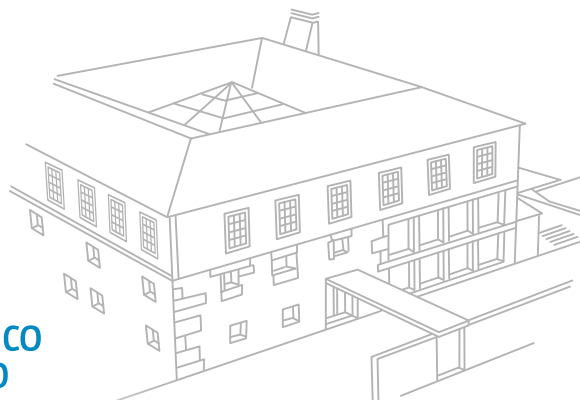


ESTGF | **POLITÉCNICO
DO PORTO**



ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO

Escalonamento de pessoal – o INRC como ponto de partida para a resolução de casos reais

DESIGNAÇÃO DO MESTRADO Mestrado em Engenharia Informática

AUTOR Tiago Fernando Ferreira Lopes

ORIENTADOR(ES) Carla Sofia Pereira

ANO 2015

www.estgf.ipp.pt

“A única maneira de fazer um trabalho extraordinário é de amares aquilo que fazes. Se ainda não o encontraste, continua a procurar. Não te acomodes. Tal como com os assuntos do coração, tu saberás quando o encontraste.”

Steve Jobs

Este trabalho é dedicado a todos os que procuram dar o melhor de si em cada projeto para que este se possa tornar extraordinário.

Agradecimentos

Desde já gostaria de agradecer à Bullet Solutions e a toda a equipa envolvida no projeto, pelo conhecimento partilhado e pelo apoio e incentivo que deram na realização deste projeto.

Gostaria de agradecer à Professora Carla Pereira por ter orientado mais uma etapa importante da minha vida académica. O meu muito obrigado pelas oportunidades e pelo conhecimento que me transmitiu ao longo dos últimos anos.

Por fim mas não menos importante, gostaria de agradecer à minha família aos meus amigos e à minha namorada pela paciência, atenção e apoio que me deram ao longo desta jornada.

Resumo

O escalonamento de pessoal é um problema muito estudado, tanto pela comunidade científica, como pelos gestores e responsáveis pelos recursos humanos das empresas. O impacto financeiro das decisões tomadas, o bem-estar dos colaboradores, ou conceitos mais subjetivos como “justiça” e “equilíbrio”, acabam por transformar este caso em muito mais do que um simples problema rotineiro. Quando se fala em escalonamento de pessoal médico, dificuldades adicionais são encontradas, como a continuidade do trabalho (24 horas por dia, 7 dias por semana), ou a qualidade de serviço que tem de ser assegurada. O trabalho apresentado neste documento teve como ponto de partida as regras definidas no INRC – *Second International Nurse Rostering Competition* – mas sempre com a visão de criar as bases necessárias para o futuro desenvolvimento de um produto genérico de escalonamento automático e otimizado de pessoal.

Palavras-chaves: heurísticas, otimização combinatória, escalonamento de pessoal, algoritmos.

Abstract

Personnel scheduling problems are widely studied both by the scientific community and by the human resource managers of the companies. The financial impact of the decisions, the welfare of the employees, or more subjective concepts like "fairness" and "balance", turn this case into something more than just a routine problem. When it comes to medical personnel scheduling, additional difficulties can be found, such as uninterrupted work (24 hours per day, 7 days per week), or the quality of service that has to be ensured. The work presented in this paper was based on the rules defined in INRC - Second International Nurse Rostering Competition - but always with the vision of creating the necessary basis for the future development of an automatic and optimized generic personnel scheduling software.

Key-words: heuristics, combinatorial optimization, personnel scheduling, algorithms.

Índice

Agradecimentos	2
Resumo	3
Abstract	4
Índice	5
Índice de Figuras	7
Índice de Tabelas	8
Índice de Abreviaturas	9
1. Introdução	10
1.1. Enquadramento e Motivação	10
1.2. Objetivos e principais resultados	10
1.3. Estrutura da tese	11
2. O problema de <i>nurse rostering</i>	12
2.1. Características gerais do problema	12
2.2. Modelação do problema de <i>nurse rostering</i>	14
2.3. Abordagens propostas na literatura	17
2.3.1. Métodos matemáticos exatos	17
2.3.2. Inteligência Artificial	17
2.3.3. Heurísticas	18
2.3.4. Meta-heurísticas	21
2.3.5. Principais conclusões da revisão da literatura	23
3. Problema	24
3.1. Conceitos fundamentais	24
3.2. Restrições	26
3.3. Objetivos	27
4. Proposta de solução	29
4.1. Abordagem	29
4.1.1. Paradigma Funcional	29
4.1.2. Dados mutáveis e imutáveis	30
4.1.3. Gestão da solução	32
4.1.4. Processamento paralelo	33

4.2. Arquitetura da solução	34
4.2.1. A camada <i>solver</i>	35
4.2.2. A camada <i>core</i>	37
4.3. Especificação dos algoritmos	39
5. Caso de Estudo	43
5.1. Descrição do problema	43
5.2. Teste e validação da solução	50
5.2.1. Soluções submetidas a concurso	52
5.2.2. Resultados submetidos pelos participantes	54
5.2.3. Resultados das soluções geradas pela organização.....	58
6. Conclusões e Trabalho Futuro.....	59
6.1. Conclusões.....	59
6.2. Trabalho Futuro	59
7. Glossário	60
8. Referências bibliográficas.....	61
9. Anexos	67
9.1. Escalas de instâncias geradas pelos algoritmos.....	67

Índice de Figuras

Figura 1 - Mapa de conceitos associados ao problema de NRP em estudo	26
Figura 2 - Declaração de dados imutáveis	30
Figura 3 - Atribuição de variáveis imutáveis	31
Figura 4 - Declaração de dados mutáveis	31
Figura 5 - Execução de alterações na solução	32
Figura 6 - Conceito de processamento paralelo	33
Figura 7 - Arquitetura global da solução	34
Figura 8 - Arquitetura da camada <i>solver</i>	35
Figura 9 - Arquitetura da camada <i>core</i>	37
Figura 10 - Fluxograma da construção da solução inicial	40
Figura 11 - Fluxograma da otimização da solução	42
Figura 12 - Exemplo do ficheiro de cenário	44
Figura 13 - Exemplo do ficheiro de dados de semana	45
Figura 14 - Exemplo do ficheiro de histórico	46
Figura 15 - Processo de geração completo (solução/simulação/validação)	47
Figura 16 - Exemplo de uma escala de 4 semanas gerada pelo simulador.....	48
Figura 17 - Exemplo da pasta do conjunto de dados de 30 recursos para 4 semanas	50
Figura 18 - Comparação de resultados por equipa	56
Figura 19 - Escala para a instância n030w4_1_6-2-9-1	68
Figura 20 - Escala para instância n030w4_1_6-7-5-3	69
Figura 21 - Escala para instância n040w4_0_2-0-6-1	70
Figura 22 - Escala para instância n040w4_2_6-1-0-6	71
Figura 23 - Escala para instância n050w4_0_0-4-8-7	72
Figura 24 - Escala para instância n050w4_0_7-2-7-2	73
Figura 25 - Escala para instância n060w4_1_6-1-1-5	74
Figura 26 - Escala para instância n060w4_1_9-6-3-8	75
Figura 27 - Escala para instância n030w8_1_2-7-0-9-3-6-0-6	76
Figura 28 - Escala para instância n040w8_0_0-6-8-9-2-6-6-4	77

Índice de Tabelas

Tabela 1 - Conceitos associados geralmente ao NRP	13
Tabela 2 - Conjuntos de dados para competição	51
Tabela 3 - Soluções das instâncias submetidas a concurso.....	52
Tabela 4 - Resultados submetidos pelos participantes: instâncias de 4 semanas	54
Tabela 5 - Resultados submetidos pelos participantes: instâncias de 8 semanas	55
Tabela 6 - Comparação de resultados	57
Tabela 7 - Classificação das equipas	58

Índice de Abreviaturas

NRP – Problema de *Nurse Rostering*

INRC – *International Nurse Rostering Competition*

FO – Função Objetivo

IA – Inteligência Artificial

MH – Meta-heurísticas

FP – Programação Funcional

DM – *Data Manager*

JSON – *JavaScript Object Notation*

1. Introdução

1.1. Enquadramento e Motivação

O escalonamento de pessoal é um problema muito estudado, tanto pela comunidade científica, como pelos gestores e responsáveis pelos recursos humanos das empresas. É um problema com especial importância no sector dos serviços, nomeadamente no escalonamento de pessoal médico, podendo-se afirmar que é um problema com grande impacto na qualidade da assistência médica [1]. Na sua resolução tem de ser considerado o impacto financeiro das decisões tomadas, o bem estar dos colaboradores, ou conceitos mais subjetivos como “justiça” e “equilíbrio”, que acabam por transformar este caso em muito mais do que um simples problema rotineiro. O escalonamento de pessoal médico é um problema complexo que surge habitualmente decomposto nos subproblemas de *nurse staffing*, *nurse rostering* e *nurse rerostering*. *Nurse staffing* determina o número de recursos necessários e os seus diferentes níveis de qualificação. *Nurse rostering*, é a tarefa de encontrar uma escala de serviço para um conjunto de recursos, respeitando regras, normas de trabalho e pedidos da administração. *Nurse rerostering* tem como objetivo criar uma escala de serviço, com as mudanças necessárias, o mais aproximada das listas geradas anteriormente [2].

Quando se fala em escalonamento de pessoal médico, dificuldades adicionais são encontradas, como a continuidade do trabalho (24 horas por dia, 7 dias por semana), ou a qualidade de serviço que tem de ser assegurada.

1.2. Objetivos e principais resultados

O projeto apresentado enquadra-se na atividade da empresa Bullet Solutions¹ e tem como objetivo apresentar uma nova abordagem ao problema de *nurse rostering* (NRP), através da implementação de uma nova solução funcional e modular. Assim, este projeto centrou-se no desenvolvimento e implementação

¹ <https://www.bulletsolutions.com/>

de novos algoritmos que, posteriormente, serão aplicados em produtos da Bullet Solutions.

Este projeto teve como ponto de partida as regras definidas no INRC – *Second International Nurse Rostering Competition* – mas sempre com a visão de criar as bases necessárias para o futuro desenvolvimento de um produto genérico de escalonamento automático e otimizado de pessoal. Os resultados obtidos foram submetidos ao concurso “Second International Nurse Rostering Competition”², que decorreu durante o ano de 2015. A participação no concurso revelou-se como uma excelente forma de avaliação dos resultados obtidos e permitiu também avaliar o nível de maturidade do trabalho desenvolvido pela equipa na resolução do problema.

1.3. Estrutura da tese

Este documento está organizado da seguinte forma. O **capítulo 2** apresenta os principais resultados da revisão da literatura e mostra uma visão geral do problema de *nurse rostering*. O **capítulo 3** descreve o problema a resolver. O **capítulo 4** descreve a proposta de solução desenvolvida para o NRP. Nesta secção é feita uma apresentação da abordagem e da arquitetura da solução. O **capítulo 5** apresenta o caso de estudo no qual foi aplicada e testada a solução proposta. Neste capítulo estão também referenciados os resultados obtidos e a comparação dos mesmos. O **capítulo 6** apresenta as principais conclusões e possibilidades de trabalho futuro.

² <http://mobiz.vives.be/inrc2/>

2. O problema de *nurse rostering*

O NRP pertence à família dos problemas de *timetabling*. É um problema que tem vindo a ser estudado à mais de 40 anos e a cuja literatura disponível é muito extensa. Em 2004, Burke et al. fizeram uma revisão do estado da arte [3] muito citada por outros artigos publicados na área do NRP, sendo que, esta revisão expressa muito bem o quão extensa é a informação disponível sobre NRP.

2.1. Características gerais do problema

Na literatura, o NRP geralmente é definido como um problema que tem como objetivo determinar os recursos, com um conjunto de competências, necessários num dado momento para responder a uma determinada necessidade [3].

Segundo a revisão do estado da arte de Burke et al. [3], encontramos associado ao NRP fatores como as características e estruturas organizacionais, categorias de competências do pessoal, preferências de trabalho [4] e as necessidades dos pacientes [3]. Na Tabela 1, são listados os conceitos que geralmente são encontrados em formulações de problemas NRP. A consideração destes fatores leva a que a geração de escalas seja uma tarefa muito complexa [5].

Conceito	Descrição
Horizonte de planeamento	O horizonte de geração é o intervalo de tempo que vai ser gerado para cada escala. Tipicamente o tamanho do horizonte de geração é de 4 semanas. [3].
Competências	Determinam a categoria do recurso. Quem tem determinada qualificação, competência ou responsabilidade.
Turnos	Os turnos são a divisão temporal dos deveres da organização. Geralmente, os turnos tem uma hora de início e de fim bem definida. Na literatura é

	tradicional encontrar a divisão em 3 turnos: manhã (ex.: 7:00 – 15:00), tarde (ex.: 15:00 – 22:00) ou noite (ex.: 22:00 – 7:00).
Necessidades	As necessidades expressam o número de recursos necessários para cada competência e para cada turno ou intervalo de tempo no horizonte de planeamento.
Regras relacionadas com tempo	Este tipo de regras, geralmente, referem-se aos pedidos dos recursos (ex.: férias ou ausências), às suas preferências (ex.: evitar alocar turnos na quarta-feira) e regras para equilibrar a carga de trabalho de pessoas que pertencem à mesma categoria (ex.: o contrato de trabalho estabelecido com a instituição).
Restrições	As restrições são um conjunto de regras que devem ser satisfeitas a todo o custo, caso contrário não se pode dizer que uma solução é viável.
Objetivos	Os objetivos são o conjunto de regras que definem o que se pretende alcançar, contudo, estes podem ser violados de modo a gerar uma solução viável.

Tabela 1 - Conceitos associados geralmente ao NRP

Em termos de procedimentos administrativos, a literatura divide os problemas propostos em três tipos: *Centralised scheduling*, *Unit scheduling* e *Self scheduling*.

O *Centralised scheduling* é utilizado para descrever um cenário em que num hospital existe um departamento administrativo central, responsável por gerar as escalas todas da organização. Este tipo de organização permite libertar os chefes das unidades da tarefa de geração das escalas [3].

O *Unit scheduling* é aplicado quando se está perante um cenário em que cada unidade do hospital tem a responsabilidade de gerar as suas escalas [3].

O *Self scheduling* é o termo usado na literatura para descrever um cenário em que as escalas são geradas manualmente pelos próprios funcionários, por vezes com pouco, ou nenhum, suporte assistido por computador [3].

O problema é ainda caracterizado por ser cíclico ou não cíclico. Segundo Howell [6], o *cyclical scheduling* (CS) é utilizado para descrever o NRP aplicado em organizações em que as pessoas trabalham ciclos de várias semanas. O CS pode ser identificado em situações em que o problema encontra-se partido em turnos distintos e as necessidades de recursos por dia e por turno obedecem a um padrão cíclico. O *non-cyclical scheduling* é o contrário do CS, o problema pode não se encontrar dividido em turnos distintos e as necessidades não obedecem a um padrão cíclico.

Na literatura, um problema cíclico é associado a um escalonamento fixo, enquanto um problema não cíclico é associado a um escalonamento flexível [7].

2.2. Modelação do problema de *nurse rostering*

A modelação do NRP envolve a especificação dos dados que vão ser utilizados, a definição de restrições, a definição de objetivos e a formulação do problema.

Os conceitos apresentados na Tabela 1, do capítulo anterior, são traduzidos para informação útil ao sistema e geram os dados que são utilizados pelos algoritmos. Os dados geralmente utilizados são: 1) itens, 2) tarefas, 3) custos e 4) blocos de tempo.

- 1) Itens – Os itens referem-se aos objetos que vão ser escalonados. No caso do NRP são os enfermeiros (recursos). Os itens têm um identificador único e podem ter um conjunto de propriedades que auxiliam o cálculo do custo da alocação do item [8].
- 2) Tarefas – As tarefas são as necessidades do problema. São uma forma geral de identificar o número e o tipo de itens necessários para um período de tempo bem definido. Por exemplo, entre as 10 da noite e as 6 da manhã são precisos dois enfermeiros para fazer o turno da noite [8].
- 3) Blocos de tempo – Um bloco de tempo é um período com início e fim bem definidos onde um item pode ser atribuído. Apesar de ser um conceito muito parecido com o turno ao nível do algoritmo, um bloco de tempo é uma

estrutura mais genérica e pode representar um turno, um intervalo, tempo de viagem ou até blocos de formação. Para além de identificar o período de tempo, esta estrutura pode ainda ter um conjunto de propriedades que determinam o custo de utilização do bloco de tempo [8].

- 4) Custos – O custo é determinado através dos itens atribuídos aos blocos de tempo. A atribuição de um item a um bloco de tempo tem custos associados, como o salário - hora do recurso ou o valor pago por horas extra se for o caso [8]. Os custos são avaliados pela função objetivo que guia a construção de resultados e a sua meta principal é a avaliação de custos por forma a tomar decisões que ajudem a ir de encontro aos objetivos [9].

Parte dos dados do NRP são gerados pelas regras do problema em específico, como é o caso dos custos. Em termos de regras, o NRP geralmente é composto por restrições (*hard constraints*) e objetivos (*soft constraints*). As restrições normalmente envolvem a cobertura de necessidades como, por exemplo, as necessidades de enfermeiros por dia, por turno ou por competência, enquanto os objetivos estão normalmente relacionados com requisitos no tempo como, por exemplo, a preferência dos enfermeiros por fazer ou não determinado turno. A seguinte lista apresenta as restrições e objetivos que são normalmente utilizados na formulação do NRP. Esta lista foi apresentada por B. Cheang et al. [9], em 2003, na revisão bibliográfica sobre o NRP:

1. Carga de trabalho (mínima/máxima);
2. Trabalho de turnos consecutivos do mesmo tipo (mínimo/máximo/número exato);
3. Trabalho de dias ou turnos consecutivos (mínimo/máximo/número exato);
4. Competências e categorias dos enfermeiros;
5. Requisitos ou preferências dos enfermeiros;
6. Dias de folga (mínimo/máximo/dias consecutivos);
7. Tempo livre entre turnos de trabalho (mínimo);
8. Atribuições de tipos de turno (máximo por tipo de turno, requisitos para cada tipo de turno);
9. Feriados e férias;
10. Fim-de-semana de trabalho como, por exemplo, trabalhar o fim-de-semana completo;

11. Restrições entre grupos/tipos de enfermeiros, por exemplo, enfermeiros que não podem trabalhar juntos ou enfermeiros que devem trabalhar juntos;
12. Padrões de turnos;
13. Histórico como, por exemplo, alocações anteriores;
14. Requisitos no horizonte de planejamento como, por exemplo, todos os dias devem ter um turno atribuído;
15. Restrições entre turnos como, por exemplo, uma pessoa não poder estar em mais do que um turno em simultâneo;
16. O número de enfermeiros para cada turno (mínimo/máximo/número exato).

A formulação do problema é a combinação dos dados com as restrições e os objetivos [10]. A formulação depende ainda do tipo de problema, que pode ser um problema de otimização, de decisão ou de otimização de restrições [9]. Os vários tipos de problemas podem ter vários tipos de abordagens à função objetivo.

No que diz respeito às funções objetivo (FO), estas podem-se dividir em dois tipos, as funções objetivo padrão e funções de avaliação. As funções objetivo padrão são utilizadas por implementações baseadas em modelos matemáticos; as funções de avaliação são utilizadas para guiar ou avaliar a geração dos resultados da solução. As FO têm como objetivo maximizar ou minimizar o valor da solução dependendo do tipo de problema que é abordado [9]. No problema de NRP apresentado neste trabalho, a FO tem como objetivo minimizar os custos da alocação dos recursos. A complexidade das FO depende do problema, que pode ser mais simples ou mais complexo como se pode ver na contribuição de K. Nonobe e T. Ibaraki [10].

2.3. Abordagens propostas na literatura

Ao longo dos últimos 40 anos, são vários os trabalhos e abordagens que podem ser encontrados nesta área. Os primeiros trabalhos sobre NRP foram publicados em 1976, por Warner e Miller [2][11].

2.3.1. Métodos matemáticos exatos

Na literatura são várias as contribuições que propuseram abordagens de otimização baseadas em modelos matemáticos lineares como Warner [2], Abernathy et al. [12], Keller e Martel [13], entre outros [11][14][15].

Uma abordagem exata avalia a totalidade das possíveis soluções e garante que a solução encontrada é a ótima. Este tipo de abordagens implica a existência de elevados recursos computacionais para explorar todas as soluções possíveis existentes no espaço de pesquisa do problema.

Os métodos exatos não são os mais apropriados para os NRP modernos, visto que estes têm um espaço de pesquisa muito complexo devido à quantidade de variáveis que têm de ser controladas. A maioria dos investigadores que propõem abordagens matemáticas restringem a dimensão do problema a um pequeno conjunto de regras, o que torna o modelo desadequado à realidade dos hospitais [3].

2.3.2. Inteligência Artificial

As propostas de soluções para o NRP com recurso a técnicas de Inteligência Artificial (IA) começaram a ser formuladas a partir de 1980 [3]. Foram aplicadas técnicas de IA ao NRP, como programação por restrições [16][17] e sistemas periciais [18][19].

A programação por restrições é uma ferramenta poderosa para encontrar soluções viáveis. É uma técnica muito útil quando o número de restrições é elevado e uma solução viável que não a ótima é o suficiente. Esta técnica é a

menos provável de produzir uma solução ótima ou aproximada à solução ótima [20].

