



UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ADMINISTRAÇÃO
DISSERTAÇÃO DE MESTRADO



Otimização da Escala de Trabalho de Policiais Penais no Rio Grande Do Sul

Autor: Andrei Felipe da Silva Nunes

Orientador: Prof. Dr. Luciano Ferreira

Porto Alegre, 4 de agosto de 2022

Sumário

Agradecimentos	iii
Resumo	v
Abstract	vii
Lista de Figuras	ix
Lista de Tabelas	xi
Lista de Abreviaturas e Siglas	xii
Lista de Símbolos	xiv
1 INTRODUÇÃO	1
2 OBJETIVOS	7
2.1 Objetivo Geral	7
2.2 Objetivos Específicos	7
3 REFERENCIAL TEÓRICO	9
3.1 O Problema de Escalonamento de Pessoal e suas etapas	10
3.2 Principais Aplicações do Problema de Escalonamento de Pessoal (PEP)	11
3.3 Resolução do PEP por meio da Meta-heurística Greedy Randomized Adaptive Search Procedure (GRASP)	14
4 METODOLOGIA	15
4.1 Formulação Específica	15
4.1.1 Definição da Demanda	15
4.1.2 Designação de Pessoal	17
4.2 Modelo	19
4.2.1 Função Objetivo 1 - FO1	24
4.2.2 Função Objetivo 2 - FO2	24
4.2.3 Restrições	25
4.3 Resolução	27
4.4 Validação	28
5 ALGORITMOS	29
5.1 Solvers	29
5.1.1 Formulação dos Algoritmos Específicos	29

5.2	Heurística	31
5.2.1	Formulação dos Algoritmos Específicos	32
6	EXPERIMENTOS COMPUTACIONAIS	39
6.1	Solvers	39
6.2	Heurística	40
6.2.1	Ajuste dos parâmetros	40
6.2.2	Resolução das instâncias	46
7	AVALIAÇÃO DO MODELO	53
7.1	Desempenho relacionado aos custos	53
7.2	Desempenho relacionado a demanda e disponibilidade de horas	56
7.3	Desempenho relacionado a FO2	58
7.4	Desempenho dos Métodos Exatos x Heurística	60
8	CONCLUSÃO	63
9	Referências	65

Agradecimentos

Ao Prof. Luciano Ferreira, pela orientação, competência, profissionalismo e dedicação. Foi valiosa a contribuição em todo percurso, sendo fundamental o incentivo para a finalização do trabalho. Aos membros da banca examinadora, Prof. Pablo Cristini Guedes, Prof. Denis Borenstein e Prof. Adiel Teixeira de Almeida Filho que gentilmente aceitaram participar e colaborar com esta dissertação. A todos professores do PPGA que puderam, de alguma forma, colaborar nessa caminhada dividindo conhecimento e experiências.

Aos colegas e amigos da Pesquisa Operacional, com os quais dividi a experiência de prosseguir em um mestrado junto à uma inédita pandemia mundial, a qual, além de todo sofrimento, ainda tornou o curso mais desafiador em todos sentidos. Especial lembrança aos amigos Márcio, Vilmar e Marcos que dividiram os trabalhos em grupo e sempre estiveram prontos a ajudar, cada um de sua maneira e com seu conhecimento, sendo sempre generosos.

Aos colegas de trabalho e chefes, que sempre apoiaram e incentivaram a capacitação profissional, inclusive fornecendo dados para que o estudo pudesse ser realizado. Nesse ponto, especial agradecimento a todos da SEAPEN e da SUSEPE pela compreensão e disponibilização de dados, objetivando a melhora de sua instituição. Ao agradecer nominalmente o Secretário César Luis de Araújo Faccioli, que me deu oportunidade de realizar o estudo e sempre acreditou no meu trabalho, agradeço a todos servidores da SEAPEN. No mesmo sentido, agradeço a Delegada Regional Samantha, servidora dedicada e sempre pronta a dividir suas experiências, e a todos servidores da SUSEPE.

Por fim, a minha família, meu porto seguro que sempre está pronta a me incentivar e apoiar. Minha esposa Greicy, paciente e amorosa, deu todo suporte nos momentos que precisei me ausentar. Meus filhos amados, Caio e Nicolás, são meu incentivo pra tudo que faço e me ensinam mais a cada dia que passa. Meus pais, Otacílio e Fátima, a quem sempre busco honrar. Meu irmão, Tiago, e irmã, Ariane que são fontes de amor e inspiração. Minhas afilhadas, Giovana e Cecília, e sobrinhas, Olívia e Clara, que trazem muita alegria a família. Meus sogros, Marco e Vera, que me acolhem como filho e a quem desejo orgulhar. Aos meus cunhados, Bruno e Jéssica, que junto com seus companheiros, Tiago e Laís, trazem alegrias e boa conversa pra amenizar as dificuldades do dia.

Resumo

O presente trabalho tem por objetivo propor uma modelagem da alocação de Policiais Penais da Superintendência dos Serviços Penitenciários (SUSEPE) e desenvolver um método de resolução. Tal objetivo será resolvido por meio da utilização de métodos de otimização, os quais podem ser encontrados na literatura com diversas propostas de resolução desse tipo de problema, considerando os métodos exatos, utilizando solvers, e aproximativos, utilizando a metaheurística GRASP. Apesar de ser amplamente estudada, tal problemática não apresenta uma formulação genérica, sendo necessária uma proposição para casos específicos. Geralmente é utilizada a proposta de redução de custos financeiros, ou o aumento da satisfação de seus integrantes, porém esse trabalho visa aplicar uma resolução experimental em dois passos, abarcando os dois vieses.

Palavras-chaves: Programação inteira. Otimização linear. Escala de Trabalho. GRASP.

Abstract

The present work aims to propose a modeling to allocation of Public Servants from SUSEPE and to develop a resolution method. This objective will be solved through the use of optimization methods, which can be found in the literature with several proposals for solving this type of problem, considering the exact methods, using solvers, and approximations, using the GRASP metaheuristic. Despite being widely studied, this issue does not have a generic formulation, requiring a proposition for specific cases. Generally, the proposal to reduce financial costs or increase the satisfaction of its members is used, but this work aims to apply an experimental solution in two steps, covering both objectives.

Keywords: Integer Programming. Linear optimization. Work Scheduling. GRASP.

Lista de Figuras

1	Gráfico Liquidação da Despesa Poder Executivo 2020.	2
2	Evolução de Vagas x População Prisional x Policiais	3
3	Execução Orçamentária Secretaria de Administração Penitenciária (SEAPEN) 2020	5
4	Despesa de Pessoal SEAPEN 2020	6
5	Metodologia de definição de demanda	16
6	Fluxo Atual	19
7	Fluxo de Otimização	27
10	FO1 e FO2 - Instância 33	44
11	Tempo e Iteração - Instância 33	45
12	Dinâmica de vizinhança	49
8	Saída de informações e status do problema	51
9	Cria a Escala Otimizada	52

Lista de Tabelas

1	deficit	4
2	Turnos x Horas	21
3	Custo de hora Trabalhada	21
4	Preferências Pessoais	22
5	Demanda por dia e Turno	23
6	Disponibilidade	23
7	Opção por Hora extra (HE)	24
8	Habilitação como Supervisor (SP)	24
9	Tempo Limite de Execução	39
10	Exemplo de escala gerada	40
11	Definição de Parâmetros	41
12	Experimental Instância 33	42
13	Lista Custo Instância 7	47
14	Valores FO1 por método	53
15	Horas Normais e Extras utilizadas por método	54
16	% FO1 em relação ao Manual	55
17	Composição FO1 Custo Financeiro R\$	55
18	Composição FO1 Custo Descoberto "S"	56
19	Informações sobre os Estabelecimentos	57
20	Utilização de cada método do total disponível de horas	57
21	Utilização de cada método do total demandado de horas	58
22	Valores da FO2 alcançado por cada método	59
23	Desempenho de cada método em relação a Manual - FO2	59
24	Meta-heurística x Métodos exatos - FO1	60
25	Meta-heurística x Métodos exatos - FO2	61
26	Capacidade de Melhora Satisfação - FO2	61
27	Tempo de execução dos métodos - FO2	62

Lista de Abreviaturas e Siglas

ACO	Ant Colony Optimization. 64
ASTEC	Assessoria Técnica. 15, 16
BDSP	Bus Driver Scheduling Problem. 32
CA	Cultural Algorithms. 64
CNPCP	Conselho Nacional de Política Criminal e Penitenciária. 4
COIN	<i>Computational Infrastructure for Operations Research</i> . 15, 29
DPR	Delegacias Penitenciárias Regionais. 17, 18
FEE	Fundação de Economia e Estatística. 1
GA	Genetic Algorithms. 64
GRASP	Greedy Randomized Adaptive Search Procedure. i, 14, 31, 32, 34, 35, 40–42, 46, 47, 55, 58, 60, 63
JSP	Job Shop Scheduling Problem. 31
LRC	Lista Restrita de Candidatos. 31–33, 47, 48, 50
OSI	<i>Open Solver Interface</i> . 15
PEP	Problema de Escalonamento de Pessoal. i, 9–11, 14, 15, 31
PO	Pesquisa Operacional. 6, 9, 11
PPT	Problemas de Programação de Tripulações. 12
PSO	Particle Swarm Optimization. 64
RRF	Regime de Recuperação Fiscal. 2
SA	Simulated Annealing. 64
SEAPEN	Secretaria de Administração Penitenciária. ix, 5, 6, 15, 16, 18
SUSEPE	Superintendência dos Serviços Penitenciários. v, vii, 17
TS	Tabu Search. 64

1 INTRODUÇÃO

Como cenário para a presente pesquisa, cabe ressaltar a importância da Pesquisa Operacional no setor Público e Privado como ferramenta de orientação qualificada na tomada de decisão. Destacam-se os ganhos operacionais, financeiros e até mesmo de satisfação devido a sua busca por soluções ótimas em qualquer contexto posto. Além disso, apesar de seu ganho ainda maior em escala, para empresas e setor público, verifica-se uma baixa utilização da metodologia no Brasil, focando principalmente na área Pública, tomando-se por base o Estado do Rio Grande do Sul e seu setor público estadual, que não utiliza tal tipo de ferramenta. Levando em consideração que a despesa de pessoal é uma das mais relevantes no orçamento de qualquer empresa, o mesmo ocorre no serviço público. Por tal motivo, uma gestão eficiente dessa porção da despesa é primordial para a ampliação e aperfeiçoamento das políticas públicas.

Com isso em mente, é fato que o Estado do Rio Grande do Sul encontra-se em situação de profunda calamidade financeira nos últimos anos. As causas são as mais diversas e perpassaram todas as administrações do executivo, segundo a Secretaria do Planejamento (2018). Como é de conhecimento, o Estado obteve resultado fiscal positivo apenas sete vezes nas últimas quatro décadas, informação reiteradamente apresentada nas mensagens de encaminhamento da proposta de orçamento do Estado (Secretaria do Planejamento, 2018). Todas as medidas tomadas para cobrir os prejuízos foram paliativas e não se configuraram em coberturas de longo prazo. Em oposição a isso, a despesa continuava crescendo em ritmo acelerado. Na busca por alternativas para equilibrar suas finanças, recuperar sua capacidade de investimentos e manter a máquina pública, utilizaram-se os saques no caixa único, espécie de cheque especial do governo, depósitos judiciais e o endividamento por meio de atrasos de contratos e salários. Porém tais ações tem um limite que parece ter sido atingido.

De forma resumida, o enfrentamento do deficit fiscal pode ser feito em duas frentes:

- aumentando a receita;
- reduzindo a despesa.

Pelo viés da receita, muitos governos utilizaram a majoração de impostos, endividamento e antecipação de receitas. Boa parte das soluções apresentadas tinham prazos definidos, ou seja, não eram receitas de caráter continuado. Por outro lado, os governos tendiam a continuar os programas de isenções fiscais e descontos para pagamentos de débitos fiscais, diminuindo a eficácia do incremento produzido pelas medidas tomadas. O combate aos sonegadores e detentores de dívida ativa não tem sido suficiente sequer para que o estoque da dívida não aumente ainda mais, além disso, boa parte da dívida é classificada como não recuperável pelos técnicos. De qualquer forma, tal ação seguiria a ideia de medida de curto prazo e não solução estrutural, cobrindo apenas um ano e meio de deficit projetado, segundo Stein and Sulzbach (2016) em artigo publicado pela Fundação de Economia e Estatística (FEE).

Pelo lado da despesa, dentre as possibilidades para enfrentamento do problema, apresenta-se a

adesão ao Regime de Recuperação Fiscal (RRF) do Governo Federal. Esse mecanismo suspende o pagamento da dívida com a União por um período, sendo assim uma medida de curto prazo que acumula juros para pagamento futuro, contribuindo para a deterioração das condições de longo prazo. Na mesma linha estão os decretos de contingenciamento do Executivo que abarcam, basicamente, despesas de custeio. Essa medida já se mostrou pouco eficaz pelas várias exceções que contém e pelo ataque a uma despesa que gera apenas economia marginal, conforme demonstrado por Petry et al. (2018). Como alternativa a esse modelo tradicional, e tentando vislumbrar uma alternativa mais duradoura pelo lado da despesa, analisaremos os grupos que formam os gastos do Estado. No gráfico da Figura 1 a composição, por grupo de despesa, do orçamento estadual liquidado em 2020 para o Poder Executivo.

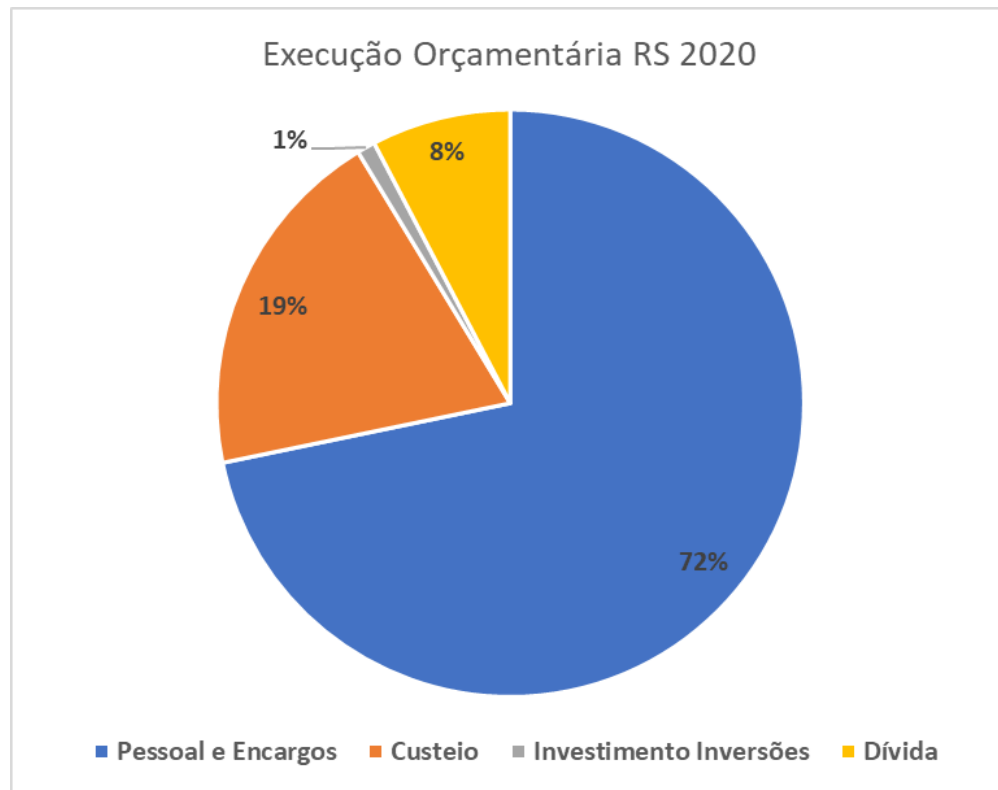


Figura 1: Gráfico Liquidação da Despesa Poder Executivo 2020.

Fonte: <http://www.transparencia.rs.gov.br/>

Dessas grandes divisões podemos concluir que:

1. **Investimento e Inversões:** representa 1% da despesa;
2. **Dívida:** parcela do orçamento que já vem sendo discutida, seja pela adesão ao RRF da União ou contestação de valores por meio da justiça;
3. **Custeio:** representa grande parte do orçamento estadual e é objeto de contingenciamento

regular pelas diversas gestões, sendo o ganho nesta economia apenas marginal, tendo em vista esta natureza de despesa manter a máquina pública funcionando;

4. **Pessoal e Encargos:** a maior despesa do Estado e a que apresentou o menor nível de gestão de todos governos, é sobre essa que nos debruçaremos de forma a buscar uma redução ou otimização em sua utilização.

Considerando, então, que a despesa com pessoal é a mais relevante e a que menos sofreu gestão, por seu claro engessamento legal, sobre a mesma vale um aprofundamento. Por ser uma despesa relevante, todo ganho em nível de eficiência nesse grupo pode gerar resultados significativos, uma vez que ela é demasiado importante para a manutenção dos serviços públicos. Por esse motivo, será dada atenção a essa parte da despesa, buscando ampliar as ferramentas de tomada de decisão para o Gestor.

Para entender as decisões que o gestor deve tomar, e suas implicações para o sistema penitenciário gaúcho, é importante se apropriar de alguns dados. A evolução histórica dos principais fatores de decisão na seara executiva é a representada na Figura 2.

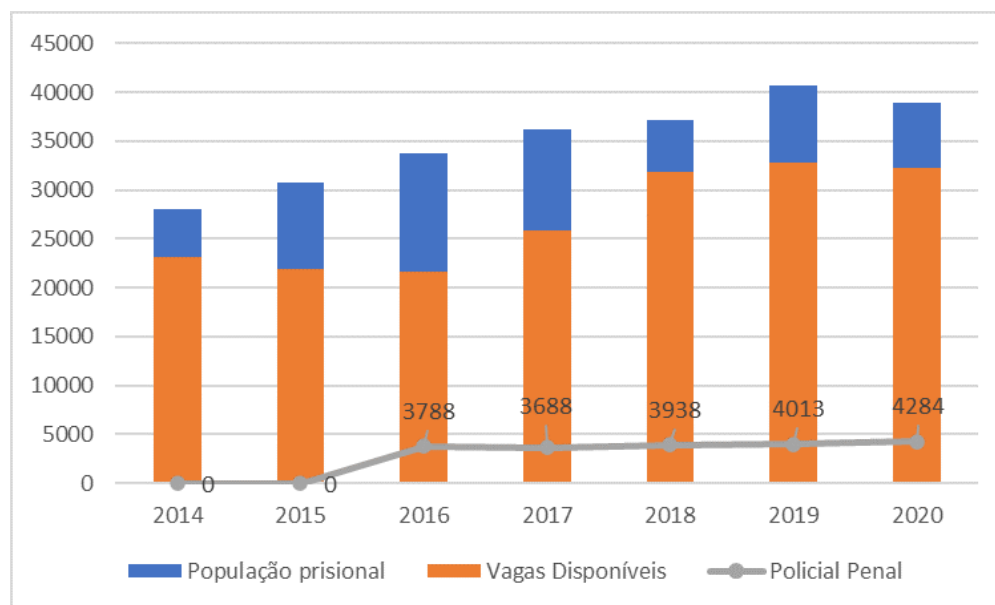


Figura 2: Evolução de Vagas x População Prisional x Policiais

Fonte: <http://www.transparencia.rs.gov.br/>

Fonte: <https://www.gov.br/depen/pt-br/sisdepen/>

Quanto ao histórico do deficit na Figura 2, fica clara a evolução do mesmo e suas variações, sendo importante referir que os dados sobre a quantidade de policiais penais não estavam disponíveis nos anos de 2014 e 2015, permanecendo zerado. Algumas variáveis dificultam a tomada de decisão, como a taxa de aumento da população carcerária e a lenta evolução no número de construções de

vagas prisionais. Existem outros fatores que não são objeto deste estudo, porém um dos que afeta a construção é a disponibilidade financeira do Estado. Este fator acaba limitando a capacidade de investimento em novas vagas.

