidatos que não provocam sobre-utilização de recursos. Este processo é resumido à Figura 4.15. Por sua vez, o Algoritmo 10 apresenta como se deve proceder na escolha do movimento e que passos são necessários para gerar a vizinhança, avalia-la, aceita-la ou rejeita-la. O Algoritmo 11 apresenta como se deve fazer o retrocesso para cada movimento.

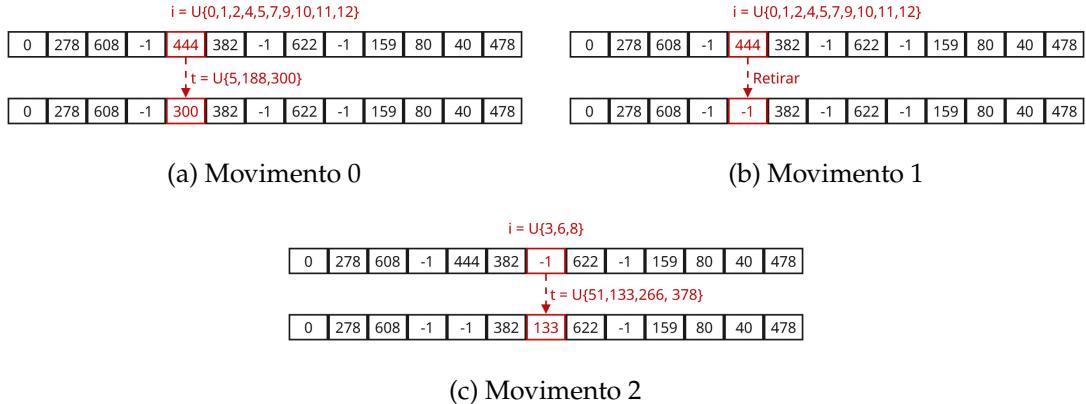


Figura 4.15: Processo de transição da solução s para a solução s' para o modelo 3 do problema 2.

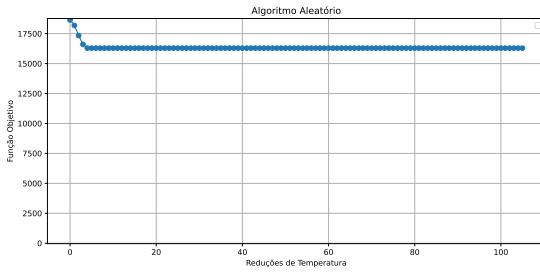
A definição da temperatura inicial tem uma nuance: de forma a inflacionar a temperatura inicial não se utilizou o movimento 0, ou seja, os movimentos 1 e 2 têm 50% de probabilidade de ocorrem, o que irá inflacionar o valor de Δ_{avg} . Se este valor for baixo demais, nunca será considerado a remoção de um trabalho, porque o impacto sobre a função objetivo é muito grande. O modelo anterior não possui este problema devido à alta temperatura inicial, proveniente da grande variabilidade da função objetivo entre soluções.

A geração da solução inicial é bastante diferente em certas condições. Na Figura 4.16 encontram-se três formas de gerar a solução inicial, na Figura 4.16a agendam-se todos os trabalhos com instante de começo aleatório, na Figura 4.16c agenda-se um número aleatório de trabalhos a instantes de começo aleatórios, na Figura 4.16e utiliza-se o algoritmo SA sem a inclusão do movimento 0 para gerar a solução inicial. As figuras não ilustram como se comporta função objetivo, mas sim $f(s) = \sum -Y + 10 \sum_{t=0}^{T_{max}-1} \sum_p v(t, p)$, ou seja, durante o algoritmo não consideramos a sobre-utilização de recursos, mas para o gráfico é necessário reportar este elemento.

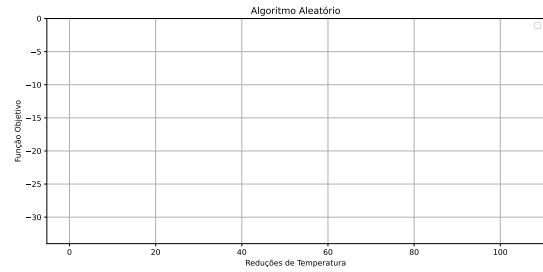
Ao comparar as Figuras 4.16a, 4.16c, 4.16e observa-se dois comportamentos distintos. Enquanto que as Figuras 4.16a, 4.16c apresentam soluções de baixa qualidade que são capazes de superar a sobre-utilização de recursos proveniente da solução inicial, na Figura 4.16e é evidente que a solução inicial não apresenta qualquer sobre-utilização, permitindo que ocorra o processo normal de minimização.

As Figuras 4.16b, 4.16d, 4.16f evidenciam o mesmo facto, apenas o algoritmo aleatório alternativo 2 é capaz de gerar uma solução inicial que origine soluções finais perto do ótimo.

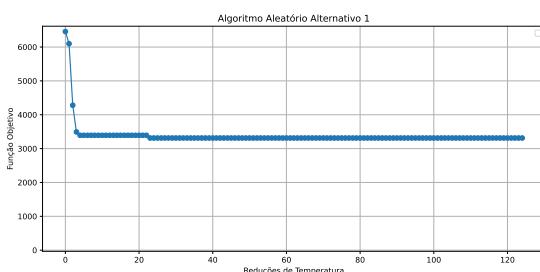
CAPÍTULO 4. DESENVOLVIMENTO DOS MODELOS



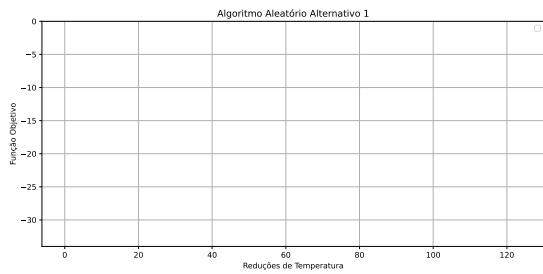
(a) Função objetivo por iteração do algoritmo aleatório



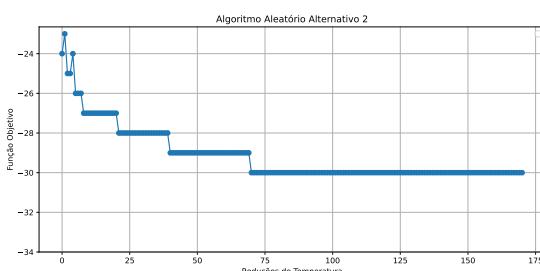
(b) Função objetivo por iteração do algoritmo aleatório limitado pelo valor máximo de 0



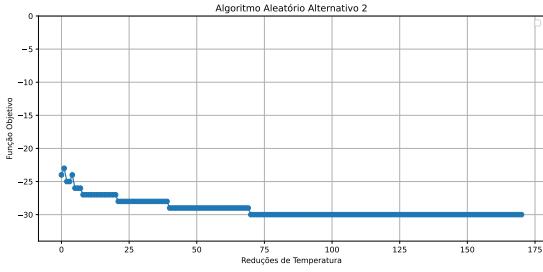
(c) Função objetivo por iteração do algoritmo aleatório alternativo 1



(d) Função objetivo por iteração do algoritmo aleatório alternativo 1 limitado pelo valor máximo de 0



(e) Função objetivo por iteração do algoritmo aleatório alternativo 2



(f) Função objetivo por iteração do algoritmo aleatório alternativo 2 limitado pelo valor máximo de 0

Figura 4.16: Evolução do valor da função objetivo ao longo da otimização do problema do número de trabalhos com o Modelo 2 ($L_k = 100$, $CP = 100$, $\alpha = 0.8$, $P = 0$, $p_0 = 0.5$).

A diferença demonstrada entre os três algoritmos deve-se a uma combinação entre baixa temperatura inicial, um valor baixo de L_k e a assumida inexistência de punição P , se aumentarmos um destes parâmetros observamos que a diferença entre os algoritmo torna-se irrelevante. A baixa temperatura torna difícil a remoção de trabalhos mesmo ao causarem sobre-utilização de recursos, o valor baixo de L_k restringe o número de vizinhos visitados no início da procura, e a inexistência de punição não consegue direcionar o algoritmo de modo a remover a sobre-utilização.

Desta forma irá ser utilizado o algoritmo de geração da solução inicial proveniente da Figura 4.16e, sendo esta a mais robusta, funcionando para qualquer combinação de níveis.

Continuaremos a utilizar os critérios utilizados no problema anterior, ou seja, NE1,

UT6, uma modificação de SC9, AC1, CS2, e TL1.

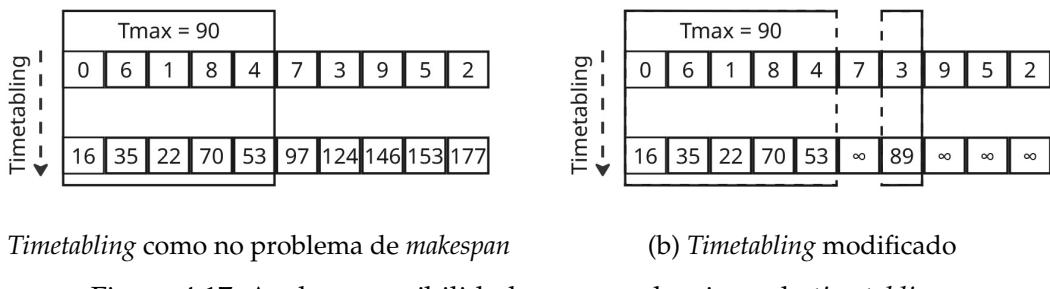
4.2.3 Modelo 3

Este modelo já utiliza uma codificação diferente, utilizando a sequência de agendamento como solução. Ao mesmo tempo, ao contrário dos modelos já descritos, não é necessário existirem vários movimentos, sendo apenas considerado a alteração da sequência. Como já previamente discutido, a alteração da sequência pode ocorrer de várias formas, pela troca de índices, pela inserção de um trabalho noutro índice, ou pela inversão de sub-sequências, mas apenas o primeiro método será utilizado.

Por sua vez, a função objetivo não é diretamente derivada pela solução. Sendo primeiro necessário utilizar um algoritmo de *timetabling* para descodificar a solução. Mesmo assim, a função objetivo é dada por:

$$f(s) = \sum -Y$$

Para a obtenção da variável Y verificamos que trabalhos acabam antes do tempo pré-definido, T_{\max} . Por isso, durante o algoritmo de *timetabling* podemos realizar o agendamento como no problema de *makespan*, ou seja, tentarmos agendar cada trabalho o mais cedo possível. Outra possibilidade será agendar o mais cedo possível cada trabalho, mas agendar o mais tarde possível aqueles trabalhos que acabariam depois de T_{\max} . A Figura 4.17 pretende demonstrar esta diferença.



(a) Timetabling como no problema de *makespan*

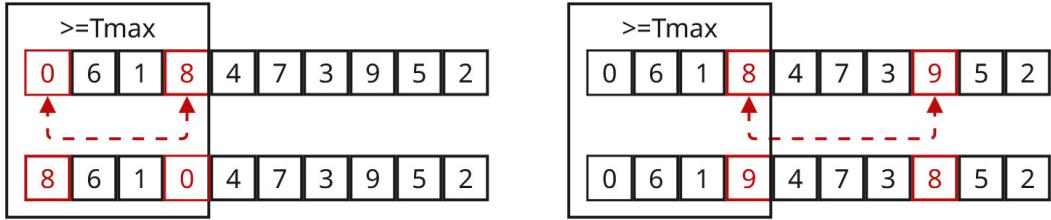
(b) Timetabling modificado

Figura 4.17: As duas possibilidades para o algoritmo de *timetabling*.

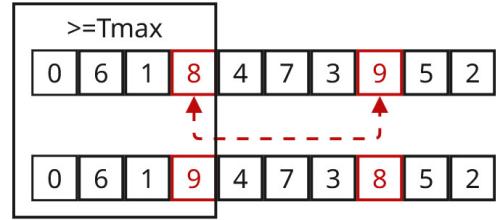
Neste exemplo ilustrativo evidencia-se a diferença entre as duas possibilidades. Na Figura 4.17a, durante o agendamento do trabalho 7 este não permite que o trabalho 3 comece mais cedo. Quando comparado com a Figura 4.17b, ao agendar o trabalho 7 para o instante T_{\max} permite-se agendar o trabalho 3 num instante que vai ser englobado na solução, o que melhora a qualidade da solução com a mesma agenda. O Algoritmo 12 apresenta o pseudo-código que define a diferença.

Como podem existir vários trabalhos iguais, deve-se assegurar que durante a geração da vizinhança não se considera a troca de dois trabalhos do mesmo tipo, de forma a evitar gerar uma nova solução igual à atual. Ao mesmo tempo existem trocas que parecem ser

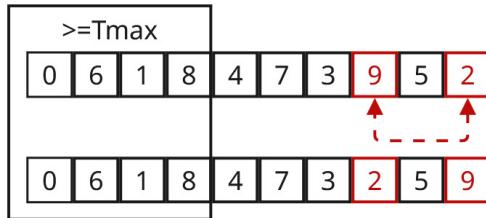
desnecessárias, este facto apresenta-se na figura 4.18.



(a) Troca de dois trabalhos pertencentes à solução



(b) Troca de um trabalho pertencente à solução e de um não pertencente



(c) Troca de dois trabalhos não pertencentes à solução

Figura 4.18: As três trocas que podem ocorrer.

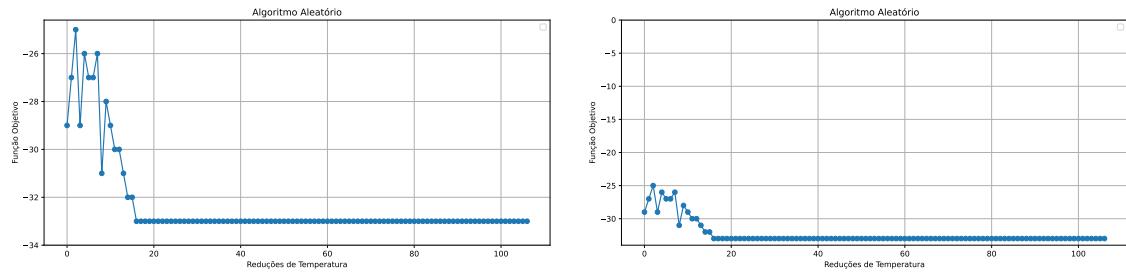
Evidencia-se o porquê de algumas das trocas possíveis não serem interessantes no processo de otimização. Para as trocas representadas pelas Figuras 4.18a, 4.18b prevê-se que a solução seja diferente da a atual, mesmo que o número de trabalhos que acabem antes de T_{\max} não se altere, os instantes de começo para estes exames são diferentes. Por outro lado, a troca apresentada na Figura 4.18c não deve ter impacto sobre os instantes de começo dos trabalhos que acabem antes de T_{\max} . Contudo, não se vai influenciar como ocorrem as trocas representadas pela Figura 4.18, isto porque é computacionalmente dispendioso verificar que trocas entre trabalhos, que acabam depois de T_{\max} , de facto têm influencia sobre a solução. O Algoritmo 13 apresenta o processo de geração da vizinhança, de avaliação, rejeição ou aceitação, atribuição da melhor solução, e possível retrocesso.

No modelo análogo do problema anterior utilizou-se *left-shift timetabling* para descodificar a solução, e o mesmo foi feito aqui. Propôs-se também a utilização de *enhanced left-shift timetabling*, para o problema anterior esta alternativa melhorava a qualidade da solução em troco de maior tempo computacional. Contudo, para este problema parece não existir benefício na utilização de *enhanced left-shift timetabling*.

A temperatura inicial será definida com a geração de uma boa solução pela heurística NEH, de seguida será realizada a caminhada aleatória com $c = \infty$ e $L_k = 1000$ e com a

temperatura inicial calculada por $c_0 = |\Delta_{avg}/\log(p_0)|$. O valor de Δ_{avg} é baixo, contudo isto não causa um problema tão grande como no modelo anterior, porque não ocorre explicitamente a remoção de trabalhos da solução.

Não existem muitas opções para a geração da solução inicial, por isso gerou-se uma sequência aleatória para tal, delimitada na Figura 4.19. Na figura 4.19a, 4.19b observa-se a evolução da função objetivo ao longo do algoritmo, segue o já observado, um aumento da função objetivo seguido da sua diminuição e por fim aciona-se o critério de paragem CP .



(a) Função objetivo por iteração do algoritmo aleatório (b) Função objetivo por iteração do algoritmo aleatório limitado pelo valor máximo de 0

Figura 4.19: Evolução do valor da função objetivo ao longo da otimização do problema do número de trabalhos com o Modelo 3 ($L_k = 500$, $CP = 100$, $\alpha = 0.8$, $P = 10$, $p_0 = 0.9$, *left-shift timetabling*, geração de vizinhos através de troca).

Continuaremos a utilizar os critérios utilizados no problema anterior, ou seja, **NE1**, **UT6**, uma modificação de **SC9**, **AC1**, **CS2**, e **TL1**.

RESULTADOS E DISCUSSÃO

Neste capítulo serão discutidos os resultados obtidos com a implementação dos vários modelos. Serão apresentados, para cada modelo, o impacto que as variáveis têm nas soluções em relação à qualidade e ao tempo computacional utilizado. Também serão utilizadas instâncias de *benchmark* da literatura.

5.1 Problema de *Makespan*

Para os três modelos já apresentados existem cinco variáveis comuns com impacto na qualidade e no tempo computacional de cada solução. Desta forma, será vital encontrar uma combinação de níveis que permite obter soluções boas de forma rápida.

Estas cinco variáveis são: L_k , o número de iterações, ou vizinhos visitados, entre arrefecimentos; CP , o critério de paragem, que representa quando deve ocorrer a paragem do algoritmo; α , a função de arrefecimento; P a punição relativa à sobre-utilização de recursos; p_0 , descrito como a probabilidade inicial de aceitar uma solução vizinha, algo que influencia a temperatura inicial.

Idealmente, utilizar-se-ia $L_k = \infty$, $CP = \infty$, $1 > \alpha \simeq 1$, $p_0 = 1$. Contudo isto implicaria um tempo computacional infinito, por isso irão ser procurados outros níveis para as variáveis.

Os níveis apresentados na Tabela 5.1 foram escolhidos de forma a existir exploração da interação entre os fatores. Por outro lado, a diferença observada nos fatores CP e L_k deve-se ao facto destes serem os principais decisores do tempo computacional, a sua alteração entre modelos permitiu que o tempo computacional permanecesse comparável.

Tabela 5.1: Níveis das variáveis dos modelos de *Simulated Annealing*

Modelo	L_k	CP	α	P	p_0
$M1$	500; 1500; 2500	10; 55; 100	0,8; 0,9; 0,975	10; 55; 100	0,5; 0,7; 0,9
$M2$	100; 550; 1000	10; 55; 100	0,8; 0,9; 0,975	0	0,5; 0,7; 0,9
$M3$	50; 150; 250	5; 30; 55	0,8; 0,9; 0,975	0	0,5; 0,7; 0,9

Para os próximos resultados, serão considerados os exames que decorrem numa segunda-feira habitual. Ou seja, 3 cintigrafia tiroideia, 5 cintigrafia pulmonar de ventilação/inalação + perfusão, 10 cintigrafia miocárdica de perfusão em repouso, 1 cintigrafia das glândulas salivares, 10 PET - estudo corpo inteiro com FDG.

Os resultados obtidos são provenientes de 100 repetições de cada combinação de variáveis. Como foi possível computar paralelamente 10 repetições, os valores apresentados são a média do valor mínimo de 10 repetições. Semelhantemente, o tempo computacional aprestado é relativo à média do valor máximo de 10 repetições.

Todos os valores foram obtidos utilizando um processador Ryzen 5600, os modelos *SA* foram executados em Python 3.11 e os modelos *MILP* em Gurobi 12.0.

5.1.1 Modelo 1

Será agora apresentada um diagrama de calor referente ao valor da função objetivo e tempo computacional. Para a função objetivo, a faixa de cores apresentada encontra-se entre 487 e 527, sendo estes os valores da melhor solução encontrada através do *MILP* e da heurística NEH, respetivamente. O tempo computacional, por sua vez teve a sua faixa de cores limitada entre 0 e 60 segundos.

Existem algumas observações que se podem fazer com a análise da Figura 5.1, em relação ao impacto que cada variável tem.

Observa-se a melhoria dos resultados e o aumento do tempo computacional com o aumento de CP e L_k . Demonstrando novamente que estes são os principais decisores da qualidade e tempo computacional das soluções.

A interação entre valores altos de temperatura inicial, devido a p_0 alto, redução lenta da temperatura, devido a α alto, e valores baixos de CP , provocam muita aceitação inicial de soluções, levando a más soluções que não são melhoradas devido à permanência alta de temperaturas e rápida paragem do algoritmo.

O aumento de CP garante que se encontra uma solução igual ou melhor quando comparado a um nível de CP menor, isto ocorre porque CP não altera o percurso feito pelo algoritmo. O impacto do fator de punição P é difícil de quantificar. Intuitivamente, quando a punição é pequena, irá ocorrer mais exploração devido ao impacto reduzido sobre a função objetivo, por outro lado, quando a punição é grande, a exploração é menor mas há mais certeza que as soluções geradas não apresentarão sobre-utilização de recursos.

Com a ajuda da Tabela A.1 do Apêndice A pode-se tecer alguns comentários sobre os valores ótimos das variáveis primárias, em relação ao *makespan*:

- L_k não apresenta efeito linear ou quadrático significativo, o nível a escolher será aquele mais conveniente;

- CP apresenta efeito linear e quadrático significativo, ambos negativos, o nível a escolher será o alto;
- α apresenta efeitos lineares e quadráticos significativos, positivo e negativo, respetivamente, o nível a escolher será o baixo;
- P apresenta efeito linear e quadrático significativo, ambos negativos, o nível a escolher será o alto;
- p_0 apresenta efeito linear significativo positivo, o nível a escolher será o baixo.

Relativamente ao tempo, pode-se fazer uma análise semelhante pela Tabela A.2:

- L_k apresenta efeito linear e quadrático significativos, ambos positivos, tornando difícil decidir que nível escolher, contudo intuitivamente sabe-se que reduzir L_k deve reduzir o tempo computacional;
- CP apresenta efeito linear e quadrático significativo, ambos positivos, a escolher torna-se difícil, mas intuitivamente deve-se escolher o nível baixo;
- α apresenta efeito linear positivo, o nível a escolher será o baixo;
- P apresenta efeito linear e quadrático significativo, ambos positivos, o que torna a escolha de nível difícil;
- p_0 não apresenta efeito linear ou quadrático significativo, o nível a escolher será aquele mais conveniente.

De forma a obter uma boa combinação de variáveis, optou-se por definir uma função com ambas as variáveis dependentes a minimizar. A função é dada por:

$$makespan^{10} \times \text{tempo}^{0,1}$$

Esta fórmula foi escolhida porque pretendemos minimizar ambas as parcelas, *makespan* e o tempo, mas também pretendemos dar mais importância a pequenas diferenças entre valores de *makespan*. Desta forma, a combinação que minimiza esta fórmula é $\{1500; 100; 0,975; 10; 0,5\}$ com *makespan* médio de 510,2 e tempo computacional médio de 18,1 segundos.

5.1.2 Modelo 2

Será apresentado um novo diagrama de calor referente ao *makespan* e ao correspondente tempo computacional obtido através do Modelo 2 utilizando as várias combinações de níveis apresentadas na Tabela 5.1. Outra vez, a faixa de cores estarão limitas aos valores já referidos. Com a análise da Figura 5.2 podemos tirar alguma conclusões acerca do impacto

5.1. PROBLEMA DE MAKESPAN

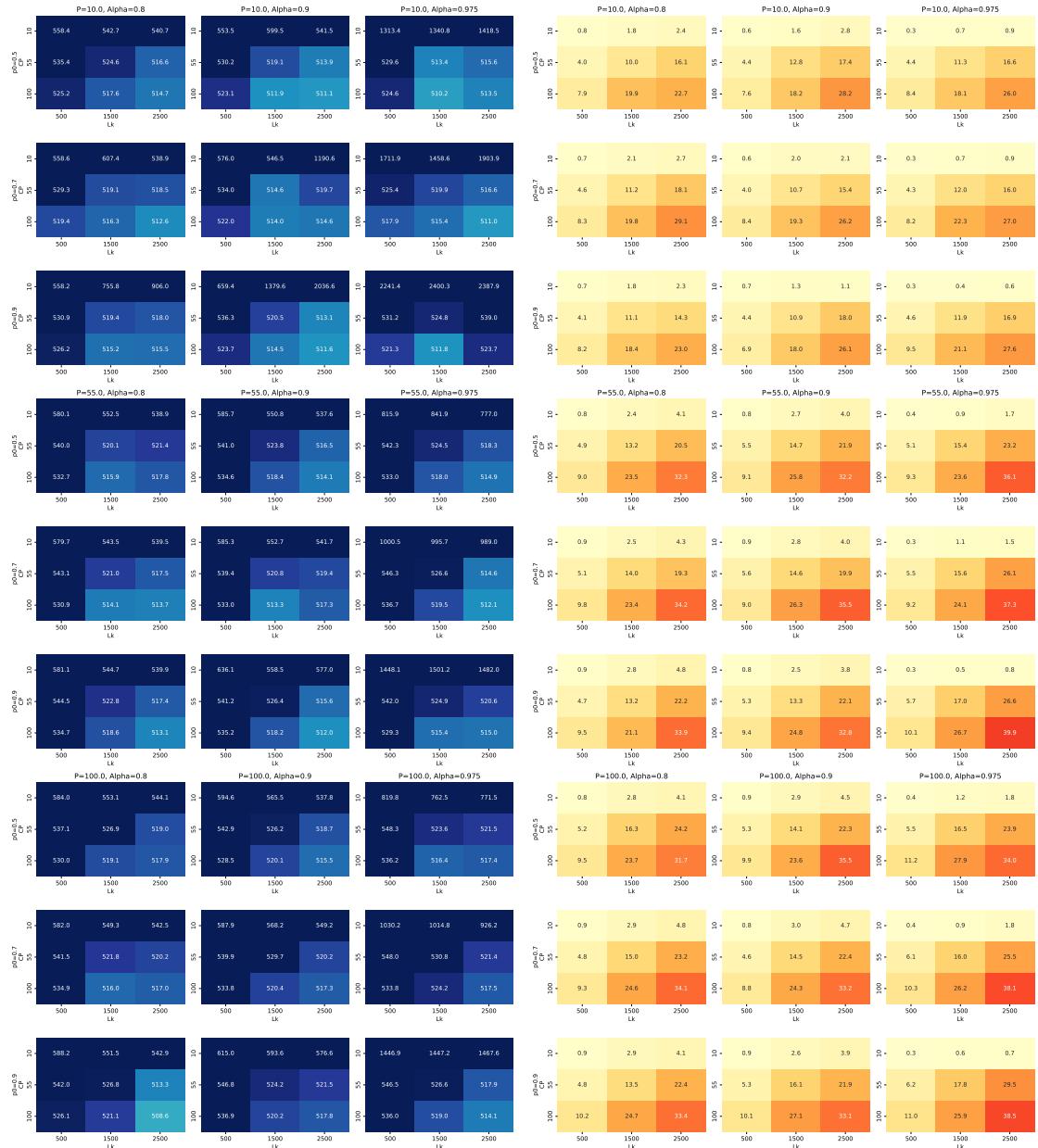


Figura 5.1: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 1.

que cada variável tem sobre a qualidade da solução e o tempo computacional necessário.

Observa-se de imediato que as soluções apresentadas são melhores do que no Modelo 1, sem aumentos desproporcionais do tempo computacional, este facto é especialmente evidente por não existirem combinações de variáveis muito superiores a 527 como foi observado no Modelo 1.

Observa-se novamente o declínio da qualidade das soluções quando CP é baixo, e p_0 e α são altos. Semelhantemente observa-se que ao aumentar CP garantidamente a solução

CAPÍTULO 5. RESULTADOS E DISCUSSÃO

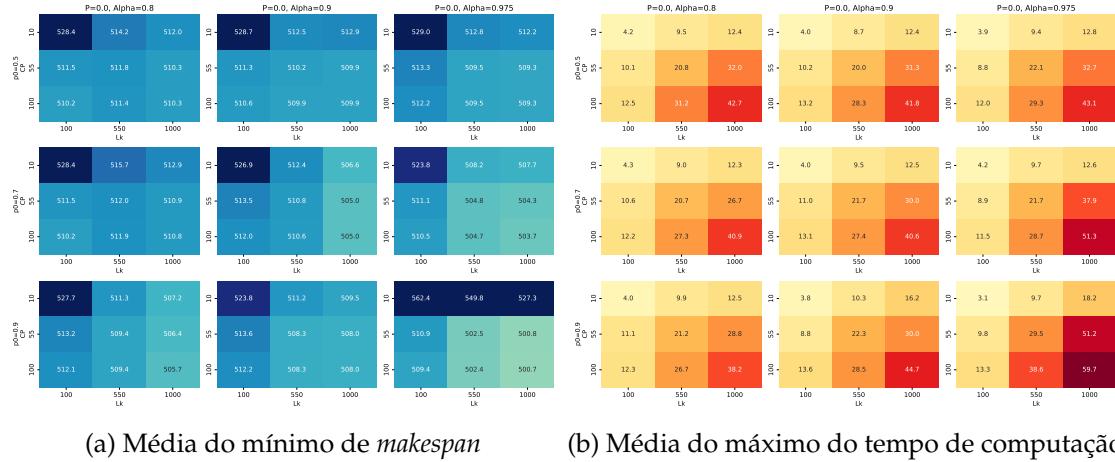


Figura 5.2: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 2.

mantém-se ou melhora, o mesmo não se verifica para L_k contudo é razoável admitir que existe a tendência para a solução melhorar.

Com a ajuda da Tabela A.3 podemos comentar como se comportam as variáveis dependentes de acordo com as variáveis independentes. Relativamente ao *makespan* temos que:

- L_k apresenta efeito linear significativo negativo, o nível a escolher será o alto;
- CP apresenta efeito linear e quadrático significativo, ambos negativos, o nível a escolher será o alto;
- α não apresenta efeito linear ou quadrático significativo, o nível a escolher será aquele mais conveniente;
- p_0 não apresenta efeito linear ou quadrático significativo, o nível a escolher será aquele mais conveniente.

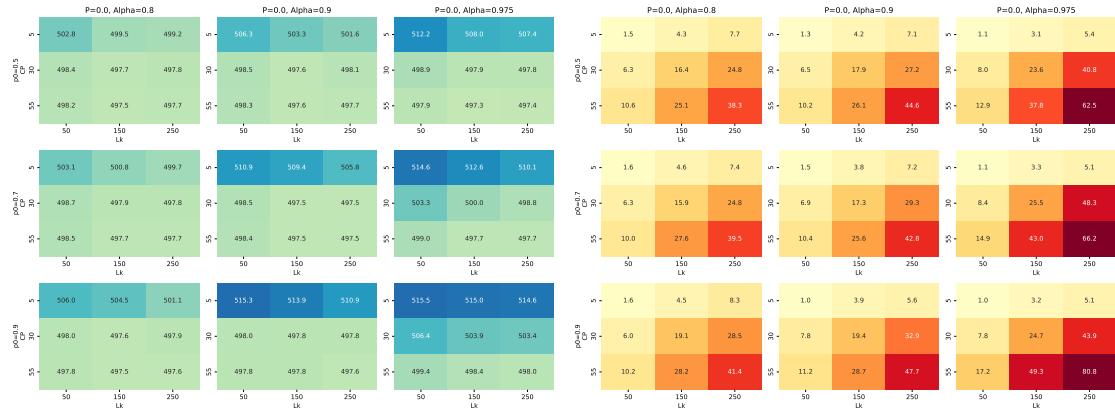
Relativamente ao tempo, poderemos fazer uma análise semelhante pela Tabela A.4:

- L_k apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- CP apresenta efeito linear e quadrático significativo, ambos positivos, a escolher torna-se difícil, mas intuitivamente devemos escolher o nível baixo;
- α apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- p_0 apresenta efeito linear significativo positivo, o nível a escolher será o baixo.

Desta forma, a combinação que minimiza a fórmula já descrita é $\{1000; 10; 0,9; 0; 0,7\}$, com *makespan* médio de 506,6 e tempo computacional médio de 12,5 segundos.

5.1.3 Modelo 3

Será outra vez apresentado um diagrama de calor referente ao *makespan* e ao tempo computacional obtido pelo Modelo 3, com os níveis descritos na Tabela 5.1. A faixa de cores continuam a ser limitadas como de antes. A Figura 5.3 permite realizar uma análise superficial do impacto das variáveis sobre a qualidade das soluções e o tempo computacional correspondente.