Os sistemas periciais permitem criar interfaces para o utilizador, integradas com metodologias de sistemas de apoio a decisão para o NRP [21]. Os sistemas de apoio à decisão são particularmente benéficos quando o problema envolve um grande conjunto de fatores humanos que não podem ser codificados no software e necessitam de ser especificados pelo utilizador no processo de geração [20].

Alguns trabalhos de investigação obtiveram resultados com algum sucesso como Chan e Weil [22] e Meyer auf'm Hofe [17].

Chan e Weil aplicaram um modelo cíclico ao mundo real, recorrendo a *Constraint Logic Programming*. Propuseram um modelo abstrato que considera múltiplos objetivos e um conjunto de várias restrições. Este modelo permite escalonar até 150 enfermeiros, controla a equidade entre eles e considera as suas preferências para ajudar a maximizar a satisfação dos mesmos e melhorar a qualidade do serviço [22].

Meyer auf'm Hofe baseou-se na sua experiência com o sistema ORBIS Dienstplan-System e propôs uma abordagem genérica baseada em processamento de restrições. Esta abordagem pode ser utilizada para implementar sistemas de escalonamento automático que podem ser adotados de maneira flexível em diferentes situações e em diferentes hospitais [17].

2.3.3. Heurísticas

Entre 1980 e 2000, por forma a resolver problemas reais, foram propostos métodos heurísticos como Smith, 1976 [23]; Smith e Wiggins, 1977 [24]; Schaerf e Meisels, 1999 [25]; entre outros [26][27].

Em 1976 Smith apresentou um algoritmo iterativo que auxilia o processo de construção de um escalonamento cíclico. O algoritmo tem em conta as necessidades e as regras de dias de folga para determinar o número de recursos a utilizar. Foi um etapa importante, contudo, esta heurística não pode

ser considerada uma solução prática para aplicar aos problemas que são propostos hoje em dia, devido à complexidade que foi sendo introduzida no desenvolvimento dos estudos em volta do NRP [3].

Smith e Wiggins desenvolveram um sistema que recorria a técnicas de processamento de listas para gerar escalas [24]. Este sistema é baseado numa abordagem mensal, não cíclica, ao contrário da proposta apresentada por Smith em 1976 [23]. Esta proposta permitia diferentes tipos de competências e diferentes tipos de trabalho. O sistema gerava escalas por pessoa, tendo em conta as necessidades do pessoal para o dia e os dias de folga dos recursos, implementava restrições como: padrões, dias de folga e tipo de contrato. Outro aspeto positivo que este sistema trouxe foi a interatividade que permitia ao utilizador fazer alterações nos horários gerados.

Schaerf e Meisels apresentaram uma descrição geral dos problemas de escalonamento de pessoal [25]. O problema trata da alocação de recursos a tarefas em turnos. Consideraram os turnos como períodos de tempo pré-definidos que podem ser colocados em qualquer parte do eixo do tempo. O modelo é rígido nas restrições de cobertura das necessidades, mas é flexível nas restrições de tempo, tentando atender às preferências dos recursos. Schaerf e Meisels introduziram uma pesquisa local que permitia alocações parciais, o que veio dar possibilidade a fazer uso de um espaço de pesquisa maior [3].

Quando se pretende resolver problemas de NRP reais, o tamanho destes, a falta de conhecimento das suas estruturas e a necessidade de obter soluções em curtos espaços de tempo, dificulta a aplicabilidade dos métodos exatos [3]. Nestas condições podem ser usadas heurísticas.

As heurísticas têm como objetivo gerar boas soluções de forma eficiente. Contudo, as heurísticas não encontram a solução ótima nem garantem que a solução encontrada é aproximada à solução ótima. Pode-se caracterizar as heurísticas como métodos apropriados para corridas de curto espaço de tempo, fáceis de implementar, flexíveis e simples. Geralmente são utilizadas para problemas complexos [28].

As heurísticas estabelecem um conjunto de etapas que têm o objetivo de gerar escalas que respeitam um conjunto de regras definidas [29]. Para aplicar algoritmos de escalonamento heurísticos ao NRP é necessária uma formulação clara das necessidades do hospital [3].

Dentro das heurísticas podem-se observar três categorias, sendo estas: 1) construtivas, 2) melhoramento e 3) compostas [30].

- 1) As heurísticas construtivas têm como objetivo construir uma solução admissível de forma iterativa. A cada iteração o método vai acrescentando novos elementos à solução, aproximando-se assim a cada etapa da solução admissível. A solução é válida quando todas as regras definidas previamente forem cumpridas. Este tipo de heurísticas é frequentemente utilizado como o método de geração de uma solução inicial, que posteriormente é utilizada como o ponto de entrada para outros métodos de pesquisa [30].
- 2) Uma heurística de melhoramento começa com uma solução inicial e a esta solução vai-se iterando e aplicando pequenas mudanças de forma a ir melhorando a solução. A procura de melhores soluções termina quando não for encontrada uma solução melhor do que a atual [30].
- 3) Uma heurística composta, é uma heurística constituída por duas fases, uma combinação de heurísticas construtivas com heurísticas de melhoramento. A primeira fase é chamada a fase construtiva onde basicamente é aplicado o conceito da heurística construtiva para gerar uma solução inicial. A segunda fase é chamada fase de melhoramento onde se aplica o conceito da heurística de melhoramento ao *input* gerado na primeira fase [30].

2.3.4. Meta-heurísticas

A partir de 1990 começaram maioritariamente a aparecer propostas de abordagens baseadas em meta-heurísticas como Bellanti et al., 2004 [31]; Burke et al., 2010 [32]; entre outros [33][34][35].

Bellanti et al. [31] apresentaram um problema de escalonamento prático que surge numa enfermaria de um hospital italiano. Para este problema, consideraram um escalonamento mensal. A solução proposta tenta gerir os requisitos contratuais e operacionais dos recursos. A solução tem em atenção ainda os meses em que a mão de obra é mais escassa devido a feriados sazonais. Bellanti et al. introduziram uma abordagem de pesquisa local baseada em vizinhanças que opera sobre soluções parcialmente completas em conjunto com o algoritmo *greedy*. O algoritmo *greedy* é responsável por evitar a geração de soluções inválidas.

Burke et al. [32] apresentaram uma abordagem de pesquisa por dispersão. A pesquisa por dispersão é um algoritmo evolutivo que tem sido aplicado com sucesso a uma série de domínios [32]. Para adaptarem a pesquisa por dispersão ao NRP, foi necessário fazerem o desenvolvimento de algumas sub-rotinas do método. Os resultados mostram que o algoritmo proposto é um método robusto e eficaz sobre uma ampla variedade de problemas reais [32].

A maioria das contribuições focam-se nas meta-heurísticas, visto que estas são consideradas como métodos modernos para a resolução de problemas de escalonamento complexos [3].

Ao longo dos últimos anos foram vários os investigadores que propuseram a definição do conceito de meta-heurísticas (MH) [36][37][38][39]. Uma MH é um método pensado para resolver problemas de otimização de forma genérica. As MH são um processo principal que guia e modifica as operações de uma heurística subordinada de forma inteligente e eficiente até chegar a uma solução de alta qualidade [39].

As MH não são desenhadas com o foco num problema em específico. Como já foi referido, são desenhadas de forma genérica para agirem de acordo com as definições estratégicas para as quais são programadas. São utilizadas como o

guia no processo de pesquisa. O objetivo base das MH é explorar o espaço de pesquisa de forma inteligente para encontrar uma solução o mais aproximada possível da ótima em tempo admissível, por isso, as MH são classificadas como algoritmos aproximados e não determinísticos [28].

Na literatura podemos encontrar várias MH como Algoritmos Genéticos [40] [41][42][43], Pesquisa em Vizinhanças Variáveis [44], Simulated Annealing [40][45], Pesquisa Scatter [32], Método Memetic [46], Algoritmos Evolutivos [47], Algoritmos Otimização de Colônias [40], Pesquisa Tabu [33][40][48], entre muitos outros. Contudo, na revisão da literatura para o NRP, as meta-heurísticas mais relevantes são o Simulated Annealing, a Pesquisa Tabu e os Algoritmos Genéticos.

O Simulated Annealing foi proposto por Kirkpatrick, Gelett e Vecchi [49] e Cerny [50]. É um método probabilístico que permite encontrar mínimos globais de uma função de custos que pode ter vários mínimos locais [51]. É em especial uma meta-heurística de otimização de aproximação global para grandes espaços de pesquisa. Este método foi inspirado no processo *annealing* da indústria metalúrgica, que é um processo que envolve o aquecimento e arrefecimento dos metais para aumentarem o tamanho de forma controlada e reduzir os seus defeitos [51].

A Pesquisa Tabu foi introduzida por Glover [48] e é uma estratégia para resolver problemas de otimização combinatória. É um procedimento adaptativo com a capacidade de fazer uso de outros métodos, tais como algoritmos de programação linear e heurísticas especializadas. A pesquisa tabu auxilia os outros métodos a escapar de ótimos locais.

Os Algoritmos Genéticos foram propostos por John Henry Holland em 1970. Têm por objetivo encontrar a melhor solução, combinando de forma iterativa um conjunto de soluções iniciais, geralmente geradas de forma aleatória. A cada iteração pretende-se que de alguma forma a solução anterior seja melhorada. É considerado um caso particular dos algoritmos evolutivos. A solução final (melhor solução) cumpre todas as restrições definidas para o problema [41].

2.3.5. Principais conclusões da revisão da literatura

Os métodos exatos, são apropriados para encontrar a solução ótima. No que toca ao NRP, a sua simplicidade limita a própria abordagem, dada a quantidade de restrições e objetivos que compõem os NRP modernos [3]. O NRP é um problema mais complexo do que o “Problema do Caixeiro Viajante” [50]. Para a maioria dos problemas, encontrar a solução ótima não é só completamente impraticável como é algo sem sentido [52]. Os administradores dos hospitais não procuram a solução ótima, mas sim gerar o mais rápido possível uma solução de qualidade que cumpra o máximo de objetivos especificados [3].

A visão dos administradores dos hospitais vai mais de encontro às abordagens das heurísticas e meta-heurísticas. As heurísticas são mais adequadas aos problemas do mundo real do que propriamente os métodos exatos, devido ao tamanho do NRP e à falta de conhecimento que existe sobre a estrutura dos mesmos. Os escalonadores heurísticos traçam uma série de etapas por forma a gerar a solução que mais respeite as restrições [29], sendo a maioria desenvolvidos de forma cíclica, ou seja, recorrem a um método de tentativa/erro por forma a encontrar a melhor solução. A maioria dos trabalhos publicados consideram as meta-heurísticas como uma abordagem mais moderna para encontrar soluções em problemas de escalonamento. As meta-heurísticas conseguem gerar soluções aceitáveis mesmo nos casos em que a carga de restrições é extremamente elevada e em casos em que encontrar a solução viável é muito difícil [3].

3. Problema

O NRP é um problema importante na área da saúde, como já foi referido anteriormente. Durante o ano (2015), decorreu o concurso “*Second International Nurse Rostering Competition*”³. A empresa Bullet Solutions⁴ entrou nesta corrida com os objetivos de resolver o problema proposto pela competição e em simultâneo criar uma base para a resolução de problemas de escalonamento reais.

Com o concurso foi introduzido o conceito de fases que exige aos concorrentes resolver uma sequência de casos referentes a semanas consecutivas num horizonte de planeamento. O problema é definido por um cenário, um conjunto de semanas e uma primeira descrição do histórico. Os problemas são resolvidos em fases de 1 semana no tempo, cobrindo o cenário total de 4 ou 8 semanas consecutivas. Após cada fase, as informações para a fase seguinte ficam disponíveis [53].

3.1. Conceitos fundamentais

O problema consiste num escalonamento por etapas para um conjunto de recursos que pode laborar num conjunto de turnos a cada dia ou então ter o próprio dia de folga. Os recursos têm um conjunto de competências e cada competência permite ao recurso dar resposta a diferentes necessidades.

De seguida são enumerados os conceitos base considerados no problema.

1. **Horizonte de planeamento.** É o intervalo de tempo sobre o qual é feito o escalonamento dos recursos. O horizonte de planeamento pode estar dividido em várias etapas. Este é utilizado para que seja possível controlar e balancear objetivos globais (ex: o total de atribuições de trabalho) nas várias etapas.

³ <http://mobiz.vives.be/inrc2/>

⁴ <https://www.bulletsolutions.com/>

2. **Competências.** As competências referem-se ao nível de qualificação, competências ou responsabilidades (ex.: enfermeiro chefe, enfermeiro, auxiliar, entre outros). A cada recurso está associado um conjunto de competências que ajudam a mapear os recursos com os turnos que estes podem realizar.
3. **Turno.** O número de turnos em que o dia está dividido é dinâmico. Os turnos considerados por defeito são Manhã, Tarde, Noite e Dia. Os turnos não têm a necessidade de ter uma hora associada pois são controlados através da formulação de sucessões de turnos proibidos. As sucessões de turnos proibidos controlam situações como, por exemplo, se um colaborador está alocado no turno da noite, este colaborador não pode ser alocado no turno da manhã do dia seguinte, para garantir que o recurso descansa entre turnos.
4. **Recursos.** São os funcionários. Cada recurso pode ter uma ou mais competências e tem um contrato associado.
5. **Contrato.** É o conjunto de regras que permitem limitar o escalonamento do recurso. Por exemplo, um contrato de tempo parcial limita a metade os dias de trabalho do recurso no horizonte de planeamento.
6. **Necessidade.** É uma combinação do dia, com o turno e a competência. Uma necessidade pode requerer que o recurso tenha uma ou mais competências para poder ser alocado.
7. **Preferências dos Recursos.** É a definição das regras pessoais de determinado recurso. O recurso pode preferir não trabalhar um turno de determinado dia ou mesmo não trabalhar o dia completo.

A Figura 1 mostra um mapa conceptual dos conceitos fundamentais, por forma a expressar visualmente as relações entre os vários conceitos incluídos no problema: horizonte de planeamento, recursos, necessidades, turnos, contratos, preferências e competências.

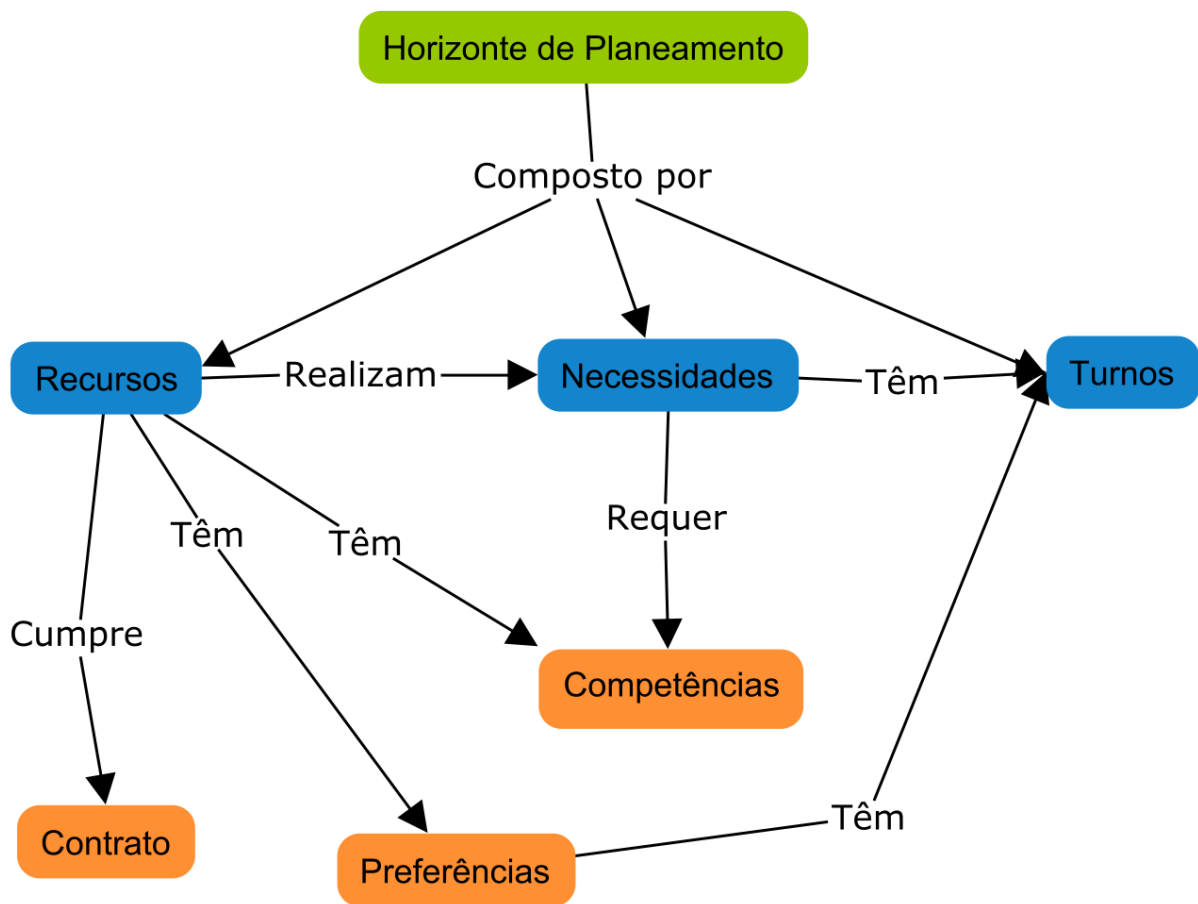


Figura 1 - Mapa de conceitos associados ao problema de NRP em estudo

3.2. Restrições

A lista abaixo apresentada identifica o conjunto de restrições que o algoritmo tem de respeitar durante o processo de geração. As restrições são verificadas durante o processo de geração em cada etapa do horizonte do planejamento.

1. Atribuir um e apenas um turno por dia a cada recurso. O turno pode ser de trabalho ou uma folga.
2. Cumprir as necessidades mínimas definidas para cada turno, por cada competência.

3. Respeitar as sucessões de turnos. Por exemplo, se um enfermeiro trabalhar no turno da noite, no dia seguinte não pode trabalhar no turno da manhã.
4. Atribuir a cada necessidade apenas recursos que tenham a competência necessária para a desempenhar.

3.3. Objetivos

Os objetivos podem ser divididos em duas categorias: semanais e globais. Os objetivos semanais são avaliados a cada semana, enquanto os objetivos globais são avaliados quando toda a solução tiver sido gerada. Na seguinte lista são apresentados os objetivos semanais, sendo os dois últimos objetivos globais (ponto 6 e 7).

1. Atribuir o número ótimo de recursos solicitados pelas necessidades. Por cada recurso a menos é atribuída uma penalização de 30 pontos. Atribuir recursos extra não penaliza.
2. Respeitar o número mínimo e máximo de dias de trabalho consecutivos. Caso não sejam respeitados, são aplicadas as seguintes penalizações: 15 pontos por cada dia que fique abaixo do mínimo; 30 pontos por cada dia que fique acima do máximo.
3. Respeitar o número mínimo e máximo de dias de descanso. Em caso de incumprimento é aplicada uma penalização de 30 pontos por cada dia a mais ou a menos.
4. Respeitar as preferências dos recursos. Por cada preferência não respeitada é atribuída uma penalização de 10 pontos.
5. Cumprir fins-de-semana completos de trabalho ou descanso. Se o contrato de um recurso especificar que prefere trabalhar o fim-de-semana completo, é atribuída uma penalização de 30 pontos se o mesmo só trabalhar um dia.

6. Respeitar os limites mínimos e máximos do total de dias de trabalho. Caso o número total de dias de trabalho não respeite o limite superior ou inferior é aplicada uma penalização de 20 pontos por cada dia a mais ou a menos.

7. Respeitar o total de fins de semana de trabalho. Não respeitar o número máximo de fins-de-semana de trabalho penaliza 30 pontos por cada fim-de-semana extra. Caso o recurso só trabalhe o sábado ou o domingo é contabilizado como 1 fim-de-semana de trabalho.

A melhor solução é aquela que, respeitando todas as restrições, apresente um total de pontos de penalização inferior.

4. Proposta de solução

4.1. Abordagem

A meta principal para os algoritmos é a possibilidade de aplicar os mesmos a casos reais. A infraestrutura de suporte tem de ser extensível e de fácil manutenção, para que os algoritmos possam ser adaptados a diferentes problemas de escalonamento de pessoal através de pequenas adaptações. Nesta abordagem utilizou-se programação funcional (FP), a pedido da Bullet Solutions. Os seguintes capítulos abordam o paradigma funcional e alguns conceitos utilizados no desenvolvimento dos algoritmos.

4.1.1. Paradigma Funcional

A PF é um paradigma que trata a lógica da programação como a avaliação de funções matemáticas. Uma função matemática define um mapeamento entre os valores do domínio e os valores da imagem. Da mesma forma, uma função em PF para um dado valor de entrada deve obter sempre o mesmo valor de saída. A PF tem como objetivo evitar estados ou dados mutáveis tornando assim mais fácil a programação paralela e a gestão de concorrência sobre os dados [54].

Tipicamente, uma linguagem funcional é identificada pelas seguintes características [54]:

1. Dados imutáveis;
2. Possibilidade de declarar funções;
3. Tratar as funções como dados;
4. Avaliação de funções *lazyloading*;
5. Correspondência de padrões.

