



SOFTWARE WEB PARA O PROBLEMA DE ESCALONAMENTO DE ENFERMEIRAS

Rafael A. M. Gomes

Departamento de Ciência da Computação, Universidade Federal de Ouro Preto
Campus Morro do Cruzeiro, Ouro Preto, Minas Gerais, 35400-000, Brasil.
rafael.amgomes@gmail.com

Túlio A. M. Toffolo

Departamento de Ciência da Computação, Universidade Federal de Ouro Preto
Campus Morro do Cruzeiro, Ouro Preto, Minas Gerais, 35400-000, Brasil.
tulio@toffolo.com.br

Haroldo Gambini Santos

Departamento de Ciência da Computação, Universidade Federal de Ouro Preto
Campus Morro do Cruzeiro, Ouro Preto, Minas Gerais, 35400-000, Brasil.
haroldo.santos@gmail.com

Nando Oliveira Coelho

Departamento de Ciência da Computação, Universidade Federal de Ouro Preto
Campus Morro do Cruzeiro, Ouro Preto, Minas Gerais, 35400-000, Brasil.
mim.noc@gmail.com

RESUMO

Este trabalho apresenta um *Software Web* capaz de especificar e resolver problemas de escalonamento de enfermeiras em hospitais. A Segunda Competição Internacional de Escalonamento de Enfermeiras estimulou o desenvolvimento de resolvedores eficientes para problemas desse tipo. Porém, esses resolvedores não possuem interface gráfica, assim, seu uso atualmente é praticamente restrito a pesquisadores da área. O presente trabalho propõe um *software Web* de fácil uso para que hospitais possam especificar e resolver problemas de escalonamento de enfermeiras através do resolvedor proposto para a competição. O *software* permite ainda a geração de novas instâncias desta classe de problemas para a literatura baseadas em situações reais.

PALAVRAS CHAVE. escalonamento, programação inteira, heurísticas, software web

Área Principal: SA, OC, MH

ABSTRACT

This work presents a Web-software able to specify and solve Nurse Rostering Problem in hospitals. The second Internation Nurse Rostering Competition stimulated the development of efficient solvers for such problems. However, these solvers don't have graphical interface, so their use is normally restricted to researchers in the field. The present work proposes an easy software tha allows the hospitals to specify and solve Nurse Rostering problems through the proposed solver for the competition. The software also allows the generation of new instances of this class of problems for the literature based on real situations.

KEYWORDS. scheduling, integer programming, heuristics, web software

Main Area: SA, OC, MH

Agradecimentos

Esta pesquisa foi apoiada pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq e pela Fundação de Apoio à Pesquisa do Estado de Minas Gerais – FAPEMIG.



1. Introdução

A criação de escalas otimizadas de trabalho do corpo técnico de enfermagem em um hospital é uma tarefa altamente complexa e onerosa para as instituições. Uma escala mal estruturada resulta em diversos fatores negativos para a administração, como: contratação excessiva, número insuficiente de profissionais, sobrecarga individual de funcionários, baixa qualidade de vida, motivação e, conseqüentemente, pior qualidade dos serviços de atendimento do hospital.

Existem diversas pesquisas referentes a métodos de solução para o Problema de Escalonamento de Enfermeiras (NRP - *Nurse Rostering Problem*) [Burke et al., 2004]. Em 2010, a Primeira Competição Internacional de Escalonamento de Enfermeiras (INRC-I - *First International Nurse Rostering Competition*) [Haspelslagh et al., 2012] foi proposta e um significativo número de algoritmos distintos foram apresentados para uma grande quantidade de instâncias propostas. Várias instâncias foram resolvidas de forma exata por Santos et al. [2014] em 2012.

Posteriormente, em 2014, a Segunda Competição Internacional de Enfermeiras (INRC-II - *Second International Nurse Rostering Competition*) foi organizada [Sara et al., 2014]. Diferentemente da primeira competição, um conjunto menor de restrições foi utilizado. No entanto, múltiplos horizontes de planejamento foram considerados, aproximando o problema ainda mais a situações do mundo real se comparada a INRC-I. [De Causmaecker et al., 2004]

[Römer e Mellouli, 2016], vencedor da INRC-II, considerou o problema como sendo um tipo multi-estágio com demanda futura incerta, ou em outras palavras, um problema de otimização estocástica de vários estágios. [Römer e Mellouli, 2016] também propôs uma abordagem determinística para o problema, onde, para cada estágio, um problema de programação linear inteira mista é resolvido para a semana de planejamento em questão, aumentada com demandas artificiais para as semanas seguintes como uma forma de antecipar as demandas das semanas futuras. O trabalho concluiu que a aproximação do valor futuro de um dado estado é difícil para este tipo de problema.

Os dados de entrada disponibilizados pela INRC-II fornecem informações suficientes para representar um problema real. Os diversos problemas possuem um determinado número de enfermeiras, horizonte de planejamento normalmente referente a um ou dois meses de trabalho e contratos com diversas características. Normalmente existem em torno de 3 a 4 contratos por problema. Cada enfermeira está vinculada a um único contrato específico e este contrato é que determina as regras de alocações a serem seguidas, portanto, as variáveis de restrições estarão sempre fazendo referência ao contrato da enfermeira.