 (a) Média do mínimo de *makespan*

(b) Média do máximo do tempo de computação

Figura 5.3: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com *left shifting*.

Na Figura 5.3 observamos de imediato que o modelo 3 é melhor que os anteriores, apresentando soluções melhor com tempo computacional semelhante. Voltamos a retirar as mesmas observações, aumentar CP garantidamente mantém ou melhora a solução, quando CP é baixo, e p_0 e α são altos observamos soluções de pior qualidade. Podemos analisar qual o efeito primário de cada variável sobre o *makespan* utilizando a Tabela A.5:

- L_k apresenta efeito linear significativo negativo, o nível a escolher será o alto;
- CP apresenta efeito linear e quadrático significativo, ambos negativos, o nível a escolher será o alto;
- α apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- p_0 apresenta efeito linear significativo positivo, o nível a escolher será o baixo.

Para a análise sobre o tempo computacional utilizamos a Tabela A.6:

- L_k apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- CP apresenta efeito linear e quadrático significativo, ambos positivos, a escolher torna-se difícil, mas intuitivamente devemos escolher o nível baixo;

CAPÍTULO 5. RESULTADOS E DISCUSSÃO

- α apresenta efeito linear e quadrático significativo, positivo e negativo, respetivamente, o nível a escolher será o baixo;
- p_0 apresenta efeito linear significativo positivo, o nível a escolher será o baixo.

Desta forma, a combinação que minimiza a fórmula já descrita é {50; 5; 0, 8; 0; 0, 5}, com *makespan* médio de 502,8 e tempo computacional médio de 1,5 segundos.

5.1.4 Comparação entre modelos

De seguida são apresentados os exames realizados em cada dia da semana no departamento de medicina nuclear em estudo. Para cada um destes dias, serão utilizados os modelos já descritos para encontrar uma solução que permita minimizar o *makespan*, cada modelo utilizará a combinação de níveis já descritas. Além disso foram introduzidas dois algoritmos de *timetabling* existentes na literatura, *non-delay* (ND) [41] e *enhanced non-delay* (END) [42]. Os valores obtidos na resolução destes problemas encontram-se na Tabela 5.2.

- Segunda-feira: 3 cintigrafia tiroideia, 5 cintigrafia pulmonar de ventilação/inalação + perfusão, 10 cintigrafia miocárdica de perfusão em repouso, 1 cintigrafia das glândulas salivares, 10 PET - estudo corpo inteiro com FDG;
- Terça-feira: 2 cintigrafia tiroideia, 10 cintigrafia óssea corpo inteiro, 1 cintigrafia para amiloidose cardíaca, 1 linfoцитografia para detecção de gânglio sentinelas, 2 cintigrafia pulmonar de ventilação/inalação + perfusão, 10 PET - estudo corpo inteiro com FDG;
- Quarta-feira: 2 cintigrafia tiroideia, 2 linfoцитografia para detecção de gânglio sentinelas, 6 cintigrafia das paratiroideias, 1 tomografia cerebral com 123I-ioflupano, 3 cintigrafia pulmonar de ventilação/inalação + perfusão, 8 PET - estudo corpo inteiro com FDG;
- Quinta-feira: 6 linfoцитografia para detecção de gânglio sentinelas, 12 cintigrafia miocárdica de perfusão em esforço/stress farmacológico, 1 tomografia cerebral com 123I-ioflupano, 2 cintigrafia miocárdica de perfusão em repouso, 10 PET - estudo corpo inteiro com FDG.

Tabela 5.2: Resultados provenientes dos três modelos em relação aos problemas de cada dia da semana.

Dia	BKS	MILP-trad			MILP- δ			Modelo 1				Modelo 2				NEH	
		BRPD	BBRPD	TCT	BRPD	BBRPD	TCT	BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT	BRPD	TCT
Segunda-Feira	487	-	-34,91	28800	0,00	-1,64	28800	2,87	4,76	104,3	18,1	2,26	4,02	94,3	12,5	8,21	0,17
Terça-Feira	441	-	-22,68	28800	0,00	0,00	73	0,00	12,15	84,9	14,1	0,45	0,98	130,8	26,0	4,54	0,16
Quarta-Feira	576	-	-27,08	28800	0,00	0,00	19547	11,28	138,57	50,0	8,1	4,17	4,83	105,8	13,4	10,24	0,06
Quinta-Feira	584	-	-37,67	28800	0,00	-4,56	28800	3,08	4,25	127,3	20,0	2,57	4,20	138,3	18,0	10,79	0,32
Média		-	-30,59	28800	0,00	-1,55	19305	4,31	39,93	91,6	15,1	2,36	3,51	117,3	17,5	8,44	0,18

Dia	BKS	Modelo 3 LS				Modelo 3 ELS				Modelo 3 ND				Modelo 3 END			
		BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT
Segunda-Feira	487	2,46	3,23	11,2	1,5	1,64	2,57	20,1	3,0	10,88	13,10	5,6	0,7	4,72	6,51	11,5	1,7
Terça-Feira	441	0,00	0,77	9,9	1,2	0,00	0,79	13,7	1,8	1,81	4,10	4,6	0,7	1,36	2,81	11,7	2,0
Quarta-Feira	576	3,12	3,59	6,2	0,8	0,00	1,00	8,9	1,4	5,56	7,01	3,0	0,4	1,74	2,62	7,5	1,1
Quinta-Feira	584	2,57	3,84	23,2	3,8	2,23	4,37	33,1	5,3	7,53	9,49	5,6	0,9	4,28	6,03	13,9	2,2
Média		2,04	2,86	12,6	1,8	0,97	2,18	19,0	2,9	6,45	8,43	4,7	0,7	3,02	4,49	11,2	1,8

Na Tabela 5.2 observa-se a melhor solução encontrada (*BKS*), tipicamente proveniente do modelo *MILP* – δ , o desvio relativo percentual da melhor solução (*BRPD*), o desvio relativo percentual do melhor *bound* encontrado (*BBRPD*), o desvio relativo percentual médio (*ARPД*), o tempo total de computação (*TCT*), e o tempo médio de computação (*ACT*). Os valores do desvio percentual foram obtidos através da fórmula:

$$RPD_h = \frac{C_{\max}^h - C_{\max}^B}{C_{\max}^B}$$

Tal que C_{\max}^h e C_{\max}^B são os valores de *makespan* obtidos pelo modelo h e *BKS*. Para a obtenção das soluções, cada modelo foi executado 100 vezes, para *BRPD* foi reportado o melhor valor encontrado e para *ARPД* utilizou-se a média do mínimo de 10 execuções. Por sua vez, *TCT* foi o tempo necessário para executar o modelo 100 vezes e para *ACT* utilizou-se a média do máximo de 10 execuções.

Ao comparar *MILP-trad* e *MILP- δ* observa-se a diferença entre as soluções encontradas, enquanto *MILP-trad* não é capaz de chegar a uma solução admissível nas oito horas previstas, *MILP- δ* origina soluções ótimas ou perto de tal devido ao valor próximo de 0 de *BBRPD*.

Apesar da formulação *MILP- δ* apresentar as melhores soluções, o tempo computacional correspondente dificulta a sua utilização na prática. Também será difícil justificar a utilização do Modelo 1 ou 2, sendo que estes, em média, apresentam soluções piores com maior tempo computacional. Por sua vez, será difícil dizer que Modelo 3 com *left-shifting*, *enhanced left-shifting*, *non-delay*, ou *enhanced non-delay* apresentam soluções ou tempos estatisticamente diferentes, mas um teste estatístico com esta amostra reduzida não é viável. A heurística NEH, considerando o baixo tempo computacional utilizado, torna-se numa escolha atraente se a obtenção de uma solução razoável e rápida seja necessário.

Os valores referentes à melhor solução e ao tempo computacional total de cada modelo com as várias combinações de níveis estão presentes no Apêndice (B), bem como o Modelo 3 utilizando *enhanced left shifting timetabling*, *non-delay timetabling*, e *enhanced non-delay timetabling*.

5.1.5 Comparação com instâncias de *benchmark*

Nesta subsecção serão utilizados os modelos descritos anteriormente, utilizando os níveis identificados, na resolução de instâncias *benchmark*. Para tal foram utilizados 21 instâncias de pequenas dimensões, com entre 6 e 10 trabalhos, e com entre 6 e 10 operações. Estas instâncias foram escolhidas devido à comparação já existente em Sundar et al. [43], onde também encontramos o valor da melhor solução de cada instância.

Apesar destes modelos não terem sido formulados para resolver as instâncias aqui apresentadas, sendo estes problemas apenas *No-Wait Job-Shop*, é possível observar que de qualquer forma apresentam boas soluções com tempo computacional razoável. Com a análise das Tabelas 5.4, 5.5, 5.6, 5.7 é possível retirar algumas conclusões sobre a superioridade de alguns modelos. Realizaram-se quatro testes de Mann–Whitney relativamente às quatro variáveis em estudo: *BRPD*, *ARPD*, *TCT*, *ACT*. Este teste pretende comparar a distribuição de duas populações para a determinar se possuem a mesma distribuição. Desta forma, o teste relativo a dois modelos for estatisticamente significante e a média de um for maior que a de outro, pode-se dizer qual destes é superior.

Mesmo assim, os modelos aqui descritos são inferiores aos apresentados em Sundar et al. [43] e em Ying & Lin [40]. O mesmo se verifica em relação ao tempo computacional, contudo esta comparação será menos relevante devido às diferentes linguagens de programação utilizadas e diferentes especificações de computadores.

Tabela 5.3: Resultados provenientes dos três modelos em relação às instâncias apresentadas em Sundar et al. [43].

Instância	BKS	Modelo 1				Modelo 2				Modelo 3 LS				Modelo 3 ELS			
		BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT
Ft06	73	0,00	0,00	27,7	3,1	0,00	0,00	6,9	0,9	0,00	0,00	0,4	0,0	0,00	0,00	0,7	0,1
La01	971	0,00	9,58	62,5	9,7	0,00	5,26	58,8	7,6	0,41	4,00	6,4	0,8	0,00	3,01	15,9	2,3
La02	937	3,31	13,24	74,6	14,4	2,13	7,72	46,4	6,9	3,09	4,51	4,1	0,8	0,00	3,02	12,9	1,9
La03	820	6,10	14,28	59,5	11,5	2,56	7,29	40,5	6,0	5,12	6,05	3,3	0,4	0,00	3,37	10,2	1,4
La04	887	4,74	22,00	53,8	10,2	0,22	4,95	41,4	6,0	0,11	1,51	3,7	0,5	0,00	0,47	11,7	1,7
La05	777	0,90	12,59	51,2	7,9	3,73	8,11	34,5	5,0	0,90	2,32	2,9	0,4	0,00	0,21	8,9	1,4
Ft10	1607	12,44	29,56	150,4	25,3	1,62	4,84	256,8	41,9	0,00	4,92	21,0	3,0	0,00	0,45	61,4	9,9
Orb01	1615	4,27	20,18	193,5	39,4	0,00	3,22	239,8	35,7	0,00	1,94	23,8	3,2	0,00	1,21	70,9	9,7
Orb02	1485	6,13	30,42	117,6	26,0	2,22	4,09	205,2	29,2	2,22	2,22	20,2	3,0	2,22	2,22	58,9	8,4
Orb03	1599	2,31	7,81	206,1	35,3	0,25	3,00	273,4	39,5	0,00	1,36	22,5	3,3	0,00	0,43	62,4	9,4
Orb04	1653	12,16	28,68	135,9	28,0	4,42	6,64	289,2	44,6	4,46	5,89	26,7	4,3	0,00	2,27	71,4	10,5
Orb05	1365	8,06	15,52	110,0	19,9	6,15	8,81	236,5	37,3	0,44	3,31	16,2	2,5	0,15	1,60	49,9	7,8
Orb06	1555	0,00	10,15	153,4	24,1	0,00	2,24	294,7	42,6	0,00	0,06	25,2	3,5	0,00	0,10	76,6	11,1
Orb08	1319	2,35	29,00	146,8	27,4	0,00	2,52	186,4	30,0	0,00	0,00	16,6	2,2	0,00	0,39	47,2	6,9
Orb09	1445	6,02	33,88	119,8	29,5	0,62	5,97	225,2	33,6	4,08	7,31	20,7	3,0	0,00	1,68	59,2	8,4
Orb10	1557	1,73	20,48	111,2	18,5	1,48	4,01	292,4	46,2	0,00	2,58	25,3	3,7	0,00	1,75	66,7	10,5
La16	1575	4,51	13,74	134,0	25,2	0,00	3,59	283,8	44,2	1,17	4,72	22,2	3,1	0,00	0,78	69,6	10,3
La17	1371	10,72	31,82	94,9	14,5	1,97	5,92	170,0	23,5	4,30	6,16	18,4	2,9	0,00	3,53	53,7	7,8
La18	1417	9,81	25,79	122,4	27,0	6,35	9,56	240,8	35,7	6,35	6,63	20,6	3,0	0,00	5,95	62,0	9,2
La19	1482	1,48	20,29	112,7	19,2	1,28	6,13	228,1	34,2	4,45	6,39	23,2	3,2	0,61	2,99	64,4	10,0
La20	1526	10,62	51,03	132,1	24,1	5,37	8,29	284,5	43,5	0,00	1,80	24,0	3,5	0,98	4,92	72,1	10,3
Média		5,13	20,95	112,9	21,0	1,92	5,34	187,4	28,3	1,77	3,50	16,5	2,4	0,19	1,92	47,9	7,1

Tabela 5.3: Resultados provenientes dos três modelos em relação às instâncias apresentadas em Sundar et al. [43] (continuação).

Instância	BKS	Modelo 3 ND				Modelo 3 END				NEH	
		BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT	BRPD	TCT
Ft06	73	0,00	0,00	0,4	0,1	0,00	0,00	0,8	0,1	0,00	0,0
La01	971	7,42	10,24	4,0	0,4	6,80	10,05	8,4	1,0	31,10	0,0
La02	937	7,15	10,28	1,7	0,2	0,00	7,03	5,7	0,8	21,02	0,0
La03	820	5,61	8,59	1,6	0,2	4,88	7,83	4,7	0,7	23,90	0,0
La04	887	0,23	6,60	1,6	0,2	1,91	6,46	4,6	0,6	5,98	0,0
La05	777	5,15	9,94	1,5	0,2	5,92	10,10	3,5	0,5	10,81	0,0
Ft10	1607	2,24	3,71	7,8	1,1	1,62	4,01	20,3	3,1	4,73	0,1
Orb01	1615	4,64	5,99	7,4	1,0	2,72	4,77	23,6	3,3	11,95	0,1
Orb02	1485	4,71	8,61	7,0	1,0	2,22	7,16	21,2	3,1	2,66	0,1
Orb03	1599	1,13	2,46	6,7	0,9	0,25	2,51	21,8	2,9	17,01	0,1
Orb04	1653	7,50	9,87	8,9	1,2	0,00	7,12	23,9	3,4	21,34	0,1
Orb05	1365	4,18	8,88	6,0	0,9	5,71	7,76	17,0	2,5	18,02	0,1
Orb06	1555	0,00	3,39	8,9	1,3	0,00	4,26	24,7	3,4	22,96	0,1
Orb08	1319	0,00	2,47	6,1	0,8	0,00	1,08	17,6	2,4	7,66	0,0
Orb09	1445	6,23	11,67	7,8	1,1	6,23	11,35	21,0	3,1	24,57	0,1
Orb10	1557	3,92	9,50	7,8	1,2	3,92	8,44	21,5	3,0	13,04	0,1
La16	1575	4,70	7,85	8,1	1,2	3,94	7,78	23,7	3,4	6,73	0,1
La17	1371	5,69	9,87	6,1	0,9	1,31	7,78	16,1	2,2	17,36	0,1
La18	1417	9,74	13,39	8,2	1,1	9,74	12,38	20,3	3,0	19,97	0,1
La19	1482	10,32	13,01	7,7	1,0	10,12	13,00	19,9	2,7	28,41	0,1
La20	1526	12,78	15,15	7,7	1,1	12,78	14,72	23,8	3,5	10,94	0,1
Média		4,92	8,17	5,9	0,8	3,81	7,41	16,39	2,32	15,25	0,1

CAPÍTULO 5. RESULTADOS E DISCUSSÃO

Tabela 5.4: Teste de Mann–Whitney de *BRPD* para os testes realizados com as instâncias em Sundar et al. [43].

	M2	M3 LS	M3 ELS	M3 ND	M3 END	NEH
M1	118,50 0,0103	108,50 0,0047	41,50 0,0000	218,50 0,9699	177,50 0,2834	69,50 0,0002
M2		203,00 0,6630	88,50 0,0003	111,00 0,0059	160,50 0,1309	28,00 0,0000
M3 LS			113,50 0,0026	100,00 0,0023	152,50 0,0848	27,00 0,0000
M3 ELS				41,50 0,0000	69,00 0,0000	12,50 0,0000
M3 ND					177,50 0,2833	63,50 0,0001
M3 END						52,50 0,0000
	U p	U p	U p	U p	U p	U p

Assim pode-se dizer que o Modelo 3 *enhanced left-shifting* é o que apresenta as melhores soluções, dado que o teste é significante e apresenta a melhor média de *BRPD* e *ARPD* ao mesmo tempo que apresenta a melhor média para estas variáveis. Contudo, em termos de tempo computacional, apesar de o teste ser significante, como a média de *TCT* e *ACT* é superior ao tempo de M3 LS, M3 ND, M3 END pode-se dizer que é pior nesta componente.

Tabela 5.5: Teste de Mann–Whitney de *ARPD* para os testes realizados com as instâncias em Sundar et al. [43].

	M2	M3 LS	M3 ELS	M3 ND	M3 END	NEH
M1	24,50 0,0000	20,00 0,0000	20,50 0,0000	56,50 0,0000	43,50 0,0000	153,50 0,0943
M2		130,00 0,0236	56,50 0,0000	114,50 0,0080	143,00 0,0527	70,50 0,0002
M3 LS			138,00 0,0391	71,00 0,0002	80,00 0,0004	44,00 0,0000
M3 ELS				39,50 0,0000	46,50 0,0000	29,50 0,0000
M3 ND					185,50 0,3854	111,50 0,0063
M3 END						106,50 0,0043
	U p	U p	U p	U p	U p	U p

Tabela 5.6: Teste de Mann–Whitney de *TCT* para os testes realizados com as instâncias em Sundar et al. [43].

	M2	M3 LS	M3 ELS	M3 ND	M3 END	NEH
M1	124,00 0,0157	0,00 0,0000	58,00 0,0000	0,00 0,0000	0,00 0,0000	0,00 0,0000
M2		15,00 0,0000	92,00 0,0013	11,00 0,0000	16,00 0,0000	0,00 0,0000
M3 LS			95,00 0,0017	95,50 0,0017	211,00 0,8209	1,00 0,0000
M3 ELS				21,00 0,0000	96,00 0,0018	1,00 0,0000
M3 ND					83,00 0,0006	1,00 0,0000
M3 END						1,00 0,0000
	U p	U p	U p	U p	U p	U p

Tabela 5.7: Teste de Mann–Whitney de *ACT* para os testes realizados com as instâncias em Sundar et al. [43].

	M2	M3 LS	M3 ELS	M3 ND	M3 END	NEH
M1	144,00 0,0559	7,50 0,0000	39,50 0,0000	0,00 0,0000	6,50 0,0000	0,00 0,0000
M2		15,00 0,0000	93,50 0,0015	12,50 0,0000	16,00 0,0000	1,00 0,0000
M3 LS			94,00 0,0015	96,00 0,0018	209,00 0,7814	25,50 0,0000
M3 ELS				20,50 0,0000	94,50 0,0016	2,00 0,0000
M3 ND					89,50 0,0010	11,00 0,0000
M3 END						6,00 0,0000
	U p	U p	U p	U p	U p	U p

5.2 Problema do número de trabalhos

Para os três modelos já apresentados deste problema, existem cinco variáveis a estudar relativamente ao seu impacto sobre a qualidade da solução e o tempo computacional necessário. Os níveis de cada variável estudados para os modelos são reportados na Tabela 5.8. Os níveis de L_k e CP utilizados foram concebidos de forma a manter o tempo computacional semelhante entre os modelos. Por sua vez, os níveis da punição P são menores que os utilizados no problema de *makespan*, devido ao menor ótimo da função objetivo.

Tabela 5.8: Níveis das variáveis dos modelos de *Simulated Annealing*

Modelo	L_k	CP	α	P	p_0
M1	1000; 3000; 5000	30; 90; 150	0,8; 0,9; 0,975	1; 5; 10	0,5; 0,7; 0,9
M2	100; 500; 1000	10; 55; 100	0,8; 0,9; 0,975	0	0,5; 0,7; 0,9
M3	50; 150; 250	5; 30; 55	0,8; 0,9; 0,975	0	0,5; 0,7; 0,9

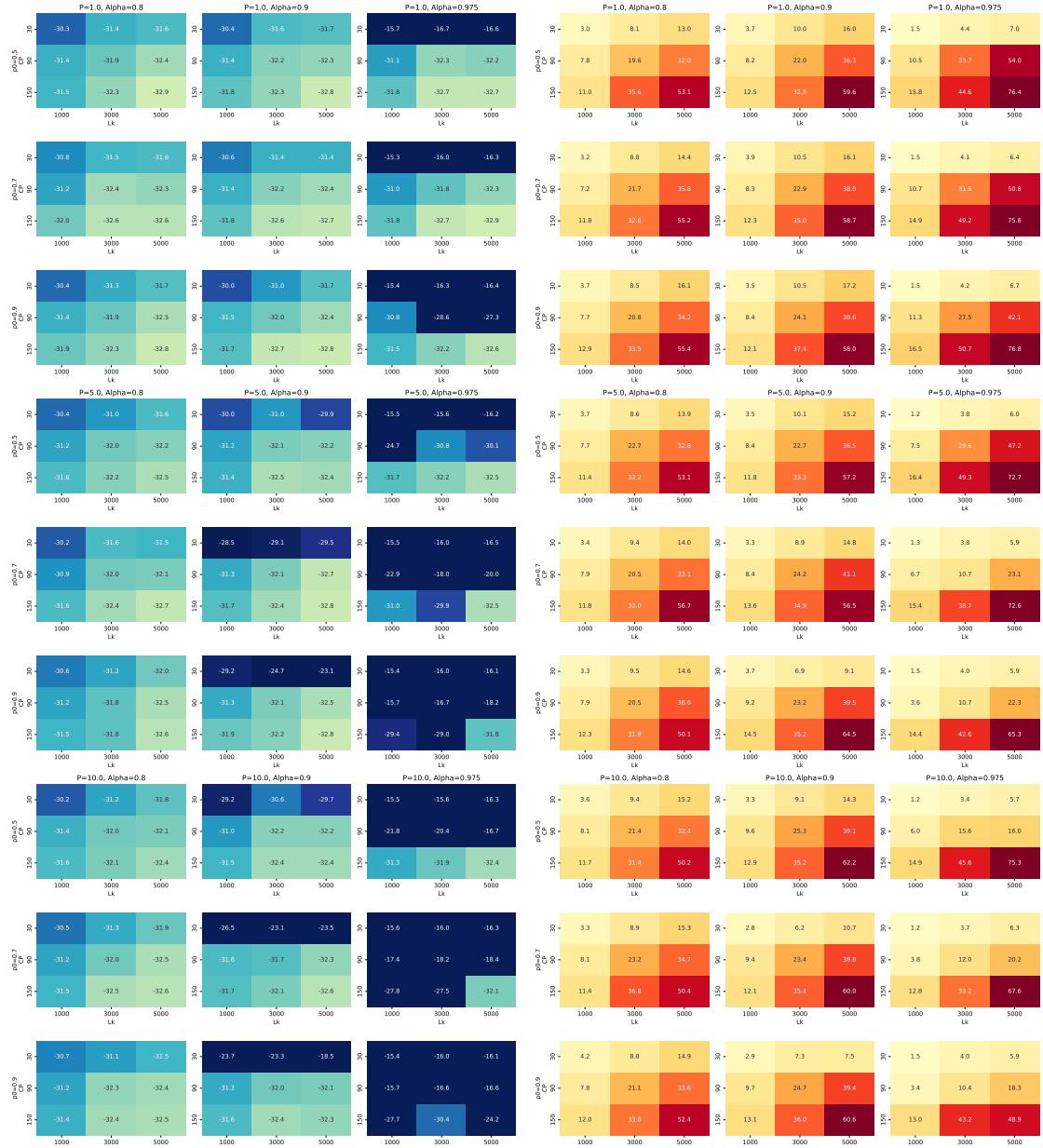
Para os próximos resultados, serão considerados os exames que decorrem numa segunda-feira habitual. Ou seja, 6 cintigrafia tiroideia, 10 cintigrafia pulmonar de ventilação/inalação + perfusão, 20 cintigrafia miocárdica de perfusão em repouso, 2 cintigrafia das glândulas salivares, 20 PET - estudo corpo inteiro com FDG. Para o valor de T_{\max} será utilizado o valor de *makespan* ótimo encontrado no problema anterior, ou seja, de 487 minutos.

Os resultados obtidos são provenientes de 100 repetições de cada combinação de variáveis. Como foi possível computar paralelamente 10 repetições, os valores apresentados são a média do valor mínimo de 10 repetições. Semelhantemente, o tempo computacional aprestado é relativo à média do valor máximo de 10 repetições.

CAPÍTULO 5. RESULTADOS E DISCUSSÃO

5.2.1 Modelo 1

Será apresentado um diagrama de calor referente ao valor da função objetivo e tempo computacional. Para a função objetivo, a faixa de cores está limitada aos valores entre -34 e -29, sendo estes os valores da melhor solução encontrada pelo *MILP*- δ e pela heurística NEH, respetivamente. O tempo computacional também está limitada entre 0 e 60 segundos.



(a) Média do mínimo de número de trabalhos (b) Média do máximo do tempo de computação

Figura 5.4: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 1.

Com a Figura 5.4 podemos tecer alguns comentários. A combinação de *CP* baixo e

redução lenta de temperatura diminui a qualidade das soluções. Ao aumentarmos o nível de CP garantimos que a solução se mantém ou melhora. Por sua vez, o aumento de L_k tende a melhorar a solução, contudo não existe a garantia de tal. A punição P não aparenta ter grande impacto sobre quando α e p_0 são baixos, mas quando estes níveis são elevados, o aumento de P parece piorar as soluções. Relativamente ao tempo computacional, os principais decisores aparentam ser L_k e Cp , quando estes níveis aumentam também aumenta o tempo computacional.

Com a ajuda da Tabela A.7 do Apêndice A podemos tecer alguns comentários sobre os valores ótimos das variáveis primárias, em relação ao número de exames:

- L_k apresenta efeito linear significativo negativo, o nível a escolher será o alto;
- CP apresenta efeito linear e quadrático significativo, ambos negativos, o nível a escolher será o alto;
- α apresenta efeitos lineares e quadráticos significativos, positivo e negativo, respetivamente, o nível a escolher será o baixo;
- P apresenta efeito linear significativo positivo, o nível a escolher será o baixo.
- p_0 apresenta efeito linear significativo positivo, o nível a escolher será o baixo.

Relativamente ao tempo, poderemos fazer uma análise semelhante pela Tabela A.8:

- L_k apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- CP apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- α apresenta efeitos lineares e quadráticos significativos, ambos positivos, tornando a escolha do nível difícil;
- P apresenta efeito linear significativo negativo, o nível a escolher será o alto;
- p_0 não apresenta efeito linear ou quadrático significativo, o nível a escolher será aquele mais conveniente.

De forma a obter uma boa combinação de variáveis utilizou-se novamente uma função a maximizar que utiliza ambas as variáveis dependentes, tal é dada por:

$$\frac{n_{exames}^4}{tempo^{0,1}}$$

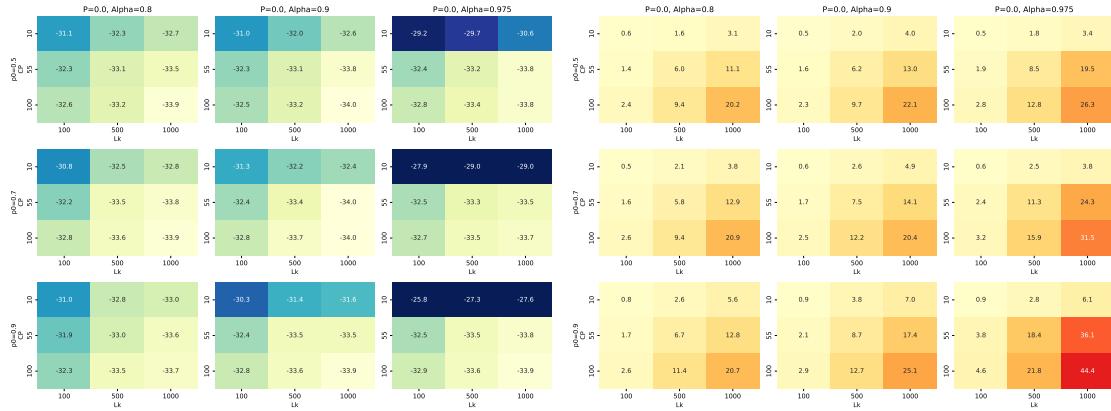
Difere da utilizada no problema anterior porque queremos maximizar o número de exames agendados enquanto se minimiza o tempo necessário, o exponente sobre o número de exames é menor de forma a não sobre-valorizar demais pequenas melhorias da qualidade da solução em troca de aumento excessivo do tempo computacional. Finalmente,

CAPÍTULO 5. RESULTADOS E DISCUSSÃO

definimos que a combinação que maximiza a fórmula é $\{1000; 150; 0,8; 1; 0,7\}$, com o número médio de exames agendados de 32,0 e tempo computacional médio de 11,8 segundos.

5.2.2 Modelo 2

De seguida, apresenta-se um novo diagrama de calor contendo os valores da função objetivo e do tempo computacional associados às combinações de níveis provenientes do Modelo 2. As cores do diagrama continuam a ser limitadas com os valores já referidos.



(a) Média do mínimo de número de trabalhos (b) Média do máximo do tempo de computação

Figura 5.5: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 2.

Com a análise da Figura 5.5 podemos comentar o impacto de cada variável sobre a função objetivo e o tempo computacional. Observamos que, em geral, a qualidade das soluções apresentadas é muito superior para um mesmo tempo computacional. Continuamos a verificar as observações já referidas para o modelo anterior.

Com a Tabela A.9 podemos tecer comentários mais concretos relativamente ao impacto de cada variável sobre a qualidade da solução:

- L_k apresenta efeito linear e quadrático significativos, ambos negativos, o nível a escolher será o alto;
- CP apresenta efeito linear e quadrático significativos, ambos negativos, o nível a escolher será o alto;
- α apresenta efeitos lineares e quadráticos significativos, positivo e negativo, respetivamente, o nível a escolher será o baixo;
- p_0 apresenta efeito linear significativo positivo, o nível a escolher será o baixo.

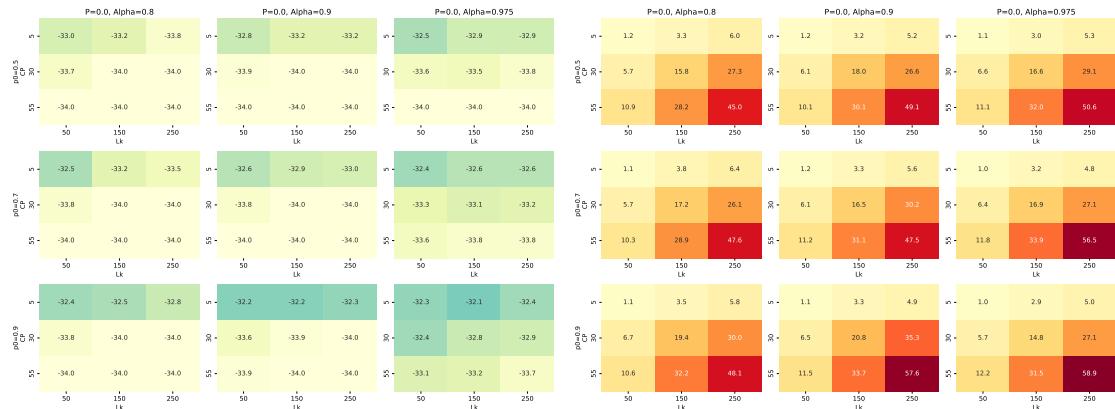
Relativamente ao tempo computacional, vamos utilizar a tabela A.10:

- L_k apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- CP apresenta efeito linear e quadrático significativo, ambos positivos, a escolher torna-se difícil, mas intuitivamente devemos escolher o nível baixo;
- α apresenta efeito linear e quadrático significativos, positivo e negativo, respetivamente, o nível a escolher será o baixo;
- p_0 apresenta efeito linear significante positivo, o nível a escolher será o baixo.