Existem várias linguagens funcionais como APL, Lisp, ML, Haskell, OCaml e F#. Por questões de maior compatibilidade com os sistemas da empresa Bullet Solutions a linguagem funcional escolhida foi o F#. A linguagem F# pertence à *framework* .NET da Microsoft. A escolha da mesma prende-se com o facto dos restantes produtos da empresa estarem desenvolvidos sobre esta *framework*.

A linguagem F# não é uma linguagem de PF pura. Pode-se dizer que é uma linguagem híbrida que nos permite aproveitar o que de melhor existe na PF e na programação imperativa. É uma linguagem que tem o objetivo de trazer as vantagens do paradigma funcional para escrever código simples na resolução de problemas complexos [55].

Na campo do desenvolvimento de produtos no contexto empresarial, é uma linguagem vista como um ponto para agilizar o desenvolvimento, ou seja, tem o objetivo de reduzir o tempo de desenvolvimento de componentes de software. A F# é uma linguagem interoperável com as restantes linguagens da *framework* .NET. Podemos tirar partido das vantagens das linguagens orientadas a objetos, como o C#, aliadas ao paradigma funcional.

4.1.2. Dados mutáveis e imutáveis

Um dos conceitos importantes aplicado neste trabalho é a imutabilidade dos dados. Como já referido, a linguagem F# não é uma linguagem de PF pura, por isso permite combinar a utilização de dados mutáveis e imutáveis. Em F#, por defeito, as variáveis declaradas são definidas como imutáveis. Isto significa que o seu valor não pode ser alterado.

```
1 let x = 3 + 4
2
3 let y = x + 6
4 |
```

Figura 2 - Declaração de dados imutáveis

Olhando para a Figura 2 observa-se que *x* tomou o valor de 7 (visto que o F# atribui a *x* o resultado da expressão $3 + 4$). A partir do momento que a *x* é atribuído um valor, já não é possível voltar a alterar o mesmo em tempo de execução.

```
1 let x = 25
2
3 let x = 48 + 10
4
```

Figura 3 - Atribuição de variáveis imutáveis

A linguagem F# não aceita a atribuição duplicada a variáveis imutáveis. Por exemplo, o caso apresentado na Figura 3 não é possível correr porque o compilador do F# alerta imediatamente da existência do erro e não deixa compilar a aplicação. Contudo, é possível contornar esta situação com a utilização de dados mutáveis. As variáveis mutáveis permitem a alteração do seu valor após a declaração da mesma.

```
1 let mutable z = 10
2
3 z <- z + 2
4
```

Figura 4 - Declaração de dados mutáveis

A alteração do valor de variáveis em F# exige que estas sejam identificadas como mutáveis aquando da sua declaração. A Figura 4 mostra como é feita essa declaração. Antes do nome da variável é acrescentada a palavra *mutable* e isto vai indicar ao compilador que é possível alterar o valor da mesma. Todas as alterações são efetuadas através do operador de atribuição (<-) específico para variáveis mutáveis e não com o operador igual (=) como é habitual na maioria das linguagens. Neste caso, z é inicializado com o valor 10 e é atualizado posteriormente para o valor 12.

Neste projeto evitou-se ao máximo o uso de dados mutáveis a nível global para se conseguir aplicar a filosofia base das funções matemáticas. Com isto, é possível paralelizar, tanto parcialmente, como na sua totalidade, os algoritmos.

4.1.3. Gestão da solução

Com o desenvolvimento de métodos de pesquisa com base em linguagens orientadas a objetos (C#) acresce a dificuldade de gerir a solução. Quando alteramos um objeto, as alterações são propagadas a todos os pontos onde este objeto é referenciado.

O processo de pesquisa de novas soluções obriga a alterar a solução para avaliarmos novas opções. A alteração de uma estrutura de uma solução traz a necessidade de armazenar todas as alterações efetuadas numa avaliação de um vizinho próximo à solução atual para que seja possível reverter, caso não seja encontrada uma solução melhor.

Para gerir a solução recorreu-se ao conceito da imutabilidade e à definição de funções. Como já referido, num paradigma funcional, uma função faz o mapeamento de um valor de entrada para um valor de saída, garantindo que para um dado valor de entrada é devolvido um determinado valor de saída.



Figura 5 - Execução de alterações na solução

A gestão da solução vai ser efetuada dessa mesma forma. Tomando o exemplo da Figura 5, ao aplicar o algoritmo à Solução A, dá origem a uma Solução A'. Seguindo este método, mantém-se o estado da Solução A e gera-se a Solução A'. Isto permite que a qualquer momento seja possível ignorar uma solução, se ela não for melhor que a anterior. Olhando para esta forma de tratar a solução, é possível ainda conseguir paralelizar a procura de novas soluções a partir de determinada solução, como mostra o ponto 4.1.4.

4.1.4. Processamento paralelo

Um dos objetivos deste projeto foi criar uma estrutura que permitisse o processamento paralelo nos algoritmos. Tendo em conta a gestão da solução apresentada no ponto 4.1.3, podem-se aplicar vários algoritmos a uma solução, visto que um algoritmo não altera diretamente a solução, mas sim gera uma nova solução.

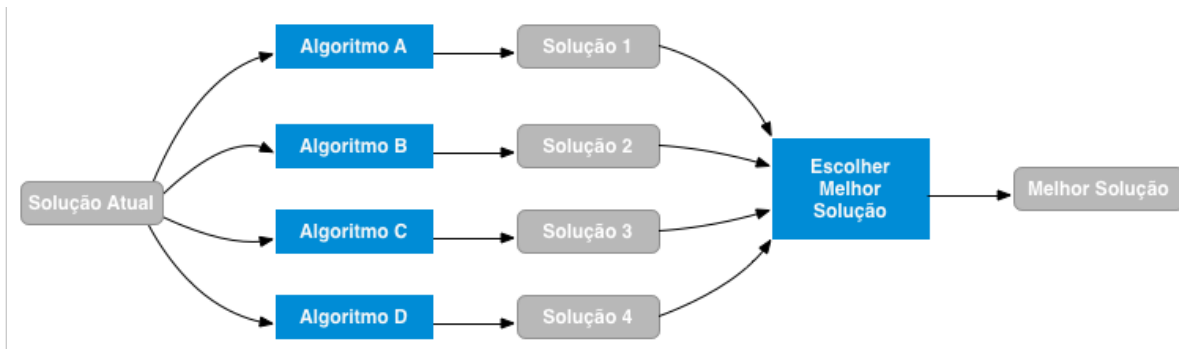


Figura 6 - Conceito de processamento paralelo

O exemplo da Figura 6 mostra o conceito de processamento paralelo que esta abordagem permite aplicar. Para uma solução de entrada, neste caso a Solução Atual, podem ser aplicados vários algoritmos em paralelo. Os resultados obtidos de cada um dos algoritmos aplicados são comparados e, a partir daí, resulta a escolha da melhor solução conhecida, passando esta a ser a solução atual.

4.2. Arquitetura da solução

No desenho desta arquitetura (Figura 7) teve-se sempre em foco a construção de um *solver* modular que permita de forma fácil desenhar e substituir módulos por outros mais evoluídos.

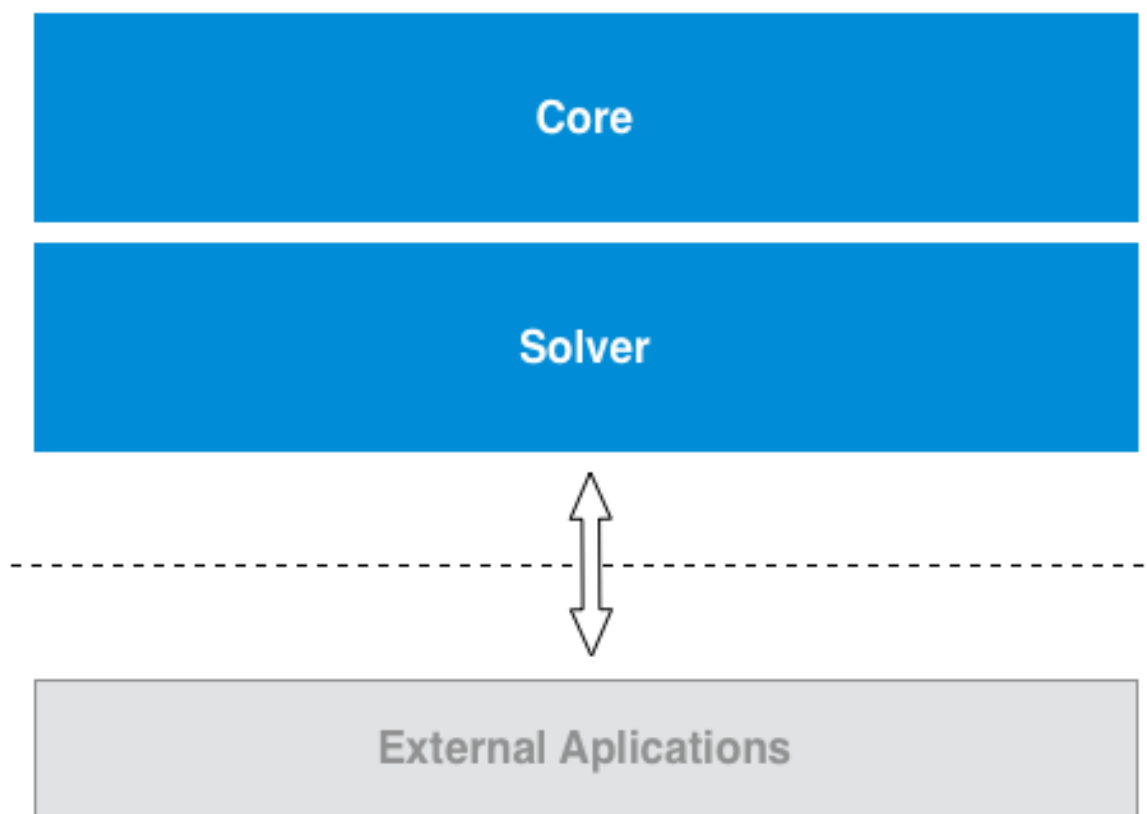


Figura 7 - Arquitetura global da solução

O software encontra-se estruturado em duas camadas: 1) *solver* e 2) *core*. A camada do *solver* é constituída pelos componentes que têm a necessidade de ser ajustados à medida do problema a ser resolvido. A forma de converter os dados pode ser adaptada a novas fontes e estruturas de informação, o que torna o *solver* mais flexível. A camada do *solver* também é responsável por comunicar com as aplicações externas à solução. A camada do *core* suporta os componentes de geração, ou seja, é nesta camada que se encontram os algoritmos de construção da solução inicial, os algoritmos de otimização, entre outros componentes que serão especificados mais à frente neste documento.

4.2.1. A camada *solver*

A camada *solver* (Figura 8) divide-se em três componentes: **Data Manager**, **Parser** e **Main Algorithm**.

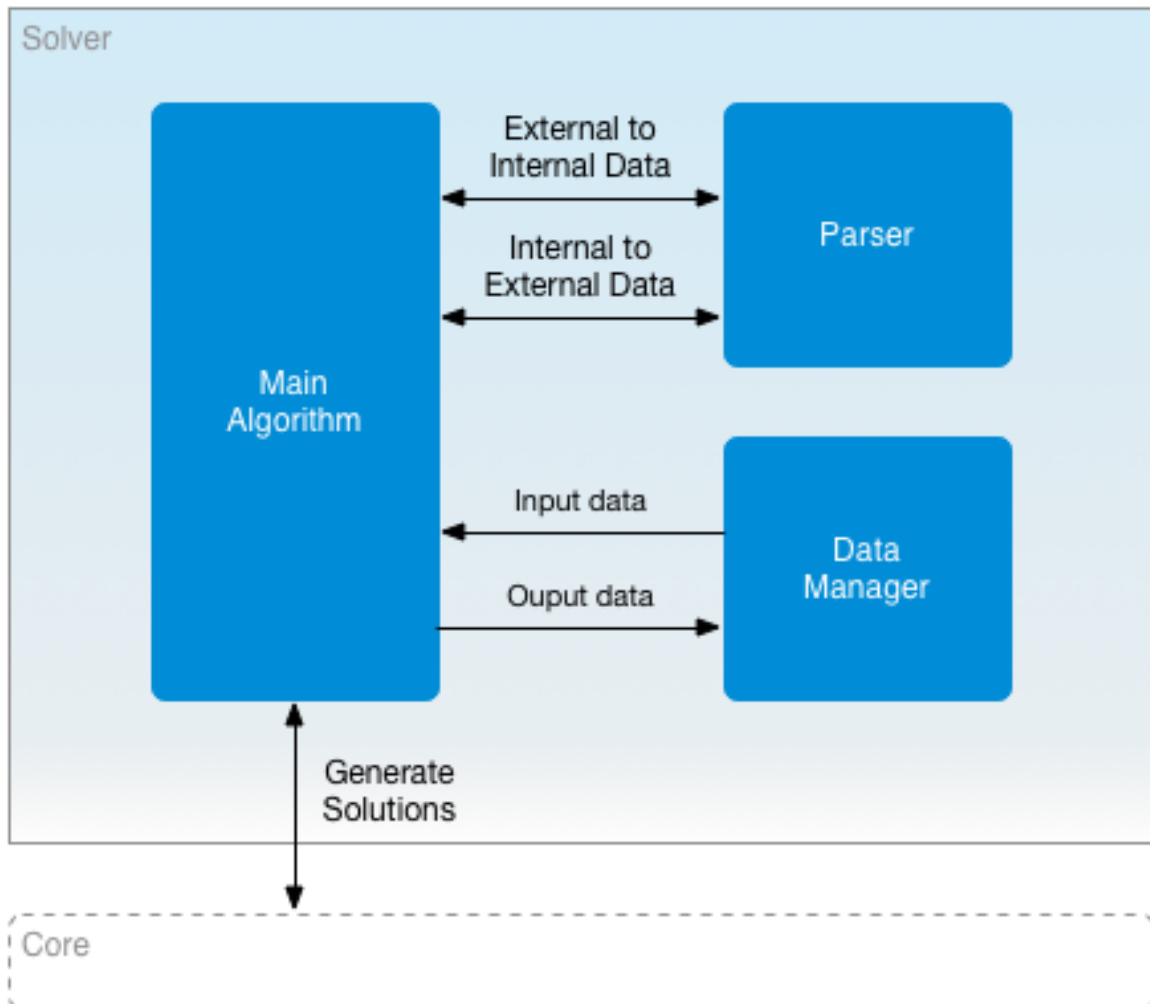


Figura 8 - Arquitetura da camada *solver*

O **Data Manager (DM)** é o componente responsável por gerir a comunicação do *solver* com o exterior. Este componente permite ler um problema para gerar a solução, publicar a solução e informação customizada sobre a geração. O DM pode ter várias implementações e, atualmente, está implementado para importar e exportar os dados para ficheiros JSON.

O **Parser** é o componente com a responsabilidade de traduzir a informação externa para informação útil ao *solver*. Os dados resultantes das conversões do

parser são a estrutura base da solução com as restrições, as necessidades e os recursos envolvidos no processo. O *parser* utiliza a biblioteca json.NET para ler a informação externa e transforma-a num mapa de dados para alimentar a solução.

O **Main Algorithm** é a peça principal desta camada. Este algoritmo começa por recolher os dados externos para começar a construir a estrutura para resolver o problema. Depois de ler os dados externos entrega os mesmos ao *parser* para que este faça as conversões necessárias e devolva o “ponto zero” da solução. O ponto zero, na sua essência, é a estrutura preenchida com as restrições, recursos, turnos e as necessidades. Após estar gerada a estrutura da solução, o algoritmo base entrega a mesma ao componente da camada *core* (analisada no ponto 4.2.2) responsável por construir a solução inicial. Em seguida, depois de obtida a solução inicial, segue para a fase de otimização. A fase de otimização termina assim que obedecer a todos os critérios de paragem.

4.2.2. A camada core

A camada core (Figura 9) é constituída por 9 componentes: *Domain*, *Domain Functions*, *Randomness*, *Urgency*, *Weight*, *Impossibility*, *Exception*, *Optimization* e *Building*.

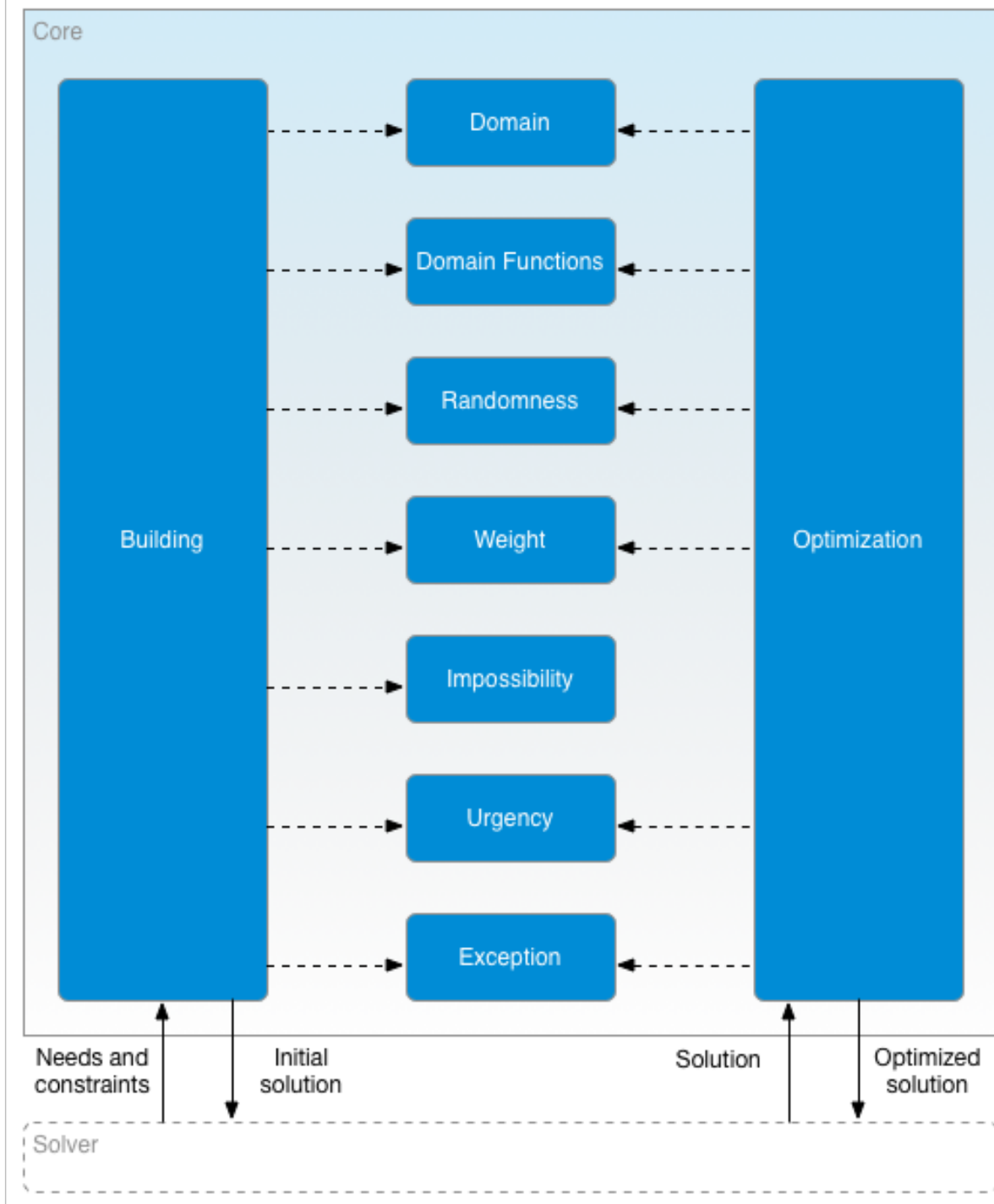


Figura 9 - Arquitetura da camada core

O componente **Domain** é o componente que agrupa todas as estruturas internas sobre as quais os algoritmos vão trabalhar. Estas estruturas auxiliam a fase de construção e otimização da solução. Ao componente **Domain** está diretamente acoplado o componente **Domain Functions**. Este permite fazer a separação das ações base que podem ser aplicadas sobre as estruturas de domínio. Optou-se por agrupar a lógica de domínio num componente à parte, por forma a facilitar futuras manutenções no código.

O componente **Randomness** é responsável pela aleatoriedade. A introdução deste componente, além de facilitar o *debug* do código na medida em que ajuda facilmente a replicar erros, veio também tornar possível a replicação exata de soluções através de uma *seed* numérica. Para ser possível replicar soluções, em cada geração só é criada uma única estrutura aleatória que é mantida até ao final da execução e, assim, através da *seed*, ela devolve sempre os mesmos números e no final obtém-se exatamente a mesma solução. Este componente é utilizado em casos de empate, para escolher recursos ou turnos aleatoriamente. O componente **Urgency** é um dos que usa mais o **Randomness**. Este componente é responsável por estabelecer a ordem de prioridade com que os recursos são colocados e, em caso de haver um empate entre recursos, este componente recorre a aleatoriedade para escolher um deles.

A avaliação das soluções é feita através do componente **Weight** e é aqui que é mantida a função objetivo. Neste componente existem dois tipos de avaliação: global e local. Para cada uma das avaliações é mantida uma lista de objetivos ativos, o que permite gerir o que é avaliado a cada momento. Sendo assim, é possível adicionar e retirar objetivos em tempo de execução. A avaliação global é feita a toda a solução e a avaliação local é efetuada somente às partes da solução afetadas por uma alteração. Este componente é utilizado na fase de construção e na fase de otimização.