Tabela 1: deficit

Dados	2014	2015	2016	2017	2018	2019	2020
População Prisional	28059	30713	33809	36149	37117	40687	38850
Vagas Disponíveis	23165	21896	21642	25813	31885	32801	32300
deficit de Vagas	4894	8817	12167	10336	5232	7886	6550

Fonte: <https://www.gov.br/depen/pt-br/sisdepen/>

A decisão por reduzir o deficit de vagas, que passa por mais do que a simples ampliação, necessita de disponibilização de recursos. Nesse caso, considerando o valor da vaga construída para o regime fechado de R\$ 41.342,14, com base na licitação da Cadeia Pública de Rio Grande para 388 vagas (CELIC), seria necessário um aporte de, aproximadamente, R\$ 270 milhões para zerar o deficit de 2020 da Tabela 1. Esse valor representa 15 anos de aporte da média dos últimos 4 anos de investimento da área prisional e 85 anos se considerarmos os últimos 2 anos, valor que representa a tendência, considerada a atual situação financeira do Estado. Obviamente a simples construção não é a solução para os problemas do encarceramento, porém serve para ilustrar a difícil tarefa na tomada de decisão quanto a alocação de recursos.

Além disso, com o aumento da população prisional, algumas necessidades, como o custeio, aumentam no sistema. Definimos o custeio como sendo, basicamente, água, luz, alimentação e contratos de serviços gerais. O atual custo para cada preso ingressante é de, em média, R\$ 23.178,12 por ano, considerando todos os gastos na área prisional.

Tratando-se de pessoal, o Conselho Nacional de Política Criminal e Penitenciária (CNPCCP), em sua resolução nº 9/2009, recomenda que existam 5 presos para cada policial penal, os dados da referida resolução remetem ao Painel Estatístico da União Européia¹ de 2006. Claro que esse número é médio e depende de outros fatores, como tipo de regime, classificação do estabelecimento e tipo de construção, por exemplo.

Os servidores penitenciários são contratados por meio de concurso público de provas e títulos. Após a seleção, os classificados passam por um curso de formação dependendo da área de atuação. Atualmente são divididos em 3 grandes áreas:

- Agentes Penitenciários - funções: Segurança e Movimentação;
- Agentes Administrativos - funções: Administração;
- Técnicos Penitenciários - Tratamento Penal.

Essas três grandes áreas formam a Polícia Penal, porém o assunto ainda encontra-se em discussão

¹https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Crime_statistics

dada a aprovação recente. O interesse pra esse estudo concentra-se na de Agentes Penais, pois esse grupo trabalha majoritariamente em escalas de serviços. Os postos são divididos dentro de cada estabelecimento, porém todos servidores são habilitados para as mesmas funções. Após alocados, os supervisores podem distribuir os Policiais em serviços como: movimentações de presos, revista, segurança externa e segurança interna, por exemplo.

Na SEAPEN do Estado do Rio Grande do Sul, responsável pela administração do sistema prisional do Estado, atualmente são geridas aproximadamente 40 mil pessoas presas em mais de 100 estabelecimentos prisionais. Para tal feito, existe uma carreira de servidores públicos específica, os Policiais Penais, que realizam atividades de segurança, reinserção social e as administrativas. Essa carreira soma mais de 5000 matrículas, quando consideramos esses 3 níveis de atendimento. Abaixo a perspectiva orçamentária da despesa da SEAPEN.

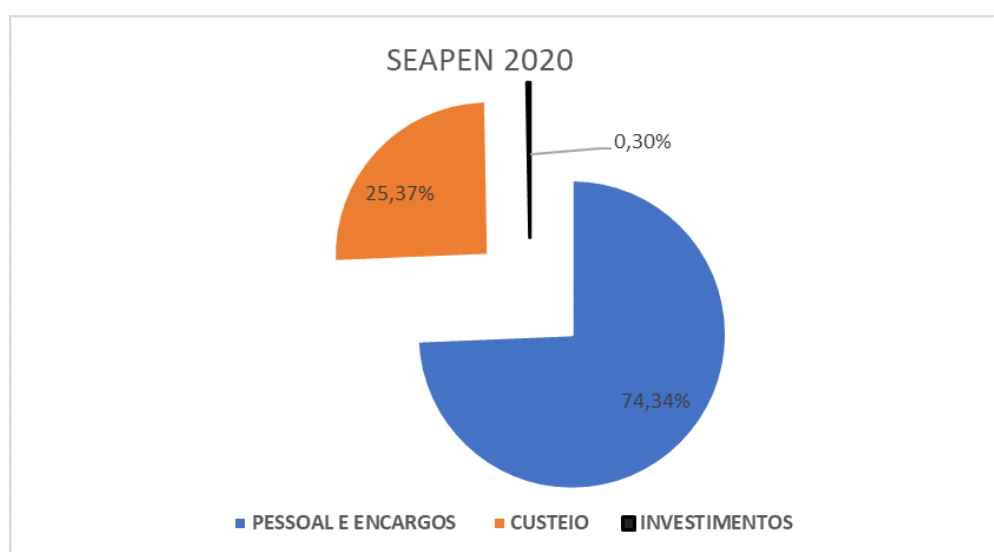


Figura 3: Execução Orçamentária SEAPEN 2020

Fonte: <http://www.transparencia.rs.gov.br/>

Pelo gráfico da Figura 3 percebe-se que, da despesa total da secretaria, a grande maioria é despesa de pessoal, o que, pelo tipo de atividade desempenhada, não é algo que cause estranheza. Porém, um crescente aumento no número de encarceramentos, em decorrência de diversos fatores, leva a necessidade de majoração em todos grupos de despesa, inclusive pessoal, conforme já evidenciado. Tendo em vista a característica engessada e vinculada da despesa de pessoal, existe uma parcela que é discricionária e pode ser alvo de tentativa de otimização, de forma a reduzir o custo atual, além de viabilizar um melhor planejamento da necessidade futura de pessoal.

Neste recorte temporal, as horas extras, conforme Figura 4, que na SEAPEN representam 9% do gasto de pessoal, ou sessenta milhões de reais no ano de 2020, poderiam sofrer uma otimização em sua utilização. Além disso, existe a possibilidade de melhor alocar os servidores de forma a atender a demanda suprimida que, mesmo não reduzindo o uso de recursos financeiros, vai melhorar o nível

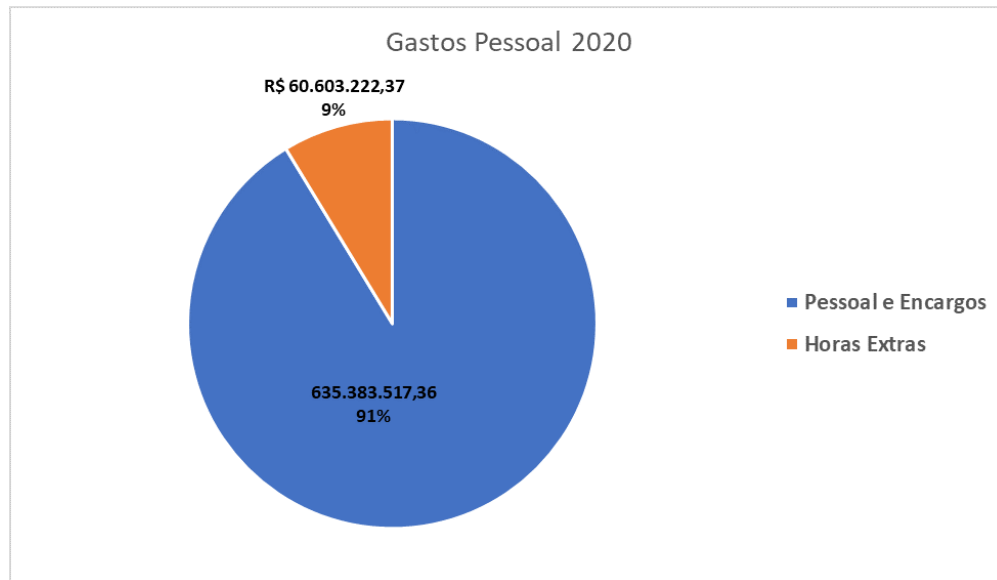


Figura 4: Despesa de Pessoal SEAPEN 2020

Fonte: <http://www.transparencia.rs.gov.br/>

dos serviços prestados, possibilitando uma menor necessidade de novas contratações para o mesmo nível de atendimento.

Nesse sentido, a Pesquisa Operacional (PO) visa instrumentalizar, por meio de modelos matemáticos e algoritmos, o Gestor para uma melhor tomada de decisão. Com isso, a intenção deste trabalho é propor um modelo que ajude a identificar a melhor decisão, a ser tomada em nível de alocação de pessoal, para resolver a problemática exposta.

2 OBJETIVOS

2.1 Objetivo Geral

Considerando a contextualização apresentada e o viés da problemática, que foi apontado como possível ponto de atenção e otimização, esta dissertação tem por objetivo propor uma modelagem para alocar os Policias Penais do Estado em escalas de serviço no âmbito de seus estabelecimentos de lotação. A solução será proposta a um problema real, que permita sua utilização pelo gestor de forma a melhorar a tomada de decisão, quanto a alocação de pessoal, reduzir custos financeiros e de falta de atendimento da demanda prevista. Importante delimitar o escopo aqui pretendido, uma vez que se produzirá uma escala para um mês, considerando as escalas de férias e licenças em matriz de dados referente a disponibilidade. O modelo não considera as escalas anteriores para sua construção e não fará projeções futuras.

2.2 Objetivos Específicos

A partir desse objetivo geral propõem-se os seguintes objetivos específicos:

- 1** Construir um modelo de designação para os Policias Penais que vise reduzir custo;
- 2** Construir um modelo secundário de designação para os Policias Penais que vise aumentar a satisfação;
- 3** Implementar o modelo na linguagem Python;
- 4** Analisar as implicações gerencias do modelo para a tomada de decisão;
- 5** Validar o modelo construído por meio de método comparativo.

3 REFERENCIAL TEÓRICO

Conforme Andrade (2002), a Pesquisa Operacional com um enfoque mais clássico, é definida como a arte de aplicar técnicas de modelagem a problemas de decisão, por meio de métodos matemáticos e estatísticos, buscando encontrar a solução ótima de maneira sistêmica. Já dentro de um enfoque atual, a PO leva em consideração as interações com o ambiente interno e externo para a formulação da modelagem de um problema. Para Duckworth (2012), existem diversas definições válidas para PO, sendo que o autor prefere considerar como o estudo dos sistemas administrativos com a mesma metodologia científica que é utilizada na Física, Química ou Biologia.

Fazer ciência é a capacidade de observar e descrever fenômenos naturais, sociais, econômicos, sendo a matemática de fundamental importância para exprimir parte da realidade na forma de modelos do problema proposto, consideradas as simplificações (Yanasse et al., 2007). Já para de Oliveira Souza et al. (2009), a modelagem de um problema envolve três aspectos:

- A definição das decisões a serem tomadas;
- Restrições que limitam as escolhas das decisões;
- O objetivo que determina preferências na escolha de decisões.

Existem diversos tipos de problemas que podem ser abstraídos da realidade para uma resolução por meio da PO. Desses, muitos já foram agrupados em determinados tipos e largamente estudados, por sua relevância, como na área da saúde, educação, segurança, indústria, etc. Um desses problemas, que pode ser encontrado em todas as áreas citadas, é o Problema de Escalonamento de Pessoal. Esse é um problema específico e teve um significativo avanço desde sua primeira propositura por Edie (1954). A definição de escala de trabalho pode ser feita como o processo de construção de horários de trabalho para os funcionários de uma organização, para que possam satisfazer a demanda por seus produtos ou serviços, conforme Ernst et al. (2004).

Inicialmente, cabe referir que a maior parte dos problemas referem-se a "staff schedule", "personel scheduling", "roster" ou "rostering". Na maior parte das vezes os termos são utilizados como sinônimos para escalonamento de pessoal. Pode-se verificar, com alguma pesquisa, que existe alguma diferença conforme o país, por exemplo, na Inglaterra, o termo "schedule" nunca é aplicado a humanos, apenas a cronogramas e maquinários, conforme o sítio especializado em consultoria VisualRota. No mesmo sentido é feita a revisão de diversos trabalhos acadêmicos sobre o assunto, nos quais utiliza-se como sinônimo para pesquisa as referidas palavras, como em "Staff scheduling and rostering: A review of applications, methods and models" (Ernst et al., 2004).

Além disso, quanto à complexidade computacional, o problema é normalmente classificado como NP-Difícil. Porém, existem poucos estudos que avaliam cada uma das formulações propostas quanto a esse aspecto. De qualquer forma, mesmo com poucos pesquisadores provando formalmente a complexidade, no trabalho de Brunner et al. (2013) prova-se a complexidade para o problema proposto com suas restrições específicas, bem como também é provado por Osogami and Imai (2000) em seu trabalho sobre PEPs com restrições específicas para um problema de designação de

enfermeiros.

3.1 O Problema de Escalonamento de Pessoal e suas etapas

O Escalonamento de Pessoal é uma tarefa composta por várias etapas, que podem variar a depender do autor do problema, conforme pode ser visto nos capítulos próprios de Ernst et al. (2004), Shiao et al. (2020), Mattia (2017). Porém, algumas etapas de escalonamento são básicas e, segundo Thompson (2003), são elas:

1. **Previsão de demanda:** consiste em definir a demanda por serviço em determinado turno de trabalho (considerando esta a menor partição). Pode refletir em clientes, movimentações, atendimentos, etc. Esta etapa vincula as demais e é primordial para o melhor resultado das posteriores;
2. **Cálculo da relação funcionário/hora para satisfazer a demanda:** etapa que traduz a demanda em número de funcionários/horas para realizar as demandas previstas. Essa etapa requer estudos prévios para determinar a correta relação entre a tarefa a ser executada e os funcionários/horas a executarem;
3. **Construção da escala de serviço:** construir a escala de serviço considerando as preferências pessoais e os objetivos corporativos, esses definidos nas restrições na própria função objetivo;
4. **Ajuste do Escalonamento dependendo da demanda:** etapa de acompanhamento e não planejamento. Consiste em verificar se a demanda está sendo atendida e ajustar a escala. Etapa em tempo real que deve ser executada com cuidado para evitar mudança no escalonamento.

O processo que envolve a determinação do número mínimo de funcionários, com habilidades específicas necessárias para atender as demandas, também deve observar todas as regulamentações trabalhistas e acordos laborais relevantes. Existe um grande número de aplicações da programação de escalas de trabalho como, por exemplo, companhias de transporte, sistemas de saúde, serviços de emergência e segurança, call centers e muitas outras.

Já no relevante trabalho de revisão da literatura sobre PEP, Ernst et al. (2004) classificaram a construção do modelo nos seguintes módulos:

1. **Modelagem de demanda:** nesse módulo busca-se traduzir algum padrão do trabalho de forma a definir uma demanda por pessoal. Dentro desse módulo classificaram-se esses padrões em três grandes áreas:
 - (a) demanda baseada em tarefas;
 - (b) demanda baseada em previsão; e
 - (c) demanda baseada em turnos.
2. **Agendamento de dias de folga:** envolve a determinação de como os dias de descanso devem ser intercalados entre os dias de trabalho;
3. **Escalonamento de Turnos:** busca atender a demanda de pessoal para cada turno, conside-

rando, ou não, os intervalos internos;

4. **Construção em linhas de trabalho:** escala construída por blocos que agrupam, turnos, tipos de trabalho, etc. Essa construção considera um horizonte temporal e busca garantir o atendimento dentro de cada bloco e, também, aquele determinado em um horizonte;
5. **Designação de Tarefa:** utilizada para designar tarefas a cada tipo de trabalhador ou dentro de cada turno;
6. **Designação de Pessoal:** utilizada para pessoas dentro das linhas de trabalho, geralmente.

Essa proposta busca listar as principais etapas encontradas na literatura revisada, porém, obviamente, a utilização de cada um vai depender da especificação do modelo que se busca construir. Isso difere do proposto por Thompson (2003) no quesito generalização.

Além das etapas descritas, a escala pode ser cíclica, quando o padrão de distribuição se repete no intervalo de tempo, ou não-cíclica, quando não existe um padrão de distribuição dos turnos no tempo. Segundo Devesse, 2016, a desvantagem da escala não-cíclica é sua não homogeneidade em relação a distribuição dos turnos. No presente estudo pretende-se adotar o modelo não-cíclico, uma vez que o problema apresentado necessita de maior liberdade para alocar os servidores nos dias e turnos, conforme a demanda e demais restrições.

3.2 Principais Aplicações do PEP

Em que pese sua clara aplicabilidade prática e os vastos estudos na área, a maior parte das instituições não utiliza a otimização de escalas como uma prática cotidiana e comercial. Mesmo com significativas publicações e métodos de solução que geram resultados excelentes, em tempo computacional aceitável, as instituições seguem organizando suas listas manualmente. Essa prática gera escalas de baixa qualidade, além de implicar em força de trabalho, muitas vezes especializada, alocada para função de escalante, aponta Wickert (2019).

Barda et al. (2003) asseguram que a programação de escalas de trabalho eficaz tornou-se um dos principais meios pelos quais as organizações de serviços podem permanecer competitivas. Escalas de trabalho mal elaboradas podem levar a um excesso de oferta de trabalhadores com muito tempo ocioso, ou uma suboferta com uma consequente perda de negócios. Burke et al. (2004) afirmam que a programação automática de escalas de trabalho não é muito comum e a importância de uma abordagem sistemática para criar boas escalas é muito elevada.

Na revisão das principais bibliografias sobre o assunto, feita por Ernst et al. (2004), foram identificadas as mais relevantes áreas de atuação e seus métodos, como seguem.

- **Sistemas de Transporte** em massa são áreas normais de estudo e aplicação da PO na busca por otimização de suas escalas de trabalho, sendo que os principais segmentos estudados são as empresas aéreas, de trem, ônibus e transportes públicos. Por sua economia de escala, existem diversos estudos nesta área, aprofundando e propondo novas metodologias a aplicações que possam abarcar as diferentes regulamentações. No estudo de revisão bibliográfica citado,

foram mais de 45 estudos revisados e alocados nesta categoria. Destes, a maioria se ateu ao escalonamento de tripulação, como em Wedelin (1994), Ryan (1992), Gamache and Soumis (1998) e Desrosiers et al. (2000).

A abordagem mais popular para resolver programação e escala da tripulação é a técnica de decomposição, dividindo o problema geral em três etapas principais:

- a emparelhamento dos meios de transporte: cria tarefas frente ao calendário fornecido, seria o itinerário;
- b otimização de emparelhamento: otimiza os emparelhamentos criados de forma a cobrir todas tarefas;
- c escala da tripulação: as tripulações são designadas no emparelhamento criado na etapa anterior.

Segundo Ahuja et al. 1993, conforme citado por Silva et al. (2005), um emparelhamento é um conjunto de pares de vértices de um grafo com pesos não nulos, e tais que cada vértice deve formar, no máximo, um par. Deste modo, um emparelhamento de um grafo $G=(X, A)$ é um subconjunto M do conjunto A de arcos de G , escolhidos de tal maneira que cada vértice é a extremidade de, no máximo, um arco de M , ou seja, não existem arcos adjacentes em M . A técnica do emparelhamento é muito usada nos Problemas de Programação de Tripulações (PPT).

Apesar dos inúmeros estudos, como já dito anteriormente, dificilmente encontramos um modelo que possa ser utilizado de forma genérica, mesmo sendo de uma área específica, pois os objetivos e as restrições acabam sendo muito variadas em cada um deles.

- **Centrais telefônicas** foram largamente estudadas e, apesar de uma resolução aparentemente mais simplificada do que o PPT, trazem dificuldades diferentes do problema anterior. A falta de previsibilidade, ou a necessidade de prever a demanda de força de trabalho, como abordado nos trabalhos de Brigandi et al. (1994), Buffa et al. (1976) e Henderson et al. (1999). Além disso, a falta de sazonalidade torna o problema difícil e ao mesmo tempo bem particular. Como os picos de demanda em um turno variam e a necessidade de pessoal também, acaba ocorrendo um ajuste bem complexo quanto à distribuição de pessoal respeitando as regras trabalhistas e contratuais de forma a buscar uma minimização de custos consistente. Para isso, os estudos de Andrews and Parsons (1993) e Thompson (1997) propuseram métodos matemáticos e heurísticas para solucionar o problema.