A combinação que maximiza a fórmula já descrita é $\{100; 100; 0, 9; 0; 0, 7\}$, com o número médio de exames agendados de 32,8 e tempo computacional médio de 2,5 segundos.

5.2.3 Modelo 3

Um novo diagrama de calor será apresentado, desta vez referente ao modelo 3, contendo as várias combinação de níveis referentes a este modelo. A faixa de cores continua a ser limitada como de antes.



(a) Média do mínimo de número de trabalhos (b) Média do máximo do tempo de computação
 Figura 5.6: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3.

Através da análise da figura 5.6 observamos que este modelo apresenta melhores soluções para o mesmo tempo computacional. Qualquer outro comentário será repetido dos já feitos.

Pela tabela A.11 podemos perceber o efeito principal de cada variável sobre a qualidade da solução.

- L_k apresenta efeito linear significativo negativo, o nível a escolher será o alto;
- CP apresenta efeito linear e quadrático significativos, ambos negativos, o nível a escolher será o alto;

CAPÍTULO 5. RESULTADOS E DISCUSSÃO

- α apresenta efeito linear e quadrático significativos, positivo e negativo, respetivamente, o nível a escolher será o baixo;
- p_0 apresenta efeito linear significativo positivo, o nível a escolher será o baixo.

Pela tabela A.12 podemos perceber o efeito principal de cada variável sobre o tempo computacional.

- L_k apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- CP apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- α apresenta efeito linear significativo positivo, o nível a escolher será o baixo;
- p_0 apresenta efeito linear significativo positivo, o nível a escolher será o baixo;

A combinação que maximiza a fórmula já descrita é {50; 5; 0,8; 0; 0,5}, com o número médio de exames agendados de 33,0 e tempo computacional médio de 1,2 segundos.

5.2.4 Comparação entre modelos

De seguida são apresentados, para cada dia da semana, os exames que podem ser realizados. Para cada um dos dias, utilizou-se o dobro do número de exames relativamente ao problema anterior. Por sua vez, T_{\max} será o melhor valor de *makespan* encontrado. Cada dia será resolvido utilizando cada um dos modelos com as variáveis já mencionadas.

- Segunda-feira: 6 cintigrafia tiroideia, 10 cintigrafia pulmonar de ventilação/inalação + perfusão, 20 cintigrafia miocárdica de perfusão em repouso, 2 cintigrafia das glândulas salivares, 20 PET - estudo corpo inteiro com FDG;
- Terça-feira: 4 cintigrafia tiroideia, 20 cintigrafia óssea corpo inteiro, 2 cintigrafia para amiloidose cardíaca, 2 linfoцитografia para detecção de gânglio sentinelas, 4 cintigrafia pulmonar de ventilação/inalação + perfusão, 20 PET - estudo corpo inteiro com FDG;
- Quarta-feira: 4 cintigrafia tiroideia, 4 linfoцитografia para detecção de gânglio sentinelas, 12 cintigrafia das paratiroides, 2 tomografia cerebral com ^{123}I -ioflupano, 6 cintigrafia pulmonar de ventilação/inalação + perfusão, 16 PET - estudo corpo inteiro com FDG;
- Quinta-feira: 12 linfoцитografia para detecção de gânglio sentinelas, 24 cintigrafia miocárdica de perfusão em esforço/stress farmacológico, 2 tomografia cerebral com ^{123}I -ioflupano, 4 cintigrafia miocárdica de perfusão em repouso, 20 PET - estudo corpo inteiro com FDG.

Tabela 5.9: Resultados provenientes dos três modelos em relação aos problemas de cada dia da semana.

Dia	BKS	MILP- δ			Modelo 1				Modelo 2				Modelo 3 LS			
		BRPD	BBRPD	TCT	BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT
Segunda-Feira	34	0,00	-8,82	28800	2,94	5,88	84,1	11,8	2,94	3,53	18,7	2,5	2,94	2,94	10,5	1,2
Terça-Feira	34	0,00	0,00	2555	5,88	6,76	89,5	11,9	0,00	3,82	18,5	2,4	2,94	2,94	19,5	2,5
Quarta-Feira	33	0,00	0,00	74	6,06	6,06	83,8	10,8	3,03	5,15	27,1	3,4	0,00	2,12	15,0	1,6
Quinta-Feira	36	0,00	-11,11	28800	8,33	8,61	92,3	12,9	2,78	5,00	24,9	3,3	0,00	2,78	40,3	5,9
Média		0,00	-4,98	15057,3	5,80	6,83	87,4	11,9	2,19	4,38	22,3	2,9	1,47	2,70	21,3	2,8

Dia	BKS	Modelo 3 ELS				Modelo 3 ND				Modelo 3 END				NEH	
		BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT	BRPD	ARPD	TCT	ACT	BRPD	TCT
Segunda-Feira	34	0,00	2,65	21,9	2,8	2,94	5,00	5,6	0,8	2,94	5,59	16,6	2,2	14,71	2,8
Terça-Feira	34	0,00	1,76	18	2,3	0,00	3,53	6,1	0,9	0,00	4,41	16,2	2,3	8,82	3,2
Quarta-Feira	33	0,00	1,52	14,6	1,7	3,03	3,33	5,7	0,6	3,03	3,33	12,9	1,9	6,06	1,6
Quinta-Feira	36	0,00	3,33	36,5	5,2	2,78	6,67	8,1	1,1	2,78	5,56	20,3	2,9	19,44	5,5
Média		0,00	2,32	22,75	3,00	2,19	4,63	6,38	0,85	2,19	4,72	16,50	2,33	12,26	3,3

Com a Tabela 5.9 podemos comparar os modelos na resolução dos quatro dias apresentados, utilizou-se a melhor solução proveniente do modelo $MILP - \delta$. Observamos que, até com 8 horas de tempo computacional alocado, nem sempre temos a garantia do valor ótimo, não temos por isso garantia que o melhor valor encontrado seja o valor ótimo. Verificamos de novo que o modelo 1 é superado pelo modelo 2 e 3 em qualidade da solução e em tempo computacional necessário. Por sua vez, o modelo 3 apresenta soluções marginalmente melhores que o modelo 2, em tempo computacional comparável. Como seria de esperar, todos os modelos apresentam soluções melhores que a heurística NEH.

5.2.5 Comparação com instâncias de *benchmark*

De seguida são apresentados os resultados provenientes das instâncias já discutidas, para cada uma utilizou-se o dobro dos trabalhos e utilizou-se T_{\max} como o melhor valor de *makespan* já reportado na literatura, ou seja, o valor *BKS* do problema anterior.

Tabela 5.10: Resultados provenientes dos três modelos em relação às instâncias apresentadas em Sundar et al. [43].

Instância	BKS	MILP			Modelo 1			Modelo 2			Modelo 3 LS					
		BRPD	BBRPD	TCT	BRPD	ARPД	TCT	ACT	BRPD	ARPД	TCT	ACT	BRPD	ARPД	TCT	ACT
Ft06	7	0,00	0,00	1	14,29	14,29	40,1	4,8	14,29	14,29	5,3	0,6	0,00	0,00	2,7	0,2
La01	11	9,09	-36,36	7200	9,09	9,09	64,8	8,5	0,00	8,18	27,3	3,8	0,00	0,00	5,0	0,7
La02	12	8,33	-25,00	7200	8,33	17,50	67,2	9,5	0,00	10,00	26,3	3,4	0,00	6,67	4,7	0,6
La03	11	0,00	-36,36	7200	0,00	10,00	54,1	8,7	0,00	7,27	24,7	3,4	0,00	0,00	4,1	0,6
La04	11	9,09	-36,36	7200	9,09	14,55	63,8	9,0	0,00	12,73	26,2	3,3	0,00	2,73	4,1	0,6
La05	11	9,09	-27,27	7200	9,09	17,27	57,3	7,7	9,09	10,91	20,0	2,4	9,09	9,09	3,7	0,5
Ft10	12	8,33	-58,33	7200	16,67	16,67	98,8	13,0	8,33	13,33	123,5	16,3	0,00	5,83	23,9	3,2
Orb01	11	9,09	-72,73	7200	9,09	11,82	99,7	13,3	0,00	8,18	127,5	15,9	0,00	0,91	24,3	3,2
Orb02	11	18,18	-72,73	7200	9,09	10,91	99,1	12,3	0,00	9,09	126,3	15,5	0,00	0,91	23,2	3,1
Orb03	11	0,00	-45,45	7200	9,09	14,55	97,6	13,1	0,00	7,27	130,7	16,2	0,00	3,64	22,4	3,3
Orb04	11	18,18	-81,82	7200	9,09	12,73	99,9	13,6	0,00	7,27	141,3	18,7	0,00	3,64	28,2	3,9
Orb05	11	27,27	-63,64	7200	18,18	19,09	90,9	11,6	9,09	14,55	97,6	12,8	0,00	6,36	18,5	2,5
Orb06	11	18,18	-72,73	7200	9,09	12,73	98,9	12,6	0,00	8,18	125,3	15,8	0,00	0,91	25,8	3,4
Orb08	11	27,27	-63,64	7200	18,18	18,18	87,1	11,9	9,09	12,73	90,4	12,6	0,00	8,18	14,8	2,0
Orb09	12	41,67	-58,33	7200	0,00	18,33	95,9	13,3	8,33	15,83	104,2	13,4	0,00	10,00	19,8	2,7
Orb10	11	9,09	-72,73	7200	9,09	13,64	100,5	13,2	9,09	9,09	133,3	17,5	0,00	0,00	24,3	3,2
La16	12	8,33	-66,67	7200	16,67	18,33	100,5	13,0	8,33	11,67	128,1	16,7	8,33	8,33	24,6	3,2
La17	12	9,09	-63,64	7200	9,09	10,00	88,4	11,9	0,00	6,36	103,7	13,3	0,00	0,00	21,2	2,6
La18	11	27,27	-72,73	7200	9,09	13,64	97,5	13,4	0,00	10,00	109,5	13,9	0,00	2,73	22,0	3,0
La19	11	18,18	-81,82	7200	9,09	18,18	99,6	13,2	9,09	12,73	130,4	18,3	0,00	3,64	25,8	3,5
La20	11	0,00	-63,64	7200	9,09	11,82	97,1	13,3	0,00	5,45	128,2	16,8	0,00	0,00	24,5	3,5
Média		13,13	-55,81	6857	10,02	14,44	85,7	11,5	4,04	10,24	91,9	11,9	0,83	3,50	17,5	2,4

Tabela 5.10: Resultados provenientes dos três modelos em relação às instâncias apresentadas em Sundar et al. [43] (continuação).

Instância	BKS	Modelo 3 ELS				Modelo 3 ND				Modelo 3 END				NEH	
		BRPD	ARPД	TCT	ACT	BRPD	ARPД	TCT	ACT	BRPD	ARPД	TCT	ACT	BRPD	TCT
Ft06	7	0,00	0,00	4,0	0,4	0,00	7,14	1,1	0,1	14,29	5,71	3,8	0,3	0,00	0,0
La01	11	0,00	0,00	11,2	1,5	0,00	1,82	3,7	0,5	0,00	1,82	5,6	0,7	9,09	0,0
La02	12	0,00	7,50	10,6	1,4	0,00	7,50	3,8	0,5	0,00	6,67	5,6	0,7	16,67	0,1
La03	11	0,00	1,82	8,2	1,1	0,00	1,82	3,2	0,4	0,00	0,00	4,9	0,7	18,18	0,1
La04	11	0,00	3,64	9,5	1,2	0,00	3,64	3,8	0,5	0,00	4,55	5,2	0,7	9,09	0,1
La05	11	0,00	8,18	7,6	0,9	0,00	7,27	2,8	0,4	0,00	7,27	4,5	0,6	9,09	0,1
Ft10	12	0,00	7,50	52,7	7,4	0,00	5,83	18,0	2,5	0,00	6,67	24,8	3,4	16,67	0,4
Orb01	11	0,00	1,82	54,4	7,3	0,00	2,73	18,4	2,5	0,00	0,91	26,8	3,6	18,18	0,4
Orb02	11	0,00	0,91	50,4	6,8	0,00	2,73	19,1	2,6	0,00	4,55	25,4	3,4	9,09	0,5
Orb03	11	0,00	1,82	53,7	7,8	0,00	1,82	17,5	2,2	0,00	1,82	25,6	3,3	18,18	0,3
Orb04	11	0,00	3,64	58,8	8,1	0,00	5,45	20,3	2,5	0,00	6,36	28,6	3,8	18,18	0,5
Orb05	11	0,00	7,27	43,1	5,6	0,00	7,27	12,4	1,7	0,00	6,36	18,4	2,5	18,18	0,3
Orb06	11	0,00	0,00	60,7	8,0	0,00	1,82	19,8	2,6	0,00	3,64	27,5	3,7	9,09	0,4
Orb08	11	0,00	8,18	35,4	4,8	0,00	8,18	10,5	1,3	0,00	7,27	16,7	2,3	18,18	0,2
Orb09	12	0,00	2,50	43,7	6,2	8,33	12,50	15,0	1,9	8,33	11,67	21,5	3,0	16,67	0,3
Orb10	11	0,00	0,91	54,8	7,4	0,00	1,82	18,9	2,5	0,00	1,82	26,0	3,6	9,09	0,5
La16	12	0,00	7,50	61,0	8,2	8,33	8,33	18,3	2,5	8,33	9,17	25,5	3,5	16,67	0,5
La17	12	0,00	0,00	45,9	6,3	0,00	1,82	13,8	2,0	0,00	1,82	19,6	2,7	18,18	0,4
La18	11	0,00	5,45	49,8	6,6	0,00	5,45	15,2	2,0	0,00	6,36	21,6	3,0	18,18	0,4
La19	11	0,00	5,45	51,9	7,2	0,00	6,36	17,8	2,4	0,00	6,36	23,9	3,3	18,18	0,5
La20	11	0,00	1,82	54,1	7,0	0,00	2,73	17,8	2,3	0,00	3,64	24,4	3,2	9,09	0,5
Média		0,00	3,61	39,1	5,3	0,79	4,95	12,9	1,7	1,47	4,97	18,4	2,5	14,00	0,3

Para a tabela 5.10 comparamos a formulação $MILP - \delta$, os três modelos, e a heurística NEH. Para a formulação $MILP-\delta$ apenas se alocou 2 horas, desta forma, verifica-se que não apresenta as melhores soluções, ao mesmo tempo apresenta grande divergência entre a solução apresentada e o melhor *bound*. O Modelo 3 apresenta-se como a escolha óbvia para a resolução deste tipo de problemas, sendo sempre capaz de gerar a melhor solução conhecida, por sua vez o tempo computacional também é muito reduzido comparando os os modelos restantes. A heurística NEH apresenta soluções semelhantes à média do Modelo 1, fazendo-o com tempo computacional muito inferior.

Outra vez, apesar destes modelos não terem sido desenhados para resolver as instâncias aqui apresentadas, é possível fazer uma comparação entre os vários modelos utilizando as Tabelas 5.11, 5.12, 5.13, 5.14. Em termos de qualidade dos resultados, para um nível de significância de 5%, não se pode rejeitar a hipótese nula de que o Modelo 3 com os vários algoritmos de *timetabling* apresentam distribuição igual. Desta forma não se pode averiguar qual destes modelos é o melhor. Entre o Modelo 3 e os restantes, como a diferença entre as distribuições já é significante, pode-se admitir que o Modelo 3 é o melhor.

Tabela 5.11: Teste de Mann–Whitney de *BRPD* para os testes realizados com as instâncias em Sundar et al. [43].

	M2		M3 LS		M3 ELS		M3 ND		M3 END		NEH	
	U	p	U	p	U	p	U	p	U	p	U	p
M1	83,50	0,0003	31,00	0,0000	21,00	0,0000	24,00	0,0000	39,50	0,0000	121,50	0,0079
M2			145,00	0,0144	126,00	0,0010	141,00	0,0100	155,50	0,0413	39,50	0,0000
M3 LS					199,50	0,1623	219,50	0,9803	210,00	0,6546	15,00	0,0000
M3 ELS							199,50	0,1622	189,00	0,0807	10,50	0,0000
M3 ND									209,00	0,6223	11,50	0,0000
M3 END											19,00	0,0000
	U	p	U	p	U	p	U	p	U	p	U	p

Tabela 5.12: Teste de Mann–Whitney de *ARPД* para os testes realizados com as instâncias em Sundar et al. [43].

	M2		M3 LS		M3 ELS		M3 ND		M3 END		NEH	
	U	p	U	p	U	p	U	p	U	p	U	p
M1	77,00	0,0003	2,50	0,0000	0,00	0,0000	6,00	0,0000	5,00	0,0000	210,50	0,8086
M2			39,00	0,0000	32,50	0,0000	46,00	0,0000	41,50	0,0000	105,00	0,0035
M3 LS					213,00	0,8589	161,50	0,1394	157,50	0,1141	28,50	0,0000
M3 ELS							164,50	0,1590	174,00	0,2445	19,00	0,0000
M3 ND									217,00	0,6223	28,00	0,0000
M3 END											34,50	0,0000
	U	p	U	p	U	p	U	p	U	p	U	p

CAPÍTULO 5. RESULTADOS E DISCUSSÃO

Tabela 5.13: Teste de Mann–Whitney de *TCT* para os testes realizados com as instâncias em Sundar et al. [43].

	M2	M3 LS	M3 ELS	M3 ND	M3 END	NEH
M1	147,50 0,0682	0,00 0,0000	22,50 0,0000	0,00 0,0000	0,00 0,0000	0,00 0,0000
M2		33,00 0,0000	95,00 0,0017	16,00 0,0000	45,00 0,0000	0,00 0,0000
M3 LS			94,00 0,0015	115,00 0,0082	187,50 0,4135	0,00 0,0000
M3 ELS				88,00 0,0009	95,00 0,0017	0,00 0,0000
M3 ND					107,50 0,0046	0,00 0,0000
M3 END						0,00 0,0000
	U p	U p	U p	U p	U p	U p

Tabela 5.14: Teste de Mann–Whitney de *ACT* para os testes realizados com as instâncias em Sundar et al. [43].

	M2	M3 LS	M3 ELS	M3 ND	M3 END	NEH
M1	151,50 0,0847	0,00 0,0000	18,50 0,0000	0,00 0,0000	0,00 0,0000	0,00 0,0000
M2		44,00 0,0000	95,00 0,0017	22,50 0,0000	54,00 0,0000	0,00 0,0000
M3 LS			95,00 0,0017	113,00 0,0070	188,50 0,4268	17,50 0,0000
M3 ELS				92,00 0,0013	95,00 0,0017	8,50 0,0000
M3 ND					108,00 0,0047	43,00 0,0000
M3 END						12,50 0,0000
	U p	U p	U p	U p	U p	U p

5.3 Discussão

Com esta dissertação foi possível criar as duas ferramentas de decisão propostas inicialmente, uma que permite minimizar *makespan* com um número fixo de exames, outra que permite maximizar o número de exames num período fixo de tempo. Para tal foram concebidos três modelos utilizando *SA* e uma formulação *MILP* indexada no tempo que visam resolver problemas do tipo *Flexible Multi-Resource Job-Shop with No-Wait*.

A nova formulação *MILP* proposta tira vantagem da restrição *No-Wait* para criar padrões de utilização dos recursos para reduzir a dimensão do problema. Este facto permite alcançar a solução ótima, ou perto, no mesmo tempo que uma formulação derivada de Kondili et al. [36] não é capaz de gerar uma solução viável. Contudo é de esperar que a utilização do modelo disjuntivo implique maior eficiência [11].

Os modelos que utilizam *SA* podem ser divididos pela sua codificação: os modelos que utilizam o instante de começo de cada exame, e o que utiliza a sequência de agendamento. Para este primeiro grupo foi ainda possível diferenciar consoante a função objetivo, se a sobre-utilização é admitida e punida, ou se através da procura da vizinhança só são aceites aqueles que não causam sobre-utilização. Este tipo de codificação parece não ser reconhecida na literatura, algo razoável quando os resultados demonstrados são tidos em conta. Por sua vez, o Modelo 3 do problema de *makespan* apresenta muito bons resultados,

necessitando de um bom algoritmo de *timetabling* para tal, os aqui testados são todos de boa qualidade mas o *enhanced left-shifting* destaca-se, mesmo que em contrapartida necessite do mais tempo computacional, algo que não foge ao observado por Deng et al. [44] para o critério de *Total Flow Time*. Contrariamente, no problema do número de trabalhos já não se observa diferença significante entre os vários algoritmos de *timetabling*.

Para cada um dos modelos desenvolvidos foram discutidas várias possibilidades na sua implementação, incluindo a geração da temperatura inicial, geração da solução inicial, os vários critérios tidos por Franzin e Stützle [38], e pequenas peculiaridades dos modelos que influenciam como se comportam. De forma semelhante, foi realizado *design of experiments* sobre as variáveis dos modelos de modo a perceber como influenciam a qualidade da solução e o tempo computacional. Para a escolha de uma boa combinação de níveis optou-se pela definição de funções que integrassem a função objetivo e o tempo computacional.

Os resultados apresentados dividem-se em dados provenientes do caso em estudo e provenientes de *benchmarks* da literatura. Desta forma é possível verificar a qualidade dos modelos perante os problemas para o qual foram desenhados e permitiu a comparação com outros modelos, mesmo que estes não pretendam resolver o problema *Flexible Multi-Resource Job-Shop with No-Wait*. Em conjunto com o teste de Mann–Whitney sobre *BRPD*, *ARPД*, *TCT*, *ACT*, e a média destas variáveis para todos os *benchmarks* foi possível concluir que modelos possuem as melhores soluções e o melhor tempo computacional.

Os resultados provenientes dos problemas do sistema não devem ser tidos como completamente fidedignos, o processo de recolha de dados não foi extenso o suficiente para existir muito confiança sobre os valores aqui utilizados. Uma recolha de maior volume será necessária para implementar os modelos aqui apresentados para que estes gerem agendas representativas do que pode ser aplicável no sistema.

Ao mesmo tempo, as extensões e restrições aqui discutidas não representam o sistema em estuda na sua totalidade, o principal responsável será a restrição *No-Wait* que, no sistema atual, deve ser substituída por uma restrição de atraso ou manter *No-Wait* mas considera alguns tempos de processamento controláveis como feito por Aschauer et al. [45]. Estas duas modificações certamente permitiriam diminuir o tempo necessário a realizar todos os exames mesmo que ao aumentar o tempo que alguns pacientes tenham de permanecer no sistema.

Por outro lado, a incerteza presente no sistema, relativamente a entregas, ao tempo de processamento, ao estado do paciente (se este é autónomo ou acamado), a avarias, e por vezes no número de recursos disponíveis, dificulta a implementação de uma só agenda. Existem para tal possíveis soluções, a utiliza por Spratt e Kozan [46] visa modificar a

CAPÍTULO 5. RESULTADOS E DISCUSSÃO

agenda original ao longo da sua execução, de forma a manter a viabilidade desta mesmo com o decorrer de disruptões. Também é possível, durante a criação da agenda, adicionar margens entre os vários exames de modo a permitir que haja absorção de certas origens de incerteza [45]. Tendo em conta os modelos aqui desenvolvidos, a atualização continua da agenda apresenta-se como a opção com mais fácil aplicação teórica. Contudo, a informatização necessária para, em tempo real, recolher a fase em que se encontra cada exame pode dificultar a sua implementação.

Na implementação destes modelos dever-se-à utilizar uma linguagem de programação mais eficiente que a utilizada. Esta modificação permitirá aumentar os valores de L_k e CP sem aumentar o tempo computacional associado. A realização paralela de várias execuções, de 10 nesta dissertação, permitiu a obtenção de soluções com maior qualidade com pouco detrimento perante o tempo computacional necessário.

6

CONCLUSÃO

BIBLIOGRAFIA

- [1] M. Meskarpour Amiri, M. Kazemian, Z. Motaghed e Z. Abdi, «Systematic Review of Factors Determining Health Care Expenditures,» *Health Policy and Technology*, vol. 10, n.º 2, p. 100 498, 2021-06, issn: 2211-8837. doi: [10.1016/j.hlpt.2021.01.004](https://doi.org/10.1016/j.hlpt.2021.01.004). acedido em 2025-05-16 (ver p. 1).
- [2] E. R. de Saúde, *ERS - Tempos Máximos de Resposta Garantidos (TMRG)*, <https://www.ers.pt/pt/utentes/pfrequentes/faq/tempo-maximo-de-resposta-garantidos-tmrg/>. acedido em 2025-05-16 (ver p. 1).
- [3] A. Almén, «Trends in Diagnostic Nuclear Medicine in Sweden (2008–2023): Utilisation, Radiation Dose, and Methodological Insights,» *EJNMMI Physics*, vol. 12, n.º 1, p. 32, 2025-04, issn: 2197-7364. doi: [10.1186/s40658-025-00747-2](https://doi.org/10.1186/s40658-025-00747-2). acedido em 2025-05-21 (ver p. 1).
- [4] S. Dauzère-Pérès, J. Ding, L. Shen e K. Tamssouet, «The Flexible Job Shop Scheduling Problem: A Review,» *European Journal of Operational Research*, vol. 314, n.º 2, pp. 409–432, 2024-04, Fala sobre as restrições extras que pode haver, e tem bastantes artigos citados com no-wait e blocking-, issn: 0377-2217. doi: [10.1016/j.ejor.2023.05.017](https://doi.org/10.1016/j.ejor.2023.05.017). acedido em 2025-04-06 (ver pp. 3–5, 8).
- [5] P. Brucker e R. Schlie, «Job-shop scheduling with multi-purpose machines,» *Computing*, vol. 45, n.º 4, pp. 369–375, 1990-12, issn: 1436-5057. doi: [10.1007/BF02238804](https://doi.org/10.1007/BF02238804). acedido em 2025-05-24 (ver p. 3).
- [6] P. Brandimarte, «Routing and scheduling in a flexible job shop by tabu search,» *Annals of Operations Research*, vol. 41, n.º 3, pp. 157–183, 1993-09, issn: 1572-9338. doi: [10.1007/BF02023073](https://doi.org/10.1007/BF02023073). acedido em 2025-04-09 (ver p. 3).
- [7] P. Brucker e J. Neyer, «Tabu-search for the multi-mode job-shop problem,» *Operations-Research-Spektrum*, vol. 20, n.º 1, pp. 21–28, 1998-03, issn: 1436-6304. doi: [10.1007/BF01545525](https://doi.org/10.1007/BF01545525). acedido em 2025-05-26 (ver p. 3).

- [8] S. Dauzère-Pérès, W. Roux e J. B. Lasserre, «Multi-Resource Shop Scheduling with Resource Flexibility,» *European Journal of Operational Research*, vol. 107, n.º 2, pp. 289–305, 1998-06, issn: 0377-2217. doi: [10.1016/S0377-2217\(97\)00341-X](https://doi.org/10.1016/S0377-2217(97)00341-X). acedido em 2025-04-08 (ver p. 3).
- [9] J. R. Callahan, *The Nothing Hot Delay Problem in the Production of Steel*. [Toronto], 1971, Open Library ID: OL19839735M (ver p. 4).
- [10] B. Jiang, Y. Ma, L. Chen, B. Huang, Y. Huang e L. Guan, «A Review on Intelligent Scheduling and Optimization for Flexible Job Shop,» *International Journal of Control, Automation and Systems*, vol. 21, n.º 10, pp. 3127–3150, 2023-10, issn: 2005-4092. doi: [10.1007/s12555-023-0578-1](https://doi.org/10.1007/s12555-023-0578-1). acedido em 2025-04-10 (ver pp. 4, 7).
- [11] W.-Y. Ku e J. C. Beck, «Mixed Integer Programming Models for Job Shop Scheduling: A Computational Analysis,» *Computers & Operations Research*, vol. 73, pp. 165–173, 2016-09, issn: 0305-0548. doi: [10.1016/j.cor.2016.04.006](https://doi.org/10.1016/j.cor.2016.04.006). acedido em 2025-07-07 (ver pp. 4, 7, 62).
- [12] A. S. Manne, «On the Job-Shop Scheduling Problem,» *Operations Research*, vol. 8, n.º 2, pp. 219–223, 1960-04, issn: 0030-364X. doi: [10.1287/opre.8.2.219](https://doi.org/10.1287/opre.8.2.219). acedido em 2025-07-11 (ver p. 4).
- [13] E. H. Bowman, «The Schedule-Sequencing Problem,» *Operations Research*, 1959-10. doi: [10.1287/opre.7.5.621](https://doi.org/10.1287/opre.7.5.621). acedido em 2025-07-11 (ver p. 5).
- [14] H. M. Wagner, «An integer linear-programming model for machine scheduling,» *Naval Research Logistics Quarterly*, vol. 6, n.º 2, pp. 131–140, 1959, issn: 1931-9193. doi: [10.1002/nav.3800060205](https://doi.org/10.1002/nav.3800060205). acedido em 2025-07-11 (ver p. 5).
- [15] G. A. Kasapidis, D. C. Paraskevopoulos, P. P. Repoussis e C. D. Tarantilis, «Flexible Job Shop Scheduling Problems with Arbitrary Precedence Graphs,» *Production and Operations Management*, vol. 30, n.º 11, pp. 4044–4068, 2021-11, issn: 1059-1478. doi: [10.1111/poms.13501](https://doi.org/10.1111/poms.13501). acedido em 2025-09-21 (ver p. 5).
- [16] A. E. Ezugwu et al., «Metaheuristics: A comprehensive overview and classification along with bibliometric analysis,» *Artificial Intelligence Review*, vol. 54, n.º 6, pp. 4237–4316, 2021-08, issn: 1573-7462. doi: [10.1007/s10462-020-09952-0](https://doi.org/10.1007/s10462-020-09952-0). acedido em 2025-07-10 (ver p. 5).
- [17] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller e E. Teller, «Equation of State Calculations by Fast Computing Machines,» *The Journal of Chemical Physics*, vol. 21, n.º 6, pp. 1087–1092, 1953-06, issn: 0021-9606. doi: [10.1063/1.1699114](https://doi.org/10.1063/1.1699114). acedido em 2025-05-28 (ver pp. 6, 18).
- [18] S. Katoch, S. S. Chauhan e V. Kumar, «A review on genetic algorithm: Past, present, and future,» *Multimedia Tools and Applications*, vol. 80, n.º 5, pp. 8091–8126, 2021-02, issn: 1573-7721. doi: [10.1007/s11042-020-10139-6](https://doi.org/10.1007/s11042-020-10139-6). acedido em 2025-09-21 (ver p. 6).