Quando uma geração é impossível de resolver na fase de construção, a solução é entregue ao componente **Impossibility**. Este componente agrupa um conjunto de métodos para tentar encontrar alternativas de forma mais inteligente. Caso a solução seja declarada impossível por este método, então a geração termina, pois não foi possível encontrar uma solução válida.

A fase de construção da solução acontece no componente **Building**. Este componente é constituído por um conjunto de funções que alocam os recursos às necessidades para formar a solução inicial. É considerada uma solução inicial quando todas as restrições do problema são satisfeitas, ou seja, quando se chega a uma solução viável.

A fase de otimização é implementada no componente **Optimization**. Este componente é construído por um conjunto de algoritmos e por um conjunto de vizinhanças que são executadas pelos algoritmos. Um algoritmo executa uma iteração sobre a solução e devolve a melhor solução encontrada para a iteração. A camada *solver* é responsável por decidir quando parar a fase de iteração, visto que é esta que dá a ordem de otimização para a solução.

4.3. Especificação dos algoritmos

Como já foi referido, o algoritmo é constituído por duas fases: construção e otimização. Estas duas fases são suportadas por uma estrutura de dados que assegura que as restrições do problema são respeitadas.

Na fase de construção da solução inicial (Figura 10) é utilizada uma heurística construtiva. É uma heurística sequencial, que resolve as necessidades de acordo com a sua urgência em cada iteração. Foram definidos 3 critérios de urgência diferentes. A ordem estabelecida por estes critérios de urgência é baseada no número de recursos disponíveis para alocar a uma necessidade (N_a) e o número de recursos que ainda faltam alocar para cumprir uma necessidade (N_r). Os 3 critérios estabelecem a ordem de prioridade calculando a distância de N_a a N_r . É utilizada a distância relativa, a distância absoluta e uma mistura das duas primeiras (Figura 10 (1)).

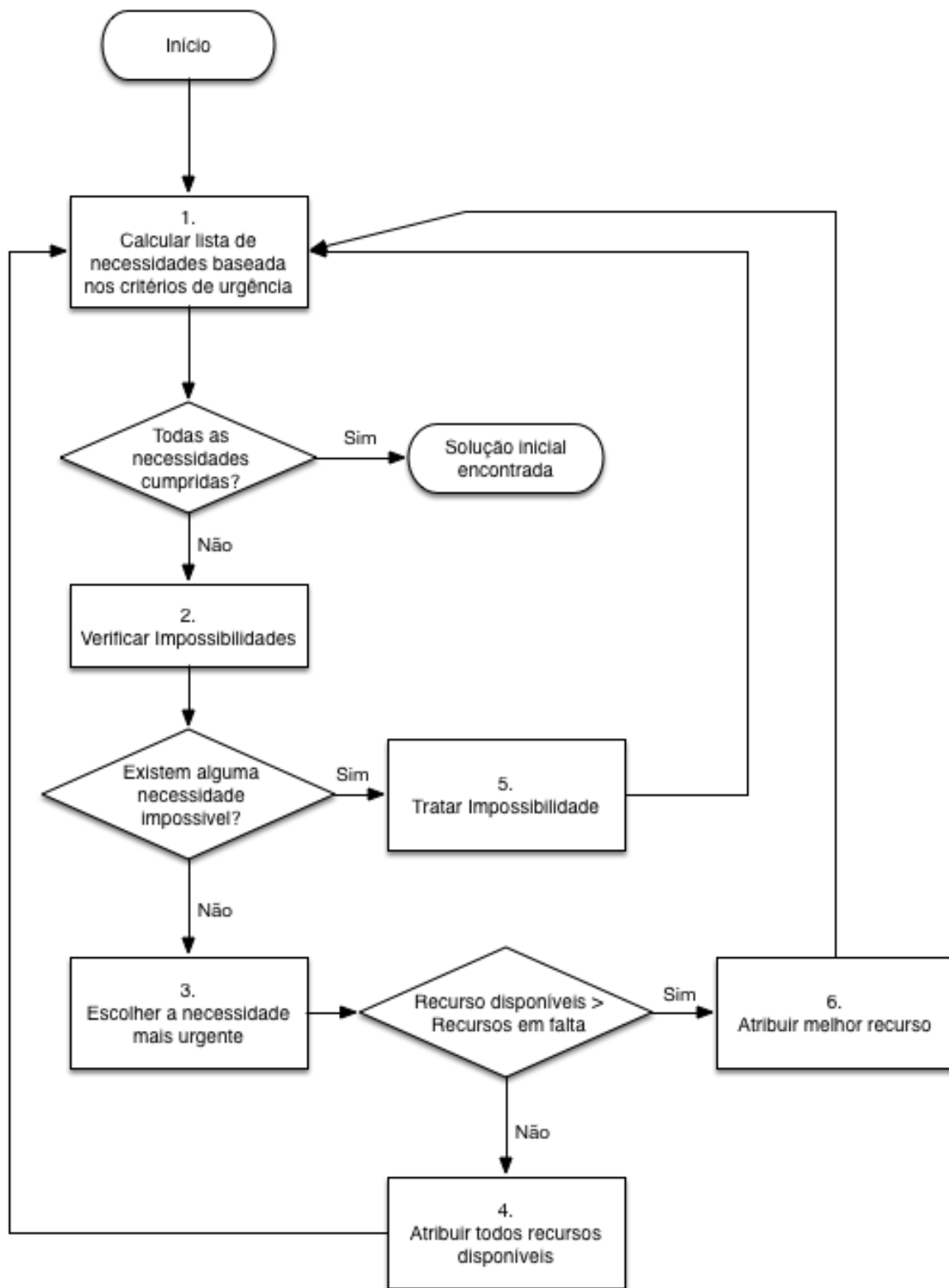


Figura 10 - Fluxograma da construção da solução inicial

São realizadas as validações necessárias para detetar as impossibilidades (Figura 10 (2)). Se existir alguma impossibilidade, tenta corrigir a mesma (Figura 10 (5)), caso contrário segue-se a seleção da necessidade mais urgente. O melhor recurso disponível para ocupar o lugar é alocado, de acordo com a função objetivo (como mencionado anteriormente, utilizando esta estrutura de dados, somente os recursos válidos estão disponíveis). Então, o critério de urgência volta a ser recalculado e é repetido o processo. Esta fase termina quando é encontrada uma solução viável, sendo que a solução é considerada viável quando são atingidos os critérios de paragem. Os critérios de paragem são todas as restrições que não podem ser controladas através da estrutura de dados como, por exemplo, a restrição de atingir os mínimos das necessidades (restrição 2, definida no capítulo 3.2).

São calculadas 3 soluções iniciais (cada uma com um dos critérios de urgência) e é utilizada a melhor para iniciar a fase de otimização. Uma vez encontrada a solução inicial para o problema (solução viável), é iniciada a fase de otimização. A Figura 11 mostra o processo de otimização da solução.

A pesquisa por novas soluções é baseada em estruturas de vizinhanças. Foram criadas 12 estruturas de vizinhanças (cada uma focada em cada um dos objetivos do problema). A cada iteração, o melhor vizinho encontrado é aceite se melhorar a solução atual. Os vizinhos são avaliados através da função objetivo, sendo que quanto menor for o valor da função objetivo, melhor é a solução. As 12 vizinhanças correm em modo sequencial até que a solução comece a ficar estável (Figura 11 (1)).

Quando a solução se torna estável, se o tempo for suficiente, o processo é repetido desde o princípio, uma ou mais vezes. O tempo da primeira corrida completa é controlado e comparado com o tempo restante, o que ajuda a determinar o número de vezes que o processo deve correr. Contudo, apesar do número de vezes que o processo é repetido, pelo menos 40% do tempo é reservado para executar uma fase de diversificação. A melhor solução encontrada até este ponto do processo é utilizada como ponto de partida da diversificação.

Na etapa de diversificação (Figura 11 (2)), por meio da adulteração temporária da função objetivo, é possível alterar ligeiramente a solução, enviando-a para

outra zona do espaço de pesquisa. Em seguida, as 12 estruturas de vizinhança são executadas, de um modo sequencial, até que a solução se torne estável novamente (Figura 11 (3)). O processo de diversificação repete-se até que o tempo se esgote.

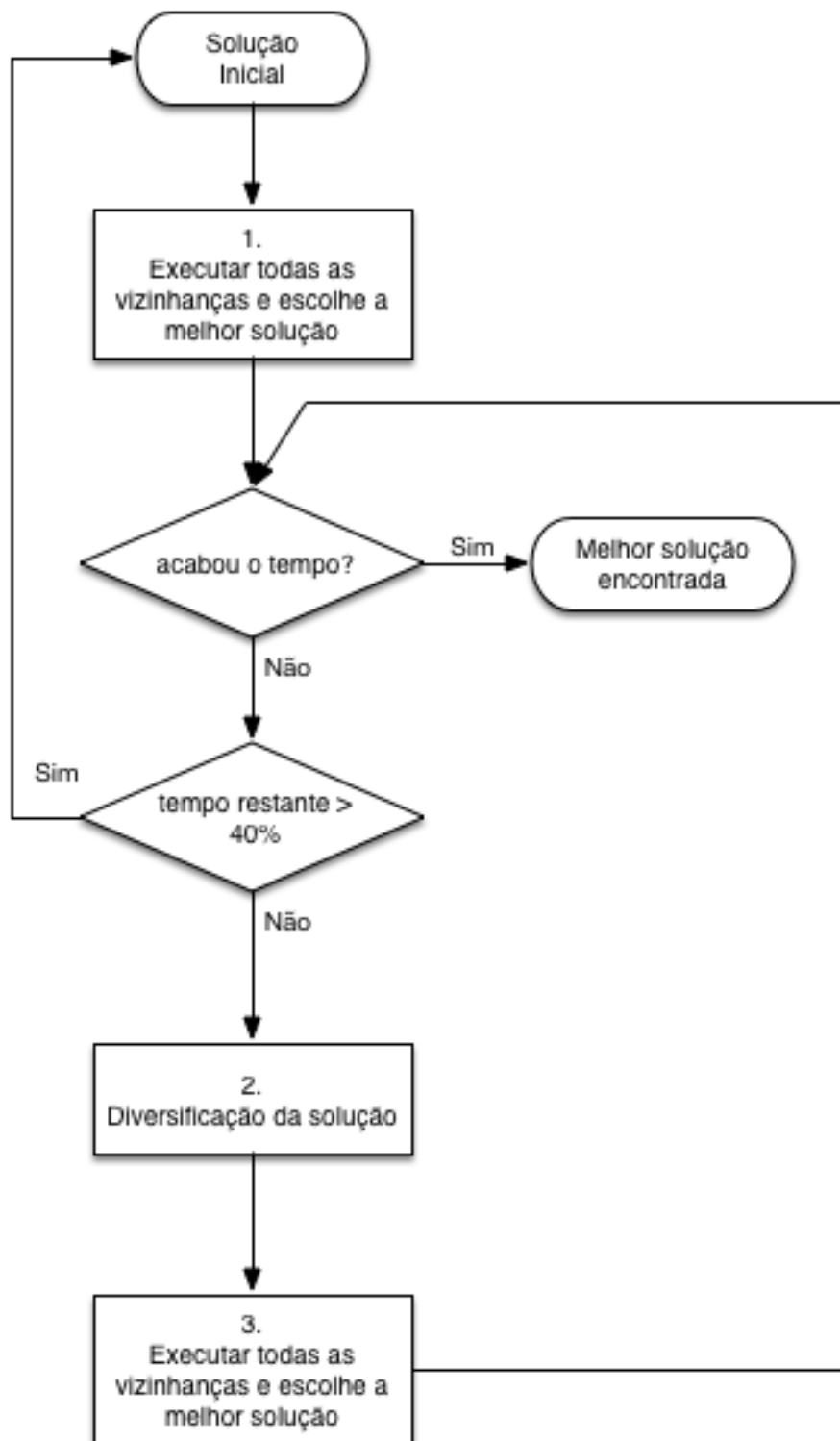


Figura 11 - Fluxograma da otimização da solução

5. Caso de Estudo

Este capítulo apresenta uma descrição do caso de estudo do concurso “*Second International Nurse Rostering Competition*”. O caso de estudo tem algumas particularidades em relação ao problema apresentado no **capítulo 3**.

Em concreto, o problema consiste em gerar uma escala semanal para um número fixo de recursos e para um conjunto de turnos, ou seja, de semana para semana não existem alterações de colaboradores nem de turnos. Diariamente só pode ser alocado a cada recurso um turno ou uma folga. Um recurso pode ter várias competências, sendo que, cada competência permite alocar o recurso a diferentes conjuntos de turnos. Cada necessidade requer apenas uma competência.

5.1. Descrição do problema

O concurso propôs aos participantes um problema com um pequeno conjunto de restrições. As mesmas restrições foram tomadas como base de trabalho para a especificação e desenvolvimento dos algoritmos. As restrições encontram-se detalhadas nos **subcapítulos 3.2 e 3.3**.

Para aumentar a dificuldade, a organização do concurso optou por propor um problema de múltiplas etapas. Cada etapa corresponde ao escalonamento do trabalho para uma semana. O problema pode ter um horizonte de planeamento (HDP) de 4 ou 8 semanas. O HDP é gerado através da chamada sequencial das várias etapas que constituem o mesmo. A ordem de geração para um HDP de 4 semanas é 1^a, 2^a, 3^a e por último a 4^a semana. Cada semana de geração corresponde a uma execução do algoritmo apresentado no **capítulo 4**. As execuções começam sempre com base nos ficheiros de histórico, do cenário, da semana e num ficheiro de dados personalizados (sendo este último opcional). Os ficheiros utilizam a notação JSON.

O cenário (Figura 12) descreve os dados comuns a todas as semanas do processo de geração. O cenário contém informações sobre o horizonte de planeamento, a lista de competências, os contratos, os recursos, os turnos e as sequências de turnos proibidas.

```

{
  "id": "n030w4",
  "numberOfWeeks": 4,
  "skills": [
    "HeadNurse",
    "Nurse",
    "Caretaker",
    "Trainee"
  ],
  "shiftTypes": [
    {
      "id": "Early",
      "minimumNumberOfConsecutiveAssignments": 2,
      "maximumNumberOfConsecutiveAssignments": 5
    },
    { },
    { }
  ],
  "forbiddenShiftTypeSuccessions": [
    { },
    {
      "precedingShiftType": "Late",
      "succeedingShiftTypes": [
        "Early"
      ]
    },
    {
      "precedingShiftType": "Night",
      "succeedingShiftTypes": [
        "Early",
        "Late"
      ]
    }
  ]
},
{
  "contracts": [
    {
      "id": "FullTime",
      "minimumNumberOfAssignments": 15,
      "maximumNumberOfAssignments": 22,
      "minimumNumberOfConsecutiveWorkingDays": 3,
      "maximumNumberOfConsecutiveWorkingDays": 5,
      "minimumNumberOfConsecutiveDaysOff": 2,
      "maximumNumberOfConsecutiveDaysOff": 3,
      "maximumNumberOfWorkingWeekends": 2,
      "completeWeekends": 1
    },
    { }
  ],
  "nurses": [
    {
      "id": "HN_0",
      "contract": "FullTime",
      "skills": [
        "HeadNurse",
        "Nurse",
        "Caretaker"
      ]
    },
    { },
    {
      "id": "TR_29",
      "contract": "HalfTime",
      "skills": [
        "Trainee"
      ]
    }
  ]
}

```

Figura 12 - Exemplo do ficheiro de cenário

O ficheiro de semana (Figura 13) contém os dados relativos a uma semana e facultar informações como as necessidades e as preferências dos recursos. As necessidades são definidas para cada turno, por cada competência, para cada dia da semana. As necessidades estabelecem o mínimo e o ótimo para atingir as metas de trabalho.

As preferências dos recursos são um conjunto de tuplos formados pelo nome do recurso, dia da semana e o nome do turno. As preferências assinalam os turnos que os recursos preferem não trabalhar. Se o nome do turno for substituído pela palavra "Any", significa que o recurso prefere não realizar nenhum dos turnos do dia.

```

{
  "scenario": "n030w4",
  "requirements": [
    {
      "shiftType": "Early",
      "skill": "HeadNurse",
      "requirementOnMonday": {
        "minimum": 0,
        "optimal": 0
      },
      "requirementOnTuesday": { },
      "requirementOnWednesday": { },
      "requirementOnThursday": { },
      "requirementOnFriday": { },
      "requirementOnSaturday": { },
      "requirementOnSunday": {
        "minimum": 0,
        "optimal": 0
      }
    },
    { },
    { }
  ],
  "shiftOffRequests": [
    {
      "nurse": "NU_15",
      "shiftType": "Late",
      "day": "Tuesday"
    },
    { }
  ]
}

```

Figura 13 - Exemplo do ficheiro de dados de semana

O histórico (Figura 14) contém informações que devem ser carregadas entre semanas, para que seja possível avaliar as restrições corretamente. O histórico para cada recurso contém informações sobre os dados fronteira e os contadores do recurso.

Nos dados fronteira, encontram-se informações sobre o último tipo de turno realizado, o número consecutivo de turnos do mesmo tipo e o número de dias consecutivos de trabalho ou de folga.

Os contadores do recurso fornecem informação do número total de dias de trabalho no HDP e o número total de fins-de-semana de trabalho.

```

{
  "week": 0,
  "scenario": "n030w4",
  "nurseHistory": [
    {
      "nurse": "HN_0",
      "numberOfAssignments": 0,
      "numberOfWorkingWeekends": 0,
      "lastAssignedShiftType": "Night",
      "numberOfConsecutiveAssignments": 4,
      "numberOfConsecutiveWorkingDays": 4,
      "numberOfConsecutiveDaysOff": 0
    },
    {
      "nurse": "HN_29",
      "numberOfAssignments": 0,
      "numberOfWorkingWeekends": 0,
      "lastAssignedShiftType": "Day",
      "numberOfConsecutiveAssignments": 2,
      "numberOfConsecutiveWorkingDays": 2,
      "numberOfConsecutiveDaysOff": 0
    },
    {
      "nurse": "NU_30",
      "numberOfAssignments": 0,
      "numberOfWorkingWeekends": 0,
      "lastAssignedShiftType": "None",
      "numberOfConsecutiveAssignments": 0,
      "numberOfConsecutiveWorkingDays": 0,
      "numberOfConsecutiveDaysOff": 2
    }
  ]
}

```

Figura 14 - Exemplo do ficheiro de histórico

Quanto ao fluxo de geração das semanas, este é suportado pelo simulador disponibilizado para o concurso. O mesmo simulador é responsável por fazer as chamadas ao algoritmo desenvolvido para gerar as semanas. O simulador faz a troca de informação entre as semanas através de ficheiros e é ainda o responsável por informar o algoritmo do tempo que pode utilizar para encontrar a solução em cada semana (na Figura 15 (a) é apresentado o processo de geração para um horizonte de planeamento de 4 semanas).

Por fim, o simulador entrega todos os ficheiros gerados no decorrer do processo ao validador (também disponibilizado pela organização) e este executa todas as avaliações e validações necessárias, gerando um relatório de avaliação que inclui a escala gerada, o peso de cada uma das restrições e

objetivos e o valor global da solução. Na Figura 15 (b), é apresentado o processo de validação para o horizonte de planeamento de 4 semanas.

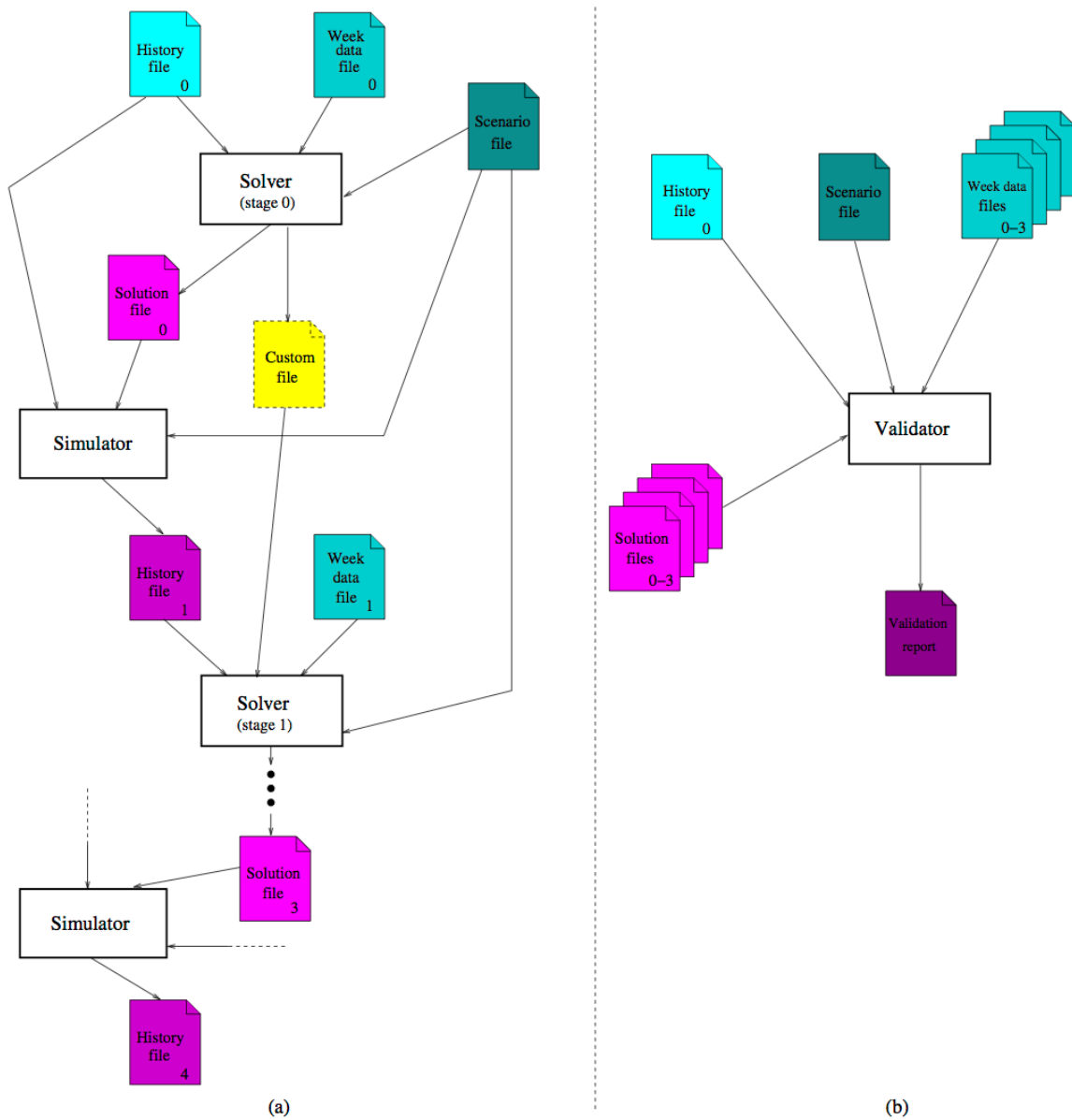


Figura 15 - Processo de geração completo (solução/simulação/validação) ⁵

Um exemplo de escala de 4 semanas gerado pelo simulador pode ser visualizado na Figura 16. Esta escala é constituída por um cabeçalho com os dias da semana e uma primeira coluna com o nome dos recursos. Para cada recurso é apresentado o turno que vai realizar.