Esse problema geralmente é resolvido com um algoritmo de designação, sendo esse um caso especial do transporte, onde cada produto (trabalhador) deve ser levado ao seu destino (tarefa) no menor custo possível. Um dos algoritmos possíveis para a solução é o conhecido método Húngaro. Geralmente utiliza-se uma matriz balanceada entre tarefas e trabalhadores, de forma que toda tarefa seja executada. Variáveis fictícias podem ser criadas para cobrir aquelas tarefas que não teriam trabalhadores de forma a viabilizar a resolução.

- **Sistemas de Saúde** são também muito explorados, principalmente na questão da formulação de escalas de enfermagem. Esses profissionais representam a maior parte da mão de obra do sistema de saúde e por isso a gestão eficiente de sua escala é de suma importância para controlar os altos custos dessas instituições. Além disso, segundo Devesse (2016), a maior parte dos estudos nessa área é multiobjetivo, pois busca reduzir custos e, ao mesmo tempo, aumentar a satisfação dos funcionários com a escala de serviços.

Diferentemente da simples designação, aqui uma restrição importante é a habilidade particular de cada profissional, elevando o número de restrições a serem aplicadas. Nesse aspecto, os trabalhos de Maier-Rothe and Wolfe (1973), Norby et al. (1977) e Ryan et al. (1975) propõem modelos para solucionar isso.

Foram propostos diversos algoritmos para se chegar a soluções mais eficientes, dentre eles estão a Busca Tabu, Algoritmo Genético, Busca local e algumas formas híbridas também. Algumas dessas proposições podem ser encontradas nos trabalhos de Warner (1976), Miller et al. (1976) e Arthur and Ravindran (1981).

Ernst et al. (2004) citam que a grande vantagem desses algoritmos propostos pelos autores, foram as habilidades de superar problemas básicos de heurísticas associadas a esse tipo de escalonamento complexo, garantindo uma qualidade de solução desvinculada a combinação de soluções parciais.

- **Serviços de proteção e emergência** como polícia, ambulância, bombeiros e segurança em geral tem especificidades a serem atendidas, devido as regulações que envolvem esses serviços, bem como a necessidade de atender demandas que podem variar dependendo de alguns fatores. Esses fatores podem ser em termos de tempo de resposta para comparecer a incidentes, o envio de determinado especialista, eventos como férias, feriados e finais de semana, dentre outros.

Assim como na área da saúde, as diferentes especialidades dos sistemas de emergências e segurança devem ser levadas em consideração. Além disso, turnos variados e diferentes restrições são aplicadas, tornando os modelos únicos. São utilizadas diversas heurísticas para solucionar os problemas que se originam, principalmente levando em conta as especificidades de turnos a serem cumpridos. Os trabalhos de Butler and Maydell (1979) e Taylor and Huxley (1989) buscam minimizar o desvio de disponibilidade de pessoal, quando houver variação da demanda, e reduzir o número de turnos descobertos, respectivamente.

Além das áreas especificadas anteriormente, o trabalho de Ernst et al. (2004) descreveu as aplicações em Centrais de Atendimento; Organizações de Serviços Privados (hotéis, restaurantes, lojas de varejo, etc.); Organizações de Serviços Públicos; Gestão Local em Serviço de Solo (Cassinos, aeroportos, terminais de Carga); Serviços Financeiros, como auditoria; Turismo e Hospitalidade; Comércio Varejista; Manufatura, tanto em escalonamento de pessoal quanto de material. Em cada um desses tópicos são apresentados os principais artigos e descritos as principais formas de resolução.

Nesse trabalho, tendo em vista ser uma área específica da segurança, nota-se uma abordagem híbrida devido a características de serviços de segurança com restrições de servidores governamentais. Não foi encontrado um caso que apresente esse tipo de estrutura dentre as fontes pesquisadas, porém a composição do modelo pode ser comparado a partes das estruturas estudadas.

3.3 Resolução do PEP por meio da Meta-heurística GRASP

Levando em consideração a divisão dos métodos de resolução entre os exatos e os de aproximação, sendo os primeiros aqueles que garantem a otimalidade do problema e os seguintes são aqueles que, mesmo não garantido o ótimo global, encontra soluções factíveis em tempo computacional razoável. Dentre os métodos aproximados, as meta-heurísticas destacam-se pela larga utilização, sendo métodos de solução que ajustam procedimentos de melhoria local e estratégias capazes de escapar de ótimos locais e realizar uma busca robusta em um espaço de solução, conforme Gendreau et al. (2010).

Segundo Burke et al. (2004), podemos diferenciar os termos heurística e Meta-heurística considerando a primeira como a que oferece orientação para encontrar boas soluções de acordo com o conhecimento do domínio e a segunda como metodologias de aprimoramento que coordenam a busca local para realizá-la de forma robusta em um espaço de solução, visando escapar do ótimo local. Conforme o mesmo autor, temos dois tipos principais de heurísticas: - as perturbativas ou de busca local, que operam em candidatos totalmente instanciados nas soluções; e - heurísticas construtivas, que expandem iterativamente soluções candidatas parciais.

Algumas das meta-heurísticas mais utilizadas, segundo Resende and Ribeiro (2016), são algoritmos genéticos (AG), simulated annealing (SA), busca tabu (BT), busca de vizinhança variável (VNS) e procedimentos de busca adaptativa randomizada gulosa (GRASP). Este último método é uma Meta-heurística de inicialização múltipla, ou seja cada iteração gera um novo ponto de início, que, segundo Gendreau et al. (2010), é usada para problemas de otimização combinatória em que cada iteração consiste basicamente em duas fases: construção e busca local. Até o limite imposto, tempo ou iterações, são construídas soluções viáveis, cuja vizinhança é investigada durante a fase de busca local.

O GRASP apresenta uma formulação simples, contendo o algoritmo principal e as duas heurísticas bases da formulação, como poderemos verificar nos pseudo-algoritmos 3 e 4 na subseção 5.2, que juntamente com a Busca Local escolhida, formam a base da solução. O algoritmo 4 é o que dá nome a Meta-heurística, uma vez que uma lista de candidatos para variar, por meio de parâmetros, entre a solução mais gulosa e a mais randômica. A escolha do GRASP, juntamente com os métodos exatos, passa pela característica simples de formulação, bem como a possibilidade de combinar outras características de meta-heurísticas, que, segundo Raidl et al. (2019), têm seus prós e contras individuais, mas explora as vantagens das técnicas de forma a obter um sistema híbrido mais eficaz, beneficiando a sinergia.

4 METODOLOGIA

No presente trabalho utilizaram-se técnicas heurísticas já largamente estudadas para resolução dos problemas de designação de escalas, de forma a propor uma solução factível para a formulação de escalas de trabalho de policiais em estabelecimentos penais do Estado do Rio Grande do Sul. Para isso, foram utilizados os dados referentes as escalas realizadas no ano de 2020 em determinados estabelecimentos de uma das regiões e, a partir disso, foi proposto um algoritmo de otimização que leva em consideração as restrições legais, bem como procura aumentar a satisfação com a escala de trabalho por meio de uma abordagem experimental de dupla otimização. Os algoritmos propostos utilizaram métodos exatos e, alternativamente, de aproximação.

O trabalho foi desenvolvido em linguagem Python, utilizando suas principais bibliotecas para ajustar os dados de entrada e saída. Para a formulação exata, foi utilizado o PuLP, que é uma biblioteca para a linguagem de script Python que permite aos usuários descrever programas matemáticos. Python é uma linguagem bem estabelecida e tem ênfase no desenvolvimento rápido e claro de código e sintaxe, conforme Mitchell et al. (2011).

Além disso, um dos solvers utilizados é o desenvolvido pelo projeto *Computational Infrastructure for Operations Research* (COIN) que, conforme explicado pelos pesquisadores Forrest and Lougee-Heimer em sítio próprio, pode ser acessado através da biblioteca PuLP. A referida biblioteca escrita em Python que pode gerar arquivos MPS ou LP e utilizar os seguintes solvers: GLPK, COIN-OR CLP / CBC, CPLEX, GUROBI, MOSEK, XPRESS, CHOCO, MIPCL, SCIP para resolver problemas lineares, conforme Roy and Mitchell. Qualquer solver de programação linear com uma interface do tipo *Open Solver Interface* (OSI) pode ser usado com o COIN-OR. Este ambiente utiliza o algoritmo branch-and-cut, além de outras variações e heurísticas, através de uma OSI. Além desse solver de código aberto, também foi testado o modelo no solver comercial CPLEX, com licença para estudante.

4.1 Formulação Específica

Considerando as etapas de formulação do PEP descritas na literatura apresentada, no presente trabalho elas serão compiladas, por simplificação, e denominadas Definição da Demanda e Designação de Pessoal.

4.1.1 Definição da Demanda

Esta seção descreve a metodologia utilizada para definir a demanda, conforme estudo proposto pelo quadro técnico da SEAPEN. Neste caso estamos considerando as etapas 1 e 2 citadas por Thompson (2003) e as etapas 1, 2 e 3 do proposto por Ernst et al. (2004).

A Assessoria Técnica (ASTECC) da SEAPEN desenvolveu trabalho de forma a definir a demanda de cada estabelecimento para subsidiar a distribuição de policiais penais que estavam em curso de formação no ano de 2021. O estudo se baseou metodologicamente no ciclo da Figura 5.



Figura 5: Metodologia de definição de demanda

Fonte: ASTEC-SEAPEN

A Figura 5 resume a forma de construção do modelo que define a distribuição dos policiais e, como um de seus subprodutos, a demanda de cada estabelecimento. As etapas podem ser descritas da seguinte maneira:

- **Planejamento, Definição de Problema e Premissas:** Nessa etapa foram definidas as características estruturais e funcionais de cada estabelecimento, sendo feita uma classificação que levou em conta o Regime (Fechado, semiaberto e aberto), a Estrutura (módulos, galerias e celas) e Funcionários (quantidade, cargo, classe). Além disso, foram averiguadas as necessidades de cada local considerando o número de presos e necessidade de movimentações externa e internas;
- **Aquisição de dados, Análise de dados e Desenvolvimento de Modelo:** etapa que reuniu e padronizou os dados definidos na etapa anterior, considerando diversas fontes internas;
- **Validação e Avaliação dos dados:** nesse ponto foram validados os dados e feita a avaliação dos cenários possíveis para construção do modelo de distribuição;
- **Correção e Implementação:** finalmente os dados foram apresentados, conforme os modelos formatados, à alta administração e demais técnicos para validação e eventuais correções.

Essa classificação chegou aos seguintes estabelecimentos a serem estudados:

- ★ Penitenciárias;
- ★ Presídios Estaduais;
- ★ Presídios regionais;

- ★ Institutos Penais;
- ★ Institutos Penais de Monitoramento Eletrônico;
- ★ Hospitais Penais;

Para cada tipo de estabelecimento foi utilizada uma fórmula que levou em consideração diversos fatores para calcular a necessidade diária de policiais. As fórmulas abaixo exemplificam os cálculos da demanda mensal.

$$Diurno = ((3 + 6 \times V)/7 + 3 \times M + (3/2) \times G + 3 \times A + (5/7) \times Mov + E) \times 31 \quad (1)$$

$$Noturno = (2 + 2 \times M + (3/2) \times G + 3 \times A + E) \times 31 \quad (2)$$

Sendo,

V= Dias de Visitas

M= Número de Módulos

G= Número de Galerias

A= Número de Anexos

Mov= Número de Movimentações

E= Especificidades

Com essa metodologia foi definida a necessidade de cada tipo de estabelecimento, consideradas ainda as peculiaridades de cada um deles. Assim, tendo sido definida a metodologia para definição da demanda, foi aplicada a cada um dos estabelecimentos penais do estado, criando a matriz de demanda de pessoal. Finalizando, desta maneira, uma parte da construção das etapas de escalas de trabalho.

4.1.2 Designação de Pessoal

Para completarmos as etapas de construção de uma escala de serviço otimizada, precisamos finalizar os passos 3 e 4 de Thompson (2003), bem como os passos 4, 5 e 6 de Ernst et al. (2004). Aqui definiremos a forma de escolha dos estabelecimentos a serem estudados e a forma atual de construção da escala de serviço.

Atualmente a SUSEPE, subordinada da SAEPEN, divide o Estado em dez Delegacias Penitenciárias Regionais (DPR), que tem sob sua administração os estabelecimentos que se encontram naquelas delimitações. São elas:

- 1ª. DPR - Vale dos Sinos e Litoral (sede em Canoas)
- 2ª. DPR - Região Central (sede em Santa Maria)
- 3ª. DPR - Missões e Noroeste (sede em Santo Ângelo)
- 4ª. DPR - Alto Uruguai (sede em Passo Fundo)

- 5ª. DPR - Sul (sede em Pelotas)
- 6ª. DPR - Campanha (sede em Santana do Livramento)
- 7ª. DPR - Serra (sede em Caxias do Sul)
- 8ª. DPR - Vale do Rio Pardo (sede em Santa Cruz do Sul)
- 9ª. DPR - Carbonífera (sede em Charqueadas)
- 10ª. DPR - Porto Alegre (sede no IPF)

Fonte: http://www.susepe.rs.gov.br/conteudo.php?cod_menu=7

De forma a comparar e avaliar a escala proposta foi escolhida uma dessas DPRs, levando em conta na escolha a disponibilidade dos dados, facilidade de coleta e comparação das informações.

Atualmente cada estabelecimento, baseado na avaliação da Direção, constrói a escala de serviço de forma manual e indica para seu Delegado Regional a necessidade de complementação, ou não, de hora de trabalho. O Delegado Regional por sua vez avalia todas as solicitações da Região e, dessa maneira, redistribui, solicita reposição de servidores, ou o equivalente em autorização de Hora Extraordinária, para o Órgão de Administração Central, qual seja a SEAPEN, conforme demonstrado na Figura 6.

Com isso temos um processo manual de distribuição e alocação de pessoal em todos os níveis organizacionais. O que limita a administração no atendimento das necessidades e envolve muitas pessoas em processos complexos e, muitas vezes, falhos.

A proposta desse trabalho é atuar nessa fase do processo, propondo um algoritmo que faça a alocação de pessoal de forma otimizada e informatizada. Com isso seria possível realocar diversos servidores que atuam em cada estabelecimento e nos demais níveis gerenciais, envolvidos na elaboração de escalas de trabalho manuais, bem como melhorar a informação para a tomada de decisão do gestor no que tange a melhor distribuição de horas extras ou até mesmo uma eventual realocação de pessoal. De forma adicional, propõem-se criar um algoritmo que aumente a satisfação dos servidores com a escala de trabalho proposta, tentando aproximar os anseios pessoais das necessidades da administração pública.

Atualmente as escalas de trabalho são formuladas de forma a atender a demanda estipulada na etapa anterior e a legislação vigente quanto a carga horária máxima e distribuição de plantões. A carga horária mensal é definida por lei e atualmente é de 160 horas para cada policial (LEI COMPLEMENTAR Nº 13.259). Utilizando uma resolução interna (nº 9/2009), instrumento infralegal, foi definida que a escala será cumprida em plantões de 24 horas de duração com, no mínimo, 72 horas de descanso entre cada turno de 24 h de duração. Além disso, são estabelecidos reforços em horário de expediente para completar a carga horária.

Os dados, de forma geral, são armazenados em planilhas, não existindo um sistema de controle de ponto. A fiel descrição das informações depende da organização de cada Delegacia que, após disponibilizar a escala aos servidores, mantém cópia em formato pdf, assinada pelo Diretor do

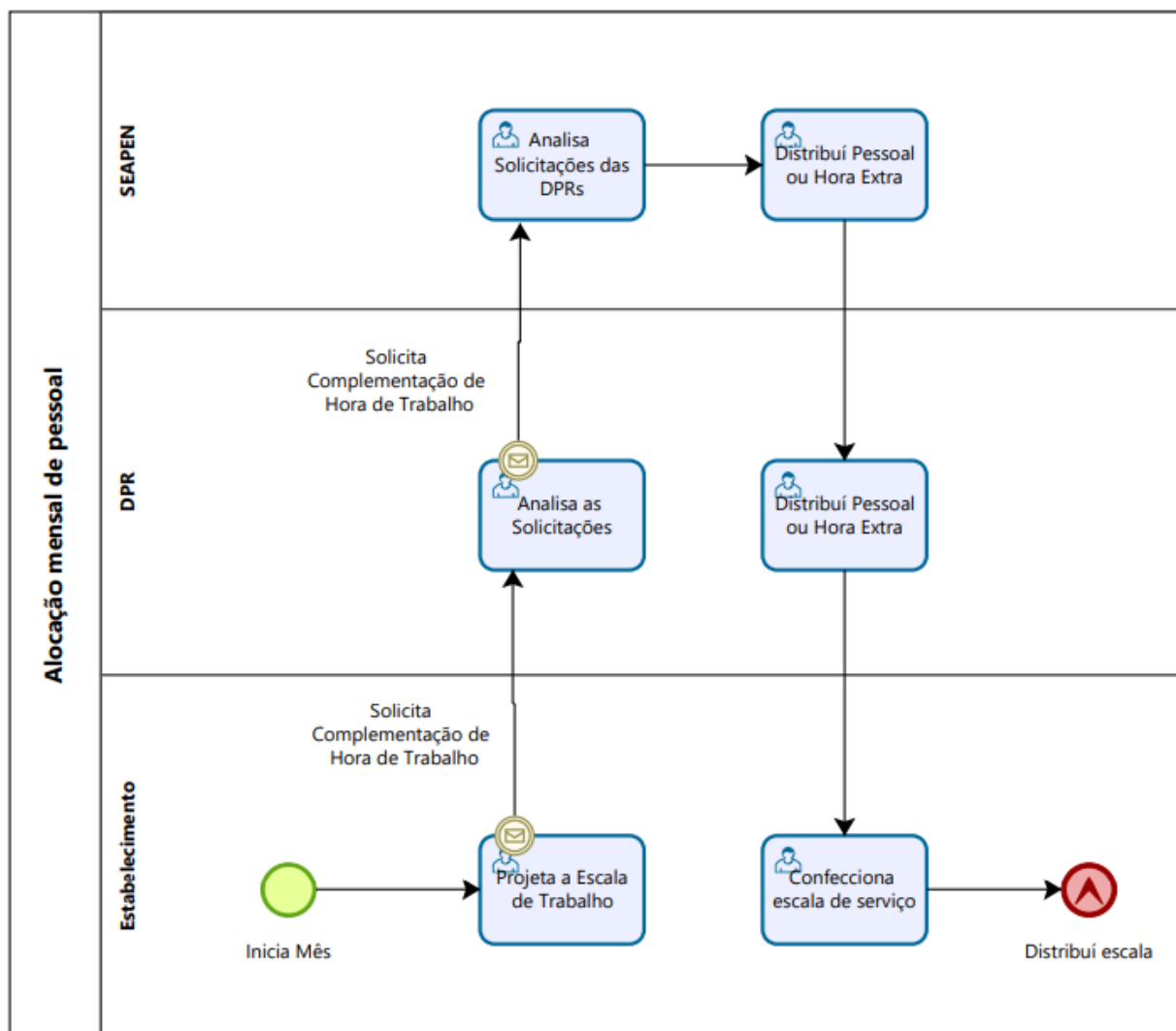


Figura 6: Fluxo Atual

Fonte: Confeção Própria

Estabelecimento. Dos dados recebidos, foram selecionados os que se encontravam mais organizados e com informações mais precisas para o estudo.

4.2 Modelo

De forma a otimizar esse problema foi proposto um modelo experimental com dupla otimização, considerando os dados a seguir. Para permitir uma comparação com as escalas realizadas, dos escolhido o mês de outubro de 2020 para definição das disponibilidades e quantidades de servidores

em cada estabelecimento.