BIBLIOGRAFIA

- [19] P. Shilpa e G. Balaraju, «Job Shop Scheduling Using Differential Evolution Algorithm,» vol. 8, pp. 327–338, 2018 (ver p. 6).
- [20] P. Sriboonchandr, N. Kriengkorakot e P. Kriengkorakot, «Improved Differential Evolution Algorithm for Flexible Job Shop Scheduling Problems,» *Mathematical and Computational Applications*, vol. 24, n.º 3, p. 80, 2019-09, issn: 2297-8747. doi: [10.3390/mca24030080](https://doi.org/10.3390/mca24030080). acedido em 2025-09-03 (ver pp. 6, 9).
- [21] «Particle Swarm Optimization: Advances, Applications, and Experimental Insights,» *Computers, Materials and Continua*, vol. 82, n.º 2, pp. 1539–1592, 2025-02, issn: 1546-2218. doi: [10.32604/cmc.2025.060765](https://doi.org/10.32604/cmc.2025.060765). acedido em 2025-09-21 (ver p. 7).
- [22] H. Xiong, S. Shi, D. Ren e J. Hu, «A Survey of Job Shop Scheduling Problem: The Types and Models,» *Computers & Operations Research*, vol. 142, p. 105731, 2022-06, issn: 0305-0548. doi: [10.1016/j.cor.2022.105731](https://doi.org/10.1016/j.cor.2022.105731). acedido em 2025-04-16 (ver p. 7).
- [23] P. Fattahi, M. Saidi Mehrabad e F. Jolai, «Mathematical modeling and heuristic approaches to flexible job shop scheduling problems,» *Journal of Intelligent Manufacturing*, vol. 18, n.º 3, pp. 331–342, 2007-06, issn: 1572-8145. doi: [10.1007/s10845-007-0026-8](https://doi.org/10.1007/s10845-007-0026-8). acedido em 2025-07-15 (ver p. 8).
- [24] C. Özgüven, L. Özbakır e Y. Yavuz, «Mathematical Models for Job-Shop Scheduling Problems with Routing and Process Plan Flexibility,» *Applied Mathematical Modelling*, vol. 34, n.º 6, pp. 1539–1548, 2010-06, issn: 0307-904X. doi: [10.1016/j.apm.2009.09.002](https://doi.org/10.1016/j.apm.2009.09.002). acedido em 2025-05-26 (ver p. 8).
- [25] K. Thörnblad, A.-B. Strömberg, M. Patriksson e T. Almgren, «A competitive iterative procedure using a time-indexed model for solving flexible job shop scheduling problems,» 2013. URL: <https://api.semanticscholar.org/CorpusID:16184058> (ver p. 8).
- [26] H. Samarghandi, «Solving the No-Wait Job Shop Scheduling Problem with Due Date Constraints: A Problem Transformation Approach,» *Computers & Industrial Engineering*, vol. 136, pp. 635–662, 2019-10, issn: 0360-8352. doi: [10.1016/j.cie.2019.07.054](https://doi.org/10.1016/j.cie.2019.07.054). acedido em 2025-07-15 (ver p. 8).
- [27] R. Behmanesh e I. Rahimi, «Improved ant colony optimization for multi-resource job shop scheduling: A special case of transportation,» *Economic Computation and Economic Cybernetics Studies and Research*, vol. 55, n.º 4, pp. 277–294, 2021, issn: 0424-267X. doi: [10.24818/18423264/55.4.21.18](https://doi.org/10.24818/18423264/55.4.21.18) (ver p. 8).
- [28] M. Nawaz, E. E. Enscore e I. Ham, «A Heuristic Algorithm for the M-Machine, n-Job Flow-Shop Sequencing Problem,» *Omega*, vol. 11, n.º 1, pp. 91–95, 1983-01, issn: 0305-0483. doi: [10.1016/0305-0483\(83\)90088-9](https://doi.org/10.1016/0305-0483(83)90088-9). acedido em 2025-08-14 (ver p. 8).

- [29] K.-C. Ying e S.-W. Lin, «Minimizing Total Completion Time in the No-Wait Jobshop Scheduling Problem Using a Backtracking Metaheuristic,» *Computers & Industrial Engineering*, vol. 169, p. 108238, 2022-07, issn: 0360-8352. doi: [10.1016/j.cie.2022.108238](https://doi.org/10.1016/j.cie.2022.108238). acedido em 2025-08-14 (ver pp. 8, 9).
- [30] M. Xu, S. Zhang e G. Deng, «No-Wait Job Shop Scheduling Using a Population-Based Iterated Greedy Algorithm,» *Algorithms*, vol. 14, n.º 5, p. 145, 2021-05, issn: 1999-4893. doi: [10.3390/a14050145](https://doi.org/10.3390/a14050145). acedido em 2025-08-14 (ver p. 8).
- [31] A. Aschauer, F. Roetzer, A. Steinboeck e A. Kugi, «An Efficient Algorithm for Scheduling a Flexible Job Shop with Blocking and No-Wait Constraints*,» *IFAC-PapersOnLine*, 20th IFAC World Congress, vol. 50, n.º 1, pp. 12490–12495, 2017-07, issn: 2405-8963. doi: [10.1016/j.ifacol.2017.08.2056](https://doi.org/10.1016/j.ifacol.2017.08.2056). acedido em 2025-04-11 (ver p. 8).
- [32] W. Xia e Z. Wu, «An Effective Hybrid Optimization Approach for Multi-Objective Flexible Job-Shop Scheduling Problems,» *Computers & Industrial Engineering*, vol. 48, n.º 2, pp. 409–425, 2005-03, issn: 0360-8352. doi: [10.1016/j.cie.2005.01.018](https://doi.org/10.1016/j.cie.2005.01.018). acedido em 2025-08-14 (ver p. 9).
- [33] S. Jonathan, C. E. Nugraheni e L. Abednego, «A Whale Optimization Algorithm Based Solver for No-wait Flexible Flow Shop Scheduling Problems,» em 2024 Ninth International Conference on Informatics and Computing (ICIC), 2024-10, pp. 1–6. doi: [10.1109/ICIC64337.2024.10956970](https://doi.org/10.1109/ICIC64337.2024.10956970). acedido em 2025-08-14 (ver p. 9).
- [34] Q. Perrachon, A.-L. Olteanu, M. Sevaux, S. Fréchengues e J.-F. Kerviche, «Industrial Multi-Resource Flexible Job Shop Scheduling with Partially Necessary Resources,» *European Journal of Operational Research*, vol. 320, n.º 2, pp. 309–327, 2025-01, issn: 0377-2217. doi: [10.1016/j.ejor.2024.07.023](https://doi.org/10.1016/j.ejor.2024.07.023). acedido em 2025-08-13 (ver p. 9).
- [35] A. Caumond, P. Lacomme e N. Tchernev, «A Memetic Algorithm for the Job-Shop with Time-Lags,» *Computers & Operations Research*, Part Special Issue: Includes Selected Papers Presented at the ECCO'04 European Conference on Combinatorial Optimization, vol. 35, n.º 7, pp. 2331–2356, 2008-07, issn: 0305-0548. doi: [10.1016/j.cor.2006.11.007](https://doi.org/10.1016/j.cor.2006.11.007). acedido em 2025-08-14 (ver p. 9).
- [36] E. Kondili, C. C. Pantelides e R. W. H. Sargent, «A General Algorithm for Short-Term Scheduling of Batch Operations—I. MILP Formulation,» *Computers & Chemical Engineering*, An International Journal of Computer Applications in Chemical Engineering, vol. 17, n.º 2, pp. 211–227, 1993-02, issn: 0098-1354. doi: [10.1016/0098-1354\(93\)80015-F](https://doi.org/10.1016/0098-1354(93)80015-F). acedido em 2025-07-11 (ver pp. 13, 62).
- [37] D. Delahaye, S. Chaimattanan e M. Mongeau, «Simulated Annealing: From Basics to Applications,» em *Handbook of Metaheuristics*, M. Gendreau e J.-Y. Potvin, eds., Cham: Springer International Publishing, 2019, pp. 1–35, isbn: 978-3-319-91086-4. doi: [10.1007/978-3-319-91086-4_1](https://doi.org/10.1007/978-3-319-91086-4_1). acedido em 2025-05-26 (ver p. 16).

- [38] A. Franzin e T. Stützle, «Revisiting Simulated Annealing: A Component-Based Analysis,» *Computers & Operations Research*, vol. 104, pp. 191–206, 2019-04, issn: 0305-0548. doi: [10.1016/j.cor.2018.12.015](https://doi.org/10.1016/j.cor.2018.12.015). acedido em 2025-08-18 (ver pp. 18, 19, 22, 27, 63).
- [39] G. Deng, Q. Su, Z. Zhang, H. Liu, S. Zhang e T. Jiang, «A Population-Based Iterated Greedy Algorithm for No-Wait Job Shop Scheduling with Total Flow Time Criterion,» *Engineering Applications of Artificial Intelligence*, vol. 88, p. 103 369, 2020-02, issn: 0952-1976. doi: [10.1016/j.engappai.2019.103369](https://doi.org/10.1016/j.engappai.2019.103369). acedido em 2025-08-15 (ver p. 24).
- [40] K.-C. Ying e S.-W. Lin, «Solving No-Wait Job-Shop Scheduling Problems Using a Multi-Start Simulated Annealing with Bi-Directional Shift Timetabling Algorithm,» *Computers & Industrial Engineering*, vol. 146, p. 106 615, 2020-08, issn: 0360-8352. doi: [10.1016/j.cie.2020.106615](https://doi.org/10.1016/j.cie.2020.106615). acedido em 2025-08-14 (ver pp. 25, 47).
- [41] C. J. Schuster e J. M. Framinan, «Approximative Procedures for No-Wait Job Shop Scheduling,» *Operations Research Letters*, vol. 31, n.º 4, pp. 308–318, 2003-07, issn: 0167-6377. doi: [10.1016/S0167-6377\(03\)00005-1](https://doi.org/10.1016/S0167-6377(03)00005-1). acedido em 2025-08-15 (ver p. 44).
- [42] J. M. Framinan e C. Schuster, «An Enhanced Timetabling Procedure for the No-Wait Job Shop Problem: A Complete Local Search Approach,» *Computers & Operations Research*, vol. 33, n.º 5, pp. 1200–1213, 2006-05, issn: 0305-0548. doi: [10.1016/j.cor.2004.09.009](https://doi.org/10.1016/j.cor.2004.09.009). acedido em 2025-08-15 (ver p. 44).
- [43] S. Sundar, P. N. Suganthan, C. T. Jin, C. T. Xiang e C. C. Soon, «A hybrid artificial bee colony algorithm for the job-shop scheduling problem with no-wait constraint,» *Soft Computing*, vol. 21, n.º 5, pp. 1193–1202, 2017-03, issn: 1433-7479. doi: [10.1007/s00500-015-1852-9](https://doi.org/10.1007/s00500-015-1852-9). acedido em 2025-09-01 (ver pp. 46–51, 59–62).
- [44] G. Deng, Z. Zhang, T. Jiang e S. Zhang, «Total Flow Time Minimization in No-Wait Job Shop Using a Hybrid Discrete Group Search Optimizer,» *Applied Soft Computing*, vol. 81, p. 105 480, 2019-08, issn: 1568-4946. doi: [10.1016/j.asoc.2019.05.007](https://doi.org/10.1016/j.asoc.2019.05.007). acedido em 2025-08-14 (ver p. 63).
- [45] A. Aschauer, F. Roetzer, A. Steinboeck e A. Kugi, «Efficient Scheduling of a Stochastic No-Wait Job Shop with Controllable Processing Times,» *Expert Systems with Applications*, vol. 162, p. 113 879, 2020-12, issn: 0957-4174. doi: [10.1016/j.eswa.2020.113879](https://doi.org/10.1016/j.eswa.2020.113879). acedido em 2025-04-11 (ver pp. 63, 64).
- [46] B. Spratt e E. Kozan, «A real-time reactive framework for the surgical case sequencing problem,» *Flexible Services and Manufacturing Journal*, vol. 33, n.º 1, pp. 183–211, 2021-03, issn: 1936-6590. doi: [10.1007/s10696-019-09371-6](https://doi.org/10.1007/s10696-019-09371-6). acedido em 2025-09-24 (ver p. 63).

A

TABE~~L~~AS

A.1 Problema de *Makespan*

APÊNDICE A. TABELAS

Tabela A.1: Effect Estimates; Var.:Makespan; R-sqr=0,85919; Adj:0,82251; 5 3-level factors, 1 Blocks, 243 Runs; MS Residual=17657,26; DV: Makespan. Modelo 1, problema de *makespan*

Factors	Effect	Std.Err	t(192)	p	-95% Cnf.Limt	+95% Cnf.Limt	Coeff.	Std.Err.Coeff	-95% Cnf.Limt	+95% Cnf.Limt
Mean/Interc.	636,032	8,52430	74,6140	0,000000	619,219	652,845	636,032	8,52430	619,219	652,845
(1)Lk (L)	13,658	20,88017	0,6541	0,513821	-27,526	54,842	6,829	10,44009	-13,763	27,421
Lk (Q)	-10,417	18,08276	-0,5761	0,565253	-46,083	25,250	-5,208	9,04138	-23,042	12,625
(2)CP (L)	-340,247	20,88017	-16,2952	0,000000	-381,431	-299,063	-170,123	10,44009	-190,715	-149,531
CP (Q)	-163,604	18,08276	-9,0475	0,000000	-199,270	-127,937	-81,802	9,04138	-99,635	-63,969
(3)Alpha (L)	249,080	20,88017	11,9290	0,000000	207,896	290,264	124,540	10,44009	103,948	145,132
Alpha (Q)	-89,869	18,08276	-4,9698	0,000001	-125,535	-54,202	-44,934	9,04138	-62,767	-27,101
(4)P (L)	-118,298	20,88017	-5,6655	0,000000	-159,482	-77,114	-59,149	10,44009	-79,741	-38,557
P (Q)	-59,417	18,08276	-3,2858	0,001209	-95,083	-23,750	-29,708	9,04138	-47,542	-11,875
(5)p0 (L)	125,273	20,88017	5,9996	0,000000	84,089	166,457	62,636	10,44009	42,044	83,228
p0 (Q)	-27,413	18,08276	-1,5160	0,131171	-63,079	8,253	-13,706	9,04138	-31,540	4,127
1L by 2L	-45,407	25,57289	-1,7756	0,077382	-95,847	5,032	-22,704	12,78644	-47,924	2,516
1L by 2Q	-25,496	22,14677	-1,1512	0,251064	-69,179	18,186	-12,748	11,07338	-34,589	9,093
1Q by 2L	7,307	22,14677	0,3300	0,741795	-36,375	50,990	3,654	11,07338	-18,187	25,495
1Q by 2Q	3,433	19,17966	0,1790	0,858119	-34,397	41,263	1,717	9,58983	-17,198	20,632
1L by 3L	4,441	25,57289	0,1737	0,862323	-45,999	54,881	2,220	12,78644	-23,000	27,440
1L by 3Q	27,604	22,14677	1,2464	0,214137	-16,079	71,286	13,802	11,07338	-8,039	35,643
1Q by 3L	-4,280	22,14677	-0,1932	0,846976	-47,962	39,403	-2,140	11,07338	-23,981	19,701
1Q by 3Q	-4,022	19,17966	-0,2097	0,834114	-41,852	33,808	-2,011	9,58983	-20,926	16,904
1L by 4L	-61,296	25,57289	-2,3969	0,017492	-111,736	-10,856	-30,648	12,78644	-55,868	-5,428
1L by 4Q	-29,091	22,14677	-1,3135	0,190567	-72,773	14,591	-14,545	11,07338	-36,386	7,296
1Q by 4L	7,759	22,14677	0,3504	0,726455	-35,923	51,441	3,880	11,07338	-17,961	25,721
1Q by 4Q	3,664	19,17966	0,1910	0,848704	-34,166	41,494	1,832	9,58983	-17,083	20,747
1L by 5L	35,800	25,57289	1,3999	0,163151	-14,640	86,240	17,900	12,78644	-7,320	43,120
1L by 5Q	-4,652	22,14677	-0,2100	0,833854	-48,334	39,030	-2,326	11,07338	-24,167	19,515
1Q by 5L	1,704	22,14677	0,0769	0,938761	-41,979	45,386	0,852	11,07338	-20,989	22,693
1Q by 5Q	-12,406	19,17966	-0,6468	0,518529	-50,235	25,424	-6,203	9,58983	-25,118	12,712
2L by 3L	-372,176	25,57289	-14,5535	0,000000	-422,616	-321,736	-186,088	12,78644	-211,308	-160,868
2L by 3Q	134,521	22,14677	6,0741	0,000000	90,839	178,204	67,261	11,07338	45,420	89,102
2Q by 3L	-185,366	22,14677	-8,3699	0,000000	-229,048	-141,684	-92,683	11,07338	-114,524	-70,842
2Q by 3Q	66,885	19,17966	3,4873	0,0000605	29,055	104,715	33,442	9,58983	14,527	52,357
2L by 4L	186,502	25,57289	7,2930	0,000000	136,062	236,942	93,251	12,78644	68,031	118,471
2L by 4Q	91,432	22,14677	4,1285	0,000054	47,750	135,115	45,716	11,07338	23,875	67,557
2Q by 4L	93,573	22,14677	4,2251	0,000037	49,891	137,255	46,787	11,07338	24,945	68,628
2Q by 4Q	45,562	19,17966	2,3756	0,018507	7,733	83,392	22,781	9,58983	3,866	41,696
2L by 5L	-187,011	25,57289	-7,3129	0,000000	-237,451	-136,571	-93,506	12,78644	-118,725	-68,286
2L by 5Q	40,696	22,14677	1,8376	0,067671	-2,986	84,379	20,348	11,07338	-1,493	42,189
2Q by 5L	-92,741	22,14677	-4,1876	0,000043	-136,423	-49,059	-46,370	11,07338	-68,211	-24,529
2Q by 5Q	20,197	19,17966	1,0531	0,293640	-17,633	58,027	10,099	9,58983	-8,816	29,014
3L by 4L	-109,100	25,57289	-4,2662	0,000031	-159,540	-58,660	-54,550	12,78644	-79,770	-29,330
3L by 4Q	-51,641	22,14677	-2,3318	0,020751	-95,323	-7,959	-25,820	11,07338	-47,661	-3,979
3Q by 4L	11,998	22,14677	0,5418	0,588615	-31,684	55,680	5,999	11,07338	-15,842	27,840
3Q by 4Q	3,328	19,17966	0,1735	0,862437	-34,502	41,158	1,664	9,58983	-17,251	20,579
3L by 5L	119,317	25,57289	4,6657	0,000006	68,877	169,757	59,658	12,78644	34,438	84,878
3L by 5Q	-19,582	22,14677	-0,8842	0,377688	-63,265	24,100	-9,791	11,07338	-31,632	12,050
3Q by 5L	-21,955	22,14677	-0,9913	0,322775	-65,637	21,728	-10,977	11,07338	-32,818	10,864
3Q by 5Q	2,487	19,17966	0,1297	0,896944	-35,342	40,317	1,244	9,58983	-17,671	20,159
4L by 5L	-71,591	25,57289	-2,7995	0,005641	-122,031	-21,151	-35,795	12,78644	-61,015	-10,575
4L by 5Q	16,668	22,14677	0,7526	0,452614	-27,015	60,350	8,334	11,07338	-13,507	30,175
4Q by 5L	-35,960	22,14677	-1,6237	0,106076	-79,642	7,722	-17,980	11,07338	-39,821	3,861
4Q by 5Q	6,540	19,17966	0,3410	0,733476	-31,290	44,370	3,270	9,58983	-15,645	22,185

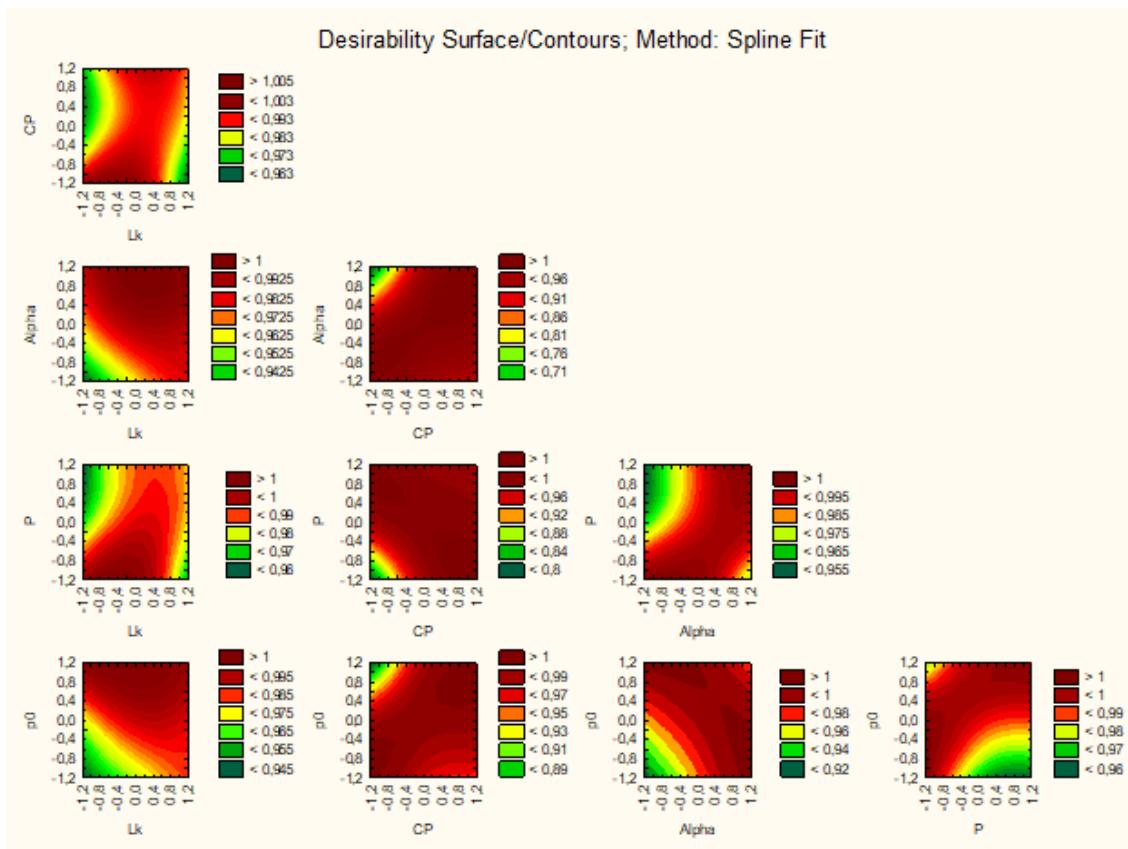


Figura A.1: Gráfico de contorno das cinco variáveis (L_k , CP , α , P , p_0), relativamente ao *makespan*, do Modelo 1 do problema de *makespan*

APÊNDICE A. TABELAS

Tabela A.2: Effect Estimates; Var.:Tempo; R-sqr=0,98992; Adj:0,98729; 5 3-level factors, 1 Blocks, 243 Runs; MS Residual=1,459374; DV: Tempo. Modelo 1, problema de *makespan*

Factors	Effect	Std.Err	t(192)	p	-95% Cnf.Limt	+95% Cnf.Limt	Coeff.	Std.Err.Coeff	-95% Cnf.Limt	+95% Cnf.Limt
Mean/Interc.	12,13253	0,077496	156,5565	0,000000	11,97967	12,28538	12,13253	0,077496	11,97967	12,28538
(1)Lk (L)	13,62214	0,189826	71,7612	0,000000	13,24772	13,99655	6,81107	0,094913	6,62386	6,99827
Lk (Q)	1,16790	0,164394	7,1043	0,000000	0,84365	1,49215	0,58395	0,082197	0,42182	0,74607
(2)CP (L)	19,62236	0,189826	103,3702	0,000000	19,24795	19,99678	9,81118	0,094913	9,62398	9,99839
CP (Q)	1,66684	0,164394	10,1393	0,000000	1,34259	1,99109	0,83342	0,082197	0,67130	0,99555
(3)Alpha (L)	0,68372	0,189826	3,6018	0,000402	0,30931	1,05813	0,34186	0,094913	0,15465	0,52907
Alpha (Q)	-0,11882	0,164394	-0,7228	0,470684	-0,44307	0,20543	-0,05941	0,082197	-0,22154	0,10271
(4)P (L)	3,48910	0,189826	18,3805	0,000000	3,11468	3,86351	1,74455	0,094913	1,55734	1,93175
P (Q)	1,27850	0,164394	7,7771	0,000000	0,95425	1,60276	0,63925	0,082197	0,47713	0,80138
(5)p0 (L)	0,23432	0,189826	1,2344	0,218574	-0,14010	0,60873	0,11716	0,094913	-0,07005	0,30436
p0 (Q)	0,15427	0,164394	0,9384	0,349205	-0,16998	0,47852	0,07714	0,082197	-0,08499	0,23926
1L by 2L	10,28914	0,232488	44,2565	0,000000	9,83058	10,74770	5,14457	0,116244	4,91529	5,37385
1L by 2Q	1,77548	0,201341	8,8183	0,000000	1,37836	2,17261	0,88774	0,100670	0,68918	1,08630
1Q by 2L	1,16728	0,201341	5,7975	0,000000	0,77016	1,56440	0,58364	0,100670	0,38508	0,78220
1Q by 2Q	-0,23543	0,174366	-1,3502	0,178536	-0,57935	0,10849	-0,11772	0,087183	-0,28968	0,05424
1L by 3L	0,45376	0,232488	1,9517	0,052424	-0,00480	0,91232	0,22688	0,116244	-0,00240	0,45616
1L by 3Q	-0,09508	0,201341	-0,4722	0,637297	-0,49220	0,30204	-0,04754	0,100670	-0,24610	0,15102
1Q by 3L	-0,02559	0,201341	-0,1271	0,898978	-0,42272	0,37153	-0,01280	0,100670	-0,21136	0,18576
1Q by 3Q	0,17401	0,174366	0,9980	0,319545	-0,16991	0,51793	0,08701	0,087183	-0,08495	0,25897
1L by 4L	2,39385	0,232488	10,2966	0,000000	1,93529	2,85241	1,19692	0,116244	0,96764	1,42620
1L by 4Q	1,09805	0,201341	5,4537	0,000000	0,70092	1,49517	0,54902	0,100670	0,35046	0,74759
1Q by 4L	0,11748	0,201341	0,5835	0,560255	-0,27965	0,51460	0,05874	0,100670	-0,13982	0,25730
1Q by 4Q	-0,11948	0,174366	-0,6852	0,494026	-0,46340	0,22444	-0,05974	0,087183	-0,23170	0,11222
1L by 5L	0,17933	0,232488	0,7713	0,441453	-0,27923	0,63789	0,08966	0,116244	-0,13962	0,31894
1L by 5Q	0,22357	0,201341	1,1104	0,268218	-0,17356	0,62069	0,11178	0,100670	-0,08678	0,31035
1Q by 5L	-0,11632	0,201341	-0,5777	0,564118	-0,51345	0,28080	-0,05816	0,100670	-0,25672	0,14040
1Q by 5Q	0,02635	0,174366	0,1511	0,880036	-0,31757	0,37027	0,01318	0,087183	-0,15878	0,18514
2L by 3L	1,74960	0,232488	7,5255	0,000000	1,29104	2,20816	0,87480	0,116244	0,64552	1,10408
2L by 3Q	-0,54684	0,201341	-2,7160	0,007211	-0,94397	-0,14972	-0,27342	0,100670	-0,47198	-0,07486
2Q by 3L	0,75124	0,201341	3,7312	0,000251	0,35411	1,14836	0,37562	0,100670	0,17706	0,57418
2Q by 3Q	-0,33750	0,174366	-1,9356	0,054385	-0,68142	0,00642	-0,16875	0,087183	-0,34071	0,00321
2L by 4L	2,25720	0,232488	9,7089	0,000000	1,79864	2,71576	1,12860	0,116244	0,89932	1,35788
2L by 4Q	0,96528	0,201341	4,7942	0,000003	0,56815	1,36240	0,48264	0,100670	0,28408	0,68120
2Q by 4L	0,53284	0,201341	2,6465	0,008809	0,13572	0,92997	0,26642	0,100670	0,06786	0,46498
2Q by 4Q	-0,01009	0,174366	-0,0579	0,953909	-0,35401	0,33383	-0,00505	0,087183	-0,17701	0,16691
2L by 5L	0,40344	0,232488	1,7353	0,084290	-0,05512	0,86200	0,20172	0,116244	-0,02756	0,43100
2L by 5Q	0,16409	0,201341	0,8150	0,416084	-0,23303	0,56122	0,08205	0,100670	-0,11652	0,28061
2Q by 5L	0,07408	0,201341	0,3680	0,713316	-0,32304	0,47121	0,03704	0,100670	-0,16152	0,23560
2Q by 5Q	-0,25198	0,174366	-1,4451	0,150051	-0,59590	0,09194	-0,12599	0,087183	-0,29795	0,04597
3L by 4L	0,33507	0,232488	1,4412	0,151145	-0,12349	0,79363	0,16754	0,116244	-0,06174	0,39682
3L by 4Q	0,23495	0,201341	1,1669	0,244687	-0,16217	0,63207	0,11748	0,100670	-0,08109	0,31604
3Q by 4L	-0,21669	0,201341	-1,0763	0,283162	-0,61382	0,18043	-0,10835	0,100670	-0,30691	0,09021
3Q by 4Q	0,07438	0,174366	0,4266	0,670168	-0,26954	0,41830	0,03719	0,087183	-0,13477	0,20915
3L by 5L	0,50616	0,232488	2,1771	0,030691	0,04760	0,96472	0,25308	0,116244	0,02380	0,48236
3L by 5Q	-0,29040	0,201341	-1,4423	0,150845	-0,68752	0,10673	-0,14520	0,100670	-0,34376	0,05336
3Q by 5L	-0,34815	0,201341	-1,7292	0,085384	-0,74528	0,04897	-0,17408	0,100670	-0,37264	0,02448
3Q by 5Q	-0,19125	0,174366	-1,0968	0,274093	-0,53517	0,15267	-0,09562	0,087183	-0,26758	0,07634
4L by 5L	0,19665	0,232488	0,8458	0,398697	-0,26191	0,65521	0,09832	0,116244	-0,13096	0,32760
4L by 5Q	-0,28064	0,201341	-1,3938	0,164975	-0,67776	0,11649	-0,14032	0,100670	-0,33888	0,05824
4Q by 5L	0,16679	0,201341	0,8284	0,408467	-0,23033	0,56392	0,08340	0,100670	-0,11517	0,28196
4Q by 5Q	-0,02010	0,174366	-0,1153	0,908333	-0,36402	0,32382	-0,01005	0,087183	-0,18201	0,16191

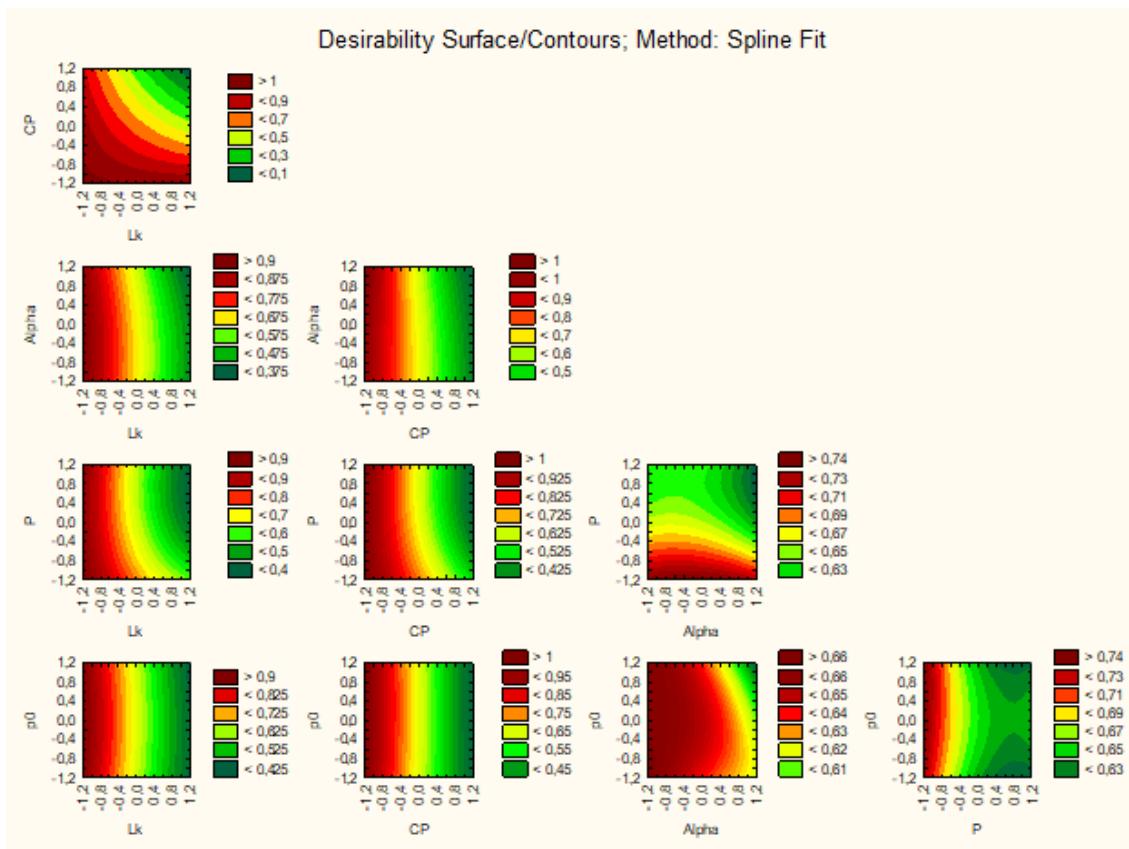


Figura A.2: Gráfico de contorno das cinco variáveis (L_k, CP, α, P, p_0), relativamente ao tempo, do Modelo 1 do problema de makespan

APÊNDICE A. TABELAS

Tabela A.3: Effect Estimates; Var.:Makespan; R-sqr=0,79905; Adj:0,66508 ; 4 3-level factors, 1 Blocks, 81 Runs; MS Residual=29,81784; DV: Makespan. Modelo 2, problema de *makespan*