Este trabalho apresenta um *software web* para o problema de escalonamento de enfermeiras, que se comunica em tempo real a outro módulo do sistema, neste caso, um resolvidor que utiliza uma formulação de programação linear inteira para o problema da INRC-II. Algumas estratégias de geração de colunas são avaliadas para o tratamento desta formulação presente no resolvidor. Além de resolver parte do problema usando Programação Inteira, o resolvidor mescla com heurísticas para acelerar significativamente a convergência do método.

O presente trabalho está organizado da seguinte forma. Seção 2 apresenta a contextualização do Problema NRP. Na Seção 3 é apresentado o resolvidor utilizado pelo software durante a etapa de otimização, detalhando a formulação matemática utilizada e as técnicas heurísticas empregadas. Seção 4 apresenta o software desenvolvido com detalhes sobre as etapas de utilização. Finalmente, conclusões trabalhos futuros são apresentados na Seção 5.

2. O Problema de Escalonamento de Enfermeiras

O Problema de escalonamento de enfermeiras pode ser descrito por uma visão enfermeira-dia, enfermeira-tarefa ou até mesmo uma visão do padrão enfermeira-turno [Cheang et al., 2003]. As alocações são indexadas para cada enfermeira e dia para uma visão enfermeira-dia. Isto permite uma solução ser representada diretamente por uma matriz onde cada célula m_{ij} contém um conjunto de turnos (*day*, *late*, *early* e *night*) e especialidade (*headnurse*, *nurse*, *trainee* e *caretaker*) para ser exercida por uma enfermeira i no dia j . A Figura 1 ilustra uma escala parcial de uma enfermeira que indica a alocação de um turno e especialidade em uma visão enfermeira-dia.



					weekend							weekend							weekend	
mon	tue	wed	thu	fri	sat	sun	mon	tue	wed	thu	fri	sat	sun	mon	tue	wed	thu	fri	sat	sun
	Day Nurse	Day Nurse			Early Nurse	Early Nurse			Night Care- taker	Night Care- taker		Day Care- taker	Day Care- taker		Day Trainee	Day Trainee		Night Trainee	Night Trainee	Night Trainee

Figura 1: Visão de alocação enfermeira-dia - Fonte: Autoria própria

A visão enfermeira-tarefa indexa cada variável decisão para cada enfermeira e tarefa realizada durante o período de escala. Estas variáveis assumem o valor 1 quando a enfermeira é alocada para uma certa tarefa e 0, caso contrário. Um levantamento bibliográfico de modelos e metodologias para resolver o NRP é apresentado por Cheang et al. [2003].

Três restrições *fortes*, que devem ser obrigatoriamente atendidas, são consideradas na INRC-II [Sara et al., 2014]: (i) uma enfermeira pode trabalhar no máximo um turno por dia; (ii) determinadas sequências de alocações de turnos não são permitidas; e (iii) dias e turnos possuem demandas associadas para diferentes enfermeiras com diferentes especialidades. Somente enfermeiras que atendem às especialidades exigidas podem trabalhar. Adicionalmente, a demanda mínima de enfermeiras por dia, turno e especialidade também são especificadas.

As soluções são avaliadas com respectivos custos de violação de sete restrições *fracas* presentes no problema : (i) quantidade insuficiente de equipe para a cobertura ótima; (ii) número mínimo/máximo de alocações consecutivas; (iii) número mínimo/máximo de folgas consecutivas; (iv) não atendimento das preferências solicitadas pela enfermeira; (v) finais de semana incompletos; (vi) número mínimo/máximo de alocações totais; e (vii) número mínimo/máximo de finais de semana trabalhados.

3. Resolvedor para o problema NRP

O presente trabalho apresenta um resolvedor que utiliza técnicas de programação linear inteira combinadas com métodos heurísticos e geração de colunas para otimizar o problema. Este resolvedor recebe os dados de entrada do problema, otimiza a escala e retorna a solução como resultado. Este resolvedor foi desenvolvido na linguagem java e utiliza *websocket* para estabelecer a comunicação com o *software* para receber os dados de entrada e retornar com a solução após o processo de otimização.

3.1. Formulação implementada no Resolvedor

A formulação presente no resolvedor foi proposta para o problema da INRC-II e contém um grande número de variáveis. A relaxação deste problema é resolvida pela técnica de Geração de Colunas Dantzig e Wolfe [1960] e geralmente fornece limites inferiores fortes.

Os conjuntos N , D , S e K_n representam as enfermeiras, dias, turnos e especialidades, respectivamente. Ω_n representa o conjunto de todos os possíveis padrões de alocações para enfermeira n , λ_{np} indica se a alocação $p \in \Omega_n$ é selecionada para a enfermeira n ($\lambda_{np}=1$) ou não ($\lambda_{np}=0$) e v_{dsk} mede a diferença entre a alocação mínima necessária e o número de enfermeiras alocadas no dia d e turno s com a especialidade k . Equações e inequações (1)–(5) apresentam a formulação proposta.