- Dados os seguintes índices:
 - i é o índice que representa o policial designado;
 - j é o índice que representa o turno designado;
 - k é o índice que representa o dia do mês designado;
- Dados os seguintes conjuntos:
 - P é o conjunto de Policiais do estabelecimento estudado;
 - T é o conjunto de turnos previstos na escala;
 - M é o conjunto de dias no mês de referência;
- Dadas as seguintes variáveis Principais:
 - X_{ijk} é variável que indica se o policial i trabalha no turno j do dia k ;
 - S_{jk} é variável que indica se o turno j do dia k está descoberto;
- Dadas as seguintes variáveis Auxiliares:
 - $HnUi$ é variável que indica a quantidade de horas normais utilizadas pelo policial i ;
 - $HeUi$ é variável que indica a quantidade de horas extras utilizadas pelo policial i ;
- Dados os seguintes parâmetros:
 - H_j é a quantidade de horas atrelada ao turno j ;
 - CN_i é o custo por hora normal trabalhada atrelada ao servidor i ;
 - CE_i é o custo por hora extra trabalhada atrelada ao servidor i ;
 - Pe é a penalidade imposta a não designação de servidores para atender a demanda;
 - PP_{ij} é a preferência do servidor i trabalhar no turno j ;
 - D_{jk} é a demanda de policiais no turno j do dia k ;
 - $Disp_{ik}$ é a disponibilidade do policial i no dia k ;
 - $CNMAX_i$ é a carga horária máxima normal mensal do policial i ;
 - $CEMAX_i$ é a carga horária máxima extra mensal do policial i ;
 - HE_i é a opção por receber hora extra mensal do policial i ;
 - SP_i é a habilitação do policial i em exercer a Supervisão;

Além disso, as seguintes variáveis auxiliares, conjuntos e parâmetros foram utilizados na modelagem:

*Variáveis Auxiliares:

- $HnUi = \sum_{i \in P} \sum_{j \in T} \sum_{k \in M} X_{ijk} \times H_j$ sendo $0 \leq HnUi \leq CNMAX_i$;
- $HeUi = \sum_{i \in P} \sum_{j \in T} \sum_{k \in M} X_{ijk} \times H_j$ sendo $CNMAX_i < HeUi \leq CEMAX_i$.

*Conjuntos:

- P é formado pelo seguinte conjunto: (07, 12, 14, 15, 33, 59, 130, 147). Onde cada número representa a quantidade de policias no estabelecimento, definindo assim a instância do mesmo. Por exemplo E_7 é o estabelecimento com instância 7, sendo esses os servidores disponíveis para formar a escala;
- T é composto por (EXP, P, SP), sendo EXP o período de expediente composto por 8 horas de trabalho, P é o plantão de 24 horas e SP é a Supervisão de 24 horas;
- M é o conjunto de 31 dias, considerando que o mês de referência é outubro de 2020;

*Parâmetros:

- H_j é a quantidade de horas atrelada ao turno j tendo a tabela 2 como base:

Tabela 2: Turnos x Horas

Turno	Duração (Hs)
Supervisor (SP)	24
Plantonista (P)	24
Expediente (EXP)	8

Fonte: confecção própria

- CN_i e CE_i são os custos por hora trabalhada atrelada a cada servidor i , conforme sua posição na carreira, resumidos na tabela 3:

Tabela 3: Custo de hora Trabalhada

Classe do Policial	Custo Normal (CN)	Custo Extra (CE)
A	34,38	51,56
B	50,00	75,00
C	62,50	93,75
D	75,00	112,50
E	93,75	140,63

Fonte: confecção própria

- Pe é a penalidade imposta a não designação de servidores para atender a demanda dada, esta foi definida como 1000 unidades;
- PP_{ij} é a preferência do servidor i trabalhar no turno j e foi construída com base em pesquisa de preferência realizada com uma amostra de servidores. Com base nesta pesquisa foi construída

a tabela 4, onde 0 significa sem interesse e 1 significa com interesse, sendo um exemplo de disposição desse tipo de matriz. Para definir a preferência dos servidores e assim construir

Tabela 4: Preferências Pessoais

Policial	SP	P	EXP
1	1	0	1
2	0	0	1
3	0	1	1
4	1	0	0
5	1	1	1
...
n	1	1	1

Fonte: confecção própria

a matriz, foi utilizada um pesquisa de preferência dos servidores quanto a cada turno e realização de horas extraordinárias.

Tal pesquisa foi feita de forma virtual, por meio da ferramenta formulários da Google. As perguntas realizadas foram as seguintes:

- Considerando a necessidade de disponibilização de serviços durante as 24 horas do dia, qual arranjo da sua preferência?
- Considerando a carga horária mensal de 160 hs , qual arranjo da sua preferência?
- Considerando o serviço em três turnos, qual período de sua preferência?
- Você aceita trabalhar além da carga normal em troca de remuneração por horas extras?

De posse do resultado da pesquisa, foi calculada a proporção de servidores que prefere cada uma das situações. Essa proporção foi aplicada a cada uma das escalas reais, criando assim a matriz de preferência representativa, quanto ao pesquisado.

- D_{jk} é a demanda de policiais no turno j do dia k , sendo definida por uma matriz que correlaciona os dias com os turnos de trabalho. A definição de demanda foi explicitada no capítulo 3.1.1, seguindo a tabela 5 como exemplo da construção:
- $Disp_{jk}$ é a disponibilidade do policial i no dia k , sendo que tal matriz representa situações que podem inviabilizar que o policial concorra à escala como férias, dispensas e atestados. Essa matriz utilizou a lógica de 1 para disponível e -1 para indisponível. A tabela 6 busca exemplificar um caso com n policiais:
- $CNMAX_i$ é a carga horária máxima normal mensal do policial i , sendo definida legalmente em 160 horas, aos servidores disponíveis por 15 dias ou mais, e em 80 horas mensais aos servidores cuja disponibilidade é menor que 15 dias;

Tabela 5: Demanda por dia e Turno

Dia do Mês	SP	P	EXP
1	1	3	1
2	1	3	1
3	1	1	1
4	1	1	1
5	1	3	1
...
n	1	3	1

Fonte: confecção própria

Tabela 6: Disponibilidade

Policia	Dias do Mês de Outubro					
	1	2	3	4	...	31
1	1	1	1	1	...	1
2	1	1	1	1	...	1
3	-1	-1	-1	-1	...	1
4	-1	-1	-1	-1	...	1
...
n	-1	-1	-1	-1	...	-1

Fonte: confecção própria

- $CEMAX_i$ é a carga horária máxima extra mensal do policial i , sendo definida legalmente em 40 horas, aos servidores disponíveis por 15 dias ou mais, e em 24 horas mensais aos servidores cuja disponibilidade é menor que 15 dias;;
- HE_i é a opção por receber hora extra mensal do policial i , sendo construída com a lógica de 1 para disponível e -1 para indisponível, sendo exemplificado na tabela 7.

A construção de cada tabela levou em consideração a pesquisa realizada sobre as preferências dos servidores.

- SP_i é a habilitação como Supervisor do policial i , sendo construída com a lógica de 1 para habilitado e 0 para não habilitado, sendo exemplificado na tabela 8.

Tabela 7: Opção por Hora extra (HE)

Policial	HE
1	1
2	1
3	-1
4	-1
...	...
n	-1

Fonte: confecção própria

Tabela 8: Habilitação como Supervisor (SP)

Policial	SP
1	1
2	1
3	0
4	0
...	...
n	1

Fonte: confecção própria

4.2.1 Função Objetivo 1 - FO1

$$Min_Z1 = \sum_{i \in P} \sum_{j \in T} \sum_{k \in M} CN_i \times HnU_i + CE_i \times HeU_i + S_{jk} \times Pe \times H_j \quad (3)$$

A equação 3 representa a primeira função objetivo e tem como diretriz a minimização, buscando reduzir os custos das designações de cada policial e sendo atrelada as diversas restrições legais e de equalização da escala. Além da meta de redução do custo com Hora Extraordinária ela pode identificar locais com excesso de pessoal alocado ou a falta de pessoal. Isso pode ser visto, principalmente, pela forte penalização da não alocação de pessoal, quando utilizada a variável "S".

4.2.2 Função Objetivo 2 - FO2

$$Max_Z2 = \sum_{i \in P} \sum_{j \in T} \sum_{k \in M} X_{ijk} \times PP_{ij} \quad (4)$$

A equação 4 representa a segunda Função Objetivo e terá a meta de aumentar a satisfação dos funcionários com a escala proposta, uma vez que ela tentará maximizar a alocação feita com as preferências pessoais de cada policial. O principal objetivo é reduzir o custo, sendo esse um objetivo

secundário, porém desejável.

4.2.3 Restrições

Aqui apresentaremos as restrições que particularizarão o modelo.

- Garante que toda demanda de policiais do turno seja atendida

$$\left(\sum_{i \in P} X_{ijk}\right) + S_{jk} \geq D_{jk} \quad \forall j, k \quad (5)$$

- Garante que só sejam designados para Supervisão aqueles servidores habilitados

$$SP_i - X_{ijk} \geq 0 \quad \forall i, k \quad j = SP \quad (6)$$

- Garante que o servidor está disponível

$$\sum_{j \in T} (X_{ijk}) \times Disp_{ik} \geq 0 \quad \forall i, k \quad (7)$$

- Garante que o mesmo servidor não trabalhe mais de 1 turno no mesmo dia

$$\sum_{j \in T} X_{ijk} \leq 1 \quad \forall i, k \quad (8)$$

- Garante a distribuição no limite legal de horas do servidor no mês, levando em conta a disponibilidade

Sendo a Disponibilidade, conforme a tabela ex 6, 1 para disponível e -1 par indisponível, quando o servidores está disponível por 15 ou mais dias no mês ele tem sua carga distribuída normalmente. Caso contrário, o mesmo servidor sofre redução de 50% em sua carga horária máxima, conforme equações 9 e 10 .

$$HnU_i = CNMAX_i \quad \forall i \quad se \sum_{j \in T} \sum_{k \in M} Disp_{ik} < 15 \quad (9)$$

$$HnU_i = CNMAX_i - 80 \quad \forall i \quad se \sum_{j \in T} \sum_{k \in M} Disp_{ik} \geq 15 \quad (10)$$

Importante referir que aqui a igualdade se faz necessária, uma vez que, por serem servidores públicos, eles devem trabalhar no mínimo as horas que foram contratados. Tal situação é gerada pelo tipo de contratação, na qual mesmo não sendo atribuído a um turno o servidor cumprirá suas horas normais em alguma atividade e o salário será pago.

- Garante a distribuição no limite legal de horas extras do servidor no mês, levando em conta a

disponibilidade

Sendo a Disponibilidade, conforme a tabela ex 6, 1 para disponível e -1 par indisponível, quando o servidores está disponível por 15 ou mais dias no mês ele tem sua carga distribuída normalmente. Caso contrário, o mesmo servidor sofre redução de 16 hs em sua carga horária extra máxima, conforme equações 11 e 12 .

$$HeU_i \leq CEMAX_i \quad \forall i \quad se \sum_{j \in T} \sum_{k \in M} Disp_{ik} < 15 \quad (11)$$

$$HeU_i \leq CEMAX_i - 16 \quad \forall i \quad se \sum_{j \in T} \sum_{k \in M} Disp_{ik} \geq 15 \quad (12)$$

- Garante a opção por horas extras do servidor no mês

$$HeU_i \times HE_i \geq 0 \quad \forall i \quad (13)$$

- Garante folga após o Plantão e Supervisão de no mínimo 72 horas

$$\sum_{j \in T} (X_{ij(k+1)} + X_{ij(k+2)} + X_{ij(k+3)}) + \sum_{j \in T^*} (X_{ij(k)}) \leq 1 \quad (14)$$

$\forall i, k \quad sendo \quad T^* = (SP, P)$

- Restringe as Variáveis

$$X_{ijk} = (0, 1) \quad \forall i, j, k \quad (15)$$

$$S_{jk} \geq 0 \quad \forall j, k \quad (16)$$

- Restringe a FO2 ao valor da FO1

$$\sum_{i \in P} \sum_{j \in T} \sum_{k \in M} CN_i \times HnU_i + CE_i \times HeU_i + S_{jk} \times Pe \times H_j \leq Z1 \quad (17)$$

As restrições definidas pelas equações de 5 a 16, são utilizadas como delimitadoras da primeira Função Objetivo, definida pela equação 3 e com resultado dado por Z1, cujo objetivo é a minimização dos custos. Tal resultado é dado pela minimização da função, respeitadas as restrições. A segunda Função Objetivo, definida por 4 e com resultado dado por Z2, é delimitada pelas mesmas restrições da primeira, e busca uma maximização de resultados de satisfação. De forma a garantir que o resultado da Função Objetiva 1 só será melhorado, ou igualado, quando da implementação da segunda Função Objetiva, foi adicionada a restrição 17 as anteriores, possibilitando a formulação experimental da dupla otimização.

4.3 Resolução

Baseado na modelagem matemática apresentada anteriormente, formularam-se algoritmos para criar uma escala de trabalho otimizada e melhorar o fluxo de informação para a tomada de decisão do gestor. Dentre os estabelecimentos cujos dados foram coletados, selecionamos um dos tipos de estabelecimento, bem como uma Região Penitenciária. O modelo foi aplicado de forma a gerar uma escala mensal de trabalho com os servidores disponíveis, considerando todas as restrições aplicadas. Dessa forma gerou-se uma escala de trabalho automatizada em duas etapas. Aqui cabe também

O modelo de redução de custos definido na equação 3 é bem explorado em diversos trabalhos como em Barboza et al. (2003), Pereira et al. (2012), Silva et al. (2005), Souza and Silva, sendo uma variação da formulação (1) de Goldbarg and Luna (2005) para o problema dos transportes. A Função Objetiva Z1 proposta traz algumas ponderações sobre o custo, bem como uma variável de folga, ou fictícia, penalizada de forma a validar modelos que estejam desequilibrados. Essa variável também apresenta uma significativa informação sobre o resultado: onde não existe atribuição, ou seja, está descoberto. Esse caso ocorre quando existia uma demanda, definida pelo parâmetro D_{jk} e exemplificado na Tabela 5, porém não existiam horas disponíveis para atendê-la.

Aplicado o modelo e obtido o resultado que minimize Z1, inicia-se o processo de ajuste do resultado obtido à segunda Função Objetivo, Z2. O objetivo da Z2 é aumentar a satisfação em relação as preferências individuais informadas pelos policiais, opção explorada em pesquisas e aplicações como em Devesse (2016), Rangel and Évora (2007), dentre outros.

Essa resolução se difere de muitas outras que buscam atender a satisfação, além da redução de custo, pois não pondera uma em relação a outra, ao invés disso cria uma delimitação da primeira em relação a segunda. Isso define o objetivo principal que se pretende alcançar: a minimização de custo para o Estado. Mesmo assim acredita-se que será possível aumentar os níveis de satisfação dos servidores em relação a escala, considerando a maximização, ou re-otimização, posterior. A Figura 7 ilustra a ideia inicial de resolução.

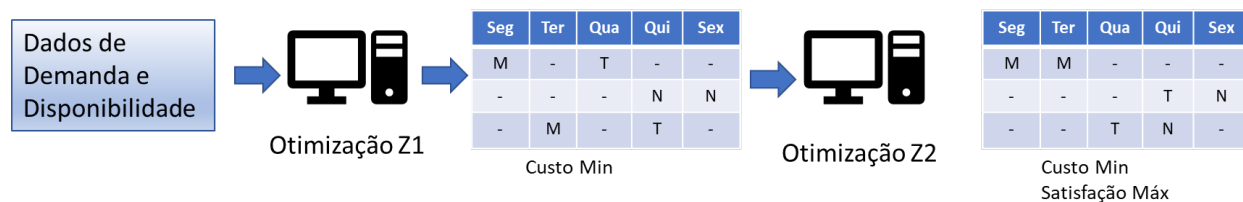


Figura 7: Fluxo de Otimização

Fonte: Confecção Própria

Para a solução do modelo, composto pelas funções dadas por 3 e 4, foi utilizado o método de Sinuany-Stern and Teomi (1986), pois o mesmo resolve objetivos separados e os acrescenta como limitantes do problema posterior, sendo que cada resolução anterior, ou objetivo anterior, tem precedência sobre o próximo.

Para tanto, o referido trabalho define que, considerando M objetivos ordenados conforme a prioridade de cada um, com H restrições, teremos:

$$Z = \text{Min}F_M(X)$$

s. t.

$$F_g(X) \leq Z_g \quad g = 1, \dots, M - 1$$

$$g_h(X) \leq b_h \quad h = 1, \dots, H$$

Onde apenas a primeira solução de Z não estará sujeita a restrição das funções anteriores quando resolvermos os modelos de forma subsequente.

4.4 Validação

De forma a validar o modelo proposto, pretende-se comparar os resultados obtidos pelo algoritmo frente aqueles executados de forma manual pelos servidores da administração pública no período de 2020. Para isso serão analisadas as informações das escalas elaboradas e executadas pelos policiais penais do estabelecimento que for escolhido como paradigma. As escalas atualmente utilizadas, e informadas para efeitos desse estudo, são formuladas em planilhas eletrônicas ou editor de texto, não havendo padronização quanto a apresentação ou forma.

Os quesitos que serão comparados para efeitos de validação são:

- Atendimento da demanda mínima;
- Dimensionamento de pessoal;
- Utilização de Horas Extraordinárias; e
- Nível de satisfação dos servidores.

Para cálculo de satisfação dos servidores, será utilizada uma pesquisa quanto as preferências em relação ao tipo de escala a ser utilizada, ao turno de preferência e a realização ou não de horas extraordinárias. A referida pesquisa gerará uma matriz relacionando o policial penal a suas preferências.

5 ALGORITMOS

Neste capítulo é apresentada a formulação dos algoritmos para resolver o modelo proposto, considerando a formulação e a resolução de cada instância.

5.1 Solvers

Conforme descrito anteriormente, os solvers escolhidos para implementar a solução foram o CPLEX, solver comercial da empresa IBM, e o CBC, do projeto COIN. Existem diversas formas de acessar os solvers para implementar o modelo, neste caso foi utilizada a biblioteca Python PuLP. Para entender e utilizar a biblioteca foi consultada a documentação do projeto, seguindo assim suas definições e características de programação. Os pseudo-códigos de implementação desses modelos podem ser descrito da seguinte maneira, conforme a documentação e o caso específico para cada função objetivo.

5.1.1 Formulação dos Algoritmos Específicos

Algorithm 1 Implementação PuLP FO 1 (CPLEX e COIN)

Require: bibliotecas: Pulp, pandas, numpy, timeit

Dados de entrada:

$Dados \leftarrow P_n, H_j, CN_i, CE_i, Pe, PP_{ij}, D_{jk}, Disp_{ik}, CNMAX_i, CEMAX_i$
considerando ($i \in P_n, j \in T, k \in M$)

Define problema:

$FO1 \leftarrow (nome, tipo(max))$

► Primeiro objetivo

Variáveis:

$FO1 \leftarrow$ Variáveis principais

$FO1 \leftarrow$ Variáveis auxiliares

for $i \in P_n$ **do**

for $j \in T$ **do**

for $k \in M$ **do**

$FO1 \leftarrow \sum CN_i \times H_n U_i + CE_i \times H_e U_i + S_{jk} \times Pe \times H_j$ ► Inse a função objetivo no

problema

$FO1 \leftarrow$ Restrições de 5 a 16

► Inse as restrições ao problema

end for

end for

end for

Resolve FO1 (Solver)

► utiliza o Solver escolhido

return Valor FO1, Valor das Variáveis, Status do Problema

As únicas diferenças entre o a implementação 1 e 2 são as definições das funções objetivos e a

Algorithm 2 Implementação PuLP FO 2 (CPLEX e COIN)**Require:** bibliotecas: Pulp, pandas, numpy, timeit

Dados de entrada:

$Dados \leftarrow P_n, H_j, CN_i, CE_i, Pe, PP_{ij}, D_{jk}, Disp_{ik}, CNMAX_i, CEMAX_i$
 considerando ($i \in P_n, j \in T, k \in M$)

Define problema:

 $FO2 \leftarrow (nome, tipo(max))$

▷ Segundo objetivo

Variáveis:

 $FO2 \leftarrow$ Variáveis principais $FO2 \leftarrow$ Variáveis auxiliares**for** $i \in P_n$ **do** **for** $j \in T$ **do** **for** $k \in M$ **do** $FO2 \leftarrow \sum X_{ijk} \times PP_{ij} + Y_{ijk} \times PP_{ij}$

▷ Insere a função objetivo no problema

 $FO2 \leftarrow$ Restrições de 5 a 17

▷ Insere as restrições ao problema

end for **end for****end for**

Resolve FO2 (Solver)

▷ utiliza o Solver escolhido

return Valor FO2, Valor das Variáveis, Status do Problema

inclusão da restrição 17 no algoritmo 2.