Factors	Effect	Std.Err	t(48)	p	-95% Cnf.Limt	+95% Cnf.Limt	Coeff.	Std.Err.Coeff	-95% Cnf.Limt	+95% Cnf.Limt
Mean/Interc.	512,7222	0,606730	845,0581	0,000000	511,5023	513,9421	512,7222	0,606730	511,5023	513,9421
(1)Lk (L)	-9,3259	1,486179	-6,2751	0,000000	-12,3141	-6,3378	-4,6630	0,743090	-6,1570	-3,1689
Lk (Q)	-2,1111	1,287069	-1,6402	0,107492	-4,6989	0,4767	-1,0556	0,643535	-2,3495	0,2384
(2)CP (L)	-10,9111	1,486179	-7,3417	0,000000	-13,8993	-7,9229	-5,4556	0,743090	-6,9496	-3,9615
CP (Q)	-4,9667	1,287069	-3,8589	0,000340	-7,5545	-2,3788	-2,4833	0,643535	-3,7772	-1,1894
(3)Alpha (L)	0,5667	1,486179	0,3813	0,704670	-2,4215	3,5548	0,2833	0,743090	-1,2107	1,7774
Alpha (Q)	-1,2167	1,287069	-0,9453	0,349240	-3,8045	1,3712	-0,6083	0,643535	-1,9022	0,6856
(4)p0 (L)	0,6815	1,486179	0,4585	0,648630	-2,3067	3,6696	0,3407	0,743090	-1,1533	1,8348
p0 (Q)	-2,0889	1,287069	-1,6230	0,111144	-4,6767	0,4989	-1,0444	0,643535	-2,3384	0,2495
1L by 2L	7,4889	1,820190	4,1143	0,000152	3,8292	11,1486	3,7444	0,910095	1,9146	5,5743
1L by 2Q	3,2444	1,576331	2,0582	0,045020	0,0750	6,4139	1,6222	0,788166	0,0375	3,2069
1Q by 2L	2,3500	1,576331	1,4908	0,142556	-0,8194	5,5194	1,1750	0,788166	-0,4097	2,7597
1Q by 2Q	0,9083	1,365143	0,6654	0,508995	-1,8365	3,6531	0,4542	0,682571	-0,9182	1,8266
1L by 3L	-2,2556	1,820190	-1,2392	0,221302	-5,9153	1,4042	-1,1278	0,910095	-2,9576	0,7021
1L by 3Q	0,5111	1,576331	0,3242	0,747164	-2,6583	3,6805	0,2556	0,788166	-1,3292	1,8403
1Q by 3L	-0,6667	1,576331	-0,4229	0,674240	-3,8361	2,5028	-0,3333	0,788166	-1,9180	1,2514
1Q by 3Q	-0,0417	1,365143	-0,0305	0,975777	-2,7865	2,7031	-0,0208	0,682571	-1,3932	1,3516
1L by 4L	-2,9222	1,820190	-1,6054	0,114955	-6,5820	0,7375	-1,4611	0,910095	-3,2910	0,3688
1L by 4Q	0,2444	1,576331	0,1551	0,877415	-2,9250	3,4139	0,1222	0,788166	-1,4625	1,7069
1Q by 4L	0,3889	1,576331	0,2467	0,806189	-2,7805	3,5583	0,1944	0,788166	-1,3903	1,7792
1Q by 4Q	0,2250	1,365143	0,1648	0,869779	-2,5198	2,9698	0,1125	0,682571	-1,2599	1,4849
2L by 3L	-5,8333	1,820190	-3,2048	0,002404	-9,4931	-2,1736	-2,9167	0,910095	-4,7465	-1,0868
2L by 3Q	3,3500	1,576331	2,1252	0,038745	0,1806	6,5194	1,6750	0,788166	0,0903	3,2597
2Q by 3L	-2,9667	1,576331	-1,8820	0,065905	-6,1361	0,2028	-1,4833	0,788166	-3,0680	0,1014
2Q by 3Q	1,6500	1,365143	1,2087	0,232712	-1,0948	4,3948	0,8250	0,682571	-0,5474	2,1974
2L by 4L	-5,1444	1,820190	-2,8263	0,006844	-8,8042	-1,4847	-2,5722	0,910095	-4,4021	-0,7424
2L by 4Q	2,9167	1,576331	1,8503	0,070432	-0,2528	6,0861	1,4583	0,788166	-0,1264	3,0430
2Q by 4L	-2,5111	1,576331	-1,5930	0,117723	-5,6805	0,6583	-1,2556	0,788166	-2,8403	0,3292
2Q by 4Q	1,4667	1,365143	1,0744	0,288029	-1,2781	4,2115	0,7333	0,682571	-0,6391	2,1057
3L by 4L	3,7111	1,820190	2,0389	0,046992	0,0514	7,3708	1,8556	0,910095	0,0257	3,6854
3L by 4Q	-4,2167	1,576331	-2,6750	0,010189	-7,3861	-1,0472	-2,1083	0,788166	-3,6930	-0,5236
3Q by 4L	-1,5944	1,576331	-1,0115	0,316853	-4,7639	1,5750	-0,7972	0,788166	-2,3819	0,7875
3Q by 4Q	1,0167	1,365143	0,7447	0,460064	-1,7281	3,7615	0,5083	0,682571	-0,8641	1,8807

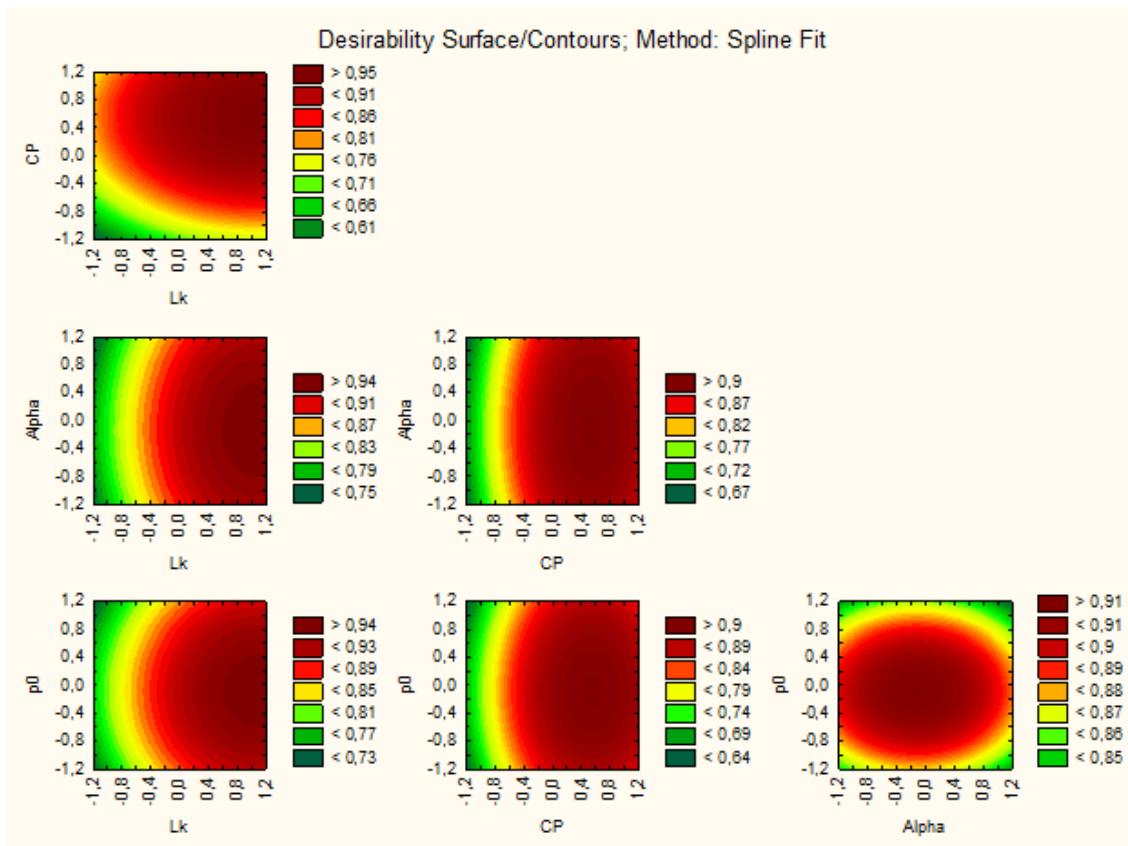


Figura A.3: Gráfico de contorno das quatro variáveis relevantes (L_k, CP, α, p_0), relativamente ao *makespan*, do Modelo 2 do problema de *makespan*

APÊNDICE A. TABELAS

Tabela A.4: Effect Estimates; Var.:Tempo; R-sqr=0,97985; Adj:0,96642; 4 3-level factors, 1 Blocks, 81 Runs; MS Residual=5,912985; DV: Tempo. Modelo 2, problema de *makespan*

Factors	Effect	Std.Err	t(48)	p	-95% Cnf.Limt	+95% Cnf.Limt	Coeff.	Std.Err.Coeff	-95% Cnf.Limt	+95% Cnf.Limt
Mean/Interc.	19,94905	0,270185	73,83485	0,000000	19,40581	20,49229	19,94905	0,270185	19,40581	20,49229
(1)Lk (L)	21,72384	0,661815	32,82465	0,000000	20,39317	23,05451	10,86192	0,330907	10,19659	11,52725
Lk (Q)	0,72651	0,573148	1,26758	0,211063	-0,42588	1,87890	0,36326	0,286574	-0,21294	0,93945
(2)CP (L)	19,97500	0,661815	30,18215	0,000000	18,64433	21,30566	9,98750	0,330907	9,32216	10,65283
CP (Q)	2,84850	0,573148	4,96991	0,000009	1,69611	4,00089	1,42425	0,286574	0,84805	2,00045
(3)Alpha (L)	3,31104	0,661815	5,00297	0,000008	1,98037	4,64170	1,65552	0,330907	0,99018	2,32085
Alpha (Q)	-1,14631	0,573148	-2,00001	0,051174	-2,29870	0,00609	-0,57315	0,286574	-1,14935	0,00304
(4)p0 (L)	2,11030	0,661815	3,18866	0,002517	0,77964	3,44097	1,05515	0,330907	0,38982	1,72049
p0 (Q)	-1,01754	0,573148	-1,77534	0,082182	-2,16993	0,13486	-0,50877	0,286574	-1,08496	0,06743
1L by 2L	11,26922	0,810554	13,90310	0,000000	9,63949	12,89895	5,63461	0,405277	4,81975	6,44947
1L by 2Q	1,29787	0,701961	1,84892	0,070633	-0,11352	2,70926	0,64894	0,350980	-0,05676	1,35463
1Q by 2L	0,03951	0,701961	0,05629	0,955346	-1,37187	1,45090	0,01976	0,350980	-0,68594	0,72545
1Q by 2Q	-0,11595	0,607916	-0,19073	0,849543	-1,33824	1,10635	-0,05797	0,303958	-0,66912	0,55318
1L by 3L	4,36745	0,810554	5,38822	0,000002	2,73772	5,99718	2,18372	0,405277	1,36886	2,99859
1L by 3Q	-1,49005	0,701961	-2,12270	0,038963	-2,90144	-0,07867	-0,74503	0,350980	-1,45072	-0,03933
1Q by 3L	-0,62145	0,701961	-0,88530	0,380406	-2,03284	0,78994	-0,31072	0,350980	-1,01642	0,39497
1Q by 3Q	-0,04126	0,607916	-0,06787	0,946169	-1,26356	1,18104	-0,02063	0,303958	-0,63178	0,59052
1L by 4L	2,06900	0,810554	2,55257	0,013929	0,43927	3,69873	1,03450	0,405277	0,21963	1,84936
1L by 4Q	-0,88475	0,701961	-1,26039	0,213621	-2,29613	0,52664	-0,44237	0,350980	-1,14807	0,26332
1Q by 4L	-0,12064	0,701961	-0,17186	0,864271	-1,53202	1,29075	-0,06032	0,350980	-0,76601	0,64537
1Q by 4Q	-0,25861	0,607916	-0,42540	0,672449	-1,48090	0,96369	-0,12930	0,303958	-0,74045	0,48185
2L by 3L	2,11549	0,810554	2,60993	0,012043	0,48576	3,74522	1,05774	0,405277	0,24288	1,87261
2L by 3Q	-0,84142	0,701961	-1,19868	0,236538	-2,25281	0,56996	-0,42071	0,350980	-1,12641	0,28498
2Q by 3L	0,88890	0,701961	1,26630	0,211515	-0,52249	2,30028	0,44445	0,350980	-0,26125	1,15014
2Q by 3Q	-0,55794	0,607916	-0,91779	0,363316	-1,78024	0,66436	-0,27897	0,303958	-0,89012	0,33218
2L by 4L	0,63012	0,810554	0,77740	0,440738	-0,99960	2,25985	0,31506	0,405277	-0,49980	1,12993
2L by 4Q	-0,42321	0,701961	-0,60290	0,549415	-1,83460	0,98818	-0,21161	0,350980	-0,91730	0,49409
2Q by 4L	0,48816	0,701961	0,69542	0,490147	-0,92323	1,89954	0,24408	0,350980	-0,46162	0,94977
2Q by 4Q	-0,17075	0,607916	-0,28088	0,780006	-1,39305	1,05154	-0,08538	0,303958	-0,69653	0,52577
3L by 4L	3,86791	0,810554	4,77193	0,000017	2,23818	5,49763	1,93395	0,405277	1,11909	2,74882
3L by 4Q	-0,61189	0,701961	-0,87169	0,387719	-2,02328	0,79950	-0,30594	0,350980	-1,01164	0,39975
3Q by 4L	-0,88585	0,701961	-1,26197	0,213058	-2,29724	0,52553	-0,44293	0,350980	-1,14862	0,26277
3Q by 4Q	0,41730	0,607916	0,68645	0,495734	-0,80499	1,63960	0,20865	0,303958	-0,40250	0,81980

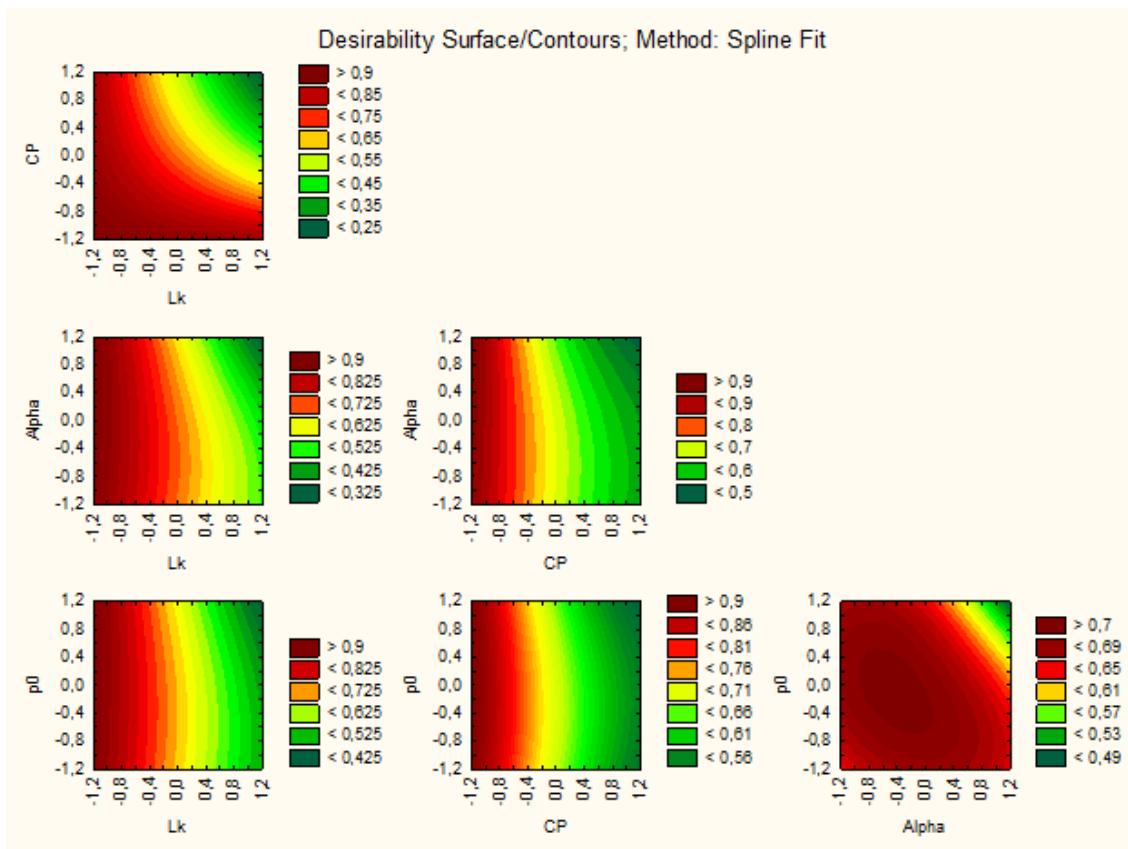


Figura A.4: Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao tempo, do Modelo 2 do problema de makespan

APÊNDICE A. TABELAS

Tabela A.5: Effect Estimates; Var.:Makespan; R-sqr=0,97227; Adj:0,95378; 4 3-level factors, 1 Blocks, 81 Runs; MS Residual=1,415756; DV: Makespan. Modelo 3, problema de *makespan*

Factors	Effect	Std.Err	t(48)	p	-95% Cnf.Limt	+95% Cnf.Limt	Coeff.	Std.Err.Coeff	-95% Cnf.Limt	+95% Cnf.Limt
Mean/Interc.	501,4914	0,132206	3793,252	0,000000	501,2255	501,7572	501,4914	0,132206	501,2255	501,7572
(1)Lk (L)	-2,0185	0,323838	-6,233	0,000000	-2,6696	-1,3674	-1,0093	0,161919	-1,3348	-0,6837
Lk (Q)	-0,3537	0,280452	-1,261	0,213335	-0,9176	0,2102	-0,1769	0,140226	-0,4588	0,1051
(2)CP (L)	-9,6630	0,323838	-29,839	0,000000	-10,3141	-9,0118	-4,8315	0,161919	-5,1570	-4,5059
CP (Q)	-3,7093	0,280452	-13,226	0,000000	-4,2731	-3,1454	-1,8546	0,140226	-2,1366	-1,5727
(3)Alpha (L)	4,6111	0,323838	14,239	0,000000	3,9600	5,2622	2,3056	0,161919	1,9800	2,6311
Alpha (Q)	-0,0759	0,280452	-0,271	0,787762	-0,6398	0,4880	-0,0380	0,140226	-0,3199	0,2440
(4)p0 (L)	2,8481	0,323838	8,795	0,000000	2,1970	3,4993	1,4241	0,161919	1,0985	1,7496
p0 (Q)	-0,0870	0,280452	-0,310	0,757641	-0,6509	0,4768	-0,0435	0,140226	-0,3255	0,2384
1L by 2L	1,6611	0,396619	4,188	0,000120	0,8637	2,4586	0,8306	0,198309	0,4318	1,2293
1L by 2Q	0,5306	0,343482	1,545	0,129001	-0,1601	1,2212	0,2653	0,171741	-0,0800	0,6106
1Q by 2L	-0,0861	0,343482	-0,251	0,803116	-0,7767	0,6045	-0,0431	0,171741	-0,3884	0,3023
1Q by 2Q	-0,1431	0,297464	-0,481	0,632760	-0,7411	0,4550	-0,0715	0,148732	-0,3706	0,2275
1L by 3L	-0,3889	0,396619	-0,981	0,331750	-1,1863	0,4086	-0,1944	0,198309	-0,5932	0,2043
1L by 3Q	0,0556	0,343482	0,162	0,872188	-0,6351	0,7462	0,0278	0,171741	-0,3175	0,3731
1Q by 3L	-0,1167	0,343482	-0,340	0,735595	-0,8073	0,5739	-0,0583	0,171741	-0,4036	0,2870
1Q by 3Q	0,1944	0,297464	0,654	0,516441	-0,4036	0,7925	0,0972	0,148732	-0,2018	0,3963
1L by 4L	0,0833	0,396619	0,210	0,834472	-0,7141	0,8808	0,0417	0,198309	-0,3571	0,4404
1L by 4Q	-0,3528	0,343482	-1,027	0,309538	-1,0434	0,3378	-0,1764	0,171741	-0,5217	0,1689
1Q by 4L	0,3639	0,343482	1,059	0,294714	-0,3267	1,0545	0,1819	0,171741	-0,1634	0,5273
1Q by 4Q	0,0403	0,297464	0,135	0,892859	-0,5578	0,6384	0,0201	0,148732	-0,2789	0,3192
2L by 3L	-5,0389	0,396619	-12,705	0,000000	-5,8363	-4,2414	-2,5194	0,198309	-2,9182	-2,1207
2L by 3Q	-0,8528	0,343482	-2,483	0,016586	-1,5434	-0,1622	-0,4264	0,171741	-0,7717	-0,0811
2Q by 3L	-1,0750	0,343482	-3,130	0,002975	-1,7656	-0,3844	-0,5375	0,171741	-0,8828	-0,1922
2Q by 3Q	-1,1764	0,297464	-3,955	0,000252	-1,7745	-0,5783	-0,5882	0,148732	-0,8872	-0,2891
2L by 4L	-3,0111	0,396619	-7,592	0,000000	-3,8086	-2,2137	-1,5056	0,198309	-1,9043	-1,1068
2L by 4Q	0,1389	0,343482	0,404	0,687747	-0,5517	0,8295	0,0694	0,171741	-0,2759	0,4148
2Q by 4L	-0,6278	0,343482	-1,828	0,073816	-1,3184	0,0628	-0,3139	0,171741	-0,6592	0,0314
2Q by 4Q	-0,0806	0,297464	-0,271	0,787700	-0,6786	0,5175	-0,0403	0,148732	-0,3393	0,2588
3L by 4L	1,7000	0,396619	4,286	0,000087	0,9025	2,4975	0,8500	0,198309	0,4513	1,2487
3L by 4Q	0,0333	0,343482	0,097	0,923094	-0,6573	0,7239	0,0167	0,171741	-0,3286	0,3620
3Q by 4L	0,1889	0,343482	0,550	0,584921	-0,5017	0,8795	0,0944	0,171741	-0,2509	0,4398
3Q by 4Q	0,0694	0,297464	0,233	0,816401	-0,5286	0,6675	0,0347	0,148732	-0,2643	0,3338

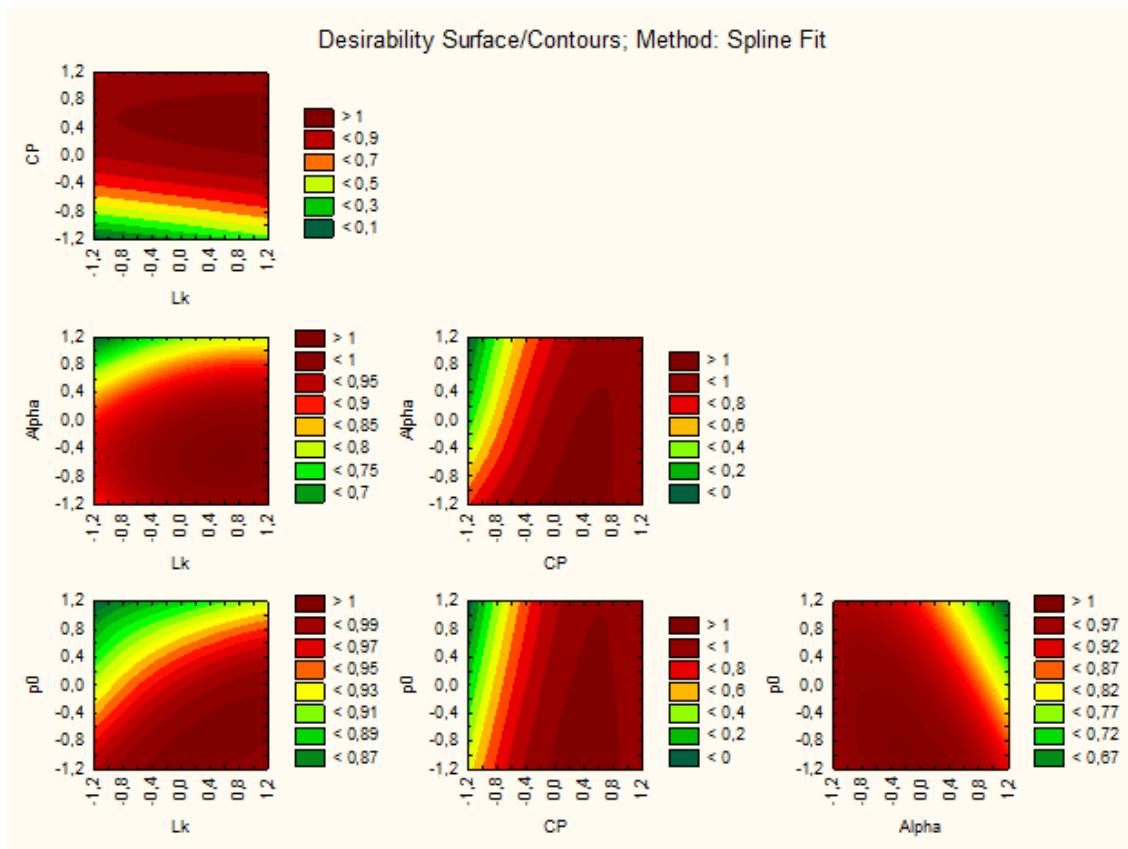


Figura A.5: Gráfico de contorno das quatro variáveis relevantes (L_k, CP, α, p_0), relativamente ao *makespan*, do Modelo 3 do problema de *makespan*

APÊNDICE A. TABELAS

Tabela A.6: Effect Estimates; Var.:Tempo; R-sqr=0,97945; Adj:0,96576; 4 3-level factors, 1 Blocks, 81 Runs; MS Residual=10,33885; DV: Tempo. Modelo 3, problema de *makespan*

Factors	Effect	Std.Err	t(48)	p	-95% Cnf.Limt	+95% Cnf.Limt	Coeff.	Std.Err.Coeff	-95% Cnf.Limt	+95% Cnf.Limt
Mean/Interc.	18,67782	0,357268	52,27963	0,000000	17,95948	19,39615	18,67782	0,357268	17,95948	19,39615
(1)Lk (L)	23,68873	0,875123	27,06902	0,000000	21,92917	25,44828	11,84436	0,437562	10,96459	12,72414
Lk (Q)	0,10397	0,757879	0,13718	0,891460	-1,41985	1,62779	0,05198	0,378939	-0,70993	0,81389
(2)CP (L)	28,05016	0,875123	32,05281	0,000000	26,29061	29,80971	14,02508	0,437562	13,14530	14,90486
CP (Q)	2,21749	0,757879	2,92592	0,005233	0,69368	3,74131	1,10875	0,378939	0,34684	1,87066
(3)Alpha (L)	8,28376	0,875123	9,46582	0,000000	6,52421	10,04331	4,14188	0,437562	3,26210	5,02166
Alpha (Q)	-3,11745	0,757879	-4,11338	0,000152	-4,64127	-1,59363	-1,55872	0,378939	-2,32063	-0,79682
(4)p0 (L)	2,35684	0,875123	2,69316	0,009720	0,59729	4,11640	1,17842	0,437562	0,29865	2,05820
p0 (Q)	-0,32421	0,757879	-0,42778	0,670724	-1,84802	1,19961	-0,16210	0,378939	-0,92401	0,59981
1L by 2L	17,15184	1,071803	16,00280	0,000000	14,99684	19,30685	8,57592	0,535901	7,49842	9,65342
1L by 2Q	1,94505	0,928208	2,09549	0,041427	0,07876	3,81134	0,97252	0,464104	0,03938	1,90567
1Q by 2L	0,34062	0,928208	0,36697	0,715257	-1,52567	2,20691	0,17031	0,464104	-0,76283	1,10345
1Q by 2Q	-0,27891	0,803852	-0,34697	0,730133	-1,89516	1,33734	-0,13945	0,401926	-0,94758	0,66867
1L by 3L	6,60764	1,071803	6,16497	0,000000	4,45263	8,76264	3,30382	0,535901	2,22632	4,38132
1L by 3Q	-2,15015	0,928208	-2,31645	0,024848	-4,01644	-0,28386	-1,07508	0,464104	-2,00822	-0,14193
1Q by 3L	-0,57409	0,928208	-0,61850	0,539172	-2,44038	1,29219	-0,28705	0,464104	-1,22019	0,64610
1Q by 3Q	-0,38687	0,803852	-0,48127	0,632512	-2,00312	1,22938	-0,19343	0,401926	-1,00156	0,61469
1L by 4L	1,69692	1,071803	1,58324	0,119934	-0,45808	3,85193	0,84846	0,535901	-0,22904	1,92596
1L by 4Q	-0,30746	0,928208	-0,33124	0,741905	-2,17375	1,55883	-0,15373	0,464104	-1,08687	0,77941
1Q by 4L	0,10229	0,928208	0,11020	0,912713	-1,76400	1,96857	0,05114	0,464104	-0,88200	0,98429
1Q by 4Q	-0,01460	0,803852	-0,01816	0,985586	-1,63085	1,60165	-0,00730	0,401926	-0,81543	0,80083
2L by 3L	9,26165	1,071803	8,64119	0,000000	7,10665	11,41665	4,63082	0,535901	3,55332	5,70833
2L by 3Q	-3,39010	0,928208	-3,65231	0,000642	-5,25639	-1,52381	-1,69505	0,464104	-2,62819	-0,76191
2Q by 3L	0,71289	0,928208	0,76803	0,446234	-1,15340	2,57918	0,35644	0,464104	-0,57670	1,28959
2Q by 3Q	0,30248	0,803852	0,37629	0,708356	-1,31377	1,91874	0,15124	0,401926	-0,65688	0,95937
2L by 4L	2,67906	1,071803	2,49958	0,015906	0,52406	4,83407	1,33953	0,535901	0,26203	2,41703
2L by 4Q	-0,68407	0,928208	-0,73698	0,464724	-2,55036	1,18222	-0,34203	0,464104	-1,27518	0,59111
2Q by 4L	-0,22473	0,928208	-0,24211	0,809723	-2,09102	1,64155	-0,11237	0,464104	-1,04551	0,82078
2Q by 4Q	0,39673	0,803852	0,49354	0,623881	-1,21952	2,01299	0,19837	0,401926	-0,60976	1,00649
3L by 4L	1,37885	1,071803	1,28648	0,204446	-0,77615	3,53386	0,68943	0,535901	-0,38807	1,76693
3L by 4Q	0,29705	0,928208	0,32002	0,750341	-1,56924	2,16333	0,14852	0,464104	-0,78462	1,08167
3Q by 4L	-0,68893	0,928208	-0,74222	0,461572	-2,55522	1,17735	-0,34447	0,464104	-1,27761	0,58868
3Q by 4Q	-0,32955	0,803852	-0,40996	0,683656	-1,94580	1,28670	-0,16478	0,401926	-0,97290	0,64335

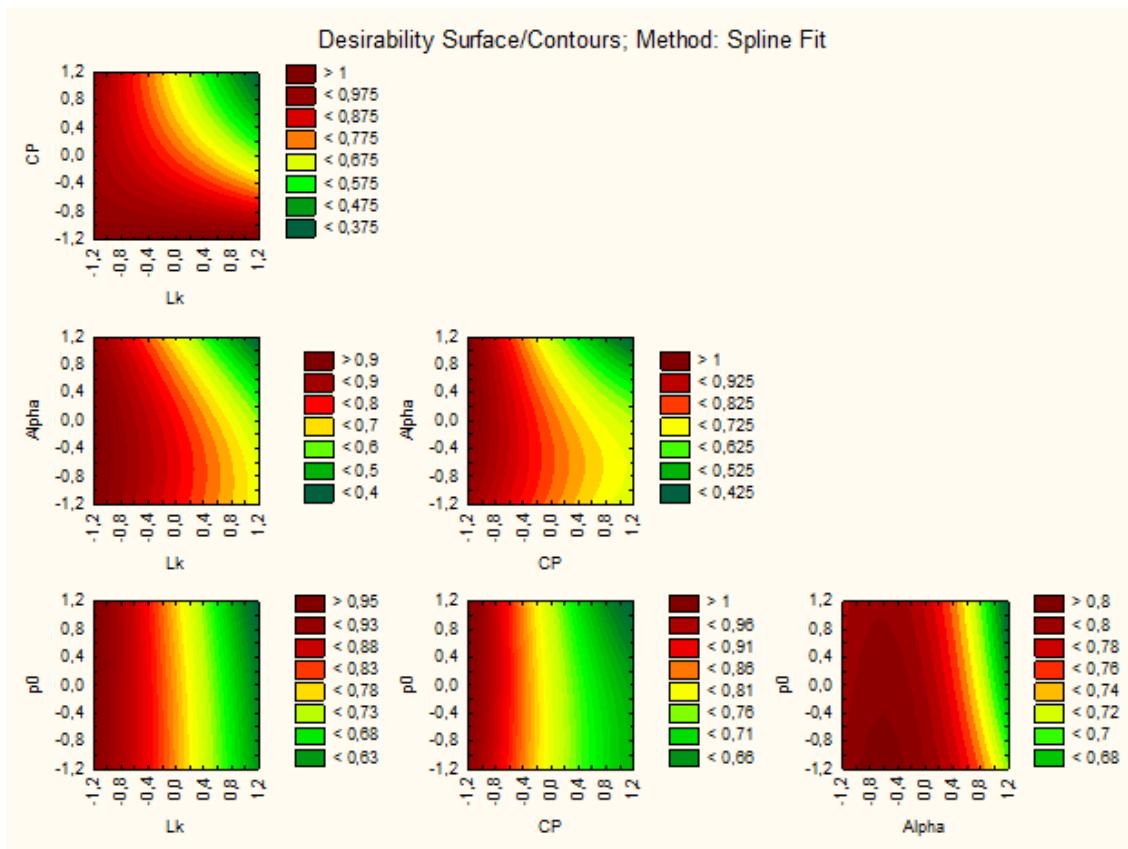


Figura A.6: Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao tempo, do Modelo 3 do problema de makespan