⁵ Imagem retirada do artigo de apresentação do concurso INRCII [57]

	M T W T F S S	M T W T F S S	M T W T F S S	M T W T F S S
HN_0	- - L L L L L	- - N N N N N	- - D D D L L	- - N N N N N
HN_1	E E E E - -	- D D D - -	D D D - -	- D D - - N N
HN_2	N N N N - -	L L - - L L	- - N N N - -	L L - - D D
HN_3	D D - - D D D	- - E E E E E	- - L L L - -	E E E - -
NU_4	- - E E E - -	E E - - - -	L L - - E E E	E - - D D - -
NU_5	D D - - - -	E E D D - -	L L L - -	- E E - -
NU_6	- - L L N N N	N - - D D D D	- - D D D D D	- - D D D D L
NU_7	L L - - E E E	E - - L L L L	- - - - N N N	N N - - E E E
NU_8	D D D D D - -	N N N - - N N	N N N - D D D	D - - E E L L
NU_9	- - - D D - -	D D D D - -	D D D - -	- D D - -
NU_10	E E L L - -	L L L - -	D D D L L - -	- E E E E - -
NU_11	- - E E E E E	- - - D D D N	N - - E E E E	E - - E E E E
NU_12	L L L - - N N	N N N - -	L L L - -	N N N N - -
NU_13	L - - - D D L	L - - L L - -	E E E - -	L L - -
NU_14	N N - - L L N	N N - E E E E	- - E E E D D	- - L L L L L
NU_15	- - L L L L L	- - E E L L L	- - L L L - -	D D L L L - -
NU_16	L L L L - -	D D - - - -	D D - - - -	E E E - -
CT_17	- D D L L - -	D D D D D - -	N N N - - L L	N N N - - E E
CT_18	E E - - - -	E E N N - -	E E - - N N N	- - E E E - -
CT_19	E E E D D - -	N N N N N - -	D D L L L - -	D D D L L - -
CT_20	- D D D D - -	L L L L - -	N N N - -	D D L L - -
CT_21	E E E E - -	L L L - -	E E E L L - -	N N N N - -
CT_22	L L N N N - -	D D L L L - -	- D D D D D N	N - - N N N N
CT_23	- - E E E D D	- - E E E - -	E E E E E - -	L L L L L - -
CT_24	N N N N N - -	L L N N N - -	D N N N N - -	E E D D D - -
TR_25	N N N N - -	N N N - - L L	L N N - -	D N N - - E E
TR_26	L L L - - D D	D - - - E E E	E - L N N - -	E E E - -
TR_27	- - - - E E	- - - N N N N	- - - E E E D	- - - E E - -
TR_28	- E E D D - -	E E D D D D D	- - D D L L L	- - D D D D N
TR_29	- - - L L L L	- - - L L - -	D D - - - -	L L - - L L L

Figura 16 - Exemplo de uma escala de 4 semanas gerada pelo simulador

Os dados do simulador são apresentados no idioma inglês, visto ser uma competição internacional.

No cabeçalho, os dias da semana repetem-se para cada uma das semanas do horizonte de planeamento. A correspondência entre as letras e os dias da semana é a seguinte:

- M – segunda-feira (Monday)
- T – terça-feira (Tuesday)
- W – quarta-feira (Wednesday)
- T – quinta-feira (Thursday)
- F – sexta-feira (Friday)
- S – sábado (Saturday)
- S – domingo (Sunday)

Na primeira coluna é apresentado o código de identificação do recurso. O cruzamento das linhas com as colunas apresenta o respetivo turno que o colaborador vai realizar num determinado dia do horizonte de planeamento.

Na apresentação dos turnos a correspondência das letras é a seguinte:

- E – manhã (Early)
- L – tarde (Late)
- D – dia (Day)
- N – noite (Nigth)

Em resumo, o processo completo de gerar a solução requer como entrada um cenário, um histórico inicial e 4 ou 8 ficheiros de dados semanais. Este processo produz ficheiros de histórico entre as semanas, 4 ou 8 soluções de semanas e um ficheiro final com os resultados.

5.2. Teste e validação da solução

Os dados necessários para a validação da solução desenvolvida foram todos disponibilizados pela organização do concurso, tanto os dados para a fase de desenvolvimento, como os dados para a competição.

A organização disponibilizou no *site* do concurso alguns conjuntos de dados para teste juntamente com possíveis soluções, para servirem de métrica dos algoritmos no processo de desenvolvimento. Foram disponibilizados ainda 14 conjuntos de dados para competição. Cada conjunto é composto por 1 ficheiro de cenário, 3 históricos iniciais e 10 ficheiros de dados de semana. A Figura 17 mostra o conteúdo da pasta de um conjunto de dados.

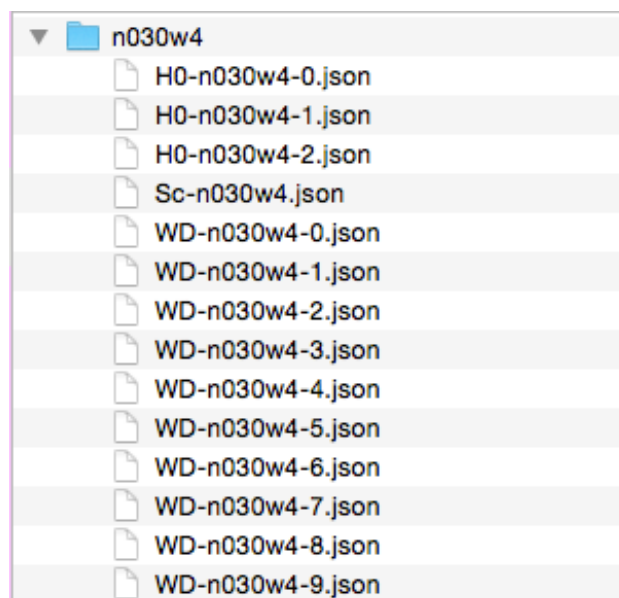


Figura 17 - Exemplo da pasta do conjunto de dados de 30 recursos para 4 semanas

A combinação dos vários ficheiros do conjunto de dados dá origem às instâncias de problemas a resolver. Uma instância é composta por um histórico inicial, um cenário e 4 ou 8 semanas, sendo que as semanas podem ser repetidas.

Os conjuntos de dados fornecidos para competir estão listados na Tabela 2. Esta tabela mostra o número de recursos e semanas utilizados em cada conjunto. Cada instância tem um tempo de resolução associado. Segundo a documentação do concurso, para um computador relativamente moderno, o

tempo de execução (em segundos) pode ser encontrado através da fórmula $10+30*(N-20)$, sendo N o número de recursos. Contudo, devido à variedade de hardware e à dificuldade em controlar a capacidade de processamento das máquinas, a organização disponibilizou uma ferramenta de *benchmark*. A ferramenta de *benchmark* calcula o tempo de execução dos problemas de acordo com as capacidades das máquinas, para garantir a igualdade entre todos os participantes.

Na Tabela 2, para cada conjunto, é apresentado o tempo de execução por semana (coluna “Tempo”) atribuído a uma das máquinas utilizadas (as restantes máquinas tiveram tempos semelhantes). A Tabela 2 apresenta também os tempos calculados pela fórmula (coluna “Tempo fórmula”) fornecida. Comparando os tempos conclui-se que as máquinas utilizadas para correr os algoritmos têm uma capacidade superior aos resultados da fórmula, pelo que é possível executar o mesmo processamento em menos tempo.

Conjunto	Número de recursos	Número de Semanas	Tempo * (seg.)	Total tempo (seg.)	Tempo fórmula ** (seg.)	Total tempo fórmula (seg.)
#1	30	4	45,34	181,36	310	1240
#2	30	8	45,34	362,72	310	2480
#3	40	4	79,35	317,4	610	2440
#4	40	8	79,35	634,8	610	4880
#5	50	4	113,36	453,44	910	3640
#6	50	8	113,36	906,88	910	7280
#7	60	4	147,37	589,48	1210	4840
#8	60	8	147,37	1178,96	1210	9680
#9	80	4	215,39	861,56	1810	7240
#10	80	8	215,39	1723,12	1810	14480
#11	100	4	283,41	1133,64	2410	9640
#12	100	8	283,41	2267,28	2410	19280
#13	120	4	351,42	1405,68	3010	12040
#14	120	8	351,42	2811,36	3010	24080

Tabela 2 - Conjuntos de dados para competição

* - tempo relativo à execução de uma semana, calculado pela ferramenta de *benchmark*.

** - tempo relativo à execução de uma semana, calculado pela fórmula fornecida na documentação do concurso.

5.2.1. Soluções submetidas a concurso

A Tabela 3 apresenta as soluções submetidas ao concurso INRC.

Instância	Restrições					Objetivos						
	Cobertura mínima	Competências necessárias	Sequências de turnos proibidas	Atribuição única por dia	Total alocações	Consecutivos	Dias de folga	Preferências	Max. fins semana de trabalho	Fins de semana completos	Cobertura ótima	Total custos
n030w4_1_6-2-9-1	0	0	0	0	520	360	90	140	480	0	900	2490
n030w4_1_6-7-5-3	0	0	0	0	700	390	120	210	420	90	750	2680
n030w8_1_2-7-0-9-3-6-0-6	0	0	0	0	1540	825	210	350	540	60	510	4035
n030w8_1_6-7-5-3-5-6-2-9	0	0	0	0	1500	660	150	170	450	90	450	3470
n040w4_0_2-0-6-1	0	0	0	0	780	570	210	320	570	0	300	2750
n040w4_2_6-1-0-6	0	0	0	0	920	720	150	480	540	0	270	3080
n040w8_0_0-6-8-9-2-6-6-4	0	0	0	0	2300	1335	150	530	960	120	570	5965
n040w8_2_5-0-4-8-7-1-7-2	0	0	0	0	2400	840	240	460	930	30	480	5380
n050w4_0_0-4-8-7	0	0	0	0	680	660	150	280	600	0	600	2970
n050w4_0_7-2-7-2	0	0	0	0	740	690	150	330	750	30	360	3050
n050w8_1_1-7-8-5-7-4-1-8	0	0	0	0	2700	1710	510	750	720	30	2070	8490
n050w8_1_9-7-5-3-8-8-3-1	0	0	0	0	2880	1575	240	790	1020	30	2070	8605
n060w4_1_6-1-1-5	0	0	0	0	1700	960	60	300	390	0	750	4160
n060w4_1_9-6-3-8	0	0	0	0	1820	930	330	350	240	60	1140	4870
n060w8_0_6-2-9-9-0-8-1-3	0	0	0	0	2100	1725	480	840	720	90	780	6735
n060w8_2_1-0-3-4-0-3-9-1	0	0	0	0	2200	1695	450	830	720	60	1080	7035
n080w4_2_4-3-3-3	0	0	0	0	1840	1110	180	240	810	0	1410	5590
n080w4_2_6-0-4-8	0	0	0	0	2020	750	480	260	750	60	1050	5370
n080w8_1_4-4-9-9-3-6-0-5	0	0	0	0	5240	1455	300	560	1170	90	720	9535
n080w8_2_0-4-0-9-1-9-6-2	0	0	0	0	4800	1785	300	770	1290	30	1350	10325
n100w4_0_1-1-0-8	0	0	0	0	2260	660	210	230	390	0	90	3840
n100w4_2_0-6-4-6	0	0	0	0	2520	615	180	280	510	60	240	4405
n100w8_0_0-1-7-8-9-1-5-4	0	0	0	0	4760	1365	510	840	660	120	210	8465
n100w8_1_2-4-7-9-3-9-2-8	0	0	0	0	4720	1620	420	920	390	30	450	8550
n120w4_1_4-6-2-6	0	0	0	0	1580	810	540	900	870	210	570	5480
n120w4_1_5-6-9-8	0	0	0	0	1440	1200	450	730	780	90	720	5410
n120w8_0_0-9-9-4-5-1-0-3	0	0	0	0	2340	1965	720	1690	1020	150	1350	9235
n120w8_1_7-2-6-4-5-2-0-2	0	0	0	0	2420	2100	780	730	1230	120	1740	9120
Total de custos por objetivo					61420	31080	8760	15280	19920	1650	22980	

Tabela 3 - Soluções das instâncias submetidas a concurso

Foram propostas 28 instâncias para submeter ao concurso. Para gerar as soluções para todas as instâncias, criou-se uma pequena ferramenta que permite gerar soluções sem necessidade de intervenção humana. Esta ferramenta começa por escolher uma das instâncias e construir as instruções necessárias para a lançar no simulador. Só é gerada a solução para uma instância de cada vez. Quando é encontrada a solução, a ferramenta repete o processo para outra instância e assim sucessivamente. A ferramenta fica a gerar soluções até que seja fechada.

A Tabela 3 apresenta as melhores soluções encontradas. Para cada instância é apresentado o resumo da violação de restrições e os custos relacionados com a quebra de objetivos.

Analisando as restrições, pode-se concluir que a estrutura de dados desenvolvida para o controlo de restrições é adequada, uma vez que não foi violada qualquer restrição nas soluções geradas pelos algoritmos implementados. Como podemos ver no grupo de colunas das restrições da tabela, todas as soluções tiveram 0 violações para cada uma das restrições avaliadas.

Olhando para os objetivos, percebe-se que podem ser agrupados em 3 grupos. Um grupo de dificuldade elevada que inclui o total de alocações e os consecutivos. Um grupo de dificuldade média que inclui a cobertura do ótimo das necessidades, o máximo de fins-de-semana de trabalho e as preferências dos recursos. Por fim, um grupo de menor dificuldade que inclui os fins-de-semana completos com descanso ou trabalho e os dias de folga.

O objetivo que os algoritmos têm mais dificuldade em controlar é o total de alocações. No âmbito do concurso não é possível alterar semanas anteriores, o que torna difícil ajustar este objetivo no decorrer da geração.

5.2.2. Resultados submetidos pelos participantes

A Tabela 4 e a Tabela 5, apresentam as soluções submetidas ao concurso pelas várias equipas. No total foram 15 equipas que submeteram soluções. Segundo a organização do concurso foram submetidas soluções geradas com métodos heurísticos e métodos exatos.

Instâncias	INF_UFRGS	Nurse_Juglers	Hust_Smart	ORTEC	eatts_inrc2-2	GOAL	SSH	LabGOL	in-optima-inc-ii	ThreeJohns	Team Puma	Bullet Solutions	NurseOptmizers	Polytechnique Montreal	ScheduleNurse	Valor médio
n030w4_1_6-2-9-1	2355	9850	2010	2000	3280	2690	1930	2065	5680	2070	3300	2490	1745	1780	2385	3042
n030w4_1_6-7-5-3	2640	10605	2135	2100	3275	2965	2130	2230	6370	2160	3330	2680	1935	1950	2415	3261
n040w4_0_2-0-6-1	2555	14680	1870	2075	3230	2965	1895	2275	8470	2120	4065	2750	1725	1765	2160	3640
n040w4_2_6-1-0-6	2710	14460	2080	2235	3385	3295	2165	2365	9065	2340	3860	3080	1910	2010	2580	3836
n050w4_0_0-4-8-7	2650	17745	1730	1890	3360	3225	1845	2135	9720	2040	3975	2970	1525	1630	2455	3926
n050w4_0_7-2-7-2	2735	15830	1785	1955	2975	3160	1875	2185	9510	2140	3965	3050	1480	1645	2435	3782
n060w4_1_6-1-1-5	4045	19230	3010	3455	4710	4575	3005	3440	12310	3330	4950	4160	2830	2830	3320	5280
n060w4_1_9-6-3-8	4120	20400	3240	3540	5195	4755	3160	3715	11560	3670	6070	4870	3015	2950	3775	5602
n080w4_2_4-3-3-3	5350	26935	4055	4130	5630	5475	3820	4485	21295	4495	6565	5590	3475	3615	4985	7327
n080w4_2_6-0-4-8	5290	27210	3995	4130	5875	5880	3955	4530	19865	4440	6060	5370	3535	3650	5320	7274
n100w4_0_1-1-0-8	3390	33740	1970	2350	3605	4195	1690	2550	15140	2400	5445	3840	1535	1445	2800	5740
n100w4_2_0-6-4-6	4085	33465	2440	2915	4050	5030	2305	3090	15685	2955	5875	4405	2070	2100	3265	6249
n120w4_1_4-6-2-6	4940	36235	3095	3395	5295	5255	2555	3650	20400	3320	5945	5480	2470	2515	4700	7283
n120w4_1_5-6-9-8	5110	36320	3090	3435	5535	5360	2865	3700	22120	3510	6490	5410	2530	2535	5200	7547

Tabela 4 - Resultados submetidos pelos participantes: instâncias de 4 semanas

Instâncias	INF_UFRGS	Nurse_Jugglers	Hust.Smart	ORTEC	eatts_inrc2-2	GOAL	SSH	LabGOL	in-optima-inrc-ii	ThreeJohns	Team Puma	Bullet Solutions	NurseOptimizers	Polytechnique Montreal	ScheduleNurse	Valor médio
n030w8_1_2-7-0-9-3-6-0-6	3965	21185	2945	2610	4475	4125	2940	3125	9355	3080	4945	4035	2295	2325	3585	4999
n030w8_1_6-7-5-3-5-6-2-9	3440	21145	2585	2370	3615	4045	2380	2630	8725	2640	5120	3470	1900	1990	2840	4593
n040w8_0_0-6-8-9-2-6-6-4	4940	35010	3765	4050	6030	6790	3755	4505	16185	4650	7525	5965	3105	3315	4535	7608
n040w8_2_5-0-4-8-7-1-7-2	4755	33000	3470	3735	5675	6060	3630	4140	15010	4295	7435	5380	3010	2975	3820	7093
n050w8_1_1-7-8-5-7-4-1-8	7985	43040	6790	6630	10015	9180	6105	7225	25690	7000	9855	8490	5560	5730	8035	11155
n050w8_1_9-7-5-3-8-8-3-1	7930	42765	6640	6630	9790	9120	5805	7105	24080	7140	9865	8605	5475	5600	8030	10972
n060w8_0_6-2-9-9-0-8-1-3	5890	44130	3715	4010	6015	7245	3770	4725	24245	4285	8360	6735	2840	2930	4895	8919
n060w8_2_1-0-3-4-0-3-9-1	6420	44430	4185	4505	6565	7230	3885	5230	24470	4925	8520	7035	3200	3280	5615	9300
n080w8_1_4-4-9-9-3-6-0-5	8375	64915	5375	6735	8400	9515	5860	6840	36000	6845	11885	9535	4845	5040	7410	13172
n080w8_2_0-4-0-9-1-9-6-2	8890	66515	6005	6765	9320	10265	6325	7480	36850	7430	12055	10325	5105	5415	7260	13734
n100w8_0_0-1-7-8-9-1-5-4	6810	85260	4140	5115	6545	9140	3550	5290	31875	4915	11505	8465	3195	3095	4885	12919
n100w8_1_2-4-7-9-3-9-2-8	7365	87445	4310	5505	6870	9640	4165	5595	31715	5590	12410	8550	3135	3355	5370	13401
n120w8_0_0-9-9-4-5-1-0-3	8595	83590	4885	6145	8185	9785	4055	6315	46870	5700	12115	9235	3555	4030	7835	14726
n120w8_1_7-2-6-4-5-2-0-2	8080	82145	5310	6315	9590	9770	4230	6040	44565	6320	14020	9120	3435	3975	8750	14778

Tabela 5 - Resultados submetidos pelos participantes: instâncias de 8 semanas

Olhando para o valor médio do custo das soluções, a equipa conseguiu ficar abaixo do custo médio em todas as instâncias (Tabela 4 e Tabela 5). É um resultado importante que mostra que o algoritmo está no bom caminho, mas ainda com espaço para melhorar.