Minimizar:

$$\sum_{n \in N} \sum_{p \in \Omega_n} c_{np} \lambda_{np} + \sum_{d \in D} \sum_{s \in S} \sum_{k \in K} \omega v_{dsk} \quad (1)$$

Sujeito a:



$$\sum_{p \in \Omega_n} \lambda_{np} = 1 \quad \forall n \in N \quad (2)$$

$$\sum_{p \in \Omega_n} \alpha_{npdsk} \lambda_{np} - x_{ndsk} = 0 \quad \forall n \in N, d \in D, s \in S, k \in K_n \quad (3)$$

$$v_{dsk} - (r_{dsk}^* - r_{dsk}^-) \leq 0 \quad \forall d \in D, s \in S, k \in K_n \quad (4)$$

$$\sum_{n \in N} x_{ndsk} + v_{dsk} - r_{dsk}^* = 0 \quad \forall d \in D, s \in S, k \in K_n \quad (5)$$

A Função Objetivo (1) minimiza todas as violações das restrições fracas através das variáveis λ_{np} e custos c_{np} . As variáveis v_{dsk} são penalizadas pelo peso ω para cada unidade abaixo da cobertura ótima. Restrições (2) requerem a seleção de exatamente um padrão para cada enfermeira. Restrições (3) garantem que a variável x seja ativada somente se um padrão no qual a respectiva alocação existente esteja ativa. Finalmente, Restrições (4) e (5) medem o quão bem satisfeitas estão as demandas mínimas e ótimas, onde r_{dsk}^- e r_{dsk}^* representam o mínimo e ótimo número de enfermeiras necessárias no dia d , turno s e especialidade k , respectivamente.

3.1.1. O Problema de Pricing

O Problema de Pricing consiste em encontrar padrões de alocação com custo reduzido negativo para cada enfermeira, considerando todos os dias do horizonte de planejamento. Para cada enfermeira, um subproblema é definido considerando variáveis duais μ e π_{dsk} para Restrições (2) e (3), respectivamente. As principais variáveis de decisão são x_{dsk} , que recebem valor um se a enfermeira trabalha no turno s do dia d usando a especialidade k e zero caso contrário.

O Problema de Pricing pode ser resolvido de forma exata usando uma Formulação de Programação Inteira similar a utilizada em Santos et al. [2014]. Esta formulação inclui variáveis binárias adicionais relativas a seleção de janelas de dias contíguos de trabalho e folga. A formulação provou ser extremamente eficiente para resolver problemas médios das instâncias de NRP, conforme apresentado em Santos et al. [2014].

3.2. Variable Neighborhood Search: Geração de colunas acelerada

Procedimentos heurísticos são utilizados para acelerar a produção de colunas factíveis com custos reduzido negativo. Nesta fase, uma solução é dado por um vetor S de tamanho $|D|$ com cada célula d_i representando a alocação (s/k) em um dia especificado onde o índice s representa o turno e k a especialidade. O valor ($-/_$) corresponde a um dia de folga.

As soluções iniciais factíveis inicializam a partir de uma solução vazia onde, para demanda não satisfeita, uma enfermeira compatível é aleatoriamente selecionada e alocada. O procedimento continua para todos os dias/turnos/especialidades necessários. A fase de busca local inicia-se assim que uma solução factível é obtida.

A fase de busca local é melhorada pela meta-heurística *Variable Neighborhood Search* (VNS), proposta por Mladenovic e Hansen [1997]. VNS perturba sistematicamente a solução atual (procedimento de *shake*) antes de executar a fase de descida em multi-vizinhança. A técnica implementada no resolvidor utiliza a variação mais comum do VNS, conhecida como *Sequential Variable Neighborhood Descent* (SVND) Mjirda et al. [2016], que aplica o método VND como busca local dentro do VNS.

Quatro estruturas de vizinhanças são consideradas dentro do VNS:

ALTERAR ALOCAÇÃO DIÁRIA - $\mathcal{N}^{CA}(s)$: alterando a alocação de uma solução em um dia específico ajustando o turno e especialidade.

ALTERAR ALOCAÇÕES DE JANELAS DE TRABALHO - $\mathcal{N}^{CW}(s)$: consiste na mudança de alocações de dias contíguos (janelas de trabalho) para um mesmo padrão de alocação.



INVERTER JANELAS DE TRABALHO - $\mathcal{N}^{IW}(s)$: janelas de trabalho e folga são inicialmente computadas e então trocadas sequencialmente; alocações uniformes são inseridas nas novas janelas de trabalho.

TROCAR ALOCAÇÃO - $\mathcal{N}^{SA}(s)$: Trocar a alocação entre dois dias diferentes.

Tabela 3.2 representa um exemplo com uma solução inicial s e uma solução modificada por cada tipo de vizinhança considerada. As quatro vizinhanças são apresentadas na mesma ordem em que são utilizadas dentro do procedimento VND.