5.2 Heurística

De forma a criar uma alternativa aos métodos exatos utilizados pelos solvers, construiu-se uma heurística inspirada no algoritmo GRASP proposto por Talbi (2009), conforme a representação do pseudo-algoritmo 3.

Algorithm 3 GRASP

Require: Critério de parada (iterações, tempo)

Melhor_Sol $\leftarrow +\infty$

repeat

S_0 =Heurística Inicial Gulosa-Randômica

 Solução=Busca Local (S_0)

if Solução \leq Melhor_Sol **then**

 Melhor_Sol \leftarrow Solução

end if

until Critério de parada atingido

return Melhor_Sol

No algoritmo 3 verificam-se duas etapas: Heurística Inicial Gulosa-Randômica e Busca Local. Sendo a primeira responsável pela diversificação e a segunda pela intensificação da construção da solução. Nessa heurística a etapa mais importante, e muitas vezes custosa computacionalmente, é a Heurística Inicial Gulosa-Randômica, pois nela se utiliza a lista restrita de candidatos e define-se o quão gulosa ou randômica será a solução, garantindo ainda sua viabilidade dentro das restrições. No algoritmo 4, podemos verificar a ideia de implementação dessa heurística.

O algoritmo 4, foi baseado no proposto por Resende and Ribeiro (2016) em um problema de minimização, demonstrando ainda a formação da Lista Restrita de Candidatos (LRC), que basicamente restringe os candidatos da lista dentre aqueles que estão contidos em um determinador intervalo de valores. O tamanho do intervalo é calculado com base em um percentual de afastamento entre um valor mínimo e um valor máximo e por meio de um parâmetro $\alpha \in [0, 1]$.

$$c^{min} \leq \text{Candidato} \leq c^{min} + \alpha \times (c^{max} - c^{min})$$

A inequação acima garante que os candidatos a ingressarem na lista seguem um parâmetro de terminado por α , que quando 0 torna o processo puramente guloso e quando 1 torna o processo totalmente aleatório, uma vez que, após formada a lista, a designação do candidato que a compõem é aleatória.

O trabalho de Binato et al. (2002) busca utilizar essa heurística para solucionar o problema do Job Shop Scheduling Problem (JSP), que basicamente designa trabalhos a máquinas na mesma ideia do PEP. No trabalho, a heurística obteve um bom desempenho, apresentando "gap" máximo para solução ideal de 10%. Já De Leone et al. (2011), propõem a utilização do GRASP para solucionar o

Algorithm 4 Heurística Inicial Gulosa-Randômica

```

Sol ← ∅
E ← n                                     ▶ conjunto de candidatos
LRC ← 0
while E ≠ ∅ do
  if  $c_{E(i)}^{min} \leq E(i) \leq c_{E(i)}^{min} + \alpha \times (c_{E(i)}^{max} - c_{E(i)}^{min})$  then           ▶ Cria a LRC
    LRC ← E(i)
  end if
  Escolhe um candidato E(i) aleatoriamente da LRC
  Solteste ← E(i)
  if Solteste ∈ Factiveis then                                     ▶ testa se a solução é viável frente as restrições
    Sol ← Solteste
    Atualiza o conjunto E sem o candidato E(i)
    Recalcula  $c^{min}$  e  $c^{max}$ 
  else
    Solteste ← Sol
    Atualiza o conjunto E sem o candidato E(i)
    Recalcula  $c^{min}$  e  $c^{max}$ 
  end if
end while
return Sol

```

Bus Driver Scheduling Problem (BDSP), obtendo resultados muito satisfatórios frente a resolução com métodos exatos. Festa and Resende (2002) faz um revisão da utilização do GRASP até o início do ano 2001, apresentando desde as primeiras aparições, tutoriais e aplicações. Festa and Resende (2009a), tratam das construções desse algoritmos feitas até 2008 e em Festa and Resende (2009b), de forma a complementar o trabalho, foi publicado um estudo sobre o campo de aplicação no qual o GRASP foi utilizado no mesmo período.

5.2.1 Formulação dos Algoritmos Específicos

Nesta seção, será apresentados os pseudos-algoritmos que forma implementados para viabilizar o GRASP nesta formulação específica. No caso em tela, a heurística construtiva inicial gulosa/randômica foi construída de forma a produzir uma solução viável, considerando todas as restrições aplicadas ao problema. Com isso essa etapa acaba levando um tempo maior, devido a validação de cada condição.

O algoritmo 5, conforme referido anteriormente, é uma das partes mais importantes do GRASP, justamente por conter a formulação da LRC, que possibilita o ajuste entre as características gulosas ou randômicas da aplicação. A definição cuidadosa de α definirá o melhor funcionamento do modelo. Essa escolha é feita baseado na experiência empírica de cada problema, conforme demonstra

Algorithm 5 Heurística Construtiva Inicial - Gulosa/Randômica**Require:** $\alpha \in [0, 1]$, $T \leftarrow \text{tempo limite}$, $I \leftarrow \text{limite iterações}$

Dados de entrada:

 $Dados \leftarrow P_n, H_j, CN_i, CE_i, Pe, PP_{ij}, D_{jk}, Disp_{ik}, CNMAX_i, CEMAX_i$
considerando ($i \in P_n, j \in T, k \in M$)

Inicializa Solução:

 $Sol \leftarrow [0] * nxm$ ($n \in P, m \in M$)**while** $T \leq \text{tempo}$ ou $I \leq \text{Iter}$ **do** início \leftarrow Inicia contagem $Sol \leftarrow \text{heuSP_GRASP}(\text{Dados})$ \triangleright cria a solução testando restrições $Sol \leftarrow \text{heuP_GRASP}(Sol)$ \triangleright aperfeiçoa a solução com nova restrição, testando restrições $Sol \leftarrow \text{heuEXP_GRASP}(Sol)$ \triangleright aperfeiçoa a solução com nova restrição, testando restrições fim \leftarrow termina contagem Tempo \leftarrow (fim-início)**end while** **return** Sol, Iter, Tempo

Resende and Ribeiro (2016). As etapas do algoritmo podem ser descritas da seguinte maneira:

- Inserção dos parâmetros (α , tempo, iterações) conforme definido posteriormente na seção 6.2.1 de experimentos computacionais;
- Inserção dos dados definidos na subseção 4.2;
- Inicialização da matriz Solução conforme o tamanho dos dados;
- Início do laço *while* dividida em três etapas, utilizando a dinâmica do algoritmo 6, inserindo as restrições:
 1. heuSP_GRASP: separa apenas o grupo SP e designa na matriz Sol, testando todas demais restrições com um encadeamento de laços *ifs* percorrendo toda matriz criada;
 2. heuP_GRASP: separa apenas o grupo P e designa nas posições livres da matriz anterior, testando todas demais restrições com um encadeamento de laços *ifs* percorrendo toda matriz;
 3. heuEXP_GRASP: separa apenas o grupo EXP e designa nas posições livres da matriz anterior, testando todas demais restrições com um encadeamento de laços *ifs* percorrendo toda matriz.
- Retorna a matriz solução encontrada e informações de desempenho (Tempo e Iterações).

No pseudo-código 6 nós podemos observar melhor o funcionamento específico da formação da LRC, na qual a lista é formada utilizando a inequação matemática que insere o critério escolhido, nesse caso o custo da hora de trabalho, entre dois extremos. Sendo que, no caso de problema

de minimização, o limite superior é ponderado pelo fator α . Assim, cada vez que um candidato é utilizado na construção da escala o mesmo é retirado da lista de candidatos a ser analisada subsequentemente.

Algorithm 6 Heurística Base Ini_SP_P_EXP

Require: $\alpha \in [0, 1]$

Dados de entrada:

$Dados \leftarrow P_n, H_j, CN_i, CE_i, Pe, PP_{ij}, D_{jk}, Disp_{ik}, CNMAX_i, CEMAX_i$
considerando ($i \in P_n, j \in T, k \in M$)

Inicializa Solução:

$Sol \leftarrow [0] * n \times m$ ($n \in P, m \in M$)

Custo \leftarrow Custos de cada Servidor em CN dos disponíveis

Candidatos \leftarrow disponíveis no Mês

CustoMin \leftarrow min(Custo)

CustoMax \leftarrow max(Custo)

while $T \leq tempo$ ou $I \leq Iter$ **do**

if Candidatos=0 **then**

 sai do laço

else

 CustoMin \leftarrow min(Custo)

 CustoMax \leftarrow max(Custo)

end if

 LRC

 ▶ inicia a Lista Restrita de Candidatos

for Candidatos **do**

if $c^{min} \leq \text{Custo}(\text{Candidatos}) \leq c^{min} + \alpha \times (c^{max} - c^{min})$ **then**

 LRC \leftarrow Candidatos

end if

end for

 Escolhe Aleatoriamente um candidato da LRC

 Designa o candidato pra cada dia e turno seguindo as restrições

 Atualiza a lista de Candidatos e Custos, retirando o utilizado

end while

return Sol

Para a fase de intensificação, foi utilizada uma busca local, algoritmo 7, do tipo primeira melhora. Ou seja, sua característica principal é parar as iterações e retornar o resultado assim que obtiver uma melhora no valor da função objetivo. Essa se adéqua melhor ao GRASP, uma vez que muito tempo é dispendido na heurística 5, pois nessa fase é construída uma solução viável dentro das restrições.

A estratégia proposta para estruturar a vizinhança, utilizada para explorar melhorias na solução, foi baseada na construção de listas restritas de candidatos, conforme o método do próprio GRASP.

Porém essa lista de servidores se baseiam em duas premissas diferentes para cada FO. Para o objetivo 1, temos a seleção dos servidores que tiveram menos horas distribuídas na escala, de forma a procurar formatações nessa vizinhança que aloquem melhor esses servidores frente as horas disponíveis. Já para o segundo objetivo a ideia é redistribuir os servidores que tiveram uma satisfação baixo da média do grupo.

Com essas listas, a depender do número de candidatos, foram sorteados blocos de servidores para trocar posições entre si e sofrer novas distribuições. Dessa forma a ideia é intensificar, porém usar a aleatoriedade das escolhas para diversificar e assim escapar de possíveis ótimos locais. Existindo melhora entre a solução inicial e a ajustada pela busca local o laço é interrompido e o resultado apresentado, conforme exemplificado na Figura 12. O limite utilizado nesse sistema é o número de iterações máxima, provocando a parada assim que atingido.

Já a construção do algoritmo representado em 8 é a demonstração da simplicidade de construção do GRASP, uma vez que essa é sua grande virtude. Essa Meta-heurística utiliza duas heurísticas anteriores, mesclando a diversificação e a intensificação, buscando agilidade e eficácia. Além disso permite ajuste por parâmetros simples, aqui definidos no tempo de execução, número de iterações e a definição do α , para regular sua gulosidade ou randomicidade.

A heurística construtiva 6, por ter limitadores de tempo e iteração, pode, para instâncias grandes, gerar soluções que não respeitem alguma restrição. De forma a corrigir isso, o algoritmo 8 inclui uma etapa de teste quanto a factibilidade da solução e, caso exista algum relaxamento não provocado, executa-se uma etapa de conserto da solução.

De forma a criar uma população de soluções factíveis e explorar esse espaço para melhorar a chance de encontrar um solução de boa qualidade e próxima de um objetivo ótimo, foi proposto ao algoritmo 9. O projeto desse algoritmo é parecido com uma busca local, ou seja, vai verificando a melhoria da solução conforme se repetem as construções. Posteriormente, na seção 6.2.1 foram verificados os parâmetros ideias para as instâncias e nesse experimento verificou-se também que as soluções tendiam a sofrer melhorias após um certo número de repetição dos procedimentos, ou seja, dentro de um bloco de resoluções aumentam as chances de melhoria da solução, gerando o algoritmo 9.

Um parâmetro importante dessa heurística para aumentar a eficiência do GRASP, proposto no algoritmo 8, é o número de iterações máximas. Nesse quesito foi vinculado o número de iterações ao tempo máximo de execução definido na tabela 9, sendo assim as iterações são o arredondamento da divisão entre o tempo máximo e o tempo de parâmetro para a instância, definido na tabela 11.

Algorithm 7 Heurística Busca Local - Primeira Melhor**Require:** Sol_atual, $I \leftarrow \text{limite}$ iterações

Dados de entrada:

 $Dados \leftarrow P_n, H_j, CN_i, CE_i, Pe, PP_{ij}, D_{jk}, Disp_{ik}, CNMAX_i, CEMAX_i$
considerando ($i \in P_n, j \in T, k \in M$)Sol \leftarrow Solução

Inicializa melhora:

 $Melhora \leftarrow [0] * nxm$ ($n \in P, m \in M$)Candidatos \leftarrow candidatos viáveis**while** $Candidatos \neq 0$ ou $I \leq \text{Iter}$ **do**

Define o Bloco de troca:

 Ale \leftarrow Escolhe um grupo de candidatos de forma aleatória **for** $d \in Ale$ **do** **for** $k \in M$ **do**

Atualiza a Base de dados e zera as soluções referentes ao Ale

end for **end for** **while** Ale $\neq 0$ **do** Sol \leftarrow heuSP_GRASP (Sol) \triangleright redistribuí o bloco escolhido \triangleright cria a solução testando restrições Sol \leftarrow heuP_GRASP (Sol) \triangleright aperfeiçoa a solução com nova restrição, testando restrições Sol \leftarrow heuEXP_GRASP (Sol) \triangleright aperfeiçoa a solução com nova restrição, testando

restrições

 Sol \leftarrow VarFict (Sol) \triangleright cria a solução inserindo as fictícias, testando restrições **end while** FoAtual \leftarrow calcFO1(Sol_atual) FoTeste \leftarrow calcFO1(Sol)

Teste Restrições(Sol)

if FoAtual \geq FoTeste e Teste Restrições = Ok **then** Sol_atual \leftarrow Sol

break - sai do laço

else Sol \leftarrow Sol_atual Iter \leftarrow +1

Retira os candidatos usados do grupo Candidatos

end if**end while** **return** Sol, Iter

Algorithm 8 GRASP _ caso Específico

Require: $\alpha \in [0, 1]$, $T \leftarrow tempo\ limite$, $I \leftarrow limite$ iterações

Dados de entrada:

 $Dados \leftarrow P_n, H_j, CN_i, CE_i, Pe, PP_{ij}, D_{jk}, Disp_{ik}, CNMAX_i, CEMAX_i$
considerando ($i \in P_n, j \in T, k \in M$)

Inicializa Solução:

 $Sol \leftarrow [0] * nxm$ ($n \in P, m \in M$)inicio \leftarrow Inicia contagem**while** $T \leq tempo$ ou $I \leq Iter$ **do** $Sol_Teste \leftarrow$ Heurística Conts Ini_GRASP (Dados)

▷ fase de diversificação

Testa Restrições(Sol_Testes)

if falha no Teste Restrições **then**

Repara Solução

 $Sol \leftarrow Sol_Teste$ (reparada)**end if** $Sol_Teste \leftarrow$ Busca Local_primeira melhora (Sol)

▷ fase de intensificação

 $FoAtual \leftarrow calcFO1(Sol)$ $FoTeste \leftarrow calcFO1(Sol_Teste)$

Teste Restrições(Sol_Testes)

if $FoAtual \geq FoTeste$ e Teste Restrições = Ok **then** $Sol \leftarrow Sol_Teste$ **end if**iter $\leftarrow +1$ **end while**fim \leftarrow termina contagemTempo \leftarrow (fim-inicio)**return** Sol, Iter, Tempo

Algorithm 9 GRASP _ B Sol Factíveis

Require: $\alpha \in [0, 1]$, $T \leftarrow tempo$ $limite I \leftarrow limite$ iterações

Inicializa variáveis:

$Sol_Bloco_FO1 \leftarrow +infinito$ ▷ para minimização

inicio \leftarrow Inicia contagem

while $T \leq tempo$ ou $I \leq Iter$ **do**

$Sol_Teste \leftarrow$ GRASP_Caso específico (Dados) ▷ cria solução utilizando GRASP

 Calculo dos Valores dos Objetivos:

$FoAtual \leftarrow calcFO1(Sol_Bloco_FO1)$

$FoTeste \leftarrow calcFO1(Sol_Teste)$

if $FoAtual \geq FoTeste$ **then** ▷ para minimização

$Sol_Bloco_FO1 \leftarrow Sol_Teste$

end if

 iter $\leftarrow +1$

end while

fim \leftarrow termina contagem

Tempo \leftarrow (fim-inicio)

return $Sol_Bloco_FO1, Tempo$

6 EXPERIMENTOS COMPUTACIONAIS

De forma a resolver todas instâncias do problema, ou seja aquelas contidas no conjunto P, foi parametrizado um tempo máximo de execução. A tabela 9 define o critério de limite de tempo que foi usado para cada instância do problema.

Tabela 9: Tempo Limite de Execução

Instância	Tempo(s)		Total (s)
	FO1	FO2	
7	7200	7200	14400
12	7200	7200	14400
14	7200	7200	14400
15	7200	7200	14400
33	10800	10800	21600
59	10800	10800	21600
130	21600	21600	43200
147	43200	43200	86400

Fonte: confecção própria

Definidos esses parâmetros gerais, a resolução seguiu o proposto por Sinuany-Stern and Teomi (1986). Ou seja, foi resolvido o algoritmo 1 e na sequência o 2, no caso da resolução exata, e o algoritmo 8 duas vezes, ajustando apenas a o cálculo da segunda Função Objetiva, no caso da heurística. Para ambos os casos foi utilizado o resultado do primeiro algoritmo para restringir a nova solução. Cada uma das implementações teve o mesmo tempo limite, conforme a tabela 9.

6.1 Solvers

Para a resolução foi realizada a chamada dos respectivos métodos, com parametrização da instância e do tempo de execução. Os ajustes utilizando a biblioteca PuLP, tendo sido implementados os algoritmos 1 e 2 na linguagem Python, foram feitos da seguinte maneira:

- CBC

```
solver = pl.PULP_CBC_CMD(path=None, keepFiles=1,
mip=True, msg=True, options=None, timeLimit=21600)
```

- CPLEX

```
solver = pl.CPLEX_PY(timeLimit=21600)
```

Nestes exemplos os tempos máximos para execução foram de 21600 segundos, para instância 130. De forma a visualizar os resultados implementou-se rotina no Python para apresentação de

dados, conforme Figura 8, e geração de escala, conforme rotina da figura 9, exemplificada na Tabela 10.

Tabela 10: Exemplo de escala gerada

Policial	Dias do Mês de Outubro											
	1	2	3	4	5	6	7	8	9	10	...	31
1	SP	-	-	-	EXP	SP	-	-	-	EXP	...	SP
2	-	EXP	P	-	-	-	EXP	P	-	-	...	-
3	P	-	-	-	EXP	SP	-	-	-	P	...	SP
4	EXP	EXP	EXP	EXP	-	-	-EXP	EXP	EXP	EXP	...	-
...
n	-	-	-	-	-	-	-	-	-	-	...	-

Fonte: confecção própria

Além disso, foram feitos testes com os mesmos solvers utilizando a linguagem AMPL para algumas instâncias, de forma a validar os resultados obtidos e reduzir possíveis erros de modelagem nos casos estudados.

6.2 Heurística

6.2.1 Ajuste dos parâmetros

De forma a ajustar os parâmetros a serem inseridos para calibrar o GRASP, tendo assim um melhor funcionamento, foram feitos teste na heurística, representada no algoritmo 8, variando a constante α , por meio da definição de um "passo" de forma que o seu valor variasse de 0 até 1. O valor definido para o "passo" foi de 0,05, criando assim uma amostragem para analisar os valores de ajuste para heurística do Algoritmo 8, sendo os seguintes parâmetros:

- Constante α ;
- Número máximo de iterações;
- Tempo máximo de execução.

Para a formação desses valores foi construída a tabela 11, na qual as colunas FO1, FO2, Tempo e Iteração da FO1 contém a média dos valores da instância em questão, retirados os dois valores extremos. Além dessas colunas, temos a coluna Alfa (α) que representa o valor da constante que simultaneamente completa os seguintes critérios, nesta ordem de prioridade:

1. FO1 < Média FO1;
2. FO2 > Média FO2;
3. Tempo < Tempo Médio;
4. Iteração da FO1 < Média Iteração da FO1.

Existiram alguns casos em que mais de um α preencheu todos os requisitos acima. Nesses casos foi dada a prioridade aos que obtiveram a menor FO1, maior FO2, menor número de iterações e menor tempo, respectivamente.

Tabela 11: Definição de Parâmetros

Instância	FO1	FO2	Algoritmo 8			Algoritmo 9	
			Tempo(s)	Iter FO1	Alfa	Tempo(s)	Iter Bloco
E_7	277.284	26	40	4	0.05	7200	10
E_12	1.552.314	64	577	5	0.85	7200	10
E_14	746.717	83	690	10	0.45	7200	10
E_15	476.884	78	888	10	0.45	7200	10
E_33	1.232.212	156	4.893	9	0.8	10800	8
E_59	535.107	279	10.179	7	0,05	10800	8
E_130	-	-	15.268	7	0,45	43200	5
E_147	-	-	15.268	8	0,45	43200	5

Fonte: confecção própria

A tabela 11 foi construída baseada em experimentos realizados nas instâncias de E_7 a E_59. Para as duas últimas, e maiores instâncias, foram definidos os critérios com base em uma médias das instâncias anteriores, no que tangia ao parâmetros "Iter da FO1" e "Alfa(α)", e um acréscimo de 50% no tempo de execução. A impossibilidade de levantar os dados empíricos se deram pelo tempo necessário para rodar de forma livre as instâncias maiores, onde cada iteração com parâmetros elevados poderia durar cerca de 12 hs. Os dados do GRASP em bloco (algoritmo 9) foram baseados no tempo total de execução dado pela tabela 9 e iterações que variassem de 5 a 10, dependendo do tempo de execução interna limite definida para o algoritmo 8. De forma a exemplificar a construção da tabela 11, no quesito empírico, foi construída a Tabela 12, representando uma das instâncias.

Tabela 12: Experimental Instância 33

FO1	FO2	Alfa (α)	Tempo(s)	Iter p/ FO1	Iter Total	% do Menor FO1	α viáveis
1261300	149	0	5183,95	8	19	16,0%	0
1286250	165	0,05	5328,24	9	20	18,3%	0
1306662,5	144	0,1	3209,76	1	12	20,2%	0
1196700	150	0,15	7391,20	23	27	10,1%	0
1264187,5	152	0,2	5759,70	10	21	16,3%	0
1287237,5	159	0,25	4646,19	6	17	18,4%	0
1309937,5	146	0,3	3046,03	1	12	20,5%	0
1197100	154	0,35	4222,04	5	16	10,1%	0
1244262,5	158	0,4	2610,76	5	11	14,4%	0
1215287,5	159	0,45	7307,84	23	31	11,8%	0
1197525	161	0,5	5272,28	10	21	10,2%	0
1196450	162	0,55	6317,37	14	25	10,1%	0
1219875	157	0,6	4262,44	6	17	12,2%	0,6
1221087,5	147	0,65	5402,95	11	22	12,3%	0
1087175	152	0,7	7441,46	25	31	0,0%	0
1218575	162	0,75	4227,75	7	18	12,1%	0,75
1175325	164	0,8	3438,87	3	14	8,1%	0,8
1177050	159	0,85	4578,68	8	19	8,3%	0,85
1273050	166	0,9	4784,81	9	20	17,1%	0
1241887,5	155	0,95	3699,99	4	15	14,2%	0

Fonte: confecção própria

Dos dados da tabela 12 foi possível definir os critérios para os parâmetros da instância 33:

- Constante $\alpha = 0,8$
- Número máximo de iterações = 9
- Tempo máximo de execução = 4893 s

Esses valores são provenientes do cálculo da média, ajustada sem os pontos extremos de forma a diminuir o desvio, para o tempo e iterações máximas. Já para definição de alfa (α), foram utilizados os critérios definidos anteriormente. Nesse caso mais de um alfa retornou na coluna " α viáveis" da tabela 12, sendo assim foi utilizado a ordem de aplicação dos critérios para desempate. Essas escolhas podem ser mais facilmente visualizadas nos gráficos resultantes do correlação entre os parâmetros de definição e a variação da constante α .

Outro importante dado observado na fase de teste dos parâmetros do algoritmo 8 foi o número repetições do GRASP que garantiam melhores resultados. Na maioria dos experimentos verificou-se que a cada quatro repetições do modelo ele gerava uma melhora que ficava abaixo da média % de

distância entre o melhor resultado e cada uma das soluções.

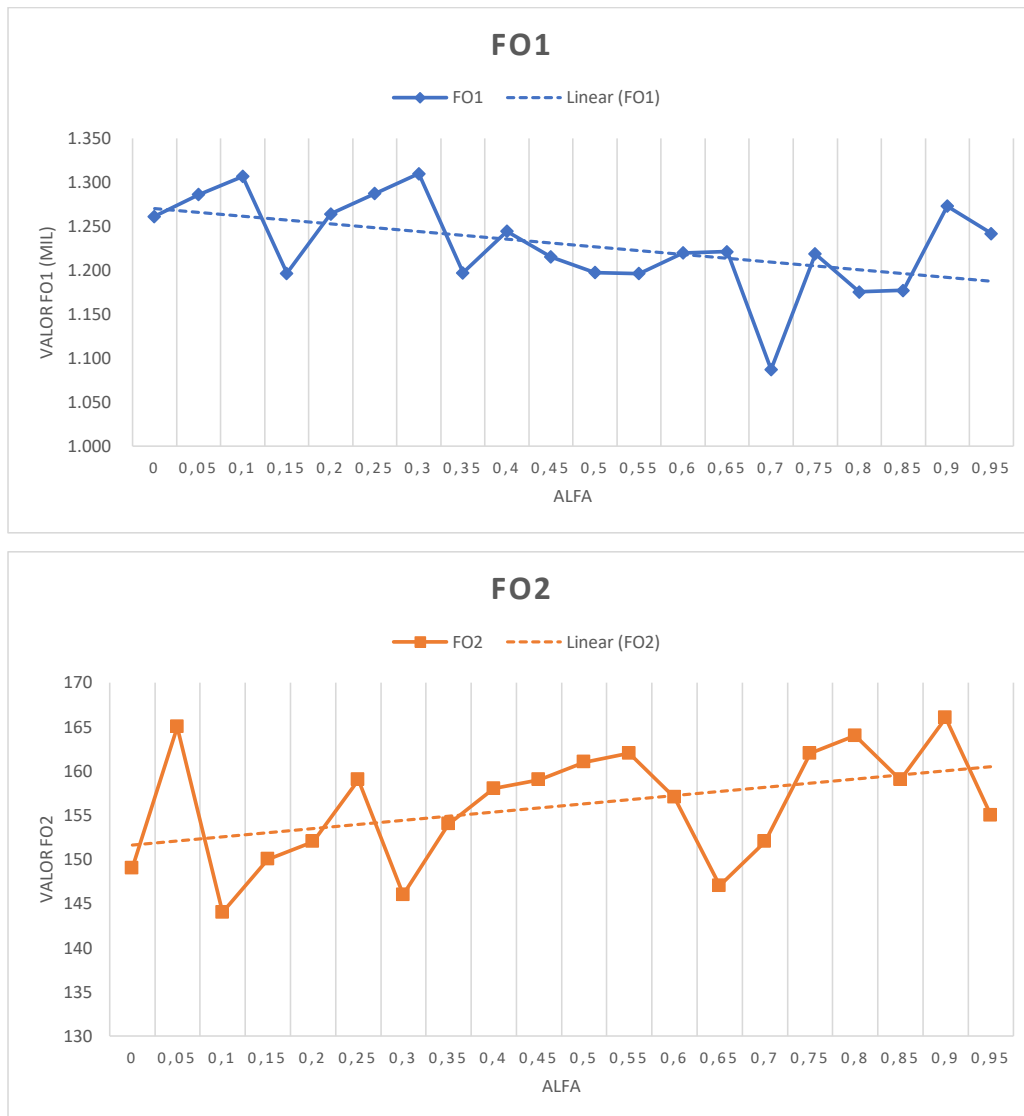


Figura 10: FO1 e FO2 - Instância 33

Fonte: confecção própria

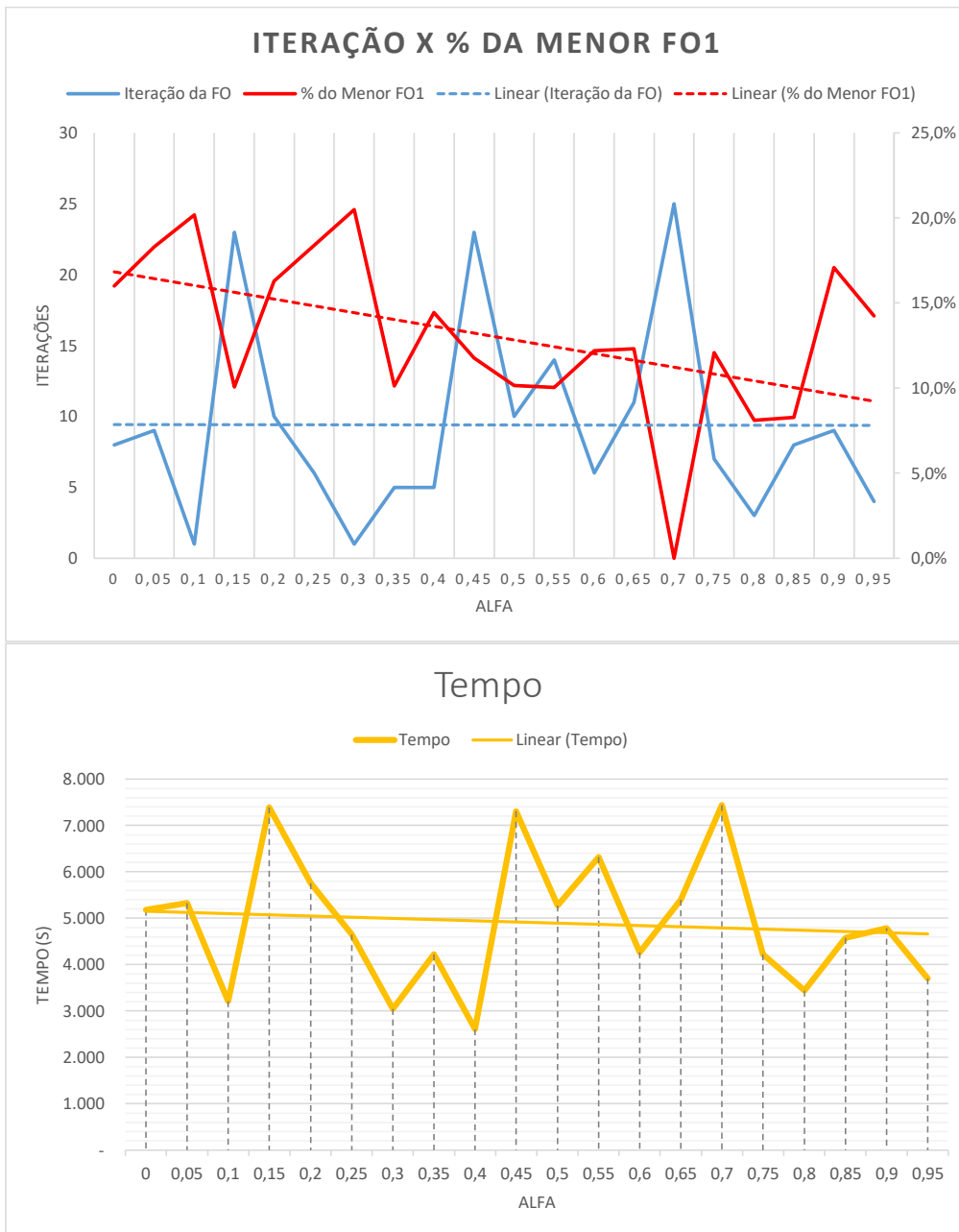


Figura 11: Tempo e Iteração - Instância 33

Fonte: confecção própria

No gráfico da FO1, na Figura10, verifica-se que, conforme Alfa (α) aumenta, o valor da função objetivo dada pela equação 3 tende a reduzir o valor, mostrando acertada a escolha do parâmetro. O mesmo ocorre no gráfico de Tempo da Figura11, pois o tempo de execução tende a cair com o valor dado por α , mostrando que a definição pela média vai permitir a busca em um tempo mais adequado.

Já o gráfico de Iterações da Figura11, permite verificar que conforme aumenta o valor de α temos uma redução no número de iterações e da distância entre a melhor solução encontrada e as soluções daquelas amostras. A equação 4 é representada pelo gráfico de FO2 da Figura10, tendo um aumento em seu valor a medida que aumentamos a constante correlacionada, o que é o desejado pra esse objetivo.

6.2.2 Resolução das instâncias

Conforme a definição empírica da seção anterior, foi possível definir os melhores ajustes para iniciar o processamento dos dados na heurística GRASP construída. De forma a aumentar a chance de sucesso, foram projetados blocos de resolução do modelo, limitados ao tempo da tabela 9 e a quatro iterações, conforme verificado nos experimentos anteriores, criando-se assim uma nova dinâmica.

Aplicados os parâmetros definidos na etapa anterior, e resumidos na tabela 11, ao algoritmo 8, procedeu-se a implementação da heurística de forma a solucionar cada instância do problema real.

Das etapas de resolução é importante destacar a seguinte sequência para cada instância:

- **tratamento dos dados de entrada:** ajuste das tabelas que contém as informações de dados no Python utilizando a biblioteca Pandas. Dessa forma criando as listas e dicionários ajustados aos algoritmos;
- **parâmetros:** ajuste conforme o estudo realizado para cada instância nos algoritmos 8 e 9;
- **inicialização do GRASP:** a execução propriamente dita da heurística, importante descrever as subetapas desse processo, o que será feito posteriormente;
- **emissão de relatório de resultados:** rotina que emite as principais informações sobre a execução do modelo e seus resultados, por meio de apresentação de resultados gerados nos algoritmos 8 e 9, com os seguintes dados:

A solução do Grasp para otimizar FO1 tem tem os seguintes valores:

- FO1= (276000.0, 33500.0, 10500.0, 232000)
- FO2= (24, [0, 0, 0, 5, 8, 5, 6])
- HN= 640 -HE: 120
- Tempo de execução= 55.21
- Iterações= 27

A solução do Grasp para otimizar FO2 tem os seguintes valores:

- FO1= (276000.0, 33500.0, 10500.0, 232000)

- FO2= (25, [0, 0, 0, 6, 8, 6, 5])

- HN= 640 -HE: 120

- Tempo de execução= 49.35

- Iterações= 21

- **tratamento dos dados de saída:** rotina que ajusta os dados de saída de forma a criar as tabelas de escala mensais para cada funcionário. Esses dados são gerados em planilhas eletrônicas do microsoft excel.

Da etapa de inicialização é importante listar as subetapas explicando as mais importantes. No algoritmo 8 ocorrem duas subetapas principais: - Heurística Construtiva Inicial; - Busca Local.

Na primeira parte é executado o algoritmo 5, cuja finalidade é apresentar uma solução inicial que respeite as restrições, ou seja, viável. Para tanto ela vai se utilizar da construção da LRC que é a característica mais importante do GRASP. Exemplificando para a instância 7 com $\alpha = 0.05$, considerando a lista de custo da tabela 13:

Tabela 13: Lista Custo Instância 7

Policial	Custo (R\$ /h)
P1	50
P2	75
P3	75
P4	50
P5	34
P6	50
P7	75

Fonte: confecção própria

→ Cálculo dos limites: $C^{min}(34) \leq Candidato \leq C^{min}(34) + \alpha(0.05) \times (C^{max}(75) - C^{min}(34)) = 34 \leq Candidato \leq 36,05$

Ou seja, será verificado cada candidato da lista ingressando apenas aqueles que satisfizerem a inequação, assim temos a seguinte lista:

→ LRC=(34)

Sendo assim o candidato com custo 34 será o escolhido para iniciar a designação, tendo alocação aleatória, conforme demanda e restrições, nas posições da escala de serviço. Nota-se que, devido ao $\alpha(0.05)$ muito próximo a zero, a heurística opera praticamente de forma gulosa pura. Finalizada essa etapa, é atualizada a lista de custos, retirando o candidato já utilizado, e feita a construção de

nova LRC, que nesse segundo passo será (50,50,50). Essa metodologia segue de forma sucessiva até que não existam mais candidatos e a escala gerada seja factível.

Finalizada essa subetapa, inicia-se a fase de intensificação, na qual é feita uma busca local na solução gerada na etapa anterior por meio do algoritmo 7. Aqui utilizou-se a estratégia da primeira melhora, para agilizar o processo. A fase mais importante desse processo é a definição da vizinhança, pois ela definirá a possível fuga de um ótimo local e uma melhor exploração do espaço de soluções. Para isso será utilizada a estratégia de criação de blocos de servidores que fazem parte da solução atual.

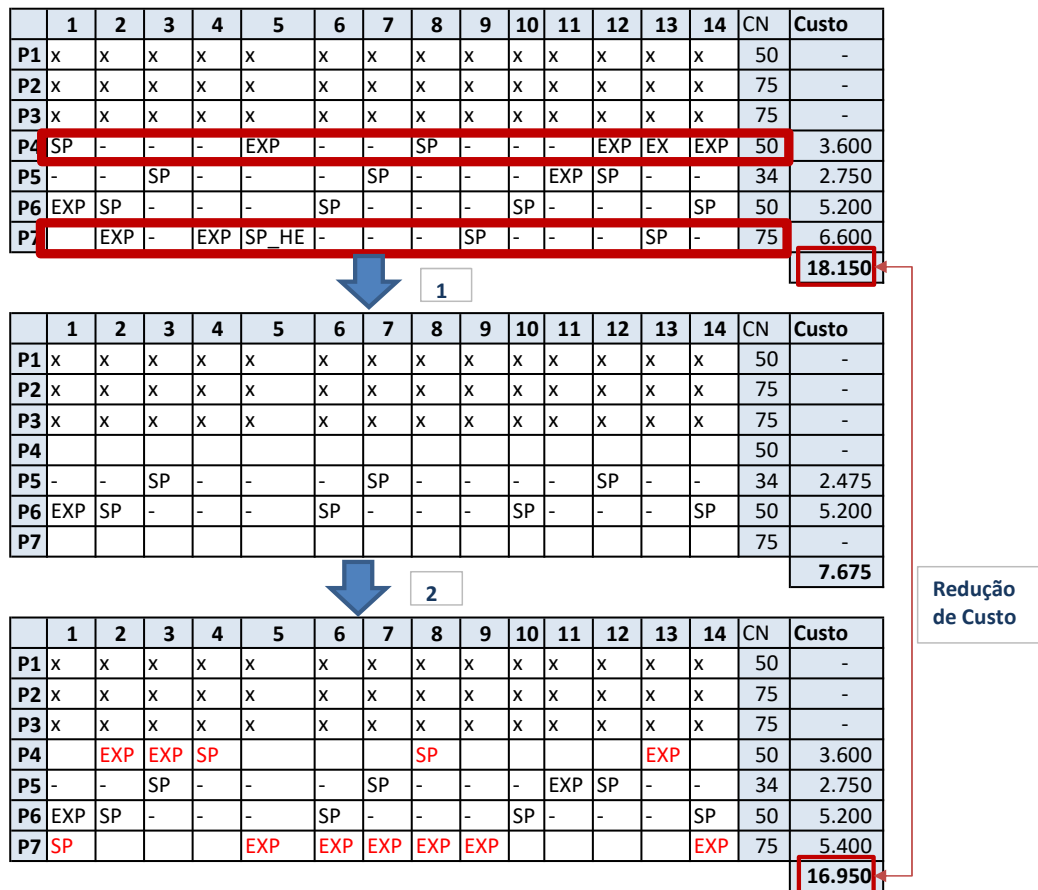


Figura 12: Dinâmica de vizinhança

Fonte: confecção própria

A Figura12 demonstra uma possível troca utilizando esse sistema de vizinhança. Nessa dinâmica são criadas lista de candidatos considerando o custo de cada um deles, similar a executado na LRC, após isso são escolhidos pares dessa lista para promover trocas. Na etapa 1 da referida figura os dois escolhidos são removidos de suas posições, reestabelecendo a situação de antes das designações. Na etapa 2 indicada na figura são feitas novas distribuições, respeitadas as restrições. Nessa etapa verifica-se se houve redução de custo.