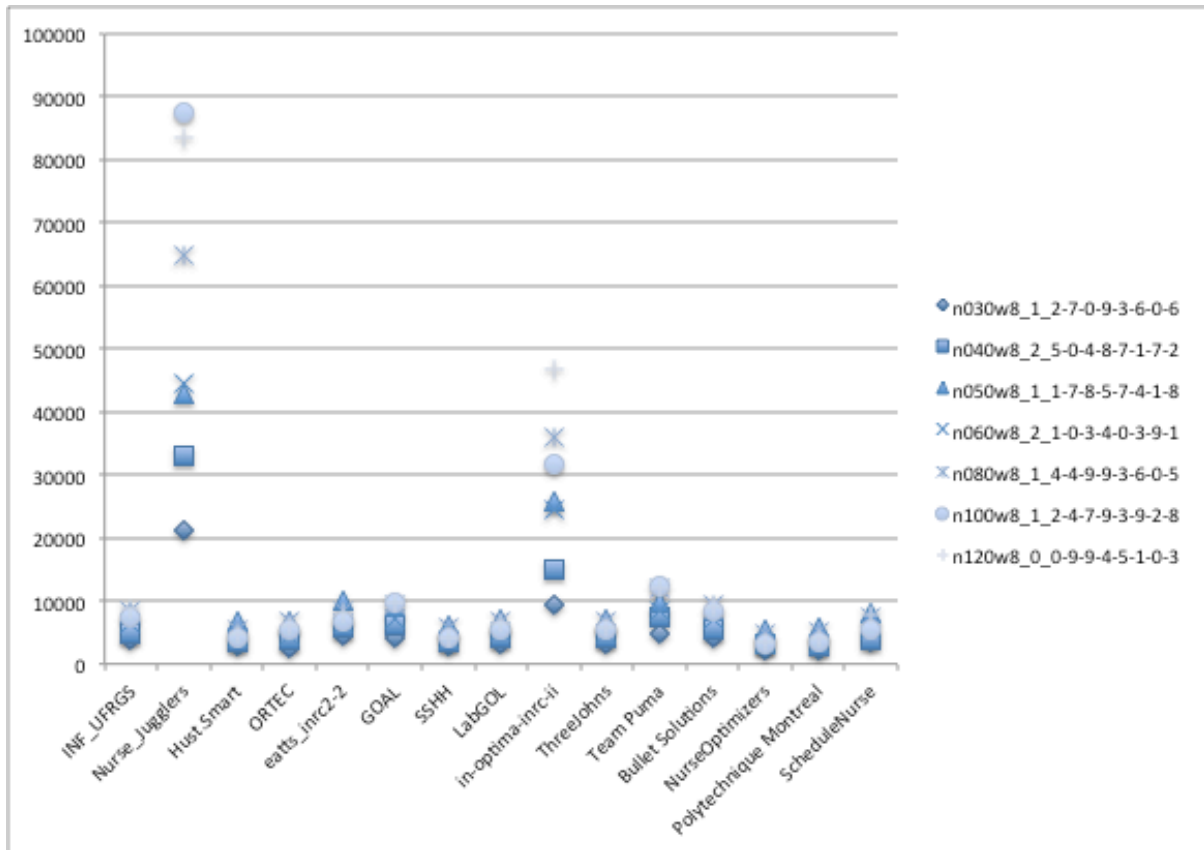


Figura 18 - Comparação de resultados por equipa

O gráfico da Figura 18 agrupa os resultados de algumas instâncias por equipa. Este gráfico ajuda a perceber o panorama geral das soluções submetidas. De notar que existe uma concentração dos resultados abaixo da linha dos 10000 pontos.

Instâncias	Pior resultado	Bullet Solutions	Melhor resultado	Diferença do pior para o melhor	Diferença Bullet Solutions para o melhor
n030w4_1_6-2-9-1	9850	2490	1745	8105	745
n030w4_1_6-7-5-3	10605	2680	1935	8670	745
n030w8_1_2-7-0-9-3-6-0-6	21185	4035	2295	18890	1740
n030w8_1_6-7-5-3-5-6-2-9	21145	3470	1900	19245	1570
n040w4_0_2-0-6-1	14680	2750	1725	12955	1025
n040w4_2_6-1-0-6	14460	3080	1910	12550	1170
n040w8_0_0-6-8-9-2-6-6-4	35010	5965	3105	31905	2860
n040w8_2_5-0-4-8-7-1-7-2	33000	5380	3010	29990	2370
n050w4_0_0-4-8-7	17745	2970	1525	16220	1445
n050w4_0_7-2-7-2	15830	3050	1480	14350	1570
n050w8_1_1-7-8-5-7-4-1-8	43040	8490	5560	37480	2930
n050w8_1_9-7-5-3-8-8-3-1	42765	8605	5475	37290	3130
n060w4_1_6-1-1-5	19230	4160	2830	16400	1330
n060w4_1_9-6-3-8	20400	4870	3015	17385	1855
n060w8_0_6-2-9-9-0-8-1-3	44130	6735	2840	41290	3895
n060w8_2_1-0-3-4-0-3-9-1	44430	7035	3200	41230	3835
n080w4_2_4-3-3-3	26935	5590	3475	23460	2115
n080w4_2_6-0-4-8	27210	5370	3535	23675	1835
n080w8_1_4-4-9-9-3-6-0-5	64915	9535	4845	60070	4690
n080w8_2_0-4-0-9-1-9-6-2	66515	10325	5105	61410	5220
n100w4_0_1-1-0-8	33740	3840	1535	32205	2305
n100w4_2_0-6-4-6	33465	4405	2070	31395	2335
n100w8_0_0-1-7-8-9-1-5-4	85260	8465	3195	82065	5270
n100w8_1_2-4-7-9-3-9-2-8	87445	8550	3135	84310	5415
n120w4_1_4-6-2-6	36235	5480	2470	33765	3010
n120w4_1_5-6-9-8	36320	5410	2530	33790	2880
n120w8_0_0-9-9-4-5-1-0-3	83590	9235	3555	80035	5680
n120w8_1_7-2-6-4-5-2-0-2	82145	9120	3435	78710	5685

Tabela 6 - Comparação de resultados

Olhando para o pior e melhor resultado, os resultados da equipa Bullet Solutions tendem a aproximar-se dos da equipa que ficou em primeiro lugar. As instancias de 80, 100 e 120 recursos para 8 semanas tiveram uma maior penalização. O algoritmo proposto mostra ser mais eficiente na resolução de problemas de 4 semanas.

5.2.3. Resultados das soluções geradas pela organização

A Tabela 7 apresenta a grelha de resultados da competição apresentados no website⁶ da mesma.

Instâncias	INF_UFRGS	Nurse_Jugglers	Hust.Smart	ORTEC	eatts_inrc2-2	GOAL	SSHH	LabGOL	in-optima-inrc-ii	Three.Johns	Team.Puma	Bullet.Solutions	Nurse.Optimizers	Polytechnique.Montreal	ScheduleNurse
n030w4_1_6-2-9-1	8	15	5	4	12	11	3	6	14	7	13	10	1	2	9
n030w4_1_6-7-5-3	9	15	5	3	12	11	4	7	14	6	13	10	1	2	8
n030w8_1_2-7-0-9-3-6-0-6	9	15	5	3	12	11	4	7	14	6	13	10	1	2	8
n030w8_1_6-7-5-3-5-6-2-9	9	15	5	3	11	12	4	6	14	7	13	10	1	2	8
n040w4_0_2-0-6-1	9	15	3	5	12	11	4	8	14	6	13	10	1	2	7
n040w4_2_6-1-0-6	9	15	3	5	12	11	4	7	14	6	13	10	1	2	8
n040w8_0_0-6-8-9-2-6-6-4	9	15	4	5	11	12	3	6	14	8	13	10	1	2	7
n040w8_2_5-0-4-8-7-1-7-2	9	15	3	5	11	12	4	7	14	8	13	10	2	1	6
n050w4_0_0-4-8-7	9	15	3	5	12	11	4	7	14	6	13	10	1	2	8
n050w4_0_7-2-7-2	9	15	3	5	10	12	4	7	14	6	13	11	1	2	8
n050w8_1_1-7-8-5-7-4-1-8	8	15	5	4	13	11	3	7	14	6	12	10	1	2	9
n050w8_1_9-7-5-3-8-8-3-1	8	15	5	4	12	11	3	6	14	7	13	10	1	2	9
n060w4_1_6-1-1-5	9	15	4	8	12	11	3	7	14	6	13	10	1,5	1,5	5
n060w4_1_9-6-3-8	9	15	4	5	12	10	3	7	14	6	13	11	2	1	8
n060w8_0_6-2-9-9-0-8-1-3	9	15	3	5	10	12	4	7	14	6	13	11	1	2	8
n060w8_2_1-0-3-4-0-3-9-1	9	15	4	5	10	12	3	7	14	6	13	11	1	2	8
n080w4_2_4-3-3-3	9	15	4	5	12	10	3	6	14	7	13	11	1	2	8
n080w4_2_6-0-4-8	8	15	4	5	11	12	3	7	14	6	13	10	1	2	9
n080w8_1_4-4-9-9-3-6-0-5	9	15	3	5	10	11	4	6	14	7	13	12	1	2	8
n080w8_2_0-4-0-9-1-9-6-2	9	15	3	5	10	11	4	8	14	7	13	12	1	2	6
n100w4_0_1-1-0-8	9	15	4	5	10	12	3	7	14	6	13	11	2	1	8
n100w4_2_0-6-4-6	10	15	4	5	9	12	3	7	14	6	13	11	1	2	8
n100w8_0_0-1-7-8-9-1-5-4	10	15	4	7	9	12	3	8	14	6	13	11	2	1	5
n100w8_1_2-4-7-9-3-9-2-8	10	15	4	6	9	12	3	8	14	7	13	11	1	2	5
n120w4_1_4-6-2-6	9	15	4	6	11	10	3	7	14	5	13	12	1	2	8
n120w4_1_5-6-9-8	8	15	4	5	12	10	3	7	14	6	13	11	1	2	9
n120w8_0_0-9-9-4-5-1-0-3	10	15	4	6	9	12	3	7	14	5	13	11	1	2	8
n120w8_1_7-2-6-4-5-2-0-2	8	15	4	6	11	12	3	5	14	7	13	10	1	2	9
Average	8,93	15	3,93	5	10,96	11,32	3,39	6,86	14	6,36	12,96	10,61	1,16	1,84	7,68
Rank	9	15	4	5	11	12	3	7	14	6	13	10	1	2	8

Tabela 7 - Classificação das equipas

A equipa Bullet Solutions ficou classificada em 10º lugar. Apesar de ser um lugar distante do primeiro, olhando para os resultados das instâncias observa-se que a equipa não se distanciou demasiado das restantes equipas, o que para a equipa é um resultado importante, visto que foi a primeira vez que se submeteu a um concurso. Os resultados permitem perceber que a abordagem tem qualidade, apesar de ainda haver espaço para melhorar. Os próximos passos no algoritmo podem incluir uma melhoria no controlo do total de alocações e dos consecutivos. Estes dois objetivos são de dificuldade acrescida devido ao problema seguir uma abordagem de múltiplas etapas.

⁶ <http://mobiz.vives.be/inrc2>

6. Conclusões e Trabalho Futuro

6.1. Conclusões

O presente documento apresenta uma revisão dos principais conceitos associados ao estudo do problema de *nurse rostering*. O documento descreve as condições, os objetivos e restrições segundo as quais a solução foi desenvolvida. Este projeto teve por base o concurso *Second International Nurse Rostering Competition*. Apesar do problema proposto pelo concurso ter um reduzido número de objetivos e restrições, foi proposta uma solução que se adaptasse o melhor possível a problemas do mundo real, ou seja, problemas com muitos objetivos e restrições que tornam a sua resolução muito complexa.

A abordagem proposta neste documento permitiu uma forma mais ágil e modular para dar resposta a problemas de escalonamento de pessoal reais. A proposta foi submetida a concurso e os resultados são aqui apresentados. Os resultados permitiram perceber que a abordagem tem qualidade, apesar de ainda haver espaço para melhorar. Enquanto empresa, a Bullet Solutions, com este projeto, tinha como objetivo criar as bases necessárias para o futuro desenvolvimento de um produto genérico de escalonamento automático e otimizado de pessoal. Esta abordagem foi de encontro aos objetivos da empresa e será utilizada como base nos algoritmos de um novo produto.

6.2. Trabalho Futuro

O projeto teve desde o início o foco em criar uma abordagem para aplicar em problemas reais. O próximo desafio é a implementação da mesma num produto empresarial de escalonamento de pessoal. Pretende-se utilizar os algoritmos para suportar a componente de geração de escalas automatizadas. Este produto visa disponibilizar aos consumidores uma interface que permite a configuração dos algoritmos e a geração de escalas de trabalho. O produto entrará em produção antes do final do ano de 2015.

7. Glossário

Debug – Em desenvolvimento de software *debug* é o termo utilizado para identificar o processo de encontrar defeitos num software, por forma a resolver e reduzir os mesmos.

JavaScript Object Notation – É uma notação para transferências de dados leve. É de fácil leitura e escrita para os humanos e é fácil de interpretar e gerar para as máquinas. É baseada em JavaScript, contudo é uma linguagem completamente independente que usa convenções familiares a programadores de várias linguagens como C, C#, Java, Javascript e muitas outras [56].

.NET framework – É um ambiente de tempo de execução, que proporciona uma variedade de serviços para as diversas aplicações em execução. É uma camada de abstração entre as aplicações e o sistema operativo. O *.NET framework* é desenvolvido pela Microsoft.

Framework – No contexto do desenvolvimento de software, uma *framework* é um conjunto de classes previamente desenvolvidas com o intuito de auxiliar o desenvolvimento do software.

Mapa Conceptual – Formalismo de representação de conhecimento, confere um grande poder de expressividade e apoio à representação de conhecimento partilhado.

Programação Funcional – É um paradigma de programação que trata a lógica da programação como a avaliação de funções matemáticas [54].

8. Referências bibliográficas

- [1] R. Groningen, "Quality in Fives: On the Analysis, Operationalization and Application of Nursing Schedule Quality," *University of Groningen*, 1996.
- [2] D. M. Warner, "Scheduling nursing personnel according to nursing preference: a mathematical programming approach," *Operations Research*, vol. 24, no. 5, pp. 842–856, 1976.
- [3] E. K. Burke, P. De Causmaecker, G. Vanden Berghe, and H. Van Landeghem, "The State of the Art of Nurse Rostering," *Journal of Scheduling*, vol. 7, no. 6, pp. 441–499, Nov. 2004.
- [4] J. J. Gray, D. McIntire, and H. J. Doller, "Preferences for specific work schedules: Foundation for an expert-system scheduling program," *Computers in Nursing*, vol. 11, no. 3, pp. 115–121, 1993.
- [5] D. A. Gillies, *Nursing Management - A Systems Approach*. Philadelphia, US America: W.B. Saunders, 1989.
- [6] J. P. Howell, "Cyclical scheduling of nursing personnel," *Hospitals*, vol. 40, pp. 77–85, 1966.
- [7] R. Silvestro and C. Silvestro, "An evaluation of nurse rostering practices in the National Health Service," *Journal of Advanced Nursing*, vol. 32, no. 3, pp. 525–535, 2000.
- [8] I. Blöchliger, "Modeling staff scheduling problems. A tutorial," *European Journal of Operational Research*, vol. 158, pp. 533–542, 2004.
- [9] B. Cheang, H. Li, A. Lim, and B. Rodrigues, "Nurse rostering problems - a bibliographic survey," *European Journal of Operational Research*, vol. 151, no. 3, pp. 447–460, Dec. 2003.
- [10] K. Nonobe and T. Ibaraki, "A tabu search approach to the constraint satisfaction problem as a general problem solver," *European Journal of Operational Research*, vol. 106, pp. 599–623, 1998.

- [11] H. E. Miller, W. P. Pierskalla, and G. J. Rath, "Nurse Scheduling Using Mathematical Programming," *Operations Research*, vol. 24, no. 5, pp. 857–870, 1976.
- [12] W. J. Abernathy, N. Baloff, J. C. Hershey, and S. Wandel, "A three-stage manpower planning and scheduling model - A service-sector example," *Operations Research*, vol. 22, pp. 693–711, 1973.
- [13] M. Warner, B. Keller, and S. Martel, "Automated Nurse scheduling," *Journal of the Society for Health Systems*, vol. 2, no. 2, pp. 66–80, 1990.
- [14] J. Bailey and J. Field, "Personnel scheduling with flexshift models," *Journal of Operations Management*, vol. 5, pp. 327–338, 1985.
- [15] M. Moz and M. Pato, "Solving the problem of rostering Nurse schedules with hard constraints: New multicommodity flow models," *Annals of Operations Research*, vol. 128, pp. 179–197, 2004.
- [16] M. Okada, "An approach to the generalised nurse scheduling problem— Generation of a declarative program to represent institution-specific knowledge," *Computers and Biomedical Research*, vol. 25, pp. 417–434, 1992.
- [17] H. Meyer auf'm Hofe, "Solving Rostering Tasks as Constraint Optimization," in *E. K. Burke and W. Erben (eds.), Practice and Theory of Automated Timetabling, Third International Conference, Konstanz, Springer, Lecture Notes in Computer Science*, 2001, vol. 2079, pp. 191–212.
- [18] L. D. Smith, A. Wiggins, and D. Bird, "Post implementation experience with computer-assisted nurse scheduling in a large hospital," *Information Systems and Operational Research*, vol. 17, pp. 309–321, 1979.
- [19] J. G. Chen and T. Yeung, "Hybrid expert system approach to Nurse scheduling," *Computers in Nursing*, pp. 183–192, 1993.
- [20] A. . Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier, "Staff scheduling and rostering: A review of applications, methods and models," *European Journal of Operational Research*, vol. 153, no. 1, pp. 3–27, Feb. 2004.

- [21] P. C. Nutt, "Decision-modeling methods used to design decision support systems for staffing," *Medical Care*, vol. 22, no. 11, pp. 1002–1013, 1984.
- [22] P. Chan and G. Weil, "Cyclical staff scheduling using constraint logic programming," in *E. K. Burke and W. Erben (eds.), Practice and Theory of Automated Timetabling, Third International Conference, Konstanz, Springer, Lecture Notes in Computer Science*, 2001, pp. 159–175.
- [23] L. D. Smith, "The application of an interactive algorithm to develop cyclical rotational schedules for Nursing personnel," *INFOR*, vol. 14, pp. 53–70, 1976.
- [24] L. D. Smith and A. Wiggins, "A computer-based Nurse scheduling system," *Computers and Operations Research*, vol. 4, no. 3, pp. 195–212, 1977.
- [25] A. Schaerf and A. Meisels, "Solving employee timetabling problems by generalised local search," in *Proceedings Italian AI*, 1999, pp. 493–502.
- [26] M. Anzai and Y. Miura, "Computer program for quick work scheduling of nursing staff," *Medical Informatics*, vol. 12, pp. 43–52, 1987.
- [27] R. Blau, "Multishift personnel scheduling with a microcomputer," *Personnel Administrator*, vol. 20, no. 7, pp. 43–58, 1985.
- [28] C. Blum and A. Roli, "Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison," *ACM Computing Surveys*, vol. 35, no. 3, pp. 268–308, 2003.
- [29] S. U. Randhawa and D. Sitompul, "A heuristic-based computerized Nurse scheduling system," *Computers and Operations Research*, vol. 20, no. 8, pp. 837–844, 1993.
- [30] B. L. Golden and W. R. Stewart, "Empirical Analysis of Heuristics," in *The Traveling Salesman Problem, E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys, eds., John Wiley*, 1985, pp. 207–250.
- [31] F. Bellanti, G. Carello, F. Della Croce, and R. Tadei, "A greedy-based neighborhood search approach to a nurse rostering problem," *European Journal of Operational Research*, vol. 153, no. 1, pp. 28–40, 2004.

- [32] E. K. Burke, T. Curtois, R. Qu, and G. Vanden Berghe, "A scatter search methodology for the nurse rostering problem," *Journal of the Operational Research Society*, vol. 61, pp. 1667–1679, 2010.
- [33] M. Chiarandini, A. Schaerf, and F. Tiozzo, "Solving employee timetabling problems with flexible workload using tabu search," in *E. K. Burke and W. Erben (eds.), Proceedings of the 3rd International Conference on the Practice and Theory of Automated Timetabling*, 2000, pp. 298–302.
- [34] Z. Lü and J.-K. Hao, "Adaptive neighborhood search for nurse rostering," *European Journal of Operational Research*, vol. 218, no. 3, pp. 865–876, 2012.
- [35] B. Maenhout and M. Vanhoucke, "An electromagnetic meta-heuristic for the nurse scheduling problem," *Journal of Heuristics*, vol. 13, pp. 359–385, 2007.
- [36] "Metaheuristics Network," 2000. [Online]. Available: <http://www.metaheuristics.net/>. [Accessed: 01-Aug-2015].
- [37] I. H. Osman and G. Laporte, "Metaheuristics: A bibliography. *Annals of Operations Research*," vol. 63, pp. 513–623, 1996.
- [38] T. Stutzle, "Local Search Algorithms for Combinatorial Problems - Analysis, Algorithms and New Applications. DISKI - Dissertationen zur Kunstlik," en Intelligenz. infix, Sankt Augustin, Germany, 1999.
- [39] S. Voß, S. Martello, I. H. Osman, and C. Roucairol, *Meta-Heuristics - Advances and Trends. Local Search Paradigms for Optimization*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999.
- [40] M. D. Goodman, K. A. Dowsland, and J. M. Thompson, "A GRASP-KNAPSACK HYBRID FOR A NURSE-SCHEDULING PROBLEM," *Cardiff: Cardiff University School of Mathematics*, vol. 44, no. 0, pp. 1–31, 2009.
- [41] U. Aickelin and K. Dowsland, "Exploiting problem structure in a genetic algorithm approach to a nurse rostering problem," *Journal of Scheduling*, vol. 3, pp. 139–153, 2000.