Tabela 1: Ilustração dos movimentos aplicadas à solução s

	0	1	2	3	4	5	6	7	8	9	10
s	-/-	D/H	L/H	L/H	N/H	-/-	-/-	-/-	-/-	E/N	E/N
$s \oplus m^{CA}$	-/-	D/H	D/H	L/H	N/H	-/-	-/-	-/-	-/-	E/N	E/N
$s \oplus m^{CW}$	-/-	D/H	D/H	D/H	D/H	-/-	-/-	-/-	-/-	D/H	D/H
$s \oplus m^{IW}$	D/H	-/-	-/-	-/-	-/-	D/H	D/H	D/H	D/H	-/-	-/-
$s \oplus m^{SA}$	-/-	D/H	L/H	L/H	N/H	-/-	-/-	E/N	-/-	-/-	E/N

O procedimento VNS proposto mantém um conjunto (Conjunto Elite) com as melhores soluções obtidas durante a busca. Sempre que um ótimo local para todas as vizinhanças é gerado, o procedimento de *shake* é acionado através do qual um percentual de alocações são aleatoriamente alteradas (perturbação) antes de continuar a busca. O algoritmo termina quando o número máximo de interações sem melhora é alcançado ou quando o tamanho do Conjunto Elite definido é atingido.

4. Software Desenvolvido

O desenvolvimento de um *Software* que permita ao usuário final informar dados de entrada e visualizar os resultados de forma simples e direta é fundamental para que a pesquisa seja aplicada no mundo real.

A arquitetura de rede do software, apresentada em Figura 2, tem como objetivo principal simplificar a estrutura necessária para o funcionamento em hospitais de qualquer porte, através do desenvolvimento de solução nas nuvens. Este tipo de arquitetura onde o hospital necessita apenas de um computador, *tablet* ou *smartphone* conectado a internet, flexibiliza e torna viável a implantação da solução em qualquer instituição, principalmente as instituições públicas que normalmente tem dificuldade financeira para viabilizar projetos que exigem grandes investimentos.

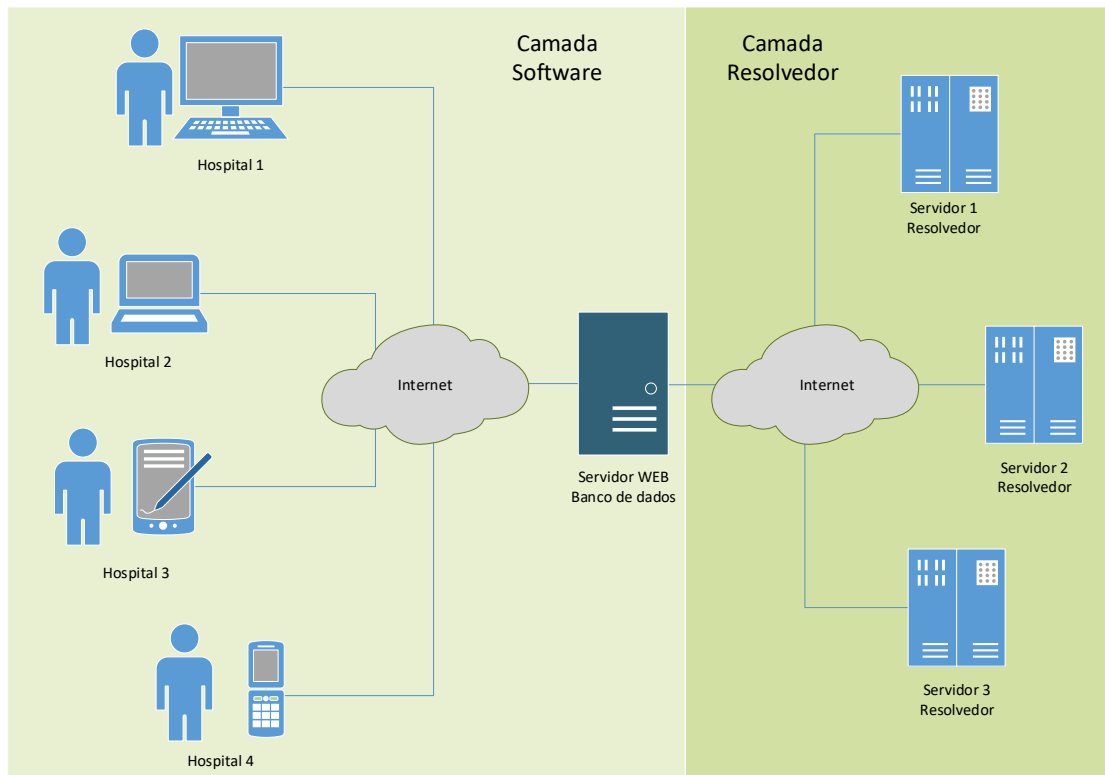


Figura 2: Arquitetura de rede. Fonte: Autoria própria

A separação da solução em duas camadas onde de um lado está o software propriamente com seu banco de dados e do outro o resolvedor, torna viável aos hospitais terem acesso à computadores com alta capacidade de processamento sem que necessariamente aconteça a aquisição deste tipo de equipamento.