A.2 Problema do número de trabalhos

Tabela A.7: Effect Estimates; Var.:Trabalhos; R-sqr=.89764; Adj.:.87098; 5 3-level factors, 1 Blocks, 243 Runs; MS Residual=4,373206; DV: Trabalhos. Modelo 1, problema do número de trabalhos

Factors	Effect	Std.Err	t(192)	p	-95,%Cnf.Limt	+95,%Cnf.Limt	Coeff.	Std.Err.Coeff.	-95,%Cnf.Limt	+95,%Cnf.Limt
Mean/Interc.	-28,6741	0,134152	-213,743	0,000000	-28,9387	-28,4095	-28,6741	0,134152	-28,9387	-28,4095
(1)Lk (L)	-0,6963	0,328604	-2,119	0,035380	-1,3444	-0,0482	-0,3481	0,164302	-0,6722	-0,0241
Lk (Q)	-0,1852	0,284579	-0,651	0,515997	-0,7465	0,3761	-0,0926	0,142290	-0,3732	0,1881
(2)CP (L)	-6,6815	0,328604	-20,333	0,000000	-7,3296	-6,0333	-3,3407	0,164302	-3,6648	-3,0167
CP (Q)	-0,6630	0,284579	-2,330	0,020866	-1,2243	-0,1017	-0,3315	0,142290	-0,6121	-0,0508
(3)Alpha (L)	8,2383	0,328604	25,071	0,000000	7,5901	8,8864	4,1191	0,164302	3,7951	4,4432
Alpha (Q)	-3,1907	0,284579	-11,212	0,000000	-3,7520	-2,6294	-1,5954	0,142290	-1,8760	-1,3147
(4)P (L)	2,4815	0,328604	7,552	0,000000	1,8333	3,1296	1,2407	0,164302	0,9167	1,5648
P (Q)	0,2815	0,284579	0,989	0,323852	-0,2798	0,8428	0,1407	0,142290	-0,1399	0,4214
(5)p0 (L)	1,3679	0,328604	4,163	0,000047	0,7198	2,0160	0,6840	0,164302	0,3599	1,0080
p0 (Q)	0,0222	0,284579	0,078	0,937839	-0,5391	0,5835	0,0111	0,142290	-0,2695	0,2918
1L by 2L	-0,3278	0,402456	-0,814	0,416399	-1,1216	0,4660	-0,1639	0,201228	-0,5608	0,2330
1L by 2Q	-0,0250	0,348537	-0,072	0,942893	-0,7125	0,6625	-0,0125	0,174268	-0,3562	0,3312
1Q by 2L	0,0361	0,348537	0,104	0,917589	-0,6513	0,7236	0,0181	0,174268	-0,3257	0,3618
1Q by 2Q	-0,0347	0,301842	-0,115	0,908538	-0,6301	0,5606	-0,0174	0,150921	-0,3150	0,2803
1L by 3L	0,2241	0,402456	0,557	0,578335	-0,5697	1,0179	0,1120	0,201228	-0,2849	0,5089
1L by 3Q	0,2861	0,348537	0,821	0,412726	-0,4013	0,9736	0,1431	0,174268	-0,2007	0,4868
1Q by 3L	0,0546	0,348537	0,157	0,875615	-0,6328	0,7421	0,0273	0,174268	-0,3164	0,3710
1Q by 3Q	-0,0431	0,301842	-0,143	0,886722	-0,6384	0,5523	-0,0215	0,150921	-0,3192	0,2761
1L by 4L	0,2889	0,402456	0,718	0,473744	-0,5049	1,0827	0,1444	0,201228	-0,2525	0,5413
1L by 4Q	-0,1444	0,348537	-0,414	0,679021	-0,8319	0,5430	-0,0722	0,174268	-0,4159	0,2715
1Q by 4L	-0,0389	0,348537	-0,112	0,911275	-0,7263	0,6486	-0,0194	0,174268	-0,3632	0,3243
1Q by 4Q	0,1694	0,301842	0,561	0,575201	-0,4259	0,7648	0,0847	0,150921	-0,2130	0,3824
1L by 5L	0,2907	0,402456	0,722	0,470917	-0,5031	1,0845	0,1454	0,201228	-0,2515	0,5423
1L by 5Q	-0,1361	0,348537	-0,391	0,696584	-0,8236	0,5513	-0,0681	0,174268	-0,4118	0,2757
1Q by 5L	0,1102	0,348537	0,316	0,752242	-0,5773	0,7976	0,0551	0,174268	-0,2886	0,3988
1Q by 5Q	0,2514	0,301842	0,833	0,405964	-0,3440	0,8467	0,1257	0,150921	-0,1720	0,4234
2L by 3L	-7,0000	0,402456	-17,393	0,000000	-7,7938	-6,2062	-3,5000	0,201228	-3,8969	-3,1031
2L by 3Q	2,0278	0,348537	5,818	0,000000	1,3403	2,7152	1,0139	0,174268	0,6702	1,3576
2Q by 3L	0,0463	0,348537	0,133	0,894467	-0,6412	0,7337	0,0231	0,174268	-0,3206	0,3669
2Q by 3Q	-0,7556	0,301842	-2,503	0,013143	-1,3509	-0,1602	-0,3778	0,150921	-0,6755	-0,0801
2L by 4L	-0,4500	0,402456	-1,118	0,264906	-1,2438	0,3438	-0,2250	0,201228	-0,6219	0,1719
2L by 4Q	-0,0194	0,348537	-0,056	0,955568	-0,7069	0,6680	-0,0097	0,174268	-0,3534	0,3340
2Q by 4L	1,3972	0,348537	4,009	0,000087	0,7098	2,0847	0,6986	0,174268	0,3549	1,0423
2Q by 4Q	0,4403	0,301842	1,459	0,146299	-0,1551	1,0356	0,2201	0,150921	-0,0775	0,5178
2L by 5L	-0,3537	0,402456	-0,879	0,380574	-1,1475	0,4401	-0,1769	0,201228	-0,5738	0,2200
2L by 5Q	-0,0194	0,348537	-0,056	0,955568	-0,7069	0,6680	-0,0097	0,174268	-0,3534	0,3340
2Q by 5L	0,4435	0,348537	1,273	0,204729	-0,2439	1,1310	0,2218	0,174268	-0,1220	0,5655
2Q by 5Q	0,0597	0,301842	0,198	0,843364	-0,5356	0,6551	0,0299	0,150921	-0,2678	0,3275
3L by 4L	2,6019	0,402456	6,465	0,000000	1,8080	3,3957	1,3009	0,201228	0,9040	1,6978
3L by 4Q	0,3491	0,348537	1,002	0,317826	-0,3384	1,0365	0,1745	0,174268	-0,1692	0,5183
3Q by 4L	-0,3250	0,348537	-0,932	0,352265	-1,0125	0,3625	-0,1625	0,174268	-0,5062	0,1812
3Q by 4Q	-0,2514	0,301842	-0,833	0,405964	-0,8467	0,3440	-0,1257	0,150921	-0,4234	0,1720
3L by 5L	1,4185	0,402456	3,525	0,000530	0,6247	2,2123	0,7093	0,201228	0,3124	1,1062
3L by 5Q	0,1074	0,348537	0,308	0,758290	-0,5800	0,7949	0,0537	0,174268	-0,2900	0,3974
3Q by 5L	0,0185	0,348537	0,053	0,957682	-0,6689	0,7060	0,0093	0,174268	-0,3345	0,3530
3Q by 5Q	-0,0194	0,301842	-0,064	0,948703	-0,6148	0,5759	-0,0097	0,150921	-0,3074	0,2880
4L by 5L	0,6593	0,402456	1,638	0,103040	-0,1345	1,4531	0,3296	0,201228	-0,0673	0,7265
4L by 5Q	0,1722	0,348537	0,494	0,621780	-0,5152	0,8597	0,0861	0,174268	-0,2576	0,4298
4Q by 5L	0,4574	0,348537	1,312	0,190964	-0,2300	1,1449	0,2287	0,174268	-0,1150	0,5724
4Q by 5Q	0,0556	0,301842	0,184	0,854164	-0,5398	0,6509	0,0278	0,150921	-0,2699	0,3255

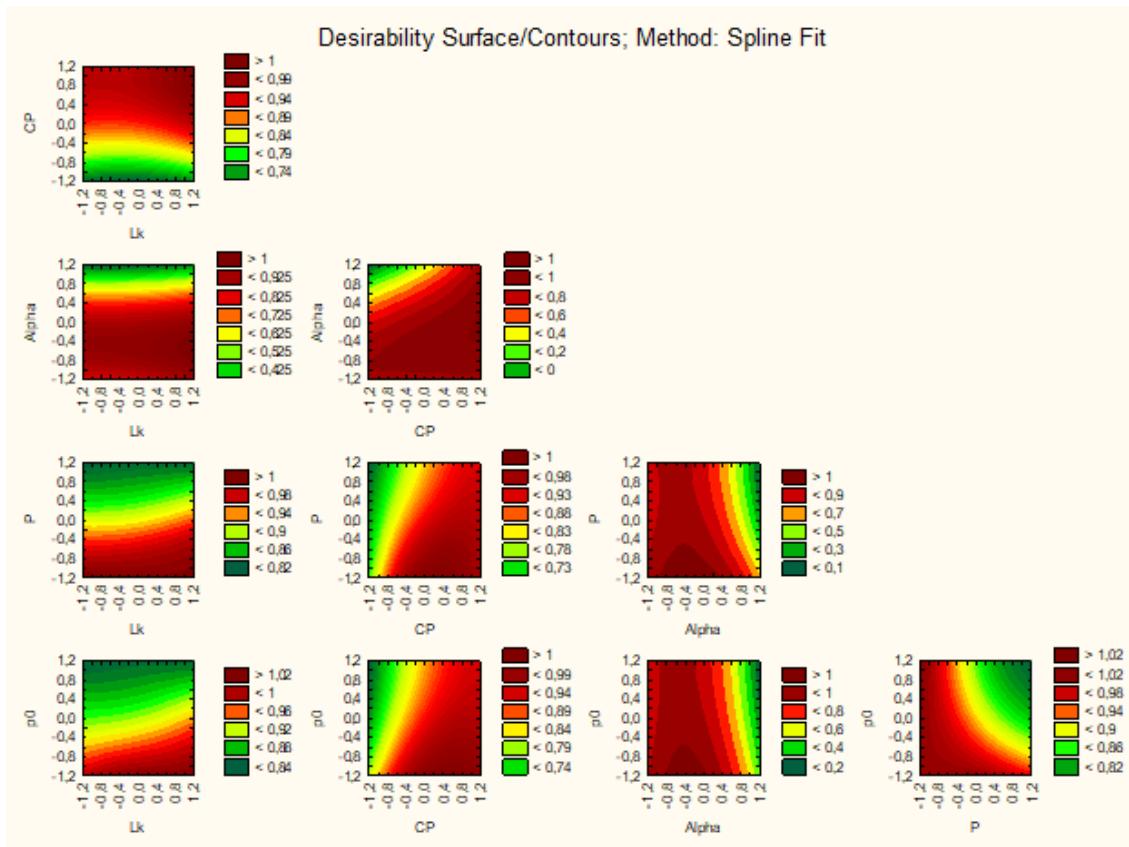


Figura A.7: Gráfico de contorno das cinco variáveis relevantes (L_k , CP , α , P , p_0), relativamente ao número de exames, do Modelo 1 do problema do número de trabalhos

APÊNDICE A. TABELAS

Tabela A.8: Effect Estimates; Var.:Tempo; R-sqr=,96619; Adj:,95739; 5 3-level factors, 1 Blocks, 243 Runs; MS Residual=15,01405; DV: Tempo. Modelo 1, problema do número de trabalhos

Factors	Effect	Std.Err	t(192)	P	-95.%Cnf.Limt	+95.%Cnf.Limt	Coeff.	Std.Err.Coeff.	-95.%Cnf.Limt	+95.%Cnf.Limt
Mean/Interc.	21,95706	0,248568	88,33411	0,000000	21,46679	22,44734	21,95706	0,248568	21,46679	22,44734
(1)Lk (L)	27,86540	0,608866	45,76610	0,000000	26,66448	29,06633	13,93270	0,304433	13,33224	14,53316
Lk (Q)	0,25520	0,527293	0,48399	0,628947	-0,78483	1,29523	0,12760	0,263647	-0,39241	0,64762
(2)CP (L)	30,05783	0,608866	49,36694	0,000000	28,85691	31,25876	15,02892	0,304433	14,42845	15,62938
CP (Q)	-0,60802	0,527293	-1,15310	0,250303	-1,64805	0,43201	-0,30401	0,263647	-0,82403	0,21600
(3)Alpha (L)	1,40762	0,608866	2,31187	0,021846	0,20669	2,60854	0,70381	0,304433	0,10335	1,30427
Alpha (Q)	1,09742	0,527293	2,08123	0,038740	0,05739	2,13745	0,54871	0,263647	0,02869	1,06872
(4)P (L)	-3,04670	0,608866	-5,00389	0,000001	-4,24762	-1,84577	-1,52335	0,304433	-2,12381	-0,92289
P (Q)	-0,45522	0,527293	-0,86332	0,389041	-1,49525	0,58481	-0,22761	0,263647	-0,74763	0,29241
(5)p0 (L)	-1,10053	0,608866	-1,80751	0,072249	-2,30145	0,10040	-0,55026	0,304433	-1,15073	0,05020
p0 (Q)	-0,22992	0,527293	-0,43605	0,663292	-1,26996	0,81011	-0,11496	0,263647	-0,63498	0,40505
1L by 2L	19,56619	0,745705	26,23851	0,000000	18,09536	21,03701	9,78309	0,372853	9,04768	10,51851
1L by 2Q	-0,52683	0,645799	-0,81578	0,415634	-1,80061	0,74694	-0,26342	0,322900	-0,90030	0,37347
1Q by 2L	0,14028	0,645799	0,21721	0,828272	-1,13350	1,41405	0,07014	0,322900	-0,56675	0,70702
1Q by 2Q	-0,03114	0,559279	-0,05569	0,955650	-1,13426	1,07198	-0,01557	0,279639	-0,56713	0,53599
1L by 3L	1,21568	0,745705	1,63024	0,104691	-0,25515	2,68650	0,60784	0,372853	-0,12757	1,34325
1L by 3Q	0,77665	0,645799	1,20261	0,230608	-0,49713	2,05042	0,38832	0,322900	-0,24856	1,02521
1Q by 3L	0,16311	0,645799	0,25257	0,800869	-1,11066	1,43688	0,08156	0,322900	-0,55533	0,71844
1Q by 3Q	-0,41793	0,559279	-0,74726	0,455820	-1,52105	0,68519	-0,20896	0,279639	-0,76052	0,34260
1L by 4L	-2,33837	0,745705	-3,13578	0,001983	-3,80919	-0,86754	-1,16918	0,372853	-1,90460	-0,43377
1L by 4Q	-0,15421	0,645799	-0,23880	0,811519	-1,42799	1,11956	-0,07711	0,322900	-0,71399	0,55978
1Q by 4L	0,14415	0,645799	0,22321	0,823609	-1,12962	1,41792	0,07207	0,322900	-0,56481	0,70896
1Q by 4Q	-0,39463	0,559279	-0,70561	0,481289	-1,49775	0,70849	-0,19732	0,279639	-0,74887	0,35424
1L by 5L	-1,04255	0,745705	-1,39807	0,163706	-2,51337	0,42828	-0,52127	0,372853	-1,25669	0,21414
1L by 5Q	0,28096	0,645799	0,43506	0,664010	-0,99281	1,55473	0,14048	0,322900	-0,49641	0,77737
1Q by 5L	0,00559	0,645799	0,08666	0,993099	-1,26818	1,27937	0,00280	0,322900	-0,63409	0,63968
1Q by 5Q	-0,44403	0,559279	-0,79394	0,428213	-1,54715	0,65909	-0,22202	0,279639	-0,77358	0,32954
2L by 3L	7,77354	0,745705	10,42442	0,000000	6,30271	9,24436	3,88677	0,372853	3,15136	4,62218
2L by 3Q	-2,08694	0,645799	-3,23157	0,001449	-3,36072	-0,81317	-1,04347	0,322900	-1,68036	-0,40659
2Q by 3L	-1,82630	0,645799	-2,82796	0,005181	-3,10007	-0,55252	-0,91315	0,322900	-1,55003	-0,27626
2Q by 3Q	1,57498	0,559279	2,81608	0,005369	0,47186	2,67810	0,78749	0,279639	0,23593	1,33905
2L by 4L	-0,78616	0,745705	-1,05426	0,293091	-2,25699	0,68466	-0,39308	0,372853	-1,12849	0,34233
2L by 4Q	-0,01159	0,645799	-0,01795	0,985700	-1,28536	1,26218	-0,00580	0,322900	-0,64268	0,63109
2Q by 4L	-1,89371	0,645799	-2,93235	0,003773	-3,16748	-0,61993	-0,94685	0,322900	-1,58374	-0,30997
2Q by 4Q	-0,39480	0,559279	-0,70591	0,481098	-1,49792	0,70832	-0,19740	0,279639	-0,74896	0,35416
2L by 5L	-0,22339	0,745705	-0,29957	0,764827	-1,69422	1,24743	-0,11170	0,372853	-0,84711	0,62372
2L by 5Q	-0,16161	0,645799	-0,25024	0,802668	-1,43538	1,11217	-0,08080	0,322900	-0,71769	0,55608
2Q by 5L	-0,73492	0,645799	-1,13801	0,256536	-2,00870	0,53885	-0,36746	0,322900	-1,00435	0,26942
2Q by 5Q	-0,06106	0,559279	-0,10917	0,913183	-1,16417	1,04206	-0,03053	0,279639	-0,58209	0,52103
3L by 4L	-4,31860	0,745705	-5,79129	0,000000	-5,78942	-2,84777	-2,15930	0,372853	-2,89471	-1,42389
3L by 4Q	-0,48524	0,645799	-0,75138	0,453342	-1,75902	0,78853	-0,24262	0,322900	-0,87951	0,39427
3Q by 4L	2,12627	0,645799	3,29246	0,001182	0,85250	3,40004	1,06314	0,322900	0,42625	1,70002
3Q by 4Q	0,23770	0,559279	0,42501	0,671302	-0,86542	1,34082	0,11885	0,279639	-0,43271	0,67041
3L by 5L	-2,28258	0,745705	-3,06096	0,002522	-3,75340	-0,81175	-1,14129	0,372853	-1,87670	-0,40588
3L by 5Q	-0,71141	0,645799	-1,10160	0,272014	-1,98519	0,56236	-0,35571	0,322900	-0,99259	0,28118
3Q by 5L	0,98076	0,645799	1,51868	0,130489	-0,29301	2,25453	0,49038	0,322900	-0,14651	1,12727
3Q by 5Q	0,07000	0,559279	0,12517	0,900521	-1,03312	1,17312	0,03500	0,279639	-0,51656	0,58656
4L by 5L	-0,89137	0,745705	-1,19534	0,233429	-2,36220	0,57946	-0,44568	0,372853	-1,18110	0,28973
4L by 5Q	-0,21557	0,645799	-0,33380	0,738896	-1,48934	1,05821	-0,10778	0,322900	-0,74467	0,52910
4Q by 5L	-0,72168	0,645799	-1,11750	0,265178	-1,99545	0,55209	-0,36084	0,322900	-0,99773	0,27605
4Q by 5Q	-0,32942	0,559279	-0,58901	0,556546	-1,43254	0,77370	-0,16471	0,279639	-0,71627	0,38685

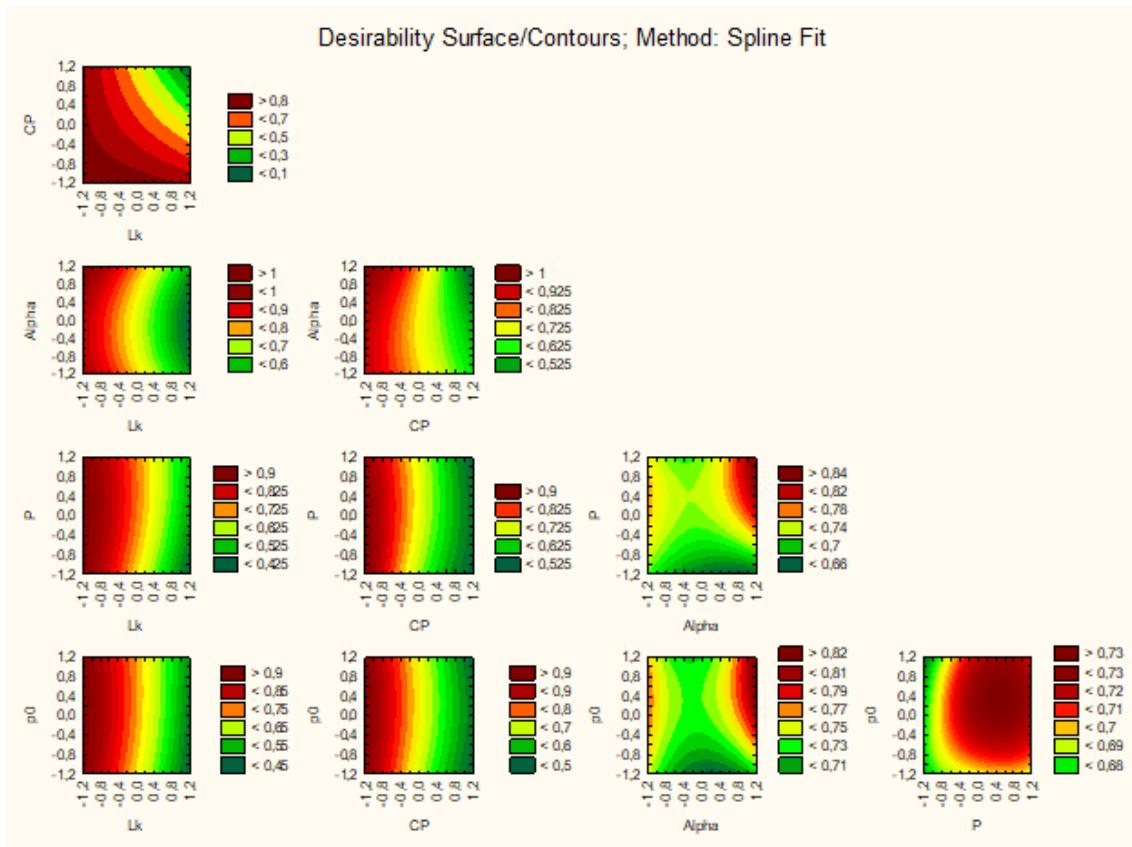


Figura A.8: Gráfico de contorno das cinco variáveis relevantes (L_k, CP, α, P, p_0), relativamente ao tempo, do Modelo 1 do problema do número de trabalhos

APÊNDICE A. TABELAS

Tabela A.9: Effect Estimates; Var.:Trabalhos; R-sqr=.9682; Adj.:.947; 4 3-level factors, 1 Blocks, 81 Runs; MS Residual=.1508179; DV: Trabalhos. Modelo 2, problema do número de trabalhos

Factors	Effect	Std.Err	t(48)	p	-95.%Cnf.Limt	+95.%Cnf.Limt	Coeff.	Std.Err.Coeff.	-95.%Cnf.Limt	+95.%Cnf.Limt
Mean/Interc.	-32,3951	0,043150	-750,749	0,000000	-32,4818	-32,3083	-32,3951	0,043150	-32,4818	-32,3083
(1)Lk (L)	-1,3667	0,105696	-12,930	0,000000	-1,5792	-1,1542	-0,6833	0,052848	-0,7896	-0,5771
Lk (Q)	-0,3019	0,091536	-3,298	0,001841	-0,4859	-0,1178	-0,1509	0,045768	-0,2429	-0,0589
(2)CP (L)	-2,6074	0,105696	-24,669	0,000000	-2,8199	-2,3949	-1,3037	0,052848	-1,4100	-1,1974
CP (Q)	-1,0630	0,091536	-11,613	0,000000	-1,2470	-0,8789	-0,5315	0,045768	-0,6235	-0,4395
(3)Alpha (L)	1,1296	0,105696	10,688	0,000000	0,9171	1,3421	0,5648	0,052848	0,4586	0,6711
Alpha (Q)	-0,5019	0,091536	-5,483	0,000002	-0,6859	-0,3178	-0,2509	0,045768	-0,3429	-0,1589
(4)p0 (L)	0,3481	0,105696	3,294	0,001861	0,1356	0,5607	0,1741	0,052848	0,0678	0,2803
p0 (Q)	-0,1407	0,091536	-1,538	0,130725	-0,3248	0,0433	-0,0704	0,045768	-0,1624	0,0217
1L by 2L	0,1833	0,129451	1,416	0,163163	-0,0769	0,4436	0,0917	0,064725	-0,0385	0,2218
1L by 2Q	-0,0083	0,112108	-0,074	0,941054	-0,2337	0,2171	-0,0042	0,056054	-0,1169	0,1085
1Q by 2L	0,1139	0,112108	1,016	0,314777	-0,1115	0,3393	0,0569	0,056054	-0,0558	0,1696
1Q by 2Q	0,0181	0,097088	0,186	0,853251	-0,1772	0,2133	0,0090	0,048544	-0,0886	0,1066
1L by 3L	0,1611	0,129451	1,245	0,219332	-0,0992	0,4214	0,0806	0,064725	-0,0496	0,2107
1L by 3Q	0,0250	0,112108	0,223	0,824482	-0,2004	0,2504	0,0125	0,056054	-0,1002	0,1252
1Q by 3L	0,0694	0,112108	0,619	0,538554	-0,1560	0,2949	0,0347	0,056054	-0,0780	0,1474
1Q by 3Q	0,0347	0,097088	0,358	0,722184	-0,1605	0,2299	0,0174	0,048544	-0,0802	0,1150
1L by 4L	-0,0111	0,129451	-0,086	0,931957	-0,2714	0,2492	-0,0056	0,064725	-0,1357	0,1246
1L by 4Q	0,0500	0,112108	0,446	0,657603	-0,1754	0,2754	0,0250	0,056054	-0,0877	0,1377
1Q by 4L	-0,1778	0,112108	-1,586	0,119357	-0,4032	0,0476	-0,0889	0,056054	-0,2016	0,0238
1Q by 4Q	-0,0611	0,097088	-0,629	0,532046	-0,2563	0,1341	-0,0306	0,048544	-0,1282	0,0670
2L by 3L	-1,8722	0,129451	-14,463	0,000000	-2,1325	-1,6119	-0,9361	0,064725	-1,0663	-0,8060
2L by 3Q	0,6472	0,112108	5,773	0,000001	0,4218	0,8726	0,3236	0,056054	0,2109	0,4363
2Q by 3L	-0,9806	0,112108	-8,747	0,000000	-1,2060	-0,7551	-0,4903	0,056054	-0,6030	-0,3776
2Q by 3Q	0,3181	0,097088	3,276	0,001960	0,1228	0,5133	0,1590	0,048544	0,0614	0,2566
2L by 4L	-0,6222	0,129451	-4,807	0,000016	-0,8825	-0,3619	-0,3111	0,064725	-0,4413	-0,1810
2L by 4Q	0,0556	0,112108	0,496	0,622470	-0,1699	0,2810	0,0278	0,056054	-0,0849	0,1405
2Q by 4L	-0,2778	0,112108	-2,478	0,016791	-0,5032	-0,0524	-0,1389	0,056054	-0,2516	-0,0262
2Q by 4Q	0,0222	0,097088	0,229	0,819929	-0,1730	0,2174	0,0111	0,048544	-0,0865	0,1087
3L by 4L	0,4500	0,129451	3,476	0,001090	0,1897	0,7103	0,2250	0,064725	0,0949	0,3551
3L by 4Q	0,0528	0,112108	0,471	0,639932	-0,1726	0,2782	0,0264	0,056054	-0,0863	0,1391
3Q by 4L	-0,1361	0,112108	-1,214	0,230646	-0,3615	0,0893	-0,0681	0,056054	-0,1808	0,0446
3Q by 4Q	-0,0986	0,097088	-1,016	0,314871	-0,2938	0,0966	-0,0493	0,048544	-0,1469	0,0483

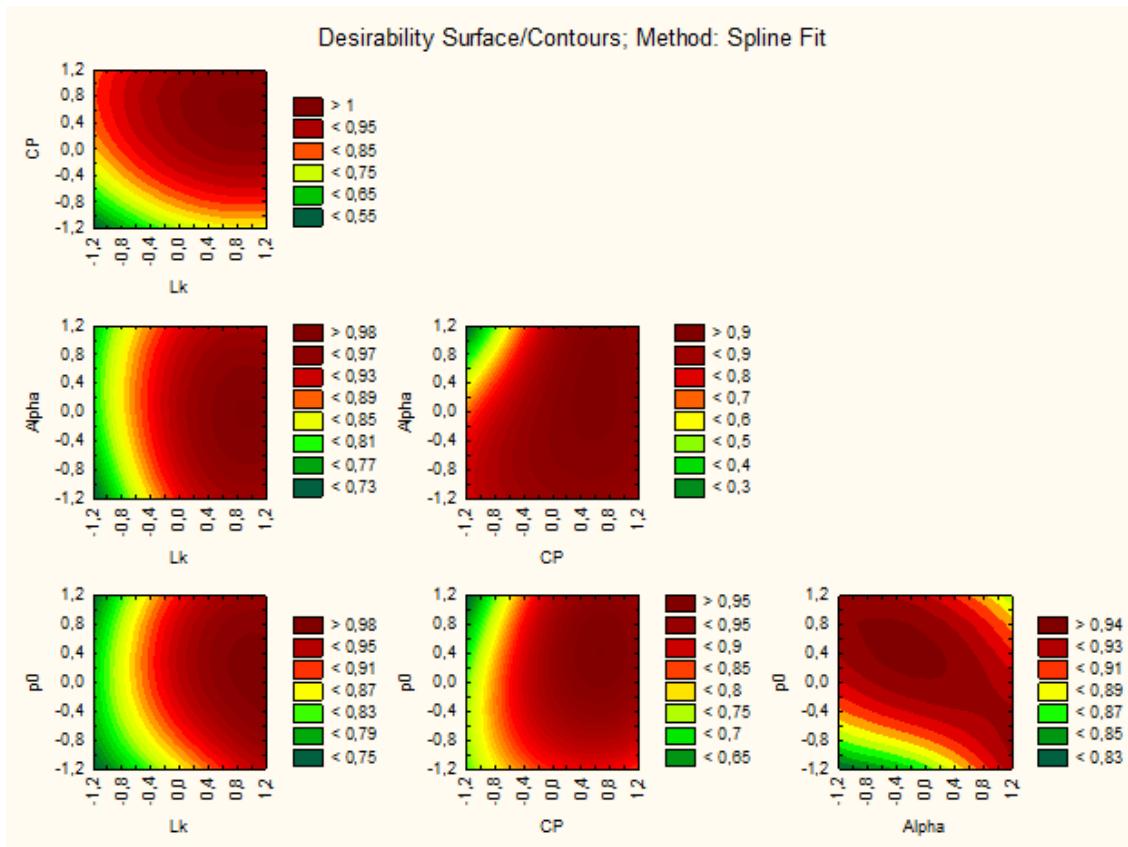


Figura A.9: Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao número de trabalhos, do Modelo 2 do problema do número de trabalhos