- [42] U. Aickelin and K. Dowsland, "An indirect genetic algorithm for a nurse scheduling problem," *Computers & Operations Research*, vol. 3, pp. 761–778, 2004.
- [43] F. F. Easton and N. Mansour, "A distributed genetic algorithm for deterministic and stochastic labor scheduling problems," *European Journal of Operational Research*, vol. 118, pp. 505–523, 1999.
- [44] E. K. Burke, P. De Causmaecker, S. Petrovic, and G. Vanden Berghe, "Variable neighborhood search for nurse rostering problems," *Metaheuristics: Computer Decision-Making*, Kluwer, 2004, pp. 153–172.
- [45] M. J. Brusco and L. W. Jacobs, "A simulated annealing approach to the cyclic staff-scheduling problem," *Naval Research Logistics*, vol. 40, pp. 69–84, 1993.
- [46] E. K. Burke, P. Cowling, P. De Causmaecker, and G. Vanden Berghe, "A memetic approach to the nurse rostering problem," *Applied Intelligence*, vol. 15, pp. 199–214, 2001.
- [47] R. Bai, E. K. Burke, G. Kendall, and B. McCollum, "An evolutionary approach to the nurse rostering problem," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 4, pp. 580–590, 2010.
- [48] F. Glover, "Tabu Search - Part I," *ORSA journal on Computing*, vol. 1, no. 3, pp. 190–206, 1989.
- [49] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [50] V. Černý, "Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm," *Journal of Optimization Theory and Applications*, vol. 45, no. 1, pp. 41–51, 1985.
- [51] D. Bertsimas and J. Tsitsiklis, "Simulated Annealing," *Statistical Science*, vol. 8, no. 1, pp. 10–15, 1993.
- [52] J. Tien and A. Kamiyama, "On manpower scheduling algorithms," *Society for Industrial and Applied Mathematics*, vol. 24, no. 3, pp. 275–287, 1982.

- [53] "INRC-II," 2014. [Online]. Available: <http://mobiz.vives.be/inrc2/>. [Accessed: 01-Aug-2015].
- [54] C. Smith, *Programming F# 3.0*, 2nd Editio. California, EUA: O'Reilly Media, Inc., 2012.
- [55] "F#," 2015. [Online]. Available: <http://fsharp.org>. [Accessed: 01-Aug-2015].
- [56] "JavaScript Object Notation," 2009. [Online]. Available: <http://www.json.org>. [Accessed: 01-Aug-2015].
- [57] S. Ceschia and N. Dang, "Second International Nurse Rostering Competition (INRC-II) 'Problem Description and Rules.'" pp. 1–18, Retrieved from <http://mobiz.vives.be/inrc2/w>, 2014.

9. Anexos

9.1. Escalas de instâncias geradas pelos algoritmos

As figuras apresentadas neste capítulo fazem parte do ficheiro gerado pelo validador do simulador, após validar os resultados gerados pelos algoritmos.

Os dados do simulador são apresentados no idioma inglês, visto ser uma competição internacional.

No cabeçalho das tabelas, os dias da semana repetem-se para cada uma das semanas do horizonte de planeamento. A correspondência entre as letras e os dias da semana é a seguinte:

- M – segunda-feira (Monday)
- T – terça-feira (Tuesday)
- W – quarta-feira (Wednesday)
- T – quinta-feira (Thursday)
- F – sexta-feira (Friday)
- S – sábado (Saturday)
- S – domingo (Sunday)

Na primeira coluna de cada tabela é apresentado o código de identificação do recurso.

O cruzamento das linhas com as colunas apresenta o respetivo turno que o colaborador vai realizar num determinado dia do horizonte de planeamento.

Na apresentação dos turnos a correspondência das letras é a seguinte:

- E – manhã (Early)
- L – tarde (Late)
- D – dia (Day)
- N – noite (Nigth)

	M T W T F S S	M T W T F S S	M T W T F S S	M T W T F S S
HN_0	- - L L L L L	- - N N N N N	- - D D D L L	- - N N N N N
HN_1	E E E E - -	- D D D - -	D D D - - -	- D D - - N N
HN_2	N N N N - -	L L - - L L	- - N N N - -	L L - - D D
HN_3	D D - - D D D	- - E E E E E	- - L L L - -	E E E - - -
NU_4	- - E E E - -	E E - - - -	L L - - E E E	E - - D D - -
NU_5	D D - - - -	E E D D - -	L L L - - -	- E E - - -
NU_6	- - L L N N N	N - - D D D D	- - D D D D D	- - D D D D L
NU_7	L L - - E E E	E - - L L L L	- - - - N N N	N N - - E E E
NU_8	D D D D D - -	N N N - - N N	N N N - D D D	D - - E E L L
NU_9	- - - D D - -	D D D D - -	D D D - - -	- D D - - -
NU_10	E E L L - -	L L L - - -	D D D L L - -	- E E E E - -
NU_11	- - E E E E E	- - - D D D N	N - - E E E E	E - - E E E E
NU_12	L L L - - N N	N N N - - -	L L L - - -	N N N N - -
NU_13	L - - - D D L	L - - L L - -	E E E - - -	L L - - - -
NU_14	N N - - L L N	N N - E E E E	- - E E E D D	- - L L L L L
NU_15	- - L L L L L	- - E E L L L	- - L L L - -	D D L L L - -
NU_16	L L L L - -	D D - - - -	D D - - - -	E E E - - -
CT_17	- D D L L - -	D D D D D - -	N N N - - L L	N N N - - E E
CT_18	E E - - - -	E E N N - -	E E - - N N N	- - E E E - -
CT_19	E E E D D - -	N N N N N - -	D D L L L - -	D D D L L - -
CT_20	- D D D D - -	L L L L - -	N N N - - -	D D L L - -
CT_21	E E E E - -	L L L - - -	E E E L L - -	N N N N N - -
CT_22	L L N N N - -	D D L L L - -	- D D D D D N	N - - N N N N
CT_23	- - E E E D D	- - E E E - -	E E E E E - -	L L L L L - -
CT_24	N N N N N - -	L L N N N - -	D N N N N - -	E E D D D - -
TR_25	N N N N - -	N N N - - L L	L N N - - -	D N N - - E E
TR_26	L L L - - D D	D - - - E E E	E - L N N - -	E E E - - -
TR_27	- - - - E E	- - - N N N N	- - - E E E D	- - - E E - -
TR_28	- E E D D - -	E E D D D D D	- - D D L L L	- - D D D D N
TR_29	- - - L L L L	- - - L L - -	D D - - - -	L L - - L L L

Figura 19 - Escala para a instância n030w4_1_6-2-9-1

	M T W T F S S	M T W T F S S	M T W T F S S	M T W T F S S
HN_0	- - L L N N N	- - E E E D D	- - N N N N N	- E E E E - -
HN_1	N N N - - -	D N N N - -	- D D - - E E	E E - - N N N
HN_2	E E E - - -	- E E - - -	- E E - - N N	- - N N N - -
HN_3	D D - - D D D	- - D D D - -	- - D D D - -	D D D - - D D
NU_4	- - N N N - -	D D - - E E E	- - - D D - -	D D D - - -
NU_5	L L - - - -	E E D D - -	D D - - - -	E E - - E E E
NU_6	- - E E E E E	- - D N N N N	- - E E L L L	- - E E E N N
NU_7	D D - - D D N	N - - - D D D	- - L L L - -	- L L L L - -
NU_8	N N N N - - E	D D L L - -	L L L L L - -	N N N N N - -
NU_9	L - - D D - -	L L - - - -	E E L L - -	- L L L - - -
NU_10	E E E E E - -	E E E E E - -	E E E - - - -	D D D - - - -
NU_11	- - L L L L L	- - L L L L L	- - E E E D D	- - D D D D D
NU_12	L L L L - - -	N N N N - - -	L L L - - L L	L - - E E L L
NU_13	L - - L L L L	- - - N N - -	N N N N - - -	N N N N - - -
NU_14	N N - - L L L	L - - E E E E	E - - E E E L	L - - L L L L
NU_15	- - D D N N N	- - L L N N N	- N N N N - -	L L L L L - -
NU_16	- D D D D - -	D L - - - -	E E - - E E N	N - - E E - -
CT_17	D D L L L - -	L L L L L - -	E E E - - E E	N N - - E E E
CT_18	E E - - - -	E E D D - - -	D D - - - - -	D D D - - E E
CT_19	E E E E E - -	D D D D D - -	- N N N N N - -	- E E E N N N
CT_20	- L L L - - -	E E E E - - -	L L L L L - -	D D D D D - -
CT_21	E E E E E - -	- - E E E E L	- - E E E - -	N N N N - - -
CT_22	- D D D D - -	N N N N - L L	N N N N - - -	E E E D D - -
CT_23	- - E E D D D	- - D D D N N	N - - D D D D	D - - D D L L
CT_24	L L N N N - -	D L L L L - -	D D D D D - -	L L L - - D D
TR_25	L L L L L - -	- E E L L L -	- E E E - - -	E E E L - - -
TR_26	E E E - - - -	D L L - - E E	L L - - - N N	N - - D D L N
TR_27	- - - D D L L	- - - E E - -	E D D D - - -	- D D - - - -
TR_28	- N N N - E E	N N N - - D D	N N N - - D D	L L - - E E D
TR_29	- - - - - D D	- - - - N N N	- - L L L - -	- N N N - - -

Figura 20 - Escala para instância n030w4_1_6-7-5-3

	M T W T F S S	M T W T F S S	M T W T F S S	M T W T F S S
HN_0	L - - E E N N	N - - N N N N	N - - L N N N	N - - E E L L
HN_1	- - L L L L	- - E E E E	- - E D D D	- N N N N
HN_2	L L L L D D	L L L L L L	L L L L L L	D D L L L L
HN_3	N N L L D D E	L L L L L L	D N N N N L L	- - D D N N N
HN_4	N - - N N N N	- - L L L L	E E E E E E	E E E E D D
HN_5	D D L L L L	- - E E E E E	- - - N N N	N N N L L L
NU_6	- - - N N N N	N - - - N N	N N L L L L	E E E E E E
NU_7	- N N N N	- E E E E	N N N N N N	- N N N N
NU_8	- N N N N	E E D D D D	E E L L L L	E E E E N N
NU_9	D D D L L L	- N N N N	- E E L L L	- D D D D
NU_10	L L L L E E D	D D L L E E E	- - D D E E E	- - - D D D E
NU_11	E E E E E E	D D D D D D	L L L L L L	L L L L L L
NU_12	- - D D D E E	- - L L L L	D D D D D D	- D D E E E D
NU_13	D - - - D D E	E E E L L L	L - - E E E D	D - - L L L N
NU_14	N N L L L L	- - E E E D D	- - D D E L L	- - L L L L L
NU_15	- L L L L L	- - L L L L L	- - - E E E E	E D D D D D
NU_16	D D L L E E D	D L L L D D D	D D L L D D E	- N N N N
NU_17	N N N L L L	N N N N L L L	N N N N N N	- L L L L L
CT_18	- - D D D D	- N N N N	- - - E E D D	D - - - - -
CT_19	- - E E E E	- D D D D	D D D D D D	- E E E E
CT_20	E E E E E E	E E L L L L	- D D L L L L	L - - - - E E
CT_21	- - E E E E E	- - - - D D	L L L L L L	N N N L L L
CT_22	E E E E E E	L L L L E E D	D D L L E L L	L - - E E D D
CT_23	- N N N N	- N N N N	- - D D D D	- D D D D
CT_24	L L L L D D	E E E E E E	L L L L D D D	- E E D D D
CT_25	E E D D D D	L L N N N N	L L L L L L	E E E E L L
CT_26	L L L L L L	L - D D D D	E E E E E E	- L L L L N N
CT_27	- - E E D D D	- - - - N N	N N L L L L	D D D D D D
CT_28	D D D L L L	D D D L L L	N N N L L L	N N L L E E E
CT_29	L L L L L L	E E E E E E	N N N N N N	D D D E E E
CT_30	E E E D D D	D D D D D D	E E L L L L	L L L L D L
CT_31	D - - E E L L	L - E D D D	D D E E E E	L L N N N N
TR_32	N - - - E E N	N N N L L L	L L L L L L	E E D D D D
TR_33	- E E L L L	- N N N N	- - D D D D L	- - - E E E
TR_34	E E E E E E	L L L L L L	E E E E E E	D L L L L L
TR_35	D - - E E D D	D - - D L L N	N N L L E E L	L L L L N N N
TR_36	- D D D D	- L L L L	D D D D D D	D L L L L L
TR_37	- D N N N	- E E E E	- E E E E	N N N N N N
TR_38	N - - D D L L	L - - E D D E	E - - - D D E	L L L L D D
TR_39	L L L L L L	- D D D D	- - N N N N N	- - - L L L L

Figura 21 - Escala para instância n040w4_0_2-0-6-1

	M T W T F S S	M T W T F S S	M T W T F S S	M T W T F S S
HN_0	N N - - N N N	N N - - L L L	- - E E E - -	D E E E E - -
HN_1	N N N N N - -	- E L L - - -	L L L L - - -	- L L L - - -
HN_2	E E D D D - -	- D E E N N N	N - - - L L L	L L - - - D D
HN_3	- - D D E E E	L L - - - N N	N N N - - - -	N N N N N - -
HN_4	D - - D D D D	D - - E E L L	L - - N N N N	N - - D D N N
HN_5	L L L L - - -	- N N N N - -	- E E D D E E	- - - - D D E
NU_6	D D D - - - -	L L L - - - -	E E E - - - -	- L L L - - -
NU_7	- E E L L - -	- L L L - - -	D D D - - - -	L L L - - - -
NU_8	D D L L L - -	E E E - - D D	D D D - - D D	L - - L L L L
NU_9	- D D L L - -	- - N N N N N	- - - - E E E	- - - E E N N
NU_10	N N N N N - -	D D D D - - -	N N N N N - -	D D D D D - -
NU_11	L L L L - - -	E E E - - - -	- E E E E - -	- - - D D D D
NU_12	- - E E E L L	- - - D D D E	E - - E E N N	N N - - N N N
NU_13	E - - D D D D	- E E E E - -	D D D L L - -	E E E L L - -
NU_14	N N - - N N N	N N - - - D D	L L - - L L L	L - - N N N N
NU_15	- D D E E - -	- D D D D E E	- - - D D D D	- - - D D E E
NU_16	- - E E E N N	- - - E E L L	- - D D D - -	D D D E E - -
NU_17	N N N N N - -	- L L L N N N	- - - N N N N	N N - - - - -
CT_18	D D D - - - -	D E E - - - -	L L L - - - -	L L L L - - -
CT_19	D D D - - - -	E E E - - - -	L L L - - - -	D D D - - - -
CT_20	- E E E E - -	- D D D L L L	L - - - - E E	D D D D - - -
CT_21	E E E - - - -	- N N N N - -	- N N N - - -	N N N - - - -
CT_22	L L - - E E E	L - - D D D E	E - - E E E D	- - E E E L L
CT_23	N N N N - - -	D D N - - - -	E E E - - - -	N N N - - - -
CT_24	- - E E E D D	- D D E E - -	D D E E E - -	E E E - - D D
CT_25	L - E E E - -	E E L L L - -	N N N N N - -	E E E L L - -
CT_26	- - L L N N N	- D D E E - -	E E D D D - -	N N N N N - -
CT_27	L L L - - - -	L L L L - - -	- D D D D - -	- N N N N - -
CT_28	L L - - D D E	E - - E E E D	D - - L L L L	L - - E E L L
CT_29	D - - L L L L	- - D D L - -	L L L L L - -	E E E E E - -
CT_30	E E D D D - -	N N N N N - -	E E E L L - -	- D D L L - -
CT_31	N N N - - L L	N N N N - - -	N N N N N - -	D D D - - E E
TR_32	L - - - D D L	L L L - - - -	D D D N N - -	- - N N N N N
TR_33	- D D D D - -	- - D D D D E	- - - E E - -	- - E E E E E
TR_34	- D D L L - -	- L L L - - -	- E E E - - -	N N N - - - -
TR_35	D L L - - N N	N N - - L L L	L - - D D D E	E - - D D D L
TR_36	N N N N - - -	D E E - - - -	N N N - - - -	D D D - - - -
TR_37	E E E - - - -	D D D - - - -	- L L L - - -	L L L - - - -
TR_38	- - E E D D E	- - N N N N N	- - D D L L N	- - D D D D L
TR_39	E - - - E E L	L - - - E E D	L L L - - - -	E E E L - - -

Figura 22 - Escala para instância n040w4_2_6-1-0-6

	M T W T F S S	M T W T F S S	M T W T F S S	M T W T F S S
HN_0	E - L L L - -	E E E L L - -	L L L - - L L	L L - - E L L
HN_1	N - - L L N N	- N N N N - -	D D E E E - -	N N N - - D D
HN_2	- - D D D D E	E - - - - L L	L L - - - -	E E E L L - -
HN_3	- - E E E - -	- - - D D E E	E - - - D D E	- - L L L - -
HN_4	L L L - - -	L L N N N - -	- - N N N N N	- - - - - E E
HN_5	D D D - - -	D D D - - -	D D L L L - -	- - N N N N N
NU_6	N N N N N - -	N N N N N - -	E E E L L - -	D D D E E - -
NU_7	- D D D D - -	- - E E E - -	- E E E - -	- E E E E - -
NU_8	D D L L - -	E E E E - -	E E E - -	- E E E E - -
NU_9	- - L L L - -	N N N N - -	- - E E E L L	- - - - D D D
NU_10	D D D D D - -	L L L L L - -	N N N N N - -	D D L L L - -
NU_11	E - - D D D D	D - - E D D L	L - L L L - -	D D D D D - -
NU_12	D D D E E - -	D D L L L - -	D D D - - D D	D - - E E D E
NU_13	L - - - - L L	L L - - - -	E E E - -	E E E - -
NU_14	E E E - -	N N N N N - -	L L L L L - -	N N N - -
NU_15	E E - - L L L	N N N - -	D D D - -	D D D - - E E
NU_16	N - - - - E E	L L L - -	E E E D D - -	- - - N N N N
NU_17	- - E E E - -	- D D D D - -	- D D D - -	N N N N N - -
NU_18	E E L - -	D D D - -	D D D - -	L L L - -
NU_19	L L - - L L L	L L - - E E E	D - - D D D E	E - - D D N N
NU_20	N - - - N N N	N N - -	L L N N N - -	D D D D D - -
NU_21	D D L - - D D	D - - N N N N	N - - E E E D	D - E E E - -
NU_22	L - - N N N N	- - E E E L L	- - D D D N N	N - - L L L N
NU_23	D - - E E E D	D - - E D D D	D - - D D D D	- - E E E - -
NU_24	N N N N N - -	E E E E E - -	N N N N N - -	E E E D D - -
NU_25	- - N N N N N	- - E E E E D	- - N N N N N	- - D D D - -
NU_26	D - - D D D E	- - L L L L L	- - E E E - -	L L N N N - -
NU_27	- E E E E - -	- D D L L - -	- - E E E - -	- L L L L - -
CT_28	E - - E E L L	L - - D D L L	L - - E E E E E	E - - L L L N
CT_29	N N - - - -	N N N N N - -	- - D D E E E E	- - - - D D D
CT_30	- E E E - -	E E L L - -	N N N N - -	D D D - -
CT_31	E E E D D - -	L L L - - N N	N N - - L L L	L - - L L N N
CT_32	- - L L L - -	E E E L L - -	- - D D D - -	- L L L L - -
CT_33	L L L L L - -	D D D D D - -	L L L - - L L	L L - - E E E
CT_34	D D D - -	E E E E - -	L L L L L - -	- - D D D - -
CT_35	N N N N N - -	- - N N N N N	- - N N N - -	E E E - -
CT_36	- D D D D - -	- D D D D - -	- - N N N - -	- N N N N - -
CT_37	L L L - - E E	L L L - - E E	E - L L L - -	N N N N N - -
CT_38	N N N N N - -	D D D D D - -	N N N N N - -	N N N - - L L
CT_39	- - - - E E E	E E - -	E E E - -	L L L - -
TR_40	D - - - E E E	- - - - D D D	D - - E E D D	- - - - D D D
TR_41	L - - - L L L	- - - - L L L	L - - - - E E	L L L - -
TR_42	- E E E - -	- - N N N N N	- - - - - N N	N N - -
TR_43	N - - D D L L	- - D D E E D	- - L L L L L	- - D D E E E
TR_44	- - L L L - -	E E L L L - -	E E E - -	D D E E - -
TR_45	L L L - - N N	N N N - - E N	N - - E N N N	- - N N N N N
TR_46	- N N N N - -	- - - E E L L	- - - - L L L	- - - L L - -
TR_47	E D D - -	- D D D D - -	- N N N N - -	D D E E - -
TR_48	N N N - -	L L N N N - -	E E E - -	E E N N N - -
TR_49	E E D - -	E E E - -	D D D D - -	- L L L - -