A carga de processamento para resolver a formulação matemática juntamente com as demais técnicas heurísticas implementadas concentram-se exclusivamente no resolvedor, ou seja, a partir do momento em que o usuário informa os dados de entrada do software com as respectivas demandas necessárias, o software dispara o processo de solução do modelo matemático em um dos servidores que estiverem disponíveis naquele momento. A partir deste ponto, o software apenas aguarda o retorno da solução final do problema de otimização para interpretar e apresentar para o usuário de forma independente do resolvedor. Importante ressaltar que a execução do resolvedor não necessita de conexão ativa entre as duas camadas durante todo o processo de otimização.

4.1. Implementação e funcionamento do software

O software desenvolvido provê uma interface WEB que utiliza um *framework* baseado em Javascript que permite ao usuário especificar as informações de acordo com a sua realidade. As interfaces foram desenvolvidas objetivando simplificar a utilização por parte dos usuários, visto que o resolvedor utiliza uma formulação matemática onde os dados de entrada não são triviais de serem especificados em um formato direto.

No desenvolvimento do *software*, foi utilizado o modelo *Model-view-controller* (MVC) que permite a construção da aplicação em três camadas, através das quais objetos de diferentes classes assumem operações relacionadas ao domínio da aplicação (modelo), a exibição do estado da aplicação (visão) e os métodos que controlam o acesso ao modelo e a visão (controlador) Krasner et al. [1988].

O usuário inicia cadastrando as informações referentes aos tipos de contratos existentes no hospital e as especialidades presentes no corpo técnico de enfermagem.



Especialidade	Profissão	Capacitados
Caretaker	Enfermeiro(a)	22
HeadNurse	Enfermeiro(a)	2
Nurse	Enfermeiro(a)	16
Trainee	Enfermeiro(a)	5

Figura 3: Cadastro de contratos e tipos de especialidades *Fonte: Autoria própria*

Após informar os cadastros básicos, o usuário cadastra no sistema, Figura 4, as informações referentes as enfermeiras, registrando o tipo de contrato e a(s) especialidade(s) que cada funcionário(a) está habilitado(a) a exercer na equipe.

#	Especialidades
<input type="checkbox"/>	Caretaker
<input checked="" type="checkbox"/>	HeadNurse
<input checked="" type="checkbox"/>	Nurse
<input type="checkbox"/>	Trainee

Figura 4: Tela de cadastro de enfermeiras (*dados fictícios*) *Fonte: Autoria própria*

Os turnos de trabalhos existentes, deverão ser informados na tela ilustrada na Figura 5. Cada turno é identificado por uma cor específica para identificação na tela de alocações. Neste mesmo cadastro, o usuário informa as sequências de turnos que são proibidas. Este cadastro é fundamental para atender principalmente as leis trabalhistas que impedem determinados tipos de jornadas de trabalho.

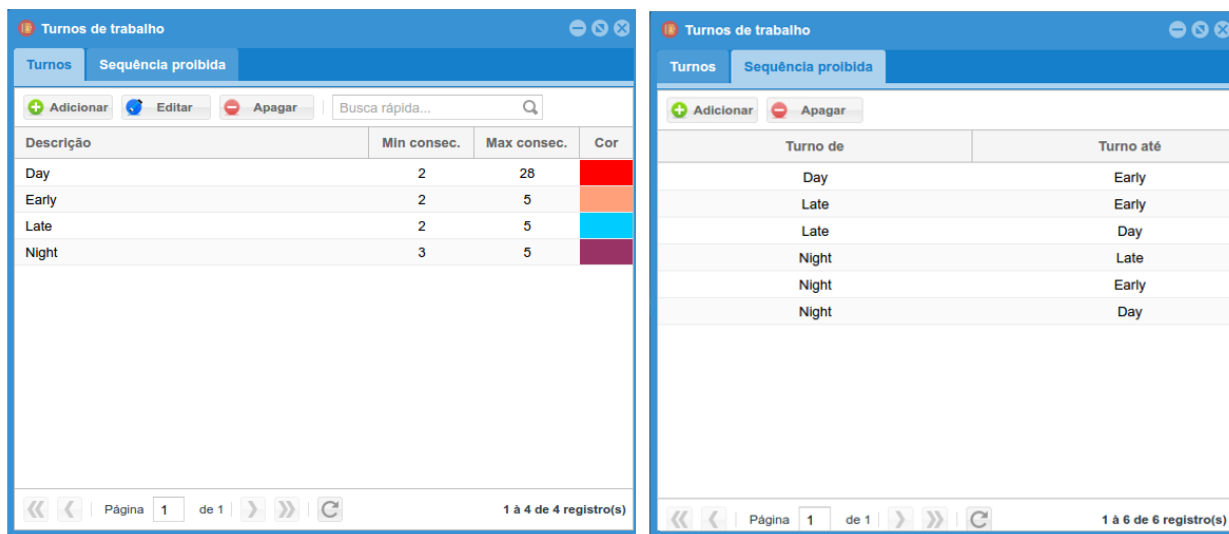


Figura 5: Tela de cadastro de turnos

A partir do momento que as informações referentes a equipe estão registradas no banco de dados, o usuário poderá começar a etapa de planejamento das escalas. A primeira etapa deste cadastro consiste na definição do horizonte de planejamento que deverá ser otimizado através da vinculação das semanas que farão parte desta escala. O horizonte de planejamento é composto por uma ou várias semanas e são informadas na tela apresentada na Figura 6.