APÊNDICE A. TABELAS

Tabela A.10: Effect Estimates; Var.:Tempo; R-sqr=,96531; Adj.,94218; 4 3-level factors, 1 Blocks, 81 Runs; MS Residual=4,781419; DV: Tempo. Modelo 2, problema do número de trabalhos

Factors	Effect	Std.Err	t(48)	p	-95,%Cnf.Limt	+95,%Cnf.Limt	Coeff.	Std.Err.Coeff.	-95,%Cnf.Limt	+95,%Cnf.Limt
Mean/Interc.	8,65433	0,242961	35,62029	0,000000	8,16582	9,14283	8,654328	0,242961	8,16582	9,142833
(1)Lk (L)	14,24014	0,595130	23,92780	0,000000	13,04356	15,43673	7,120072	0,297565	6,52178	7,718366
Lk (Q)	-0,97820	0,515397	-1,89794	0,063726	-2,01447	0,05808	-0,489098	0,257699	-1,00724	0,029040
(2)CP (L)	11,24531	0,595130	18,89557	0,000000	10,04873	12,44190	5,622657	0,297565	5,02436	6,220951
CP (Q)	1,38678	0,515397	2,69071	0,009782	0,35051	2,42306	0,693392	0,257699	0,17525	1,211530
(3)Alpha (L)	4,87821	0,595130	8,19689	0,000000	3,68162	6,07480	2,439105	0,297565	1,84081	3,037399
Alpha (Q)	-1,40244	0,515397	-2,72109	0,009037	-2,43872	-0,36616	-0,701220	0,257699	-1,21936	-0,183082
(4)p0 (L)	3,31278	0,595130	5,56648	0,000001	2,11619	4,50936	1,656388	0,297565	1,05809	2,254682
p0 (Q)	-0,65114	0,515397	-1,26338	0,212554	-1,68742	0,38513	-0,325572	0,257699	-0,84371	0,192566
1L by 2L	9,44954	0,728882	12,96443	0,000000	7,98402	10,91505	4,724768	0,364441	3,99201	5,457525
1L by 2Q	1,23266	0,631230	1,95280	0,056688	-0,03651	2,50184	0,616332	0,315615	-0,01825	1,250918
1Q by 2L	-0,64606	0,631230	-1,02349	0,311207	-1,91523	0,62312	-0,323029	0,315615	-0,95762	0,311558
1Q by 2Q	-0,15408	0,546661	-0,28185	0,779268	-1,25321	0,94506	-0,077039	0,273331	-0,62661	0,472529
1L by 3L	4,33044	0,728882	5,94121	0,000000	2,86492	5,79596	2,165220	0,364441	1,43246	2,897978
1L by 3Q	-1,26857	0,631230	-2,00967	0,050105	-2,53774	0,00061	-0,634283	0,315615	-1,26887	0,000303
1Q by 3L	-0,26415	0,631230	-0,41847	0,677472	-1,53332	1,00502	-0,132075	0,315615	-0,76666	0,502512
1Q by 3Q	0,20539	0,546661	0,37571	0,708787	-0,89375	1,30452	0,102693	0,273331	-0,44687	0,652262
1L by 4L	2,56135	0,728882	3,51408	0,0000973	1,09583	4,02686	1,280673	0,364441	0,54792	2,013431
1L by 4Q	-0,61920	0,631230	-0,98095	0,331538	-1,88838	0,64997	-0,309601	0,315615	-0,94419	0,324986
1Q by 4L	0,08346	0,631230	0,13222	0,895360	-1,18571	1,35264	0,041732	0,315615	-0,59285	0,676319
1Q by 4Q	0,14914	0,546661	0,27282	0,786159	-0,94999	1,24828	0,074571	0,273331	-0,47500	0,624139
2L by 3L	3,42415	0,728882	4,69781	0,000022	1,95864	4,88967	1,712076	0,364441	0,97932	2,444834
2L by 3Q	-1,45811	0,631230	-2,30995	0,025236	-2,72728	-0,18894	-0,729055	0,315615	-1,36364	-0,094468
2Q by 3L	1,85433	0,631230	2,93765	0,005068	0,58516	3,12351	0,927167	0,315615	0,29258	1,561754
2Q by 3Q	-0,68210	0,546661	-1,24776	0,218171	-1,78124	0,41703	-0,341052	0,273331	-0,89062	0,208516
2L by 4L	1,40334	0,728882	1,92533	0,060124	-0,06218	2,86885	0,701669	0,364441	-0,03109	1,434427
2L by 4Q	-0,33241	0,631230	-0,52661	0,600891	-1,60158	0,93676	-0,166205	0,315615	-0,80079	0,468381
2Q by 4L	0,71261	0,631230	1,12892	0,264543	-0,55657	1,98178	0,356304	0,315615	-0,27828	0,990890
2Q by 4Q	-0,07090	0,546661	-0,12970	0,897348	-1,17004	1,02824	-0,035450	0,273331	-0,58502	0,514118
3L by 4L	2,89770	0,728882	3,97555	0,000236	1,43219	4,36322	1,448852	0,364441	0,71609	2,181610
3L by 4Q	-0,66724	0,631230	-1,05704	0,295783	-1,93641	0,60194	-0,333618	0,315615	-0,96821	0,300968
3Q by 4L	-0,88979	0,631230	-1,40961	0,165101	-2,15896	0,37938	-0,444895	0,315615	-1,07948	0,189692
3Q by 4Q	0,12124	0,546661	0,22178	0,825423	-0,97790	1,22038	0,060620	0,273331	-0,48895	0,610189

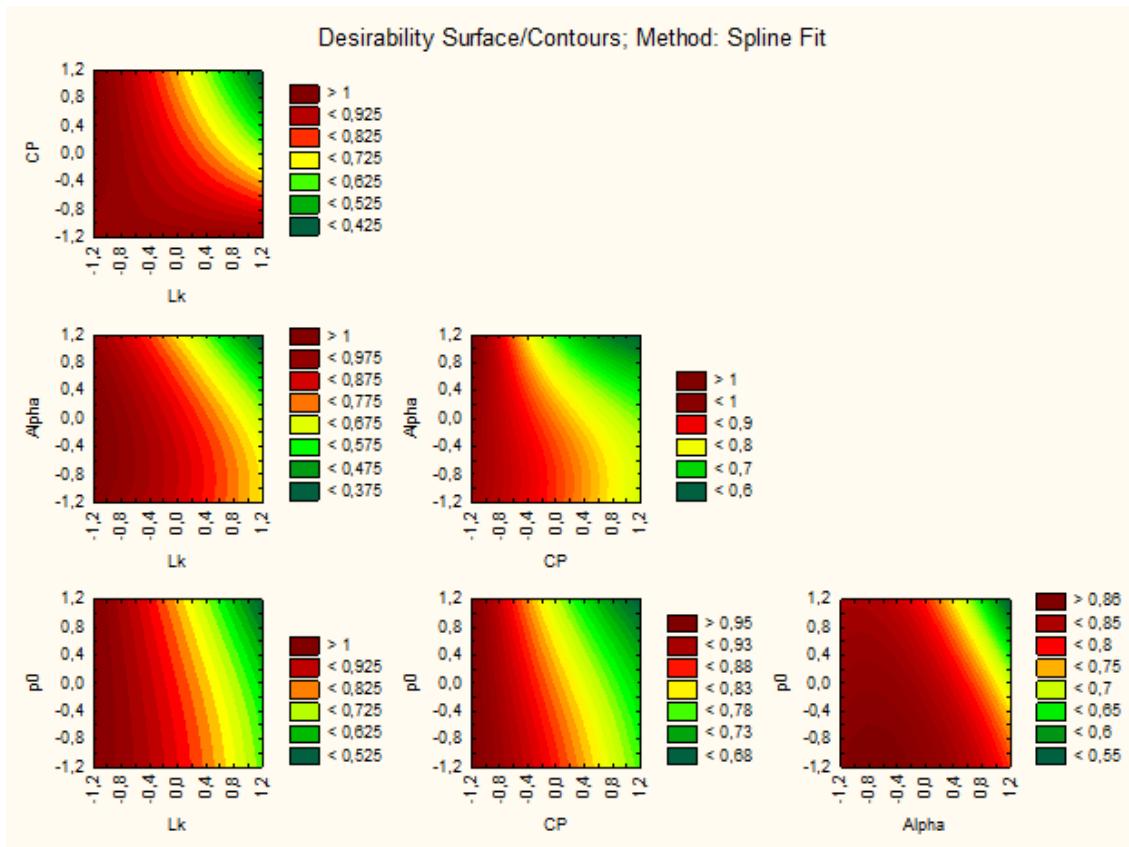


Figura A.10: Gráfico de contorno das quatro variáveis relevantes (L_k, CP, α, p_0), relativamente ao tempo, do Modelo 2 do problema do número de trabalhos

APÊNDICE A. TABELAS

Tabela A.11: Effect Estimates; Var.:Trabalhos; R-sqr=.94694; Adj.:.91156; 4 3-level factors, 1 Blocks, 81 Runs; MS Residual=.0347377; DV: Trabalhos. Modelo 3, problema do número de trabalhos

Factors	Effect	Std.Err	t(48)	p	-95.%Cnf.Limt	+95.%Cnf.Limt	Coeff.	Std.Err.Coeff.	-95.%Cnf.Limt	+95.%Cnf.Limt
Mean/Interc.	-33,4346	0,020709	-1614,50	0,000000	-33,4762	-33,3929	-33,4346	0,020709	-33,4762	-33,3929
(1)Lk (L)	-0,2481	0,050726	-4,89	0,000012	-0,3501	-0,1462	-0,1241	0,025363	-0,1751	-0,0731
Lk (Q)	-0,0204	0,043930	-0,46	0,644960	-0,1087	0,0680	-0,0102	0,021965	-0,0543	0,0340
(2)CP (L)	-1,1519	0,050726	-22,71	0,000000	-1,2538	-1,0499	-0,5759	0,025363	-0,6269	-0,5249
CP (Q)	-0,3537	0,043930	-8,05	0,000000	-0,4420	-0,2654	-0,1769	0,021965	-0,2210	-0,1327
(3)Alpha (L)	0,5074	0,050726	10,00	0,000000	0,4054	0,6094	0,2537	0,025363	0,2027	0,3047
Alpha (Q)	-0,1537	0,043930	-3,50	0,001019	-0,2420	-0,0654	-0,0769	0,021965	-0,1210	-0,0327
(4)p0 (L)	0,4259	0,050726	8,40	0,000000	0,3239	0,5279	0,2130	0,025363	0,1620	0,2640
p0 (Q)	-0,0537	0,043930	-1,22	0,227497	-0,1420	0,0346	-0,0269	0,021965	-0,0710	0,0173
1L by 2L	0,1611	0,062127	2,59	0,012566	0,0362	0,2860	0,0806	0,031063	0,0181	0,1430
1L by 2Q	0,0194	0,053803	0,36	0,719388	-0,0887	0,1276	0,0097	0,026902	-0,0444	0,0638
1Q by 2L	0,0139	0,053803	0,26	0,797401	-0,0943	0,1221	0,0069	0,026902	-0,0471	0,0610
1Q by 2Q	-0,0181	0,046595	-0,39	0,700100	-0,1117	0,0756	-0,0090	0,023298	-0,0559	0,0378
1L by 3L	0,0444	0,062127	0,72	0,477838	-0,0805	0,1694	0,0222	0,031063	-0,0402	0,0847
1L by 3Q	0,0444	0,053803	0,83	0,412861	-0,0637	0,1526	0,0222	0,026902	-0,0319	0,0763
1Q by 3L	0,0278	0,053803	0,52	0,608027	-0,0804	0,1360	0,0139	0,026902	-0,0402	0,0680
1Q by 3Q	-0,0306	0,046595	-0,66	0,515105	-0,1242	0,0631	-0,0153	0,023298	-0,0621	0,0316
1L by 4L	-0,0111	0,062127	-0,18	0,858812	-0,1360	0,1138	-0,0056	0,031063	-0,0680	0,0569
1L by 4Q	0,0111	0,053803	0,21	0,837264	-0,0971	0,1193	0,0056	0,026902	-0,0485	0,0596
1Q by 4L	0,0222	0,053803	0,41	0,681427	-0,0860	0,1304	0,0111	0,026902	-0,0430	0,0652
1Q by 4Q	-0,0306	0,046595	-0,66	0,515105	-0,1242	0,0631	-0,0153	0,023298	-0,0621	0,0316
2L by 3L	-0,0778	0,062127	-1,25	0,216665	-0,2027	0,0471	-0,0389	0,031063	-0,1013	0,0236
2L by 3Q	-0,0944	0,053803	-1,76	0,085578	-0,2026	0,0137	-0,0472	0,026902	-0,1013	0,0069
2Q by 3L	0,1778	0,053803	3,30	0,001806	0,0696	0,2860	0,0889	0,026902	0,0348	0,1430
2Q by 3Q	-0,1556	0,046595	-3,34	0,001635	-0,2492	-0,0619	-0,0778	0,023298	-0,1246	-0,0309
2L by 4L	-0,2333	0,062127	-3,76	0,000467	-0,3582	-0,1084	-0,1167	0,031063	-0,1791	-0,0542
2L by 4Q	0,0389	0,053803	0,72	0,473312	-0,0693	0,1471	0,0194	0,026902	-0,0346	0,0735
2Q by 4L	-0,0611	0,053803	-1,14	0,261670	-0,1693	0,0471	-0,0306	0,026902	-0,0846	0,0235
2Q by 4Q	0,0194	0,046595	0,42	0,678315	-0,0742	0,1131	0,0097	0,023298	-0,0371	0,0566
3L by 4L	0,2278	0,062127	3,67	0,000615	0,1029	0,3527	0,1139	0,031063	0,0514	0,1763
3L by 4Q	0,0028	0,053803	0,05	0,959039	-0,1054	0,1110	0,0014	0,026902	-0,0527	0,0555
3Q by 4L	-0,0694	0,053803	-1,29	0,202988	-0,1776	0,0387	-0,0347	0,026902	-0,0888	0,0194
3Q by 4Q	-0,0181	0,046595	-0,39	0,700100	-0,1117	0,0756	-0,0090	0,023298	-0,0559	0,0378

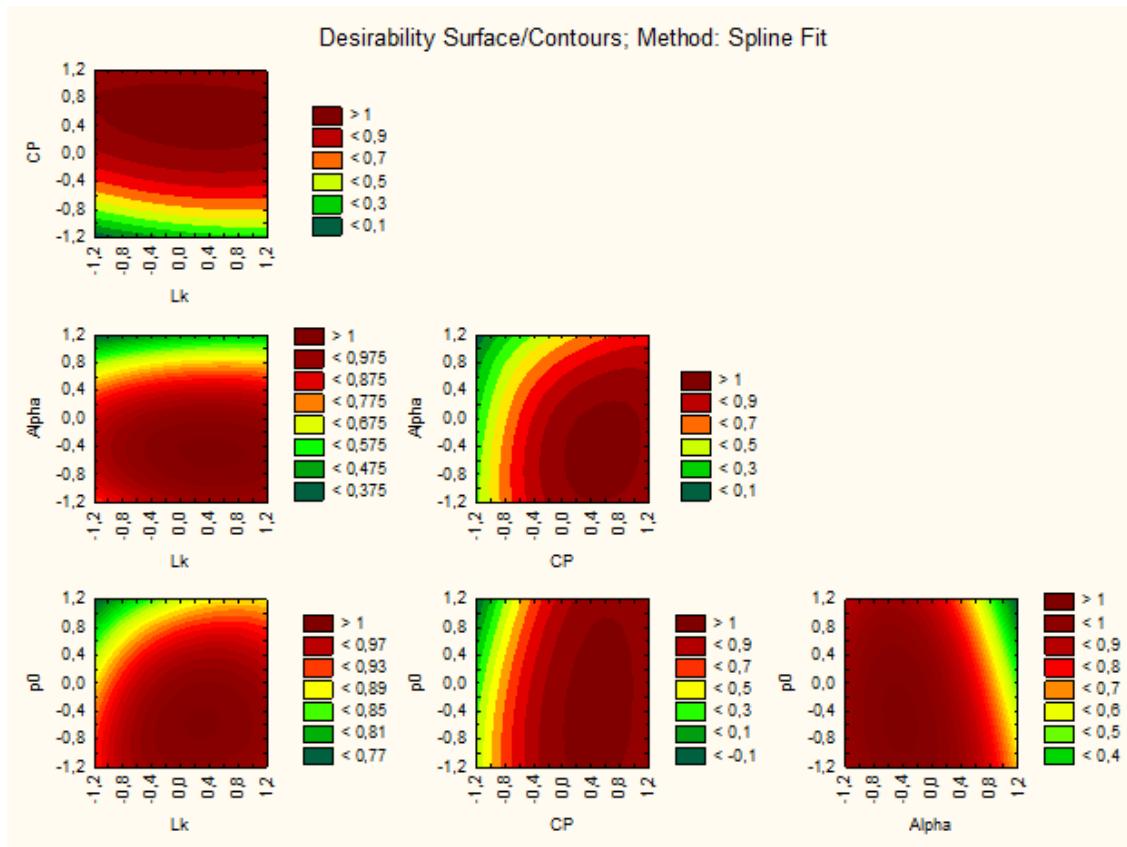


Figura A.11: Gráfico de contorno das cinco variáveis relevantes (L_k , CP , α , p_0), relativamente ao número de trabalhos, do Modelo 3 do problema do número de trabalhos

APÊNDICE A. TABELAS

Tabela A.12: Effect Estimates; Var.:Tempo; R-sqr=,99387; Adj.,98978; 4 3-level factors, 1 Blocks, 81 Runs; MS Residual=2,619281; DV: Tempo. Modelo 3, problema do número de trabalhos

Factors	Effect	Std.Err	t(48)	P	-95.%Cnf.Limt	+95.%Cnf.Limt	Coeff.	Std.Err.Coeff.	-95.%Cnf.Limt	+95.%Cnf.Limt
Mean/Interc.	17,29346	0,179824	96,16862	0,000000	16,93190	17,65502	17,29346	0,179824	16,93190	17,65502
(1)Lk (L)	22,35834	0,440478	50,75926	0,000000	21,47270	23,24398	11,17917	0,220239	10,73635	11,62199
Lk (Q)	0,01320	0,381465	0,03460	0,972539	-0,75379	0,78019	0,00660	0,190733	-0,37689	0,39009
(2)CP (L)	27,91703	0,440478	63,37895	0,000000	27,03139	28,80267	13,95852	0,220239	13,51570	14,40134
CP (Q)	0,17597	0,381465	0,46130	0,646665	-0,59102	0,94296	0,08799	0,190733	-0,29551	0,47148
(3)Alpha (L)	1,03598	0,440478	2,35194	0,022826	0,15034	1,92162	0,51799	0,220239	0,07517	0,96081
Alpha (Q)	0,55499	0,381465	1,45490	0,152206	-0,21199	1,32198	0,27750	0,190733	-0,10600	0,66099
(4)p0 (L)	1,58334	0,440478	3,59460	0,000764	0,69770	2,46898	0,79167	0,220239	0,34885	1,23449
p0 (Q)	-0,32573	0,381465	-0,85388	0,397412	-1,09271	0,44126	-0,16286	0,190733	-0,54636	0,22063
1L by 2L	17,91187	0,539473	33,20252	0,000000	16,82718	18,99655	8,95593	0,269737	8,41359	9,49828
1L by 2Q	0,18790	0,467197	0,40218	0,689335	-0,75146	1,12726	0,09395	0,233599	-0,37573	0,56363
1Q by 2L	0,07140	0,467197	0,15282	0,879183	-0,86797	1,01076	0,03570	0,233599	-0,43398	0,50538
1Q by 2Q	-0,09355	0,404605	-0,23121	0,818134	-0,90706	0,71996	-0,04677	0,202302	-0,45353	0,35998
1L by 3L	1,01771	0,539473	1,88648	0,065287	-0,06698	2,10239	0,50885	0,269737	-0,03349	1,05119
1L by 3Q	0,48880	0,467197	1,04624	0,300691	-0,45056	1,42816	0,24440	0,233599	-0,22528	0,71408
1Q by 3L	-0,57286	0,467197	-1,22616	0,226121	-1,51222	0,36651	-0,28643	0,233599	-0,75611	0,18325
1Q by 3Q	0,11609	0,404605	0,28692	0,775406	-0,69742	0,92960	0,05805	0,202302	-0,34871	0,46480
1L by 4L	1,43418	0,539473	2,65848	0,010633	0,34950	2,51886	0,71709	0,269737	0,17475	1,25943
1L by 4Q	-0,34871	0,467197	-0,74639	0,459070	-1,28808	0,59065	-0,17436	0,233599	-0,64404	0,29533
1Q by 4L	-0,20355	0,467197	-0,43569	0,665014	-1,14292	0,73581	-0,10178	0,233599	-0,57146	0,36790
1Q by 4Q	0,11673	0,404605	0,28850	0,774204	-0,69678	0,93024	0,05836	0,202302	-0,34839	0,46512
2L by 3L	2,31293	0,539473	4,28739	0,000087	1,22825	3,39761	1,15647	0,269737	0,61412	1,69881
2L by 3Q	0,14675	0,467197	0,31412	0,754794	-0,79261	1,08612	0,07338	0,233599	-0,39630	0,54306
2Q by 3L	-1,07545	0,467197	-2,30191	0,025722	-2,01481	-0,13608	-0,53772	0,233599	-1,00741	-0,06804
2Q by 3Q	0,74785	0,404605	1,84835	0,070716	-0,06566	1,56137	0,37393	0,202302	-0,03283	0,78068
2L by 4L	1,66726	0,539473	3,09054	0,003321	0,58258	2,75195	0,83363	0,269737	0,29129	1,37597
2L by 4Q	-0,23857	0,467197	-0,51065	0,611938	-1,17794	0,70079	-0,11929	0,233599	-0,58897	0,35039
2Q by 4L	0,01789	0,467197	0,03829	0,969617	-0,92148	0,95725	0,00894	0,233599	-0,46074	0,47863
2Q by 4Q	-0,32629	0,404605	-0,80644	0,423967	-1,13980	0,48722	-0,16314	0,202302	-0,56990	0,24361
3L by 4L	-0,57609	0,539473	-1,06787	0,290922	-1,66077	0,50860	-0,28804	0,269737	-0,83038	0,25430
3L by 4Q	0,43072	0,467197	0,92191	0,361184	-0,50865	1,37008	0,21536	0,233599	-0,25432	0,68504
3Q by 4L	0,90588	0,467197	1,93897	0,058396	-0,03348	1,84525	0,45294	0,233599	-0,01674	0,92262
3Q by 4Q	-0,54502	0,404605	-1,34704	0,184294	-1,35853	0,26849	-0,27251	0,202302	-0,67927	0,13425

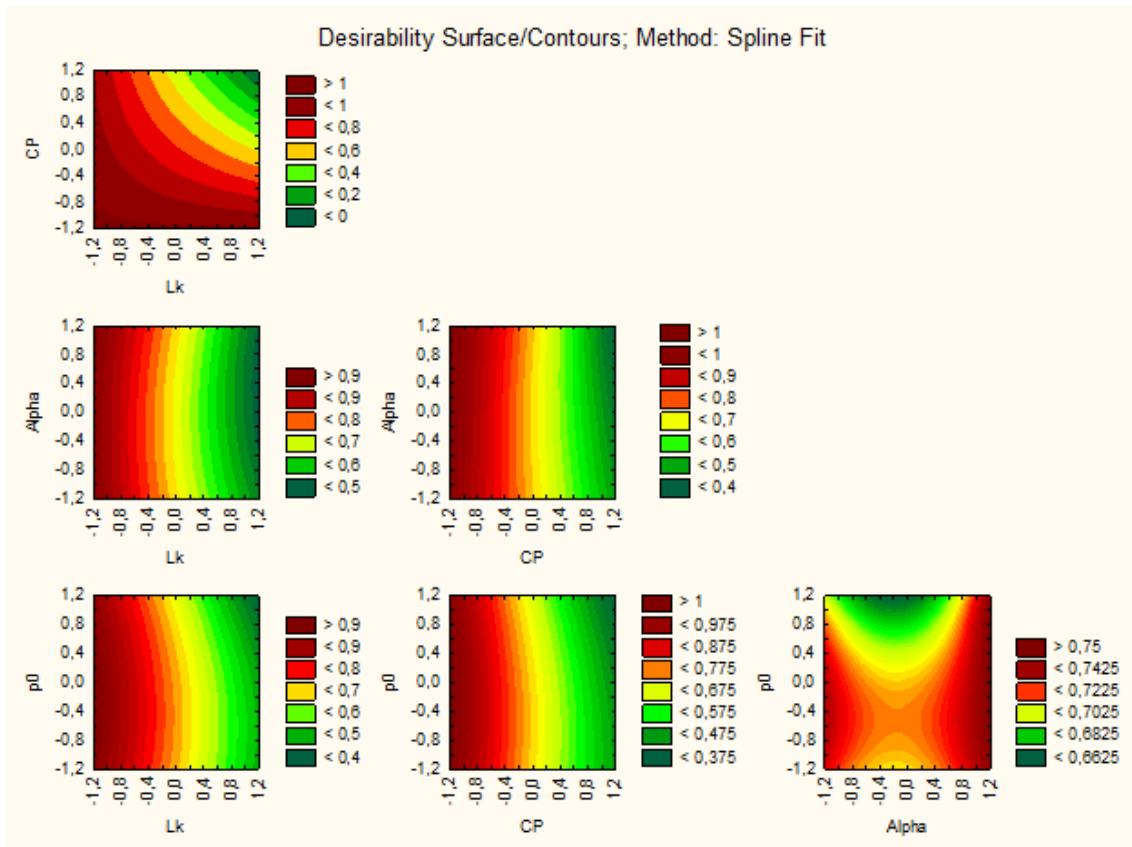


Figura A.12: Gráfico de contorno das quatro variáveis relevantes (L_k , CP , α , p_0), relativamente ao tempo, do Modelo 3 do problema do número de trabalhos

B

IMAGENS

B.1 Problema de Makespan

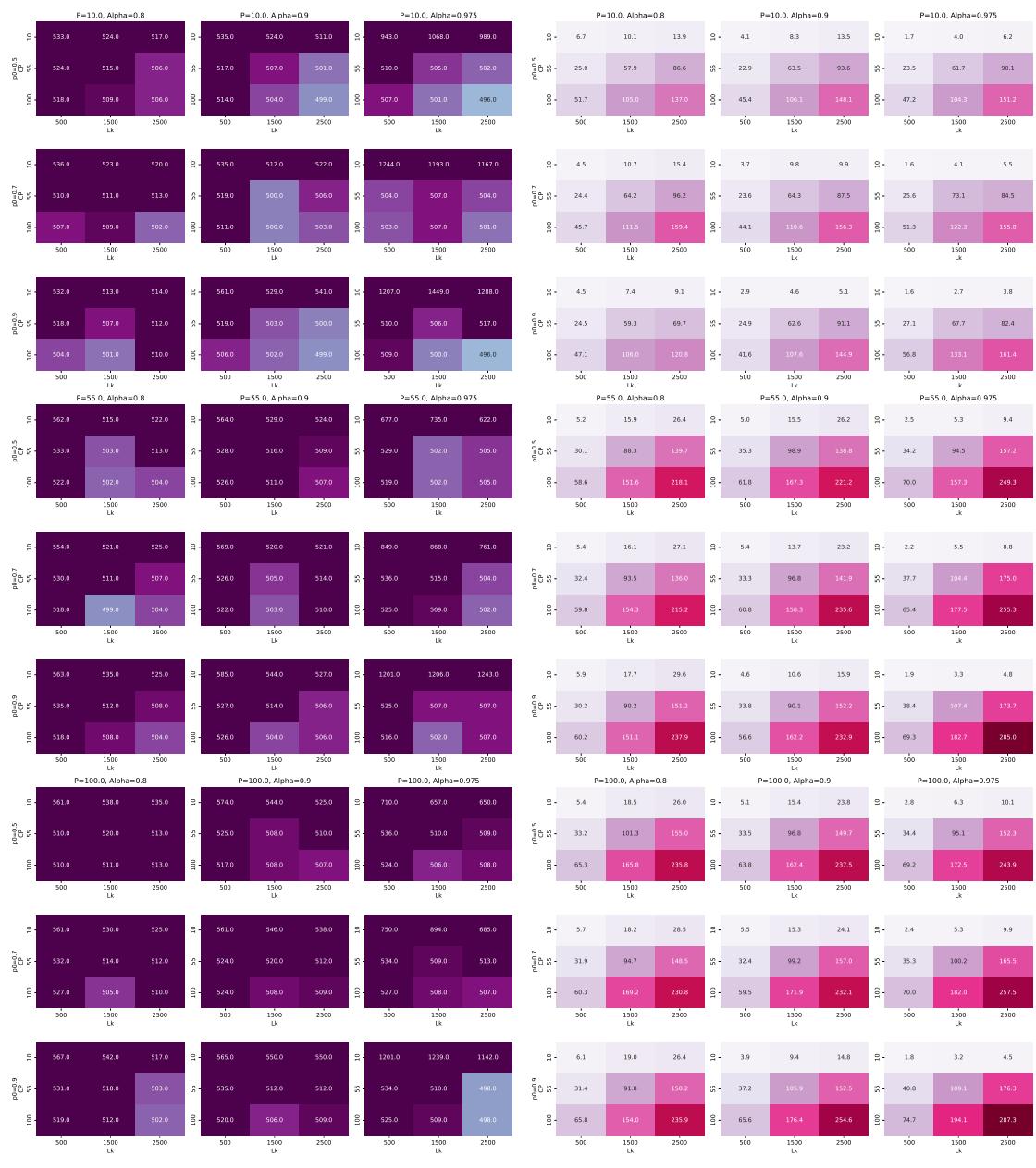


Figura B.1: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 1 do problema de makespan.

APÊNDICE B. IMAGENS

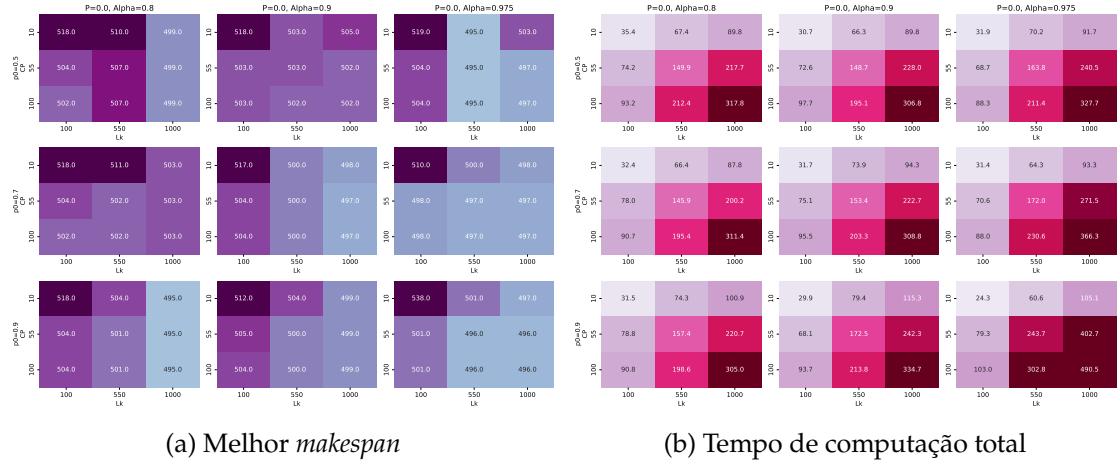


Figura B.2: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 2 do problema de *makespan*.

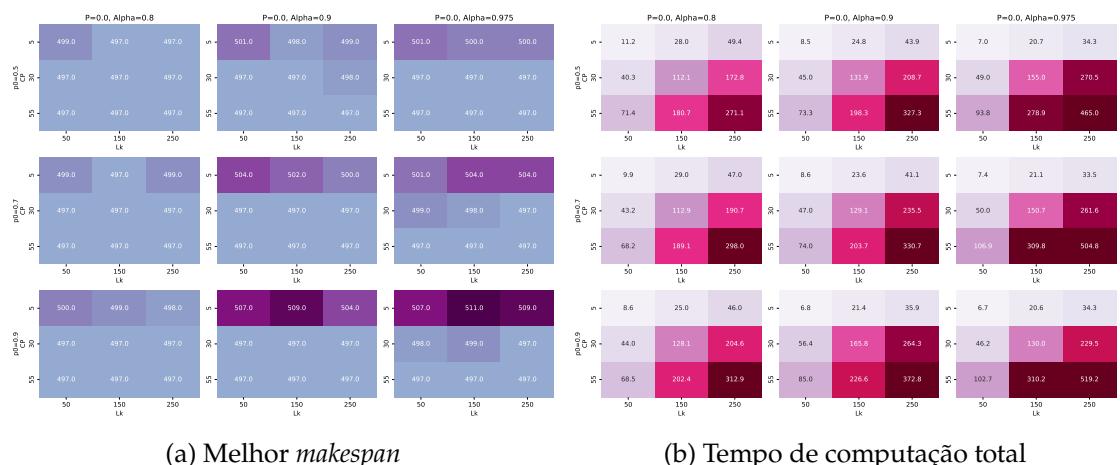


Figura B.3: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com *left shifting* do problema de *makespan*.

