

# Estudo de heurísticas e modelagem PLI para um problema de seleção e agendamento de portfólio de projetos

Acadêmico: Nicolás Herculano Pires

Orientador: prof. Dr. Cleber Valgas Gomes Mira

Área: Ciências Exatas e da Terra, Ciência da Computação

Subárea: Teoria da Computação, Análise de Algoritmos e Complexidade de Computação

# 1 Introdução

O Problema de Seleção e Agendamento de Portfólio de Projetos, do inglês *Project Portfolio Selection and Scheduling Problem* (PPSSP) [13, 11, 5, ?], consiste em dado um conjunto de projetos com custos, durações e outros parâmetros, encontrar o melhor portfólio de projetos, ou seja, um agendamento dos projetos ao longo de um horizonte de planejamento, que atenda a um certo conjunto de restrições (riscos, orçamentos, tempo, entre outros) de tal forma que algum critério seja otimizado. Este problema impacta principalmente organizações das áreas de desenvolvimento de software, finanças, planejamento militar e a área de pesquisa e desenvolvimento [5].

Na literatura, existem estratégias que buscam resolver este problema NP-difícil [5], por exemplo, é possível usar meta-heurísticas, como a Busca Tabu [4], Têmpera Simulada [3], Algoritmos Genéticos [12], GRASP [13], dentre outros. Ademais, outra estratégia utilizada para abordar este problema consiste em modelos de programação matemática como, por exemplo, a programação linear inteira [6].

A proposta deste trabalho tem como base o artigo de Mira et al. de 2015 [13] que apresenta um problema real de uma empresa geradora de energia elétrica no qual, dado um conjunto de projetos de manutenção que evitam riscos, deseja-se construir um agendamento de projetos, um portfólio, válido que otimize um critério de redução de risco da operação da empresa sem violar certas restrições. Mira e outros [13] utilizam a meta-heurística GRASP para abordar este problema; um algoritmo guloso que constrói uma solução inicial e, em seguida, realiza uma busca local para encontrar soluções do que a corrente em sua vizinhança. Posteriormente, Tanaka [15] utilizou o algoritmo BRKGA, um algoritmo genético com viés controlado [7, 10], para abordar o mesmo problema; contudo, o algoritmo GRASP encontrou melhores soluções em um tempo menor que o apresentado pelo BRKGA. Patrício [1] abordou uma versão simplificada do problema utilizando o algoritmo de colônia de formigas, que se inspira