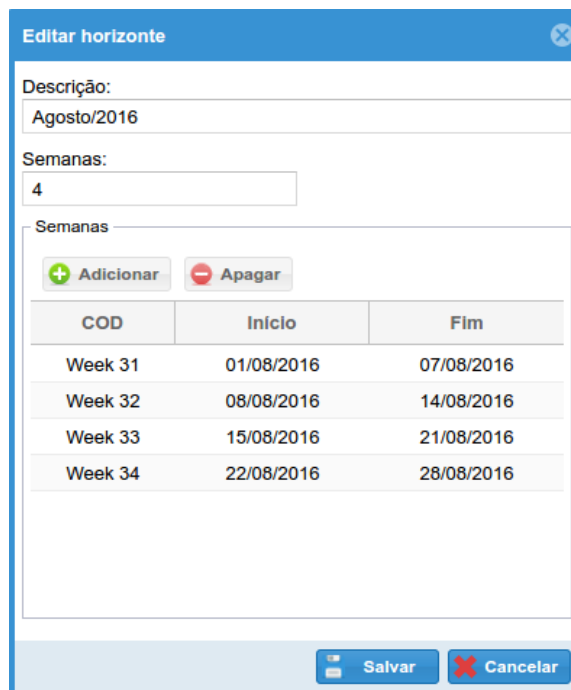


Figura 6: Tela de cadastro de horizontes de planejamento - Fonte: Autoria própria

A próxima etapa consiste em informar a demanda de profissionais em cada dia das semanas do horizonte de planejamento cadastrado, conforme tela ilustrada na Figura 7. Clicando duas vezes em qualquer uma das células das tabelas de cada semana, o usuário terá acesso ao cadastro da demanda propriamente, onde deverá ser informado o número mínimo e o ideal de funcionários,



para cada especialidade, necessário para cobrir aquele turno no dia selecionado, conforme Figura 8

Horizontes			WEEK 31	01/08/2016 Segunda	02/08/2016 Terça	03/08/2016 Quarta	04/08/2016 Quinta	05/08/2016 Sexta	06/08/2016 Sábado	07/08/2016 Domingo
Ano:	2016		Early	4 / 5	6 / 6	5 / 7	3 / 5	3 / 4	2 / 2	3 / 3
H	Descrição	Semanas	Day	3 / 3	4 / 6	2 / 3	4 / 5	5 / 6	3 / 4	3 / 3
	Julho/2016	4	Late	4 / 5	5 / 6	5 / 6	4 / 7	4 / 5	4 / 4	3 / 4
	Agosto/2016	4	Night	3 / 5	4 / 7	3 / 4	4 / 5	3 / 5	1 / 2	1 / 3
			<input type="button" value="Importar demanda"/> <input type="button" value="Pedidos de funcionários"/>							
			WEEK 32	08/08/2016 Segunda	09/08/2016 Terça	10/08/2016 Quarta	11/08/2016 Quinta	12/08/2016 Sexta	13/08/2016 Sábado	14/08/2016 Domingo
			Early	3 / 5	2 / 4	3 / 3	3 / 4	4 / 4	2 / 3	2 / 3
			Day	4 / 5	3 / 5	4 / 5	5 / 7	4 / 4	3 / 3	2 / 3
			Late	4 / 7	2 / 5	3 / 3	3 / 5	3 / 5	3 / 5	2 / 4
			Night	5 / 6	4 / 4	4 / 6	4 / 5	4 / 4	2 / 3	3 / 4
			<input type="button" value="Importar demanda"/> <input type="button" value="Pedidos de funcionários"/>							
			WEEK 33	15/08/2016 Segunda	16/08/2016 Terça	17/08/2016 Quarta	18/08/2016 Quinta	19/08/2016 Sexta	20/08/2016 Sábado	21/08/2016 Domingo
			Early	4 / 5	3 / 4	2 / 5	4 / 4	4 / 5	3 / 3	2 / 3
			Day	5 / 7	4 / 7	4 / 6	3 / 4	3 / 5	2 / 4	2 / 4
			Late	4 / 4	2 / 5	6 / 6	2 / 4	5 / 6	2 / 2	3 / 3
			Night	3 / 5	4 / 5	5 / 6	3 / 3	4 / 6	2 / 2	2 / 3
			<input type="button" value="Importar demanda"/> <input type="button" value="Pedidos de funcionários"/>							
			WEEK 34	22/08/2016 Segunda	23/08/2016 Terça	24/08/2016 Quarta	25/08/2016 Quinta	26/08/2016 Sexta	27/08/2016 Sábado	28/08/2016 Domingo
			Early	3 / 6	5 / 6	3 / 5	2 / 5	5 / 6	2 / 3	3 / 5
			Day	4 / 5	4 / 6	4 / 6	4 / 4	4 / 4	2 / 3	1 / 1
			Late	4 / 6	4 / 4	3 / 4	3 / 5	3 / 5	2 / 3	3 / 4
			Night	4 / 5	4 / 7	2 / 4	4 / 5	3 / 3	3 / 3	1 / 4
			<input type="button" value="Importar demanda"/> <input type="button" value="Pedidos de funcionários"/>							