B.1. PROBLEMA DE MAKESPAN

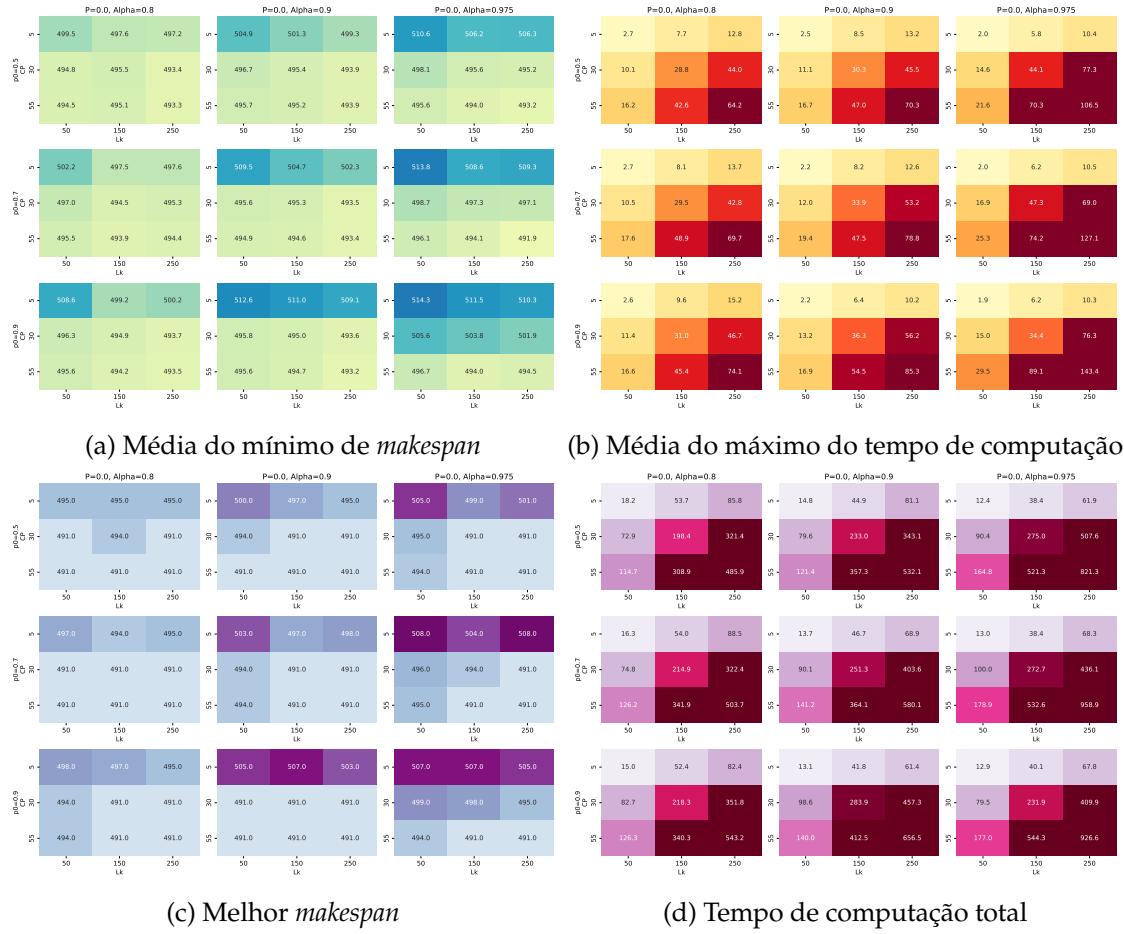


Figura B.4: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com *enhanced left shifting* do problema de makespan.

APÊNDICE B. IMAGENS

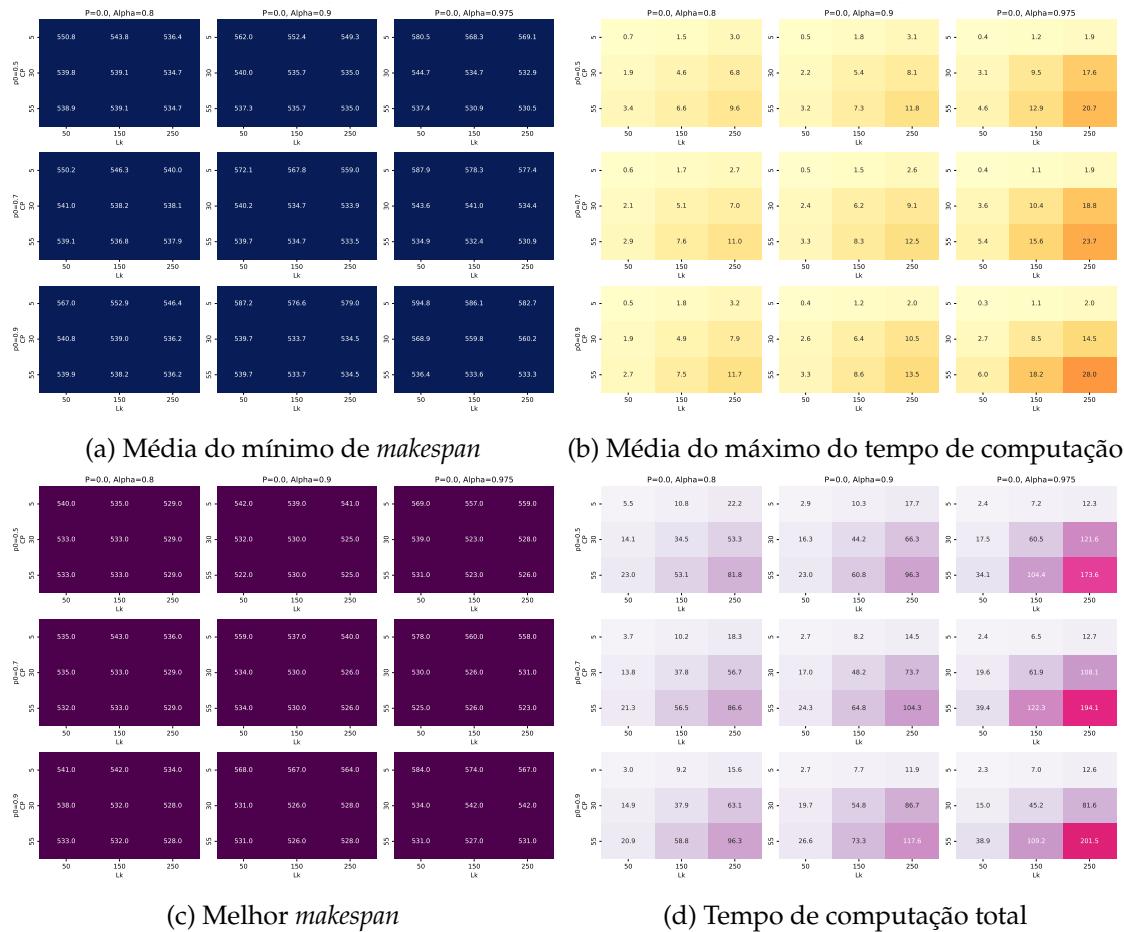


Figura B.5: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com *non-delay* do problema de *makespan*.

B.1. PROBLEMA DE MAKESPAN

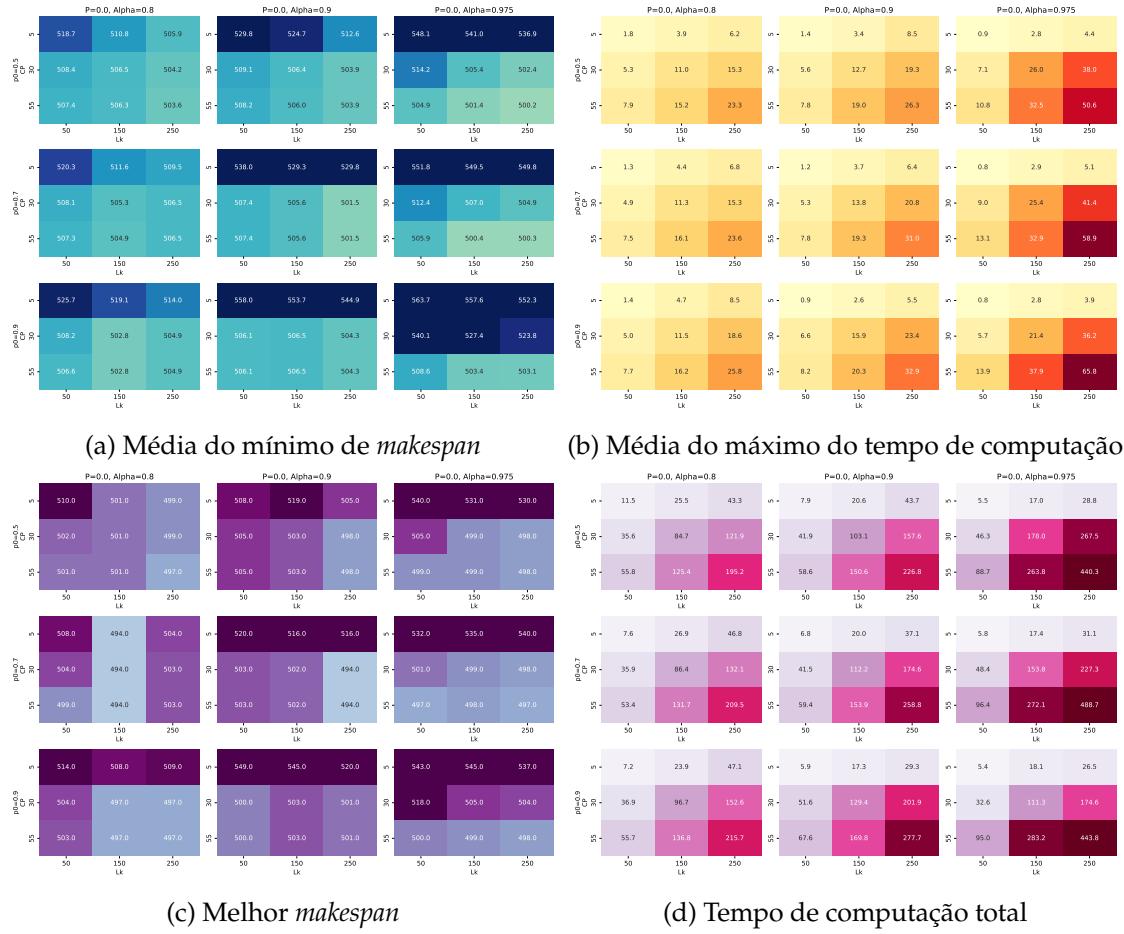


Figura B.6: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 com *enhanced non-delay* do problema de makespan.

B.2 Problema do número de trabalhos

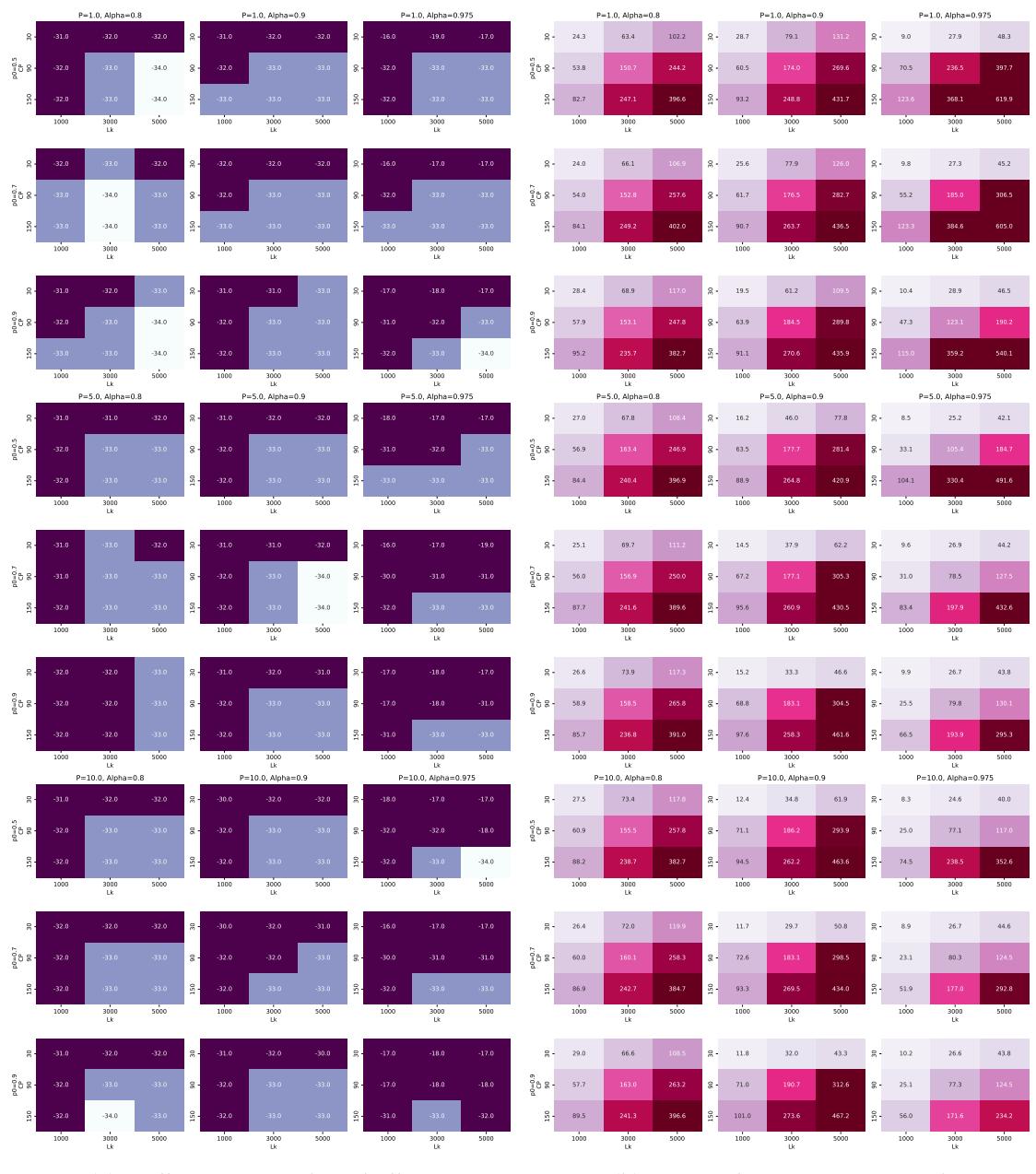


Figura B.7: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 1 do problema do número de trabalhos.

B.2. PROBLEMA DO NÚMERO DE TRABALHOS

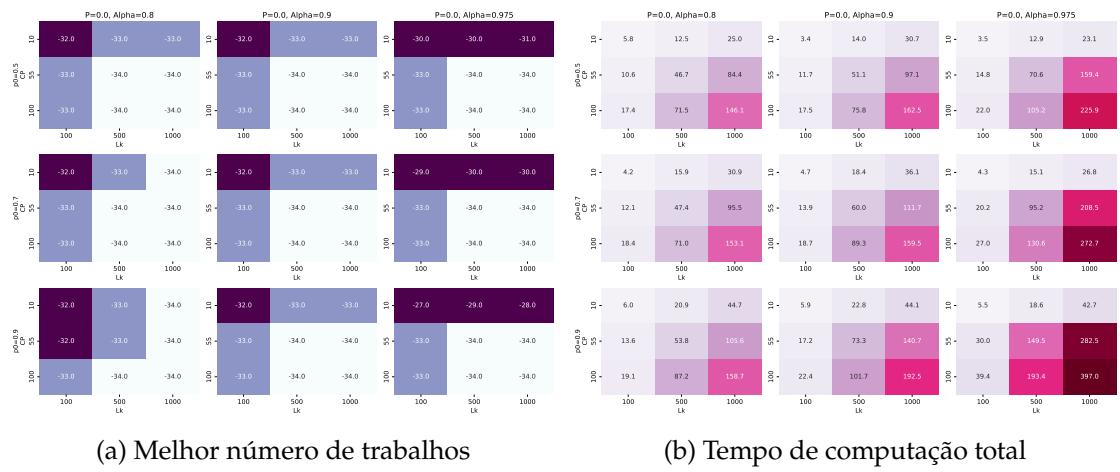


Figura B.8: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 2 do problema do número de trabalhos.

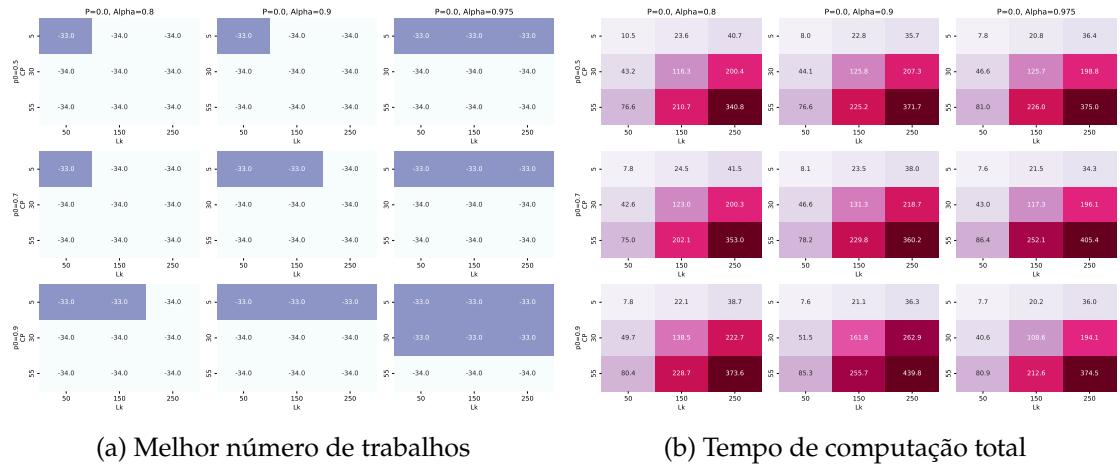


Figura B.9: Impacto dos níveis das variáveis sobre o valor da função objetivo e do tempo computacional para o Modelo 3 do problema do número de trabalhos.

C

ALGORITMOS

C.1 Problema de *Makespan*

Algorithm 1: Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição, e retrocesso. Para o problema de *makespan* com o Modelo 1.

```

i ← U(0,n);
q ← U(0,1);
d ← duração[i];
 $\delta$  ← tipo_exame_delta[i];
tvelho ← S[i];
// Retirar da matriz de utilização de recursos u, a utilização
    relativa a i
u[tvelho:tvelho+d] ← u[tvelho:tvelho+d] −  $\delta$ ;
// ini e fim provenientes da iteração anterior
tnovo ← U(ini, fim-d);
// Modificar a matriz S no índice i de forma a começar no novo
    instante tnovo
S[i] ← tnovo;
// Adicionar à matriz de utilização de recursos u, a utilização
    relativa a i
u[tnovo:tnovo+d] ← u[tnovo:tnovo+d] +  $\delta$ ;
ini ← min(S);
fim ← max(S + duração);
Cmax ← fim − ini;
f(s') ← Cmax +  $P \sum_{t=ini}^{fim-1} \sum_p v(t, p)$ ;
if f(s') < f(s) ou q < exp((f(s) - f(s'))/c) then
    s ← s';
    if f(s') < f(s*) then
        | s* ← s'
    end
end
else
    S[i] ← tvelho;
    u[tvelho:tvelho+d] ← u[tvelho:tvelho+d] +  $\delta$ ;
    u[tnovo:tnovo+d] ← u[tnovo:tnovo+d] −  $\delta$ ;

```

Algorithm 2: Pseudo-código que gera os instantes candidatos durante a criação da vizinhança.

```

Candidatos = {};
for  $t \leftarrow ini$  to  $fim - d$  do
     $u_{[t:t+d]} \leftarrow u_{[t:t+d]} + \delta;$ 
    if  $u \leq C_p$  then
        // Adicionar  $t$  à lista de candidatos
        Candidatos  $\leftarrow$  Candidatos  $\cup \{t\}$ ;
    end
end
Candidatos  $\leftarrow$  Candidatos  $\cup \{\max(ini - d, 0)\}$ ;
Candidatos  $\leftarrow$  Candidatos  $\cup \{\min(fim + 1, T_{max} - d)\}$ ;
return Candidatos

```

Algorithm 3: Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição, e retrocesso. Para o problema de *makespan* com o Modelo 2.

```

i ← U(0,n);
q ← U(0,1);
d ← duração[i];
 $\delta$  ← tipo_exame_delta[i];
tvelho ← S[i];
// Retirar da matriz de utilização de recursos u, a utilização
    relativa a i
 $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} - \delta;$ 
tnovo ← Candidatos[U(0, |Candidatos| - 1)];
// Modificar a matriz S no índice i de forma a começar no novo
    instante tnovo
S[i] ← tnovo;
// Adicionar à matriz de utilização de recursos u, a utilização
    relativa a i
 $u_{[t_{novo}:t_{novo}+d]} \leftarrow u_{[t_{novo}:t_{novo}+d]} + \delta;$ 
ini ← min(S);
fim ← max(S + duração);
Cmax ← fim - ini;
f(s') ← Cmax;
if f(s') < f(s) ou q < exp((f(s) - f(s'))/c) then
    |
    |   s ← s';
    |   if f(s') < f(s*) then
    |       |
    |       |   s* ← s'
    |   |
    |   end
    |
end
else
    |
    |   S[i] ← tvelho;
    |    $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} + \delta;$ 
    |    $u_{[t_{novo}:t_{novo}+d]} \leftarrow u_{[t_{novo}:t_{novo}+d]} - \delta;$ 
end

```

Algorithm 4: Pseudo-código do algoritmo para encontrar o instante mais cedo possível.

```

for  $t \leftarrow ini$  to  $fim - d$  do
     $u_{[t:t+d]} \leftarrow u_{[t:t+d]} + \delta;$ 
    if  $u \leq C_p$  then
        return  $t$ 
    end
end
return-1

```

Algorithm 5: Pseudo-código de *left-shift timetabling*

```

 $S^{forward} \leftarrow \{\};$ 
for  $i \in Sequência$  do
     $d \leftarrow duração[i];$ 
     $\delta \leftarrow tipo\_exame\_delta[i];$ 
     $t \leftarrow instante\_mais\_cedo;$ 
     $S^{forward} \leftarrow S^{forward} \cup \{t\};$ 
     $u_{[t:t+d]} \leftarrow u_{[t:t+d]} + \delta;$ 
end

```

Algorithm 6: Pseudo-código de *enhanced left-shift timetabling*

```

 $S^{forward} \leftarrow \{\};$ 
for  $i \in Sequência$  do
     $d \leftarrow duração[i];$ 
     $\delta \leftarrow tipo\_exame\_delta[i];$ 
     $t \leftarrow instante\_mais\_cedo;$ 
     $S^{forward} \leftarrow S^{forward} \cup \{t\};$ 
     $u_{[t:t+d]} \leftarrow u_{[t:t+d]} + \delta;$ 
end
 $S^{backward} \leftarrow \{\};$ 
Sequência  $\leftarrow$  Sequência.flip();
for  $i \in Sequência$  do
     $d \leftarrow duração[i];$ 
     $\delta \leftarrow tipo\_exame\_delta[i];$ 
     $\delta \leftarrow \delta.\text{flip}();$ 
     $t \leftarrow instante\_mais\_cedo;$ 
     $S^{backward} \leftarrow S^{backward} \cup \{t\};$ 
     $u_{[t:t+d]} \leftarrow u_{[t:t+d]} + \delta;$ 
end

```

Algorithm 7: Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição, e retrocesso. Para o problema de *makespan* com o Modelo 3.

```

// Assegurar que o trabalho representado por  $i$  é diferente do  $l$ 
 $i \leftarrow U(0,n);$ 
 $l \leftarrow U(0,n);$ 
 $q \leftarrow U(0,1);$ 
// Alterar a ordem dos índices  $i$  e  $l$  na sequência  $\pi$ 
 $temp \leftarrow \pi_{[i]};$ 
 $\pi_{[i]} \leftarrow \pi_{[l]};$ 
 $\pi_{[l]} \leftarrow temp;$ 
// Obter o instante de começo de cada trabalho e a matriz de
utilização de recursos
 $S, u \leftarrow timetabling(\pi);$ 
 $C_{\max} \leftarrow \min(S);$ 
 $f(s') \leftarrow C_{\max};$ 
if  $f(s') < f(s)$  ou  $q < \exp((f(s) - f(s'))/c)$  then
    |
    |    $s \leftarrow s';$ 
    |   if  $f(s') < f(s^*)$  then
    |       |
    |       |    $s^* \leftarrow s'$ 
    |   |
    |   end
|
end
else
    |
    |    $temp \leftarrow \pi_{[i]};$ 
    |    $\pi_{[i]} \leftarrow \pi_{[l]};$ 
    |    $\pi_{[l]} \leftarrow temp;$ 
|
end
```

C.2 Problema do número de trabalhos

Algorithm 8: Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição. Para o problema do número de trabalhos com o Modelo 1.

```

// Escolher um valor de  $m$  com probabilidade, 90%, 5%, 5%
 $m \leftarrow \{0, 1, 2\};$ 
 $q \leftarrow U(0,1);$ 
if  $m = 0$  then
     $i \leftarrow Y = 1;$ 
     $t_{velho} \leftarrow S_{[i]};$ 
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} - \delta;$ 
     $t_{novo} \leftarrow U(0, T_{\max}-d);$ 
     $S_{[i]} \leftarrow t_{novo};$ 
     $u_{[t_{novo}:t_{novo}+d]} \leftarrow u_{[t_{novo}:t_{novo}+d]} + \delta;$ 
end
else if  $m = 1$  then
     $i \leftarrow Y = 1;$ 
     $t_{velho} \leftarrow S_{[i]};$ 
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} - \delta;$ 
     $S_{[i]} \leftarrow -1;$ 
     $Y_{[i]} \leftarrow 0;$ 
else if  $m = 2$  then
     $i \leftarrow Y = 0;$ 
     $t_{novo} \leftarrow U(0, T_{\max}-d);$ 
     $S_{[i]} \leftarrow t_{novo};$ 
     $Y_{[i]} \leftarrow 1;$ 
     $u_{[t_{novo}:t_{novo}+d]} \leftarrow u_{[t_{novo}:t_{novo}+d]} + \delta;$ 
 $f(s') \leftarrow \sum(Y = 1) + P \sum_{t=0}^{T_{\max}-1} \sum_p v(t, p);$ 
if  $f(s') < f(s)$  ou  $q < \exp((f(s) - f(s'))/c)$  then
     $s \leftarrow s';$ 
    if  $f(s') < f(s^*)$  then
         $| s^* \leftarrow s'$ 
    end
end
end
else
     $| Retrocesso(m, i, t_{velho});$ 
end
```

Algorithm 9: Pseudo-código de retrocesso. Para o problema do número de trabalhos com o Modelo 1.

```

if  $m = 0$  then
     $S_{[i]} \leftarrow t_{velho};$ 
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} + \delta;$ 
     $u_{[t_{novo}:t_{novo}+d]} \leftarrow u_{[t_{novo}:t_{novo}+d]} - \delta;$ 
end
else if  $m = 1$  then
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} + \delta;$ 
     $S_{[i]} \leftarrow t_{velho};$ 
     $Y_{[i]} \leftarrow 1;$ 
else if  $m = 2$  then
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} - \delta;$ 
     $S_{[i]} \leftarrow -1;$ 
     $Y_{[i]} \leftarrow 0;$ 

```

Algorithm 10: Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição. Para o problema do número de trabalhos com o Modelo 2.

```

// Escolher um valor de  $m$  com probabilidade, 90%, 5%, 5%
 $m \leftarrow \{0, 1, 2\};$ 
 $q \leftarrow U(0,1);$ 
if  $m = 0$  then
     $i \leftarrow Y = 1;$ 
     $t_{velho} \leftarrow S_{[i]};$ 
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} - \delta;$ 
     $t_{novo} \leftarrow \text{Candidatos}[U(0, |\text{Candidatos}| - 1)];$ 
     $S_{[i]} \leftarrow t_{novo};$ 
     $u_{[t_{novo}:t_{novo}+d]} \leftarrow u_{[t_{novo}:t_{novo}+d]} + \delta;$ 
end
else if  $m = 1$  then
     $i \leftarrow Y = 1;$ 
     $t_{velho} \leftarrow S_{[i]};$ 
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} - \delta;$ 
     $S_{[i]} \leftarrow -1;$ 
     $Y_{[i]} \leftarrow 0;$ 
else if  $m = 2$  then
     $i \leftarrow Y = 0;$ 
     $t_{novo} \leftarrow \text{Candidatos}[U(0, |\text{Candidatos}| - 1)];$ 
     $S_{[i]} \leftarrow t_{novo};$ 
     $Y_{[i]} \leftarrow 1;$ 
     $u_{[t_{novo}:t_{novo}+d]} \leftarrow u_{[t_{novo}:t_{novo}+d]} + \delta;$ 
 $f(s') \leftarrow \sum(Y = 1) + P \sum_{t=0}^{T_{\max}-1} \sum_p v(t, p);$ 
if  $f(s') < f(s)$  ou  $q < \exp((f(s) - f(s'))/c)$  then
     $s \leftarrow s';$ 
    if  $f(s') < f(s^*)$  then
         $| s^* \leftarrow s'$ 
    end
end
else
     $| \text{Retrocesso}(m, i, t_{velho});$ 
end
```

Algorithm 11: Pseudo-código de retrocesso. Para o problema do número de trabalhos com o Modelo 2.

```

if  $m = 0$  then
     $S_{[i]} \leftarrow t_{velho};$ 
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} + \delta;$ 
     $u_{[t_{novo}:t_{novo}+d]} \leftarrow u_{[t_{novo}:t_{novo}+d]} - \delta;$ 
end
else if  $m = 1$  then
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} + \delta;$ 
     $S_{[i]} \leftarrow t_{velho};$ 
     $Y_{[i]} \leftarrow 1;$ 
else if  $m = 2$  then
     $u_{[t_{velho}:t_{velho}+d]} \leftarrow u_{[t_{velho}:t_{velho}+d]} - \delta;$ 
     $S_{[i]} \leftarrow -1;$ 
     $Y_{[i]} \leftarrow 0;$ 

```

Algorithm 12: Pseudo-código de *left-shift timetabling* para o problema do número de trabalhos

```

 $S^{forward} \leftarrow \{\};$ 
for  $i \in Sequência$  do
     $d \leftarrow duração[i];$ 
     $\delta \leftarrow tipo\_exame\_delta[i];$ 
     $t \leftarrow instante\_mais\_cedo;$ 
    if  $t = 1$  then
         $| \quad t \leftarrow T_{\max}$ 
    end
     $S^{forward} \leftarrow S^{forward} \cup \{t\};$ 
     $u_{[t:t+d]} \leftarrow u_{[t:t+d]} + \delta;$ 
end

```

Algorithm 13: Pseudo-código de geração de novos vizinhos, a sua avaliação, aceitação ou rejeição, e retrocesso. Para o problema do número de trabalhos com o Modelo 3.

```

 $i \leftarrow U(0,n);$ 
 $l \leftarrow U(0,n);$ 
 $q \leftarrow U(0,1);$ 
 $temp \leftarrow \pi_{[i]};$ 
 $\pi_{[i]} \leftarrow \pi_{[l]};$ 
 $\pi_{[l]} \leftarrow temp;$ 
 $S, u \leftarrow timetabling(\pi);$ 
//  $S + d$  é equivalente ao instante de término de cada trabalho
 $f(s') \leftarrow \sum(S + d \leq T_{\max});$ 
if  $f(s') < f(s)$  ou  $q < \exp((f(s) - f(s'))/c)$  then
     $s \leftarrow s';$ 
    if  $f(s') < f(s^*)$  then
         $| s^* \leftarrow s'$ 
    end
end
else
     $temp \leftarrow \pi_{[i]};$ 
     $\pi_{[i]} \leftarrow \pi_{[l]};$ 
     $\pi_{[l]} \leftarrow temp;$ 
end
```