Caso exista redução no custo, como no exemplo proposto, a troca é mantida e novos candidatos são escolhidos para realizar trocas. Sempre que resultar em custo maior, ou violação de restrição, a troca é cancelada e a solução inicial é restaurada. Ocorrendo uma melhora o algoritmo sai do laço e guarda a nova solução melhorada, caso contrário ele segue testando os candidatos restantes a é atingir o limite de iterações.

A solução gerada é, conforme esquema demonstrado na Figura7, reprocessada utilizando o algoritmo 8 ajustado para maximização da equação 4. Para isso é necessária a inserção da restrição, equação 17, que vincula o custo máximo ao encontrado na solução inicial. Dessa maneira é feita nova otimização do modelo buscando aumentar a satisfação dos servidores sem aumentar o custo já encontrado.

```
##### FO1_Minimizar_Custo #####
Status: integer optimal solution - Código Status CPLEX 101
Status do Pulp 1 e 2: Optimal e Optimal
O valor de salários pagos é R$ 1.220.448.00 por mês.
#####
##### FO2_Maximizar_Satisfação #####
Status: integer optimal solution - Código Status CPLEX 101
O valor máximo de atendimento das preferências dos policiais 796 .
#####
##### Informações #####
Tempo para solução FO1: 1.88
Tempo para solução FO2: 2.40
Custo Normal = 984500.00
Custo Extra = 235987.50
Custo Descoberto = 4984000.00
Satisfação total FO2 = 796.00
Satisfação total FO1 = 788.00
H_Normal Disponível para o Modelo = 18720.00
H_Extra Disponível para o Modelo = 4120.00
H_Normal Utilizada para o Modelo = 18720.00
H_Extra Utilizada para o Modelo = 3328.00
#####
```

Figura 8: Saída de informações e status do problema

Fonte: Confecção Própria

```

#####cria a matriz de saida da escala
matriz=[]
for i in nome:
    lista=[]
    for k in dia:
        .....
        .....if sum(x[i,j,k].varValue - for j in Turno)>=1:
        .....for j in Turno:
        .....if x[i,j,k].varValue>=1:
        .....lista.append(j)
        .....else:
        .....lista+='-'
    matriz+= [lista]
    .....
df5 = pd.DataFrame(matriz, index=[i for i in nome], columns=[k for k in dia])
#Inserção da Demanda Turno Dia na Escala
D_saida=df2.T
Saida=pd.concat([df5,D_saida])....
#Inserção da preferencia
lista2=[]
lista3=[]
for i in nome:
    .....
    .....if Preferencia_F01[i].varValue>=0.01:
    .....lista2.append(Preferencia_F01[i].varValue)
for i in nome:
    .....
    .....if Preferencia_F02[i].varValue>=0.01:
    .....lista3.append(Preferencia_F02[i].varValue)
    .....else:
    .....lista3.append(0)
    .....
while len(Saida)-len(lista2)!=0:
    ..... lista2.append(0)
while len(Saida)-len(lista3)!=0:
    ..... lista3.append(0)
    .....

Saida['Satisfação_F01']=lista2
Saida['Satisfação_F02']=lista3
# Saida Excel (Ajustar Caminho e nome)
Saida.to_excel(r'e:\Mestrado\Mestrado\Trabalho_DADOS\E_130\Resultados\Escala_E_130_V2_CEPLEX.xlsx')

```

Figura 9: Cria a Escala Otimizada

Fonte: Confecção Própria

7 AVALIAÇÃO DO MODELO

De forma a avaliar a construção do modelo, bem como o desempenho de cada método de resolução, será feita a comparação de diversos aspectos quanto a cada método empregado, utilizando como base os dados reais que motivaram a construção deste modelo. Conforme já visto anteriormente, os métodos de resolução foram o Manual, Exatos (CPLEX e CBC) e Não-Exatos (GRASP e GRASP_B). Inicialmente cabe análise quanto aos custos, sendo eles financeiros ou pelo não atendimento da demanda, o que nos leva ao ideal dimensionamento da mão de obra.

7.1 Desempenho relacionado aos custos

O resultado inicial apresentado refere-se ao principal objetivo, definido na equação 3, tendo a tabela 14 apresentado os valores alcançados. Importante lembrar que o resultado da FO1 é a soma do custo financeiro e da penalização feita por postos sem alocação de policial.

Tabela 14: Valores FO1 por método

Instância	Composição dos FO 1					Status do Solver	
	Manual	CPLEX	CBC	GRASP	GRASP_B	CPLEX	CBC
E_7	456.025	276.000	276.000	283.400	283.400	ótimo	ótimo
E_12	1.780.750	1.349.800	1.349.800	1.407.275	1.400.100	ótimo	ótimo
E_14	1.152.675	654.613	654.613	699.013	691.650	ótimo	ótimo
E_15	610.400	166.513	166.513	259.825	244.113	ótimo	ótimo
E_33	1.330.775	N/A	N/A	676.888	650.563	tempo	tempo
E_59	659.878	406.500	406.500	427.088	428.675	ótimo	ótimo
E_130	8.694.922	5.421.525	5.421.525	6.133.825	6.079.263	ótimo	ótimo
E_147	6.029.100	N/A	N/A	3.960.650	3.919.475	tempo	tempo

Fonte: confecção própria

Na primeira análise, percebe-se que os métodos de otimização apresentaram resultados melhores que o método manual em todos os casos. Essa redução não implica, necessariamente, redução de custos financeiros, pois é importante ressaltar que a equação que define o custo relaciona o custo financeiro e o operacional, sendo este último o que mais pesa nos valores pois sua penalização é mil unidades a cada hora descoberta. Na referida tabela estão destacados os métodos que foram mais eficientes no quesito avaliado. Os solvers, CPLEX e CBC, não conseguiram apresentar uma solução nas instâncias E_33 e E_147, devido ao atingimento do tempo limite de processamento definido, nesse caso o dado foi apresentado com N/A (Não Atingido). O método meta-heurístico proposto conseguiu os melhores resultados nas instâncias onde os solvers não obtiveram soluções válidas, destacando-se pela redução em relação a escala Manual. Além disso, as soluções do GRASP para as instâncias que não apresentaram solução pelos solvers foram satisfatórias, não infringindo nenhuma restrição.

Quanto a questão da otimalidade apresentada pelos solvers, frente a melhora na solução dada pelo GRASP, utilizou-se a linguagem AMPL para replicar o modelo do PuLp e testar os solvers novamente. Dessa forma buscou-se reduzir a chance de erro na modelagem, ou erros de interação entre as bibliotecas Python de tratamento de dados e formulação matemática. Como resultado desse teste, encontrou-se no AMPL o mesmo resultado, utilizando-se os solvers propostos. Além disso, quanto a falta de solução viável nessas últimas instâncias, verificou-se que ao relaxar a restrição 14 para 48 horas de folga a solução passou a situação ótima, demonstrando uma dificuldade de lidar com essa solução no tempo proposto.

Tabela 15: Horas Normais e Extras utilizadas por método

Instância	CPLEX		CBC		GRASP		GRASP_B	
	HN	HE	HN	HE	HN	HE	HN	HE
E_7	640	120	640	120	640	112	640	112
E_12	1.680	320	1.680	344	1.680	280	1.680	288
E_14	2.000	448	2.000	448	2.000	400	2.000	408
E_15	2.160	320	2.160	320	2.160	280	2.160	272
E_33	N/A	N/A	N/A	N/A	4.400	680	4.400	688
E_59	8.640	-	8.640	-	8.640	392	8.640	424
E_130	18.800	4.120	18.800	4.120	18.800	3.344	18.800	3.408
E_147	N/A	N/A	N/A	N/A	20.304	3.344	20.304	3.384

Fonte: confecção própria

De forma a validar os resultados, foi feita a comparação do GRASP com os solver em alguns quesitos, chamando atenção os resultados da tabela 15. A coluna HN (Horas Normais) de todos os métodos apresenta o mesmo resultado, o que era de se esperar uma vez que as restrições 9 e 10 exigem isso do modelo, de forma a atender questões legais. Já a coluna HE (Hora Extra) mostra que os métodos exatos utilizaram mais horas extras, dentre as disponíveis, na maioria dos casos em que conseguiu um melhor resultado na FO1.

Tabela 16: % FO1 em relação ao Manual

Instância	Manual	CPLEX	CBC	GRASP	GRASP_B
E_7	100%	61%	61%	62%	62%
E_12	100%	76%	76%	79%	79%
E_14	100%	57%	57%	61%	60%
E_15	100%	27%	27%	43%	40%
E_33	100%	N/A	N/A	51%	49%
E_59	100%	62%	62%	65%	65%
E_130	100%	62%	62%	71%	70%
E_147	100%	N/A	N/A	66%	65%

Fonte: confecção própria

Na tabela 16 fica evidenciada a redução do custo total atrelado aquela medida, o qual se buscava minimizar. O desempenho médio dos solvers, em relação as instâncias com resultados, levou a uma redução de aproximadamente 43% em relação a escala Manual. Se considerarmos a Meta-heurística GRASP, tanto o bloco como a normal, as únicas a obter resultados válido em todas instâncias, a redução média do custo foi de 38%. Sendo que a GRASP_B conseguiu um desempenho médio ligeiramente melhor.

Tabela 17: Composição FO1 Custo Financeiro R\$

Instância	Manual	CPLEX	CBC	GRASP	GRASP_B
E_7	32.025	44.000	44.000	43.400	43.400
E_12	100.750	149.800	149.800	143.275	144.100
E_14	96.675	126.613	126.613	123.013	123.650
E_15	138.400	166.513	166.513	167.825	164.113
E_33	226.775	N/A	N/A	284.888	282.563
E_59	395.878	406.500	406.500	427.088	428.675
E_130	1.034.922	1.309.525	1.309.525	1.237.825	1.247.263
E_147	1.152.100	N/A	N/A	1.312.650	1.319.475

Fonte: confecção própria

Tabela 18: Composição FO1 Custo Descoberto "S"

Instância	Manual	CPLEX	CBC	GRASP	GRASP_B
E_7	424.000	232.000	232.000	240.000	240.000
E_12	1.680.000	1.200.000	1.200.000	1.264.000	1.256.000
E_14	1.056.000	528.000	528.000	576.000	568.000
E_15	472.000	-	-	92.000	80.000
E_33	1.104.000	N/A	N/A	392.000	368.000
E_59	264.000	-	-	-	-
E_130	7.660.000	4.112.000	4.112.000	4.896.000	4.832.000
E_147	4.877.000	N/A	N/A	2.648.000	2.600.000

Fonte: confecção própria

Pela tabela 17 pode-se verificar que, considerando as melhoras obtidas pelos métodos de otimização, haveria um acréscimo no valor de dispêndio mensal com salário no valor de R\$ 627.463,00. Já a redução de horas demandadas e não trabalhadas, conforme tabela 18, seria de 8.497 horas/mês. Isso representaria um aumento de eficiência, uma vez que para reduzir a mesma quantidade de horas utilizando trabalho extraordinário, considerando o valor médio da hora, seria necessário R\$ 804.581,00 por mês, ou seja um valor 28% superior ao valor utilizado.

Isso representa uma economia potencial de aproximadamente R\$ 2,36 milhões anuais e um aumento significativo no atendimento da demanda. Isso altera, inclusive, a perspectiva de contratações futuras de servidores pois a otimização muda a dinâmica de alocações e reduz a necessidade de servidores, devido a melhor alocação.

7.2 Desempenho relacionado a demanda e disponibilidade de horas

Quanto ao atendimento da demanda e a utilização da disponibilidade de horas, tanto normais como extras, para cada escala, pode-se verificar um aumento na eficiência nesse quesito, demonstrando uma melhor distribuição de pessoal. Para isso é importante destacar as condições teóricas antes das formulações, tais dados estão resumidos na tabela 19.

Tabela 19: Informações sobre os Estabelecimentos

Instância	Disponibilidade		Demanda (h)	S (h) (1000) min Teórico
	H_Normal	H_Extra		
E_7	640	160	992	192
E_12	1680	384	3224	1160
E_14	2000	512	2976	464
E_15	2160	562	2480	-242
E_33	4400	1112	5456	-56
E_59	8640	2200	7936	-2904
E_130	19136	3824	27032	4072
E_147	20400	5136	26288	752

Fonte: confecção própria

O cálculo do valor a descoberto teórico (min teórico) é dado pela diferença entre a demanda e a disponibilidade total, sendo que, em caso de valor negativo, o estabelecimento estaria com horas a mais disponíveis para formar a escala. Porém, nem sempre é possível atingir o mínimo, mesmo com esses valores disponíveis. A principal razão são as restrições, que impõem condições para alocar os servidores, não permitindo muitas vezes que eles sejam totalmente utilizados. O relaxamento da restrição legal sobre folgas obrigatórias permitiria, por exemplo, uma melhor utilização do recurso disponibilidade de horas.

Tabela 20: Utilização de cada método do total disponível de horas

Instância	Manual	CPLEX	CBC	GRASP	GRASP_B
E_7	76%	95%	95%	94%	94%
E_12	75%	97%	97%	95%	95%
E_14	80%	97%	97%	96%	96%
E_15	78%	91%	91%	90%	88%
E_33	79%	N/A	N/A	92%	92%
E_59	78%	80%	80%	83%	84%
E_130	84%	100%	100%	96%	97%
E_147	84%	N/A	N/A	93%	93%

Fonte: confecção própria

A tabela 20 destaca, por estabelecimento, ou instância, aquele método que utilizou a maior proporção de horas disponíveis. Mais uma vez demonstrando um bom desempenho dos métodos de otimização propostos frente ao manual. Importante destacar que essa comparação tem ressalvas, uma vez que alguns estabelecimento tinham mais horas do que o necessário, sendo assim o ideal era não gastar mais horas que a demanda. Fato que ocorreu na instância E_59, cujo máximo que

poderia chegar da disponibilidade era 73%. Nesse ponto, o modelo proposto acaba utilizando toda hora disponível por servidor, uma vez que por legislação existe um mínimo de horas já contratadas e que devem ser alocadas. O modelo manual ficou abaixo da alocação mínima legal, demonstrando o possível uso em tarefas fora da escala de serviço. Essa tabela também mostra que, nas instâncias nas quais só o método não exato obteve resultado, a proporção de utilização das horas foi próxima as demais instâncias, indicando que o resultado obtido tem a mesma eficácia.

Tabela 21: Utilização de cada método do total demandado de horas

Instância	Manual	CPLEX	CBC	GRASP	GRASP_B
E_7	61%	77%	77%	76%	76%
E_12	48%	62%	62%	61%	61%
E_14	67%	82%	82%	81%	81%
E_15	86%	100%	100%	98%	97%
E_33	80%	N/A	N/A	93%	93%
E_59	106%	109%	109%	114%	114%
E_130	72%	85%	85%	82%	82%
E_147	81%	N/A	N/A	90%	90%

Fonte: confecção própria

A tabela 21 complementa a informação da eficiência em relação a utilização das horas, demonstrando se o nível de demanda foi atendido em cada modelo. Mais uma vez destacou-se a proposta implementada do GRASP, sendo que nota-se que, conforme verificado na disponibilidade, a instância E_59 foi superdimensionada nas escalas por uma questão legal. De qualquer forma a escala manual peca ao utilizar menos horas do que o disponível no contrato, tendo sido utilizadas as horas para questões administrativas, provavelmente.

7.3 Desempenho relacionado a FO2

Devido a priorização no modo de resolução de cada função objetivo, sendo que a FO1 tem precedência sobre FO2, percebe-se que não houve melhora consistente na satisfação dos funcionários. A equação 4 foi formulada buscando a maximização dos resultados. A tabela 22 resume os resultados alcançados por cada um dos métodos, conforme metodologia proposta.

Tabela 22: Valores da FO2 alcançado por cada método

Instância	Manual	CPLEX	CBC	GRASP	GRASP_B
E_7	24	33	33	26	25
E_12	104	88	88	73	69
E_14	72	102	102	82	131
E_15	78	110	110	92	97
E_33	153	N/A	N/A	157	151
E_59	327	333	N/A	281	277
E_130	850	877	N/A	786	737
E_147	278	N/A	N/A	840	855

Fonte: confecção própria

Mesmo com um resultado não tão consistente quanto na otimização da FO1, a otimização proposta conseguiu melhorar a satisfação na maioria dos casos testados, sendo que em apenas um caso ocorreu uma piora nesse quesito. A tabela 23 resume esse desempenho que, além da relação com a escala manual, mostra o melhor desempenho dos métodos exatos frente a Meta-heurística proposta.

Tabela 23: Desempenho de cada método em relação a Manual - FO2

Instância	Manual	CPLEX	CBC	GRASP	GRASP_B
E_7	100%	138%	138%	108%	104%
E_12	100%	85%	85%	70%	66%
E_14	100%	142%	142%	114%	182%
E_15	100%	141%	141%	118%	124%
E_33	100%	N/A	N/A	103%	99%
E_59	100%	102%	N/A	86%	85%
E_130	100%	103%	N/A	92%	87%
E_147	100%	N/A	N/A	302%	308%

Fonte: confecção própria

Aqui importante relembrar que os dados referem-se a otimização realizada pela segunda função objetivo (eq. 4), ou seja o não atingimento de resultados pela pelos métodos exatos em quatro instâncias referem-se apenas a segunda fase da otimização. Ou seja, caso um valor tenha sido obtido na primeira otimização para FO1, existe também um valor para a satisfação. Porém, aqui cabe analisar apenas o resultado alcançado pela maximização desse objetivo.

7.4 Desempenho dos Métodos Exatos x Heurística

Para verificar o desempenho da Meta-heurística proposta serão feitas algumas comparações quanto ao desempenho em relação as duas funções objetivos (FO1 e FO2). Nesse aspecto, é importante verificar o resultado alcançado, bem como o tempo despendido para tal. Na tabela 24 o GAP refere-se a diferença, em percentual, entre o valor obtido pela heurística e os solvers, enfatizando que trata-se de minimização da equação 3, sendo que quanto menor a variação modular melhor.

Tabela 24: Meta-heurística x Métodos exatos - FO1

Instância	Resultado FO 1				GAP			
	CPLEX	CBC	GRASP	GRASP_B	CPLEX	CBC	CPLEX	CBC
E_7	276.000	276.000	283.400	283.400	2,7%	2,7%	2,7%	2,7%
E_12	1.349.800	1.349.800	1.407.275	1.400.100	4,3%	4,3%	3,7%	3,7%
E_14	654.613	654.613	699.013	691.650	6,8%	6,8%	5,7%	5,7%
E_15	166.513	166.513	259.825	244.113	56,0%	56,0%	46,6%	46,6%
E_33	N/A	N/A	676.888	650.563	N/A	N/A	N/A	N/A
E_59	406.500	406.500	427.088	428.675	5,1%	5,1%	5,5%	5,5%
E_130	5.421.525	5.421.525	6.133.825	6.079.263	13,1%	13,1%	12,1%	12,1%
E_147	N/A	N/A	3.960.650	3.919.475	N/A	N/A	N/A	N/A

Fonte: confecção própria

Quanto aos resultado da otimização da primeira função objetiva, a meta-heurística se mostrou promissora, uma vez que conseguiu manter um valor reduzido no que tange aos GAPs em relação aos solvers, tanto o comercial quanto o aberto, na maioria dos casos. Importante perceber também, que além disso, o GRASP construído para o caso específico retornou resultados viáveis em todos os casos reais. Como tratado na sessão anterior, a otimalidade atingida pelos solvers na maior parte das instâncias, o que gera um bom parâmetro para verificar o funcionamento do GRASP, que nesse caso obteve GAP médio de 6,2%. Para o cálculo da média foi desconsidera a instância E_15 que claramente teve um mau funcionamento frente aos solvers, fato que pode ocorrer em algumas instâncias, segundo Silva et al. (2006).