Figura 7: Tela de cadastro de demanda - Fonte: Autoria própria

Adicionar demanda

Data: Turno:

Habilidades

- Caretaker
- HeadNurse
- Nurse
- Trainee

Demanda mínima:

Demanda ótima:

Figura 8: Tela de cadastro de demanda - Fonte: Autoria própria

Importante ressaltar que nesta mesma tela de cadastro de demanda, o usuário também informa o histórico de cada enfermeira. Esse histórico é importante, pois a partir desta informação, o usuário registra como foi o trabalho de cada enfermeira no último dia do horizonte anterior, como quantidade de dias trabalhados, última alocação e a quantidade de dias ou folgas que antecedem ao primeiro do horizonte atual.

Neste momento, todas as informações necessárias para o resolvidor otimizar a escala estão registradas no banco de dados. A partir deste ponto, através do módulo "Otimizar" representado



na Figura 9, o usuário seleciona o horizonte de planejamento a ser otimizado e em qual servidor deverá ser executado o resolvidor. A plataforma permite que sejam cadastrados vários servidores individualmente para cada hospital e o usuário acompanha a disponibilidade de cada um, visto que, um mesmo servidor também pode ser compartilhado por vários hospitais. Este módulo também permite o download da instância no formato padrão disponibilizado pela INRC-II, o que contribui para que novos problemas sejam gerados para comunidade acadêmica contrastar resultados.

Descrição	Semanas	Tarefas	Servidor	Status
Agosto/2016	4	1	Goal - HAWKEYE	🟢
Julho/2016	4	3	Goal - HULK	🟢
			Goal - OTHER	🟢
			Local	🟡

Início	Fim	Horizonte	Solver	Status
15/12/2016 17:23:40	15/12/2016 17:37:29	Agosto/2016	Local	🟢

Figura 9: Módulo de otimização de escalas - Fonte: *Autoria própria*

Ao clicar em otimizar, uma tarefa é aberta e o processo de envio das informações e execução do resolvidor é iniciado no servidor. Este processo de conexão é feito via tecnologia *websocket* e permite que a execução do resolvidor no servidor funcione de forma independente da sessão aberta no cliente. Todo o processo pode ser acompanhado pelo usuário em uma tela específica conforme a ilustrada pela Figura 10.

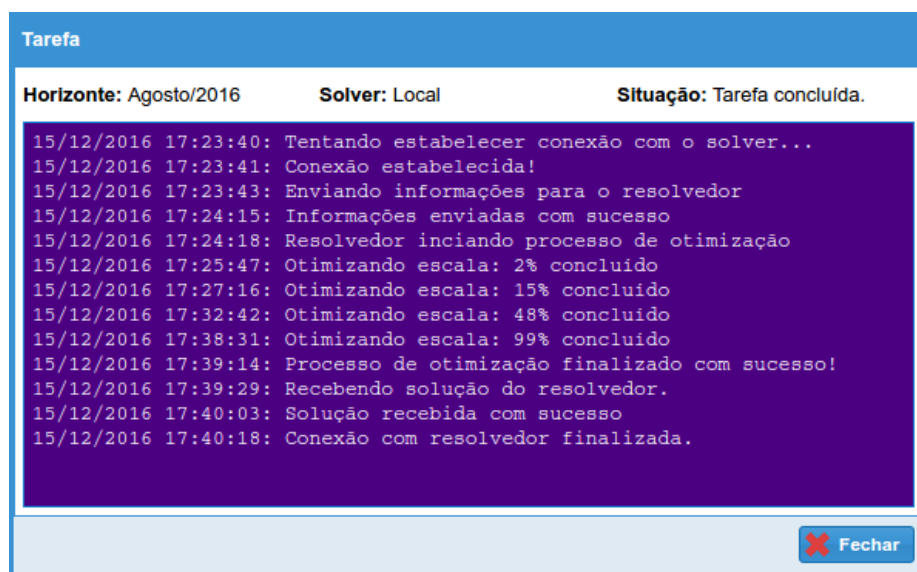


Figura 10: Acompanhamento do log de execução da tarefa - Fonte: Autoria própria

Após o processo de otimização, o resolvidor envia a solução final para o software. O sistema interpreta a solução e exibe para o usuário em uma interface de fácil interpretação. Foram implementados recursos de exportação dos resultados em outros formatos digitais como planilha eletrônica ou pdf. A Figura 11 apresenta a interface contendo o resultado da otimização de uma escala.