Figura 23 - Escala para instância n050w4_0_0-4-8-7

	M T W T F S S	M T W T F S S	M T W T F S S	M T W T F S S
HN_0	- -N N N N N N	- -N N N N N N	- N N N N - -	E E E D D - -
HN_1	N N - -D D D	N N -D D L L	- - -L L L N	N N - -E L L
HN_2	- - -E E E E	E - - -E E D	D - - -E E E	E D - - - -
HN_3	- -E E L L -	- N N N - -	- E E E E - -	L L L - -D D
HN_4	D L L L - -	L L L - -	E E E -D D	D - -E E L
HN_5	N N N N - -	E E E - -	N N N N - -	N N N N - -
NU_6	E E E D D -	N N N N N -	D D D E E -	N N N N N -
NU_7	-D D D D -	- -L L L -	- N N N N -	- N N N - -
NU_8	-E E E E -	D E E E E -	- -D D L L	L - -E E E
NU_9	- -N N N N N	- - -L L	L L - -N N	N N N - -
NU_10	L L L L L -	D N N N N -	N N N N N -	N N N N N -
NU_11	N N N -D D	D D -E E E	E -E E D D	L -D D E
NU_12	D D D E E -	N N N N N -	L L L -E E	D D -D D D
NU_13	- - -L L N	N N - -	D D D - -	L L L - -
NU_14	E L L - -	L L L -D D	D D -E E E	E E - -
NU_15	E -E E L L	L -D D L L	-D D D -	L L L - -
NU_16	N -L L N N	N N - -	E E E L L -	-L L L -
NU_17	-E E E - -	-L L L -	-E E E -	-E E E -
NU_18	E E E - -	D D D - -	E E E - -	D D L L -
NU_19	L -D D L L	L -L L N N	N -D D D D	-E N N N N
NU_20	N -D D D	E E -N N	N N N - -	E E E E -
NU_21	N N N N N -	D D E E E -	D D E E E -	E E E E -
NU_22	L L -E E E	N N -D D D	-E E D D -	D D D D -
NU_23	D -L L L L	-D D D D D	-L L N N N	N -E E E D
NU_24	D D E E -	L L L -L L	L L -E E D	D -N N N N
NU_25	-L L N N N	-E E E L L	-D D D L L	-D D N N N
NU_26	-D D D D D	-D D D E E	-L L L L L	-L L L L L
NU_27	L L L L L -	L L L L L -	-E E E -	D D D - -
CT_28	L L -E E E	D D -N N N	N -D D N N	N N -D L L
CT_29	N N - -	N N N N N -	L L L L L -	-D D D -
CT_30	-N N N N -	-D D D D -	-L L L L -	-E E E D D
CT_31	D D E E E -	L L L L L -	L L N N N -	L L L L L -
CT_32	D D D - -	D D D -E E	E -L L L	L -L L N
CT_33	D D D L L -	E E E L L -	N N N N N -	L L L L L -
CT_34	E E E - -	D D E E -	L L -D D E	D D -E E D
CT_35	E E D D D -	L L L - -	D D D - -	D D D D -
CT_36	-N N N N -	-E E E -	-D D D -	-N N N -
CT_37	D -D D L L	L -D D D D	D -L L L -	E E E D D -
CT_38	L L L L L -	E E E E E -	D D D -L L	L -L L N N
CT_39	- - -E E	E E - -	E E E L L -	-E E E -
TR_40	D -N N N N	-E E E D	D -D D	-L L L -
TR_41	-D D D	N N N -L L	N N N - -	D D D -N N
TR_42	-D D D -	-E E L -	-D D E E -	-E E E D
TR_43	N -L L L L	-D D N N N	-N N N N N	-E E E E L
TR_44	E -E E E	-D D L	L -N N N N	-N N N -
TR_45	-N N N N -	-N N N N N	-E D D E E	-D D D L L
TR_46	L L L - -	D D D - -	E L L - -	L L L - -
TR_47	D D E E -	-D D E E E	-L L L -	-N N N N
TR_48	-E E E -	L L L L -	D E E E -	N N N N -
TR_49	-L L L -	-L L L -	-L L L -	-D D D D E

Figura 24 - Escala para instância n050w4_0_7-2-7-2

	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	
HN_0	-	-	-	-	E	E	E	L	L	-	-	-	-	-	D	D	D	-	-	-	-	L	L	L	-	-	-	-	
HN_1	-	-	E	E	L	L	L	L	-	-	E	E	E	L	L	-	-	D	D	D	D	-	-	N	N	N	N	N	
HN_2	L	L	-	-	-	-	-	D	D	D	-	-	-	-	-	E	E	L	L	-	-	-	-	-	-	L	L	N	
HN_3	L	L	N	N	N	-	-	-	E	E	E	E	-	-	-	-	-	L	L	L	N	N	N	-	-	-	-	-	
HN_4	E	D	D	D	D	-	-	-	-	-	L	L	L	L	L	L	-	-	-	-	-	D	D	D	D	D	D	D	
HN_5	-	-	N	N	N	-	-	-	-	D	D	D	-	-	-	-	-	-	E	E	L	L	-	-	-	-	-	-	
HN_6	N	-	-	-	E	E	D	N	N	N	-	-	-	-	N	N	N	N	N	-	-	-	E	E	E	-	-	-	
NU_7	-	D	D	D	D	D	D	D	-	-	-	-	-	E	E	E	E	E	-	-	-	-	-	-	E	E	E		
NU_8	-	D	D	D	-	-	-	E	E	E	-	-	-	-	N	N	N	-	-	-	D	D	D	-	-	-			
NU_9	-	-	-	D	D	L	L	L	L	-	-	-	-	N	N	N	N	N	-	-	-	-	-	N	N	N	N		
NU_10	-	-	-	N	N	N	N	-	-	-	-	-	-	E	E	E	E	-	-	-	D	D	L	L	-	-			
NU_11	-	-	-	-	-	E	E	E	D	D	D	D	-	-	-	-	N	N	N	N	N	N	N	-	-	-	-		
NU_12	-	-	-	L	L	L	L	-	-	-	-	-	-	E	E	E	-	-	-	-	N	N	N	N	-	-			
NU_13	N	N	N	N	N	-	-	D	D	D	D	D	-	L	L	L	L	L	-	-	E	E	E	E	E	-			
NU_14	-	-	-	-	L	L	L	L	-	-	-	-	-	N	N	N	N	-	-	-	E	E	E	-	-	-			
NU_15	D	D	-	-	-	-	-	L	L	L	L	L	N	N	-	-	-	-	-	E	E	E	E	E	-				
NU_16	E	-	-	-	-	-	-	E	E	L	-	-	-	-	-	-	-	E	E	E	-	-	-	-	-	-			
NU_17	-	D	N	N	N	N	N	-	-	-	-	D	D	D	D	D	-	-	-	L	L	N	N	N	N				
NU_18	-	-	E	E	E	D	D	-	-	-	E	E	E	E	-	-	D	D	D	-	-	E	E	E	L	L	-		
NU_19	N	N	N	N	N	-	-	-	-	-	L	L	L	L	L	-	-	-	-	D	D	D	D	D	-				
NU_20	-	L	L	L	L	-	-	-	-	-	-	D	D	D	-	-	-	-	-	-	E	E	E	-	-	-			
NU_21	D	D	D	-	-	-	-	D	D	D	-	-	-	L	L	L	-	-	-	-	N	N	N	N	-				
NU_22	L	L	L	L	L	-	-	-	-	-	-	D	D	D	D	D	D	-	-	-	-	L	L	L	L	L			
NU_23	D	D	D	-	-	-	-	-	-	E	E	E	-	-	-	-	-	D	D	D	D	-	-	-	-	-			
NU_24	E	E	E	-	-	-	-	N	N	N	N	-	-	-	-	-	-	D	D	D	D	-	-	-	-	-			
NU_25	L	L	-	-	-	-	-	N	N	N	N	N	-	-	-	-	-	D	D	D	D	L	L	L	-	-	-		
NU_26	N	N	-	-	-	-	-	L	L	L	L	L	N	N	N	N	-	-	-	-	L	L	L	L	L	-			
NU_27	L	L	L	L	L	-	-	N	N	N	N	N	-	-	E	E	E	L	L	-	-	D	D	D	D	D			
NU_28	D	D	D	D	D	-	-	D	D	L	L	L	-	-	-	E	E	E	E	E	E	-	-	N	N	N	N		
NU_29	E	E	D	D	D	-	-	L	L	L	L	L	-	-	D	D	D	D	D	-	-	E	E	N	N	N			
NU_30	D	D	D	-	-	-	-	N	N	N	N	-	-	-	-	-	-	E	E	L	L	-	-	-	-	-			
NU_31	E	E	E	E	E	-	-	N	N	N	N	-	-	-	E	D	D	D	D	-	-	D	D	D	D	D			
NU_32	-	N	N	N	N	-	-	-	L	L	L	-	-	-	E	E	E	E	E	-	-	L	L	L	-	-			
NU_33	-	D	D	D	-	-	-	E	E	E	-	-	-	L	L	L	L	-	-	-	L	L	L	-	-				
NU_34	N	N	-	-	-	-	-	-	D	D	D	D	D	D	-	-	-	-	-	D	D	D	D	D	D				
NU_35	-	-	L	L	L	L	L	-	-	-	E	E	E	D	D	-	-	E	E	E	D	D	-	-	E	E	E	E	
NU_36	N	N	-	-	-	-	-	E	E	E	-	-	-	L	L	L	-	-	-	L	L	L	-	-	-				
CT_37	N	N	N	-	-	-	-	-	E	E	D	D	L	L	-	-	-	-	-	L	L	L	L	L	-	-	-		
CT_38	-	-	-	-	-	E	E	E	E	E	E	E	-	-	-	-	E	E	N	N	N	N	N	-	-	-			
CT_39	N	N	N	N	N	-	-	D	D	D	D	D	-	-	D	D	D	D	D	-	-	L	L	L	L	-			
CT_40	E	E	E	-	-	-	-	E	E	E	E	-	-	N	N	N	N	N	-	-	-	D	D	D	-				
CT_41	L	-	-	E	E	D	D	-	-	E	E	E	-	-	D	D	D	D	D	-	-	N	N	N	N	-			
CT_42	-	E	E	L	L	N	N	N	-	-	-	-	-	L	L	L	L	L	-	-	-	-	-	E	E	E			
CT_43	-	-	E	E	E	-	-	-	D	D	L	L	-	-	-	N	N	N	N	-	-	-	-	D	L	L	L		
CT_44	E	D	-	-	E	E	E	E	-	-	E	E	E	E	-	-	L	L	L	L	L	-	-	E	E	E	D	D	
CT_45	E	E	E	E	E	-	-	D	D	D	D	D	-	-	E	E	E	E	E	-	-	E	E	E	E	E	-		
CT_46	E	E	E	E	-	-	-	-	-	N	N	N	N	N	N	-	-	-	-	-	E	E	E	E	E	-			
CT_47	-	E	E	D	D	D	D	D	-	-	-	-	-	L	L	L	L	L	-	-	-	-	D	D	D	D			
CT_48	L	L	-	-	-	-	-	L	L	N	N	N	N	-	-	-	-	-	E	E	E	D	D	-	-	-			
CT_49	L	L	L	-	-	-	-	-	-	D	D	D	-	-	-	D	D	D	-	-	-	-	-	L	L	L			
CT_50	L	L	L	L	L	-	-	-	-	-	-	E	E	E	E	D	D	-	-	-	-	L	L	L	L	L			
CT_51	-	-	-	-	N	N	N	N	N	N	-	-	-	-	-	E	E	N	N	N	N	N	-	-	-	-			
TR_52	N	N	-	-	D	D	L	-	-	-	D	D	D	L	L	-	-	E	D	D	N	N	-	-	E	E	E	D	D
TR_53	-	-	-	L	L	-	-	-	N	N	N	-	-	-	N	N	N	-	-	-	D	D	D	-	-	-			
TR_54	L	L	N	N	-	-	-	-	E	E	E	-	-	-	-	E	E	E	-	-	-	-	-	L	L	L			
TR_55	-	-	-	L	L	N	N	N	N	-	-	-	-	-	D	D	D	D	L	-	-	-	-	L	L	N	N		
TR_56	-	E	E	E	E	-	-	-	-	-	-	D	D	D	-	-	-	-	D	D	D	-	-	-	-	-			
TR_57	-	N	N	N	N	-	-	-	-	L	L	L	-	-	-	L	L	L	-	-	-	L	L	L	-	-			
TR_58	L	-	-	-	E	E	E	E	-	-	-	-	E	E	E	-	-	-	-	L	L	L	-	-	-	E	E	E	
TR_59	D	D	D	D	D	-	-	-	-	-	-	D	D	N	N	-	-	-	-	-	E	E	N	N	N	N	-	-	-

Figura 25 - Escala para instância n060w4_1_6-1-1-5

	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	
HN_0	-	-	-	-	E	E	D	D	-	-	-	-	-	-	E	E	E	-	-	-	-	-	-	E	E	L	-	-	
HN_1	-	-	E	E	D	L	L	-	-	E	L	L	-	-	D	D	D	-	E	D	D	D	-	-	D	D	L	L	
HN_2	L	L	-	-	-	-	-	N	N	N	N	-	-	-	-	-	-	-	N	N	N	N	-	-	-	-	-	-	
HN_3	D	D	D	L	L	-	-	L	L	N	N	N	-	-	-	-	-	L	L	L	L	L	-	-	-	N	N	N	
HN_4	N	N	N	N	N	-	-	-	-	-	-	D	D	D	L	L	L	-	-	-	-	-	-	E	E	E	E	E	
HN_5	-	-	-	-	D	D	N	N	-	-	-	-	-	-	N	N	N	N	-	-	-	-	N	N	N	N	-	-	
HN_6	L	L	-	-	-	-	-	E	D	D	D	-	-	-	E	L	L	-	-	-	E	E	N	N	N	-	-	-	
NU_7	-	N	N	N	N	N	N	N	-	-	-	-	-	-	D	D	D	D	D	-	-	-	-	-	-	N	N	N	
NU_8	-	E	E	E	-	-	-	-	D	D	D	-	-	-	-	D	D	D	-	-	-	-	E	E	E	E	-	-	
NU_9	-	-	-	D	D	D	D	L	L	L	-	-	-	-	-	-	D	D	D	D	D	D	D	-	-	-	-	-	
NU_10	-	-	-	-	E	E	D	D	-	-	-	-	-	-	E	E	E	-	-	-	-	-	L	L	L	L	-	-	
NU_11	-	-	-	-	N	N	N	N	N	N	-	-	-	-	-	-	N	N	N	N	N	N	N	-	-	-	-	-	
NU_12	-	-	E	E	E	-	-	-	-	D	D	L	-	-	-	-	D	D	D	-	-	-	-	N	N	N	-	-	
NU_13	D	D	D	D	D	-	-	D	D	D	D	D	-	-	-	D	D	D	D	L	L	-	-	N	N	N	N	N	
NU_14	-	-	L	L	L	-	-	-	-	-	D	D	L	L	-	-	-	-	-	-	-	E	E	E	-	-	-	-	
NU_15	D	D	D	-	-	-	-	-	-	L	L	L	L	L	N	N	N	-	-	-	-	-	-	D	D	N	N	N	
NU_16	E	E	-	-	-	-	-	D	D	D	-	-	-	-	N	N	N	N	-	-	-	-	-	-	D	D	L	L	
NU_17	-	N	N	N	N	N	N	N	-	-	-	-	-	-	D	D	D	D	D	-	-	-	-	-	E	E	E	E	
NU_18	D	-	-	L	L	L	L	-	N	N	N	N	-	-	N	N	N	N	N	-	-	E	E	D	D	D	-	-	
NU_19	D	D	D	D	D	-	-	-	-	-	L	L	L	L	L	L	-	-	-	-	-	N	N	N	N	-	-	-	
NU_20	-	E	L	L	-	-	-	-	E	E	E	-	-	-	-	-	L	L	L	-	-	-	-	-	-	D	D	D	
NU_21	-	N	N	N	N	-	-	-	-	-	-	N	N	N	N	-	-	-	-	-	-	D	D	D	-	-	-	-	
NU_22	D	D	D	D	D	-	-	-	-	-	-	D	D	D	D	D	-	-	-	-	-	-	D	D	D	D	L	L	
NU_23	-	-	-	N	N	N	N	-	-	-	-	-	-	-	E	E	E	-	-	-	-	L	L	L	-	-	-	-	
NU_24	E	E	E	-	-	-	-	N	N	N	N	-	-	-	-	L	L	L	L	-	-	-	-	-	-	L	L	L	
NU_25	L	L	-	-	-	-	-	L	L	L	L	L	-	-	-	-	-	E	E	E	E	L	L	L	-	-	-	-	
NU_26	N	N	N	-	-	-	-	-	D	D	D	L	L	L	-	-	-	-	-	D	D	D	L	L	L	-	-	-	
NU_27	L	L	L	L	L	-	-	N	N	N	N	N	-	-	L	L	L	L	L	-	-	L	L	L	L	-	-	-	
NU_28	D	D	D	D	D	-	-	D	D	D	-	-	E	E	N	N	N	N	-	-	-	N	N	N	N	-	-	-	
NU_29	L	L	L	L	L	-	-	D	D	L	L	L	-	-	-	E	E	L	L	L	L	-	-	E	E	D	D	D	
NU_30	E	E	E	-	-	-	-	E	E	E	-	-	-	-	-	E	E	E	E	-	-	-	-	-	-	E	E	E	
NU_31	E	E	E	L	L	-	-	E	E	E	E	E	-	-	D	D	L	L	L	-	-	E	E	-	-	E	E	E	
NU_32	-	-	D	D	D	D	D	-	-	-	N	N	N	N	-	-	-	-	N	N	N	N	N	-	-	-	-	-	
NU_33	-	-	-	E	E	E	E	E	-	-	-	N	N	N	N	N	-	-	-	-	-	E	E	E	-	-	-	-	
NU_34	N	N	-	-	-	-	-	L	L	N	N	N	-	-	-	-	-	E	E	E	E	L	L	-	-	-	-	-	
NU_35	E	-	-	D	D	D	D	-	-	E	E	E	-	-	L	L	N	N	N	-	-	D	D	D	D	D	-	-	
NU_36	N	N	-	-	-	-	-	E	E	L	L	-	-	-	-	-	D	D	D	-	-	-	-	-	E	E	E	E	
CT_37	N	N	N	-	-	-	-	-	D	D	D	D	D	D	D	-	-	-	-	-	-	D	D	D	D	D	-	-	
CT_38	L	-	-	-	-	-	-	E	E	E	E	E	D	D	-	-	-	-	-	E	E	E	L	L	L	L	-	-	
CT_39	-	E	E	E	E	-	-	E	E	E	L	L	-	-	E	E	E	E	E	-	-	E	E	E	E	E	-	-	
CT_40	E	E	E	-	-	-	-	L	L	L	-	-	-	-	E	E	E	-	-	-	-	-	E	E	E	-	-	-	
CT_41	-	-	E	E	E	E	E	-	N	N	N	N	-	-	D	D	D	L	L	-	-	E	E	E	E	E	-	-	
CT_42	-	E	E	D	D	D	D	L	-	-	-	-	-	-	E	E	E	E	E	-	-	-	-	-	-	D	D	D	
CT_43	-	L	L	L	-	-	-	L	L	L	-	-	-	-	E	E	E	-	-	-	-	N	N	N	-	-	-	-	
CT_44	-	-	-	L	L	L	L	-	D	D	D	D	-	-	N	N	N	N	N	-	-	D	D	D	L	L	-	-	
CT_45	N	N	N	N	N	-	-	E	E	E	E	E	-	-	-	L	L	L	L	L	-	-	-	E	E	E	L	L	
CT_46	L	L	-	-	-	-	-	E	E	E	L	L	-	-	-	-	-	D	D	D	D	L	L	L	-	-	-	-	
CT_47	L	L	L	L	L	-	-	-	-	-	E	E	E	E	D	D	-	-	-	-	-	L	L	L	L	L	-	-	
CT_48	D	D	D	-	-	-	-	-	-	E	E	E	D	D	D	D	-	-	-	-	-	D	D	D	D	D	D	D	
CT_49	D	D	D	-	-	-	-	D	D	D	-	-	-	-	-	-	E	E	E	-	-	-	-	L	L	L	-	-	
CT_50	L	L	L	L	L	-	-	-	-	-	-	E	E	E	L	L	-	-	-	-	-	N	N	N	N	N	-	-	
CT_51	-	-	-	E	E	L	L	L	L	-	-	-	-	-	L	L	L	L	L	-	-	-	-	-	N	N	N	N	
TR_52	-	-	E	E	D	D	D	-	E	E	N	N	-	-	E	E	E	L	L	-	-	-	D	D	D	L	L	L	
TR_53	-	-	-	-	-	E	E	L	L	-	-	-	-	-	N	N	N	N	-	-	-	L	L	L	-	-	-	-	
TR_54	D	L	L	-	-	-	-	N	N	N	N	-	-	-	-	-	-	N	N	N	-	-	-	-	-	-	D	D	D
TR_55	-	E	E	D	D	-	-	-	-	-	L	L	N	N	N	-	-	-	-	D	N	N	N	N	-	-	-	-	
TR_56	-	-	-	N	N	N	N	-	-	-	-	-	-	-	E	D	D	-	-	-	-	D	D	D	-	-	-	-	
TR_57	-	L	L	L	-	-	-	D	D	D	D	D	-	-	-	-	-	E	E	L	L	-	-	-	-	E	E	E	
TR_58	L	-	-	-	L	L	L	L	-	-	-	E	E	D	D	-	-	-	-	E	E	E	E	-	-	-	-	-	
TR_59	-	-	D	D	D	D	-	-	-	-	-	-	D	D	L	L	L	L	-	-	-	-	-	-	-	D	D	N	N

Figura 26 - Escala para instância n060w4_1_9-6-3-8