Os solvers não conseguiram retornar resultado válido em duas das instâncias no tempo proposto, porém retornaram soluções nos demais casos. Eles obtiveram um desempenho superior ao da heurística em todos casos válidos, como podemos verificar no resumo da tabela 24, porém com GAP aceitável.

Tabela 25: Meta-heurística x Métodos exatos - FO2

Instância	Resultado FO 2				GAP			
	CPLEX	CBC	GRASP	GRASP_B	GRASP		GRASP_B	
					CPLEX	CBC	CPLEX	CBC
E_7	33	33	26	25	21%	21%	24%	24%
E_12	88	88	73	69	17%	17%	22%	22%
E_14	102	102	82	131	20%	20%	28%	28%
E_15	110	110	92	97	16%	16%	12%	12%
E_33	N/A	N/A	157	151	N/A	N/A	N/A	N/A
E_59	333	N/A	281	277	16%	N/A	17%	N/A
E_130	877	N/A	786	737	10%	N/A	16%	N/A
E_147	N/A	N/A	840	855	N/A	N/A	N/A	N/A

Fonte: confecção própria

No que tange ao resultado da FO2, a tabela 25 apresenta o GAP resultante da segunda otimização feita. Nesse caso a média da variação entre os resultados alcançados foi muito alta, passando dos 20%, esse fato mostra um claro ponto de possível melhoria a ser implementada.

Tabela 26: Capacidade de Melhora Satisfação - FO2

Instância	CPLEX	CBC	GRASP	GRASP_B
E_7	10%	14%	0,0%	0,0%
E_12	24%	24%	0,0%	0,0%
E_14	20%	17%	0,0%	0,0%
E_15	22%	20%	0,0%	1,0%
E_33	N/A	N/A	0,0%	0,0%
E_59	10%	N/A	0,0%	0,0%
E_130	12%	N/A	0,0%	0,0%
E_147	N/A	N/A	0,0%	0,0%

Fonte: confecção própria

A tabela 26 remete ao desempenho da rotina de re-otimização, resumida na Figura7, parametrizada para atingir o objetivo de maximização da satisfação, limitada a não ultrapassar o valor mínimo de custo já obtido na minimização da FO1. A referida tabela compara se houve melhora do valor obtido na satisfação quando minimizada a FO1, ou seja se o algoritmo 8, ajustado a maximização, consegue produzir melhoras na satisfação (objetivo da FO2), sem piorar o objetivo da FO1.

Percebe-se que os métodos exatos obtiveram maior êxito em melhorar a satisfação, após ocorrer uma primeira otimização. Já a meta-heurística proposta não alcançou bons resultados, quando comparada a primeira à segunda otimização, na questão da satisfação. Porém, cabe ressaltar que

os valores encontrados na primeira otimização já eram bons resultados, comparados a manual, conforme a tabela 23.

Tabela 27: Tempo de execução dos métodos - FO2

Instância	Tempo de Execução (s)				Metaheurística/Solver			
	CPLEX	CBC	GRASP	GRASP_B	GRASP		GRASP_B	
					CPLEX	CBC	CPLEX	CBC
E_7	0,19	0,52	2,68	131,00	14	5	689	252
E_12	0,39	1,89	31,00	1.141,00	79	16	2.926	604
E_14	0,27	288,02	39,00	1.646,00	144	0	6.096	6
E_15	1,03	3,76	38,00	2.024,72	37	10	1.966	538
E_33	N/A	N/A	187,00	6.835,00	N/A	N/A	N/A	N/A
E_59	8,00	8,30	240,00	8.715,00	30	29	1.089	1.050
E_130	68,00	863,00	3.720,00	43.935,00	55	4	646	51
E_147	N/A	N/A	4.609,00	46.664,00	N/A	N/A	N/A	N/A

Fonte: confecção própria

No que tange ao tempo de execução, a tabela 27 apresenta os resultados comparados, sendo que a relação entre métodos mostra a quantidade de vezes que o valor da Meta-heurística é maior que o do solver. O solver da IBM, obteve o melhor desempenho nas instâncias que resolveu. A heurística implementada conseguiu apenas ficar abaixo dos tempos máximos estipulados. Sendo assim, a questão temporal foi razoável, uma vez que em situações de maior exigência a heurística obteve solução em tempo ainda viável, considerando a tempestividade necessária para o resultado conforme fluxo da Figura6.

8 CONCLUSÃO

A proposta do trabalho, apresentar ferramentas de tomada de decisão em processos de gerenciamento de pessoal, mais especificamente formulação de escalas de trabalho no sistema prisional gaúcho, mostrou-se exitosa. Afirma-se isso devido a construção de um modelo matemático e sua implementação, em três ferramentas distintas, de forma a gerar escala automatizada e de qualidade superior as manuais nos aspectos estudados. Tal dado fortalece sua utilização no caso prático, ainda mais utilizando métodos que não gerariam custos adicionais, como a compra de um software, mesmo com algum custo envolvido os mesmos aparentam ser inferiores aos possíveis ganhos.

Além da situação fática, aprofunda-se a utilização da Meta-heurística GRASP com a proposta de um algoritmo baseado nessa estrutura, contribuindo ao comparar seu desempenho aos demais métodos apresentados. Mostra-se promissora a aplicação desse método na construção de escalas de trabalho dentro do setor público, tendo em vista a boa relação custo benefício entre o método não-exato e os demais. O ganho financeiro imediato é relativo, pois depreende-se da ampliação do atendimento da demanda, que hoje encontra-se suprimida. Mesmo assim, pela análise dos resultados, existe boa possibilidade de ganho financeiro no curto prazo, ampliando a capacidade de atendimento e reduzindo a necessidade de novas contratações. É possível, também, configurar a demanda a ser atendida para os postos existentes e assim buscar a otimização das escalas nas condições atuais, ou seja, otimizar a demanda atualmente definida.

Essa ideia foca na definição da demanda que, como visto no sessão 2.1, é uma das etapas do escalonamento de pessoal e poderia ser desenvolvida como um método de otimização. Além disso, amplia-se a base para novos estudos nesse diapasão, deixando espaço para melhorias importantes no funcionamento da Meta-heurística proposta, como no desempenho da segunda função objetiva e redução do tempo de execução.

Para tanto, considerando estudos futuros, outras variantes do método GRASP poderiam ser propostas de forma a comparar seus desempenhos com o aqui apresentado. Em Neto and Pedroso (2003) é feita uma comparação da Meta-heurística GRASP, com variação no método de construção inicial e intensificação, com resultados interessantes, no que tange ao comportamento do algoritmo. Nesse caso os métodos se alternaram no atingimento de melhores resultados, demonstrando o comportamento diverso a depender do problema e instância. Já em Sandoya and Aceves (2013), apresenta-se os resultados de variações do GRASP, como GRASP-PR (combinação com Path Relinking) e GRASP-VNS (combinação com Variable Neighborhood Search), em problemas teste. Os resultados mostram um aumento do GAP conforme aumenta a dificuldade das instâncias, a depender da adaptação implementada na heurística.

Além desses, Alvarenga and Rocha (2006) apresentou proposta de construção para comparar o GRASP, e algumas variações do mesmo, na solução do problema da árvore geradora de custo mínimo feita por um Algoritmo Genético da literatura. Nessa proposta o GRASP teve desempenho superior, na comparação feita, apresentando soluções de melhor qualidade, segundo o autor. Conforme avaliação da meta-heurística GRASP proposta por Silva et al. (2006), é comum que uma

heurística apresente um bom desempenho para uma classe de problemas ou para algumas instâncias e para outras tenha um desempenho ruim.

De forma a implementar melhorias futuras na meta-heurística desenvolvida neste trabalho, um dos pontos importantes é o desenvolvimento de formas de redução no tempo de execução. O tempo relativamente alto de execução tem origem no algoritmo de construção de solução inicial, algoritmo 6, pois sua implementação retorna solução apenas quando todas restrições são satisfeitas e uma solução válida é encontrada, sendo a tarefa difícil e que pode tomar mais tempo, conforme Nascimento et al. (2008). De forma a minimizar esse efeito, em estudos futuros pretende-se aplicar um banco de soluções ou uma geração inicial por meio de um solver, utilizando a solução encontrada em um tempo determinado.

Considerando os resultados apresentados pelo solvers e, principalmente, o tempo de execução a utilização de uma Meta-heurística pode parecer desnecessária. Porém, podem existir instâncias cuja resolução exata não seja possível, como ocorreu. Ainda, conforme definido escopo no objetivo, a solução apresentada não considera a relação entre escalas (férias, folgas e sobreaviso) e, temporalmente, abarca apenas um mês, sendo ideal planejar anual ou semestralmente. A inserção desses fatores aumentaria muito a complexidade do modelo, inviabilizando, provavelmente, a solução por solvers. Tais ressalvas cabem para indicar melhorias futuras na implementação para o problema em tela.

Ainda, seria válida a comparação da heurística usada neste problema com outras técnicas já consolidadas, como por exemplo Simulated Annealing (SA), Tabu Search (TS), Ant Colony Optimization (ACO), Cultural Algorithms (CA), Genetic Algorithms (GA) e Particle Swarm Optimization (PSO). Sendo assim, um bom número de trabalhos futuros pode ser desenvolvido para explorar o problema e ratificar sua importância para além do seara acadêmica.

9 Referências

- nº 9/2009, R., . Resolução. URL: <https://www.gov.br/depen/pt-br/composicao/cnpcp/resolucoes/2009/resolucao-no-9-de-13-de-novembro-de-2009.pdf/view>.
- Alvarenga, F., Rocha, M.L., 2006. Melhorando o desempenho da metaheurística grasp utilizando a técnica path-relinking: Uma aplicação para o problema da árvore geradora de custo mínimo com grupamentos. XXXVIII Simpósio Brasileiro de Pesquisa Operacional .
- Andrade, E.L., 2002. Introdução à pesquisa operacional: métodos e modelos para a análise de decisões. volume 3. LTC.
- Andrews, B., Parsons, H., 1993. Establishing telephone-agent staffing levels through economic optimization. *Interfaces* 23, 14–20.
- Arthur, J.L., Ravindran, A., 1981. A multiple objective nurse scheduling model. *AIIE transactions* 13, 55–60.
- Barboza, A.O., Carnieri, C., Steiner, M.T.A., Siqueira, P.H., 2003. Técnicas da pesquisa operacional no problema de horários de atendentes em centrais telefônicas. *Gestão & Produção* 10, 109–127.
- Barda, J.F., Binici, C., deSilva, A.H., 2003. Staff scheduling at the united states postal service. *Computers Operations Research* 30.
- Binato, S., Hery, W., Loewenstern, D., Resende, M.G., 2002. A grasp for job shop scheduling, in: *Essays and surveys in metaheuristics*. Springer, pp. 59–79.
- Brigandi, A.J., Dargon, D.R., Sheehan, M.J., Spencer III, T., 1994. At&t’s call processing simulator (caps) operational design for inbound call centers. *Interfaces* 24, 6–28.
- Brunner, J.O., Bard, J.F., Köhler, J.M., 2013. Bounded flexibility in days-on and days-off scheduling. *Naval Research Logistics (NRL)* 60, 678–701.
- Buffa, E.S., Cosgrove, M.J., Luce, B.J., 1976. An integrated work shift scheduling system. *Decision Sciences* 7, 620–630.
- Burke, E.K., Causmaecker, P.D., Berghe, G.V., Landeghem, H.V., 2004. The state of the art of nurse rostering. *Journal of Scheduling* 7.
- Butler, D., Maydell, U., 1979. Manpower scheduling in the edmonton police department. *INFOR: Information Systems and Operational Research* 17, 366–372.
- CELIC, S.D.A.C.D.L., . Edital 0040/2020. URL: https://www.compras.rs.gov.br/editais/0040_2020/261907.
- De Leone, R., Festa, P., Marchitto, E., 2011. A bus driver scheduling problem: a new mathematical model and a grasp approximate solution. *Journal of Heuristics* 17, 441–466.

- Desrosiers, J., Lasry, A., McInnis, D., Solomon, M.M., Soumis, F., 2000. Air transat uses altitude to manage its aircraft routing, crew pairing, and work assignment. *Interfaces* 30, 41–53.
- Devesse, V., 2016. Método de solução para o escalonamento de médicos. Master's thesis. Instituto de Ciências Matemáticas e de Computação ICMC/SP.
- Duckworth, W.E., 2012. A guide to operational research. Springer Science & Business Media.
- Edie, L.C., 1954. Traffic delays at toll booths. *Journal of the operations research society of America* 2, 107–138.
- Ernst, A., Jiang, H. and Krishnamoorthy, M., Sier, D., 2004. Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research* 153.
- Festa, P., Resende, M.G., 2002. Grasp: An annotated bibliography, in: *Essays and surveys in metaheuristics*. Springer, pp. 325–367.
- Festa, P., Resende, M.G., 2009a. An annotated bibliography of grasp—part i: Algorithms. *International Transactions in Operational Research* 16, 1–24.
- Festa, P., Resende, M.G., 2009b. An annotated bibliography of grasp—part ii: Applications. *International Transactions in Operational Research* 16, 131–172.
- Forrest, J., Lougee-Heimer, R., . Cbc user guide. URL: <https://www.coin-or.org/Cbc/ch01.html>.
- Gamache, M., Soumis, F., 1998. A method for optimally solving the rostering problem, in: *Operations research in the airline industry*. Springer, pp. 124–157.
- Gendreau, M., Potvin, J.Y., et al., 2010. *Handbook of metaheuristics*. volume 2. Springer.
- Goldbarg, M.C., Luna, H.P.L., 2005. *Otimização combinatória e programação linear: modelos e algoritmos*. volume 2. Elsevier.
- Henderson, S., Mason, A., Ziedins, I., Thomson, R., 1999. A heuristic for determining efficient staffing requirements for call centres. Technical Report. Citeseer.
- LEI COMPLEMENTAR Nº 13.259, D.D.O.D., . Governo do estado do rio grande do sul. URL: http://www.al.rs.gov.br/legis/M010/M0100099.ASP?Hid_Tipo=TEXT0&Hid_TodasNormas=53264&hTexto=&Hid_IDNorma=53264.
- Maier-Rothe, C., Wolfe, H.B., 1973. Cyclical scheduling and allocation of nursing staff. *Socio-Economic Planning Sciences* 7, 471–487.
- Mattia, S. and Rossi, F.S.M.S.S., 2017. Staffing and scheduling flexible call centers by two-stage robust optimization .
- Miller, H.E., Pierskalla, W.P., Rath, G.J., 1976. Nurse scheduling using mathematical programming. *Operations Research* 24, 857–870.

- Mitchell, S., OSullivan, M., Dunning, I., 2011. Pulp: a linear programming toolkit for python. The University of Auckland, Auckland, New Zealand , 65.
- Nascimento, M.C., Resende, M., Toledo, F.M., 2008. Grasp with path-relinking for the multi-plant capacitated lot sizing problem. *European Journal of Operational Research* .
- Neto, T., Pedroso, J.P., 2003. Grasp for linear integer programming, in: *Metaheuristics: Computer decision-making*. Springer, pp. 545–573.
- Norby, R.B., Freund, L.E., Wagner, B., 1977. A nurse staffing system based upon assignment difficulty. *JONA: The Journal of Nursing Administration* 7, 2–24.
- de Oliveira Souza, R., da Costa, L.P.R., Gabriel, C.P.C., Gabriel Filho, L.R.A., 2009. Pesquisa operacional: Na tomada de decisões administrativa. *ETIC-ENCONTRO DE INICIAÇÃO CIENTÍFICA-ISSN 21-76-8498* 5.
- Osogami, T., Imai, H., 2000. Classification of various neighborhood operations for the nurse scheduling problem, in: *International Symposium on Algorithms and Computation*, Springer. pp. 72–83.
- Pereira, V., Barbosa, J.M., Gomes, H.C., 2012. Modelagem em duas fases para a programação de escalas de trabalho em uma prestadora de serviços: Estudo de caso consórcio rio cidadão .
- Petry, G., Braatz, J., Martinez, P., 2018. Perspectivas para as finanças públicas do RS no período de 2019 a 2025. *Textos para Discussão TE/RS 10*. Secretaria da Fazenda RS. Porto Alegre RS. URL: https://tesouro.fazenda.rs.gov.br/upload/1529092117_Texto_discussao_TE_10_GUILHERME_et_al_perspectivas_financas_2019-2025.pdf.
- Secretaria do Planejamento, G.e.G., 2018. Proposta da Lei Orçamentária de 2018 MENSAGEM. Mensagem à Assembléia Legislativa 1. Governo do Estado. Porto Alegre RS. URL: <https://planejamento.rs.gov.br/upload/arquivos/201709/15103858-mensagem.pdf>.
- Raidl, G.R., Puchinger, J., Blum, C., 2019. Metaheuristic hybrids, in: *Handbook of metaheuristics*. Springer, pp. 385–417.
- Rangel, A.L., Évora, Y.D.M., 2007. Elaboração automática da escala periódica de trabalho dos profissionais de enfermagem por meio de um software específico. *Revista Eletrônica de Enfermagem* 9.
- Resende, M.G., Ribeiro, C.C., 2016. *Optimization by GRASP*. Springer.
- Roy, J., Mitchell, S., . Project description. URL: <https://pypi.org/project/PuLP/>.
- Ryan, D.M., 1992. The solution of massive generalized set partitioning problems in aircrew rostering. *Journal of the operational research society* 43, 459–467.
- Ryan, T., Barker, B.L., MARCIANTE, F.A., 1975. System for determining appropriate nurse staffing. *JONA: The Journal of Nursing Administration* 5, 30–38.

- Sandoya, F., Aceves, R., 2013. Grasp and path relinking to solve the problem of selecting efficient work teams. *Recent advances on meta-heuristics and their application to real scenarios* .
- Shiau, J.Y., Huang, M.K., Huang, C.Y., 2020. A hybrid personnel scheduling model for staff rostering problems. *Mathematics* 8, 1702.
- Silva, G.C.d., Ochi, L.S., Martins, S.L., 2006. Proposta e avaliação de heurísticas grasp para o problema da diversidade máxima. *Pesquisa Operacional* 26, 321–360.
- Silva, G.P., Souza, M.J.F., Júnior, A.d.C.G., 2005. Uma metodologia baseada em emparelhamentos sucessivos aplicada ao problema de programação de tripulações, in: *SBPO - Pesquisa Operacional e o Desenvolvimento Sustentável*, Gramado, Brasil. p. LGT : Logística e Transportes.
- Sinuany-Stern, Z., Teomi, Y., 1986. Multi-objective scheduling plans for security guards. *Journal of the Operational Research Society* 37, 67–77.
- Souza, D.S., Silva, G.P., . Um modelo exato para resolver o problema da escala de motoristas de ônibus urbano .
- Stein, G., Sulzbach, V.N., 2016. Dívida ativa do rio grande do sul. *Carta de Conjuntura FEE* 25.
- Talbi, E.G., 2009. *Metaheuristics: from design to implementation*. volume 74. John Wiley & Sons.
- Taylor, P.E., Huxley, S.J., 1989. A break from tradition for the san francisco police: Patrol officer scheduling using an optimization-based decision support system. *Interfaces* 19, 4–24.
- Thompson, G., 2003. Labor scheduling: A commentary. *Cornell Hotel and Restaurant Administration Quarterly* 44.
- Thompson, G.M., 1997. Labor staffing and scheduling models for controlling service levels. *Naval Research Logistics (NRL)* 44, 719–740.
- VisualRota, . Changing to new shift patterns. URL: <https://247rota.com/faq.htm>.
- Warner, D.M., 1976. Scheduling nursing personnel according to nursing preference: A mathematical programming approach. *Operations Research* 24, 842–856.
- Wedelin, D., 1994. An algorithm for 0–1 programming with application to airline crew scheduling, in: *European Symposium on Algorithms*, Springer. pp. 319–330.
- Wickert, T.I., 2019. *Personnel rostering: models and algorithms for scheduling, rescheduling and ensuring robustness*. Ph.D. thesis. Universidade Federal do Rio Grande do Sul, Instituto de Informática, Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS and KU Leuven, Arenberg Doctoral School, Faculty of Engineering Technology, Department of Computer Science, Gent, Belgium.
- Yanasse, H., Arenales, M., Morabito, R., Armentano, V., 2007. *Pesquisa Operacional para cursos de Engenharia*. volume 1. Elsevier.