n035w4							
Exportar							
EARLY DAY LATE NIGHT							
COD	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
= WEEK: 0							
HN_0	-	Nurse	HeadNurse	Caretaker	HeadNurse	-	-
HN_1	Caretaker	HeadNurse	-	-	-	-	-
HN_2	Nurse	Nurse	-	-	-	-	-
HN_3	Nurse	Nurse	Caretaker	HeadNurse	-	-	-
HN_4	-	-	Caretaker	Caretaker	Nurse	-	-
HN_5	HeadNurse	-	-	-	Caretaker	HeadNurse	Nurse
NU_6	Caretaker	-	-	-	Caretaker	Nurse	Nurse
NU_7	Caretaker	Caretaker	Caretaker	-	-	-	-
NU_8	Caretaker	-	-	Nurse	Nurse	Nurse	Nurse
NU_9	Nurse	Nurse	-	-	Nurse	Nurse	Nurse
NU_10	Nurse	Nurse	Nurse	Caretaker	Caretaker	-	-
NU_11	-	-	Caretaker	Caretaker	Nurse	-	-
NU_12	-	-	-	-	Caretaker	Nurse	Nurse
NU_13	-	Nurse	Nurse	Nurse	-	-	-
NU_14	Caretaker	Nurse	Nurse	Nurse	-	-	-
NU_15	Caretaker	Caretaker	Nurse	Caretaker	Nurse	-	-
NU_16	-	Caretaker	Nurse	Nurse	Caretaker	-	-
CT_17	Caretaker	-	-	Caretaker	Caretaker	Caretaker	Caretaker

Figura 11: Resultado do processo de otimização - Fonte: Autoria própria



5. Conclusões e trabalhos futuros

O presente trabalho apresentou um *software Web* para a solução do problema de escalonamento de enfermeiras. A solução foi dividida em duas camadas onde de um lado está a interface onde o usuário interage com a solução e do outro, o resolvidor do modelo matemático que foi baseado no problema apresentado pela INRC-II. O software se mostrou eficiente na arquitetura proposta e foi capaz de solucionar problemas que foram gerados baseando em instâncias disponibilizadas pela competição. Como trabalho futuro, pretende-se fazer a análise do comportamento da aplicação e do modelo matemático proposto para a competição em um ambiente hospitalar público, onde a solução será validada em um ambiente real. Outra ação futura a ser aplicada nesta solução é fazer com que o software desenvolvido seja capaz de resolver outros problemas de otimização na área da saúde, como o problema de sequenciamento de salas de cirurgias. Esta ação consiste basicamente em propor um novo modelo matemático na camada do resolvidor que interprete os dados informados na mesma camada do software proposto, visto que grande parte das informações necessárias para o problema de escalonamento de enfermeiras é também utilizada para outros problemas de otimização em um ambiente hospitalar, tornando o software capaz de resolver os diversos problemas de otimização relacionados à área da saúde.

Referências

- Burke, E. K., De Causmaecker, P., Berghe, G. V., e Van Landeghem, H. (2004). The state of the art of nurse rostering. *J. of Scheduling*, 7(6):441–499. ISSN 1094-6136.
- Cheang, B., Li, H., Lim, A., e Rodrigues, B. (2003). Nurse rostering problems—a bibliographic survey. *European Journal of Operational Research*, 151(3):447 – 460. ISSN 0377-2217.
- Dantzig, G. B. e Wolfe, P. (1960). Decomposition principle for linear programs. *Operations Research*, 8(1):101–111.
- De Causmaecker, P., Demeester, P., Berghe, G. V., e Verbeke, B. (2004). Analysis of real-world personnel scheduling problems. In *Proceedings of the 5th international conference on practice and theory of automated timetabling, Pittsburgh*, p. 183–197.
- Haspelslagh, S., DeCausmaecker, P., Schaerf, A., e Stølevik, M. (2012). The first international nurse rostering competition 2010. *Annals of Operations Research*, 218(1):221–236.
- Krasner, G. E., Pope, S. T., et al. (1988). A description of the model-view-controller user interface paradigm in the smalltalk-80 system. *Journal of object oriented programming*, 1(3):26–49.
- Mjirda, A., Todosijević, R., Hanafi, S., Hansen, P., e Mladenović, N. (2016). Sequential variable neighborhood descent variants: an empirical study on the traveling salesman problem. *International Transactions in Operational Research*. ISSN 1475-3995.
- Mladenovic, N. e Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11):1097 – 1100. ISSN 0305-0548.
- Römer, M. e Mellouli, T. (2016). Future demand uncertainty in personnel scheduling: Investigating deterministic lookahead policies using optimization and simulation. In *Proceedings, 30th European Conference on Modelling and Simulation, ECMS 2016: May 31st - June 3rd, 2016, Regensburg, Germany*. European Council for Modelling & Simulation. ISBN 9780993244025. URL <https://books.google.com.br/books?id=PSsRvgAACAAJ>.
- Santos, H. G., Toffolo, T. A. M., Gomes, R. A. M., e Ribas, S. (2014). Integer programming techniques for the nurse rostering problem. *Annals of Operations Research*, 239(1):225–251. ISSN 1572-9338.
- Sara, C., Dang, N. T. T., De Causmaecker, P., Haspelslagh, S., e Schaerf, A. The second International Nurse Rostering Competition. PATAT 2014, York, 26-30 August 2014, August 2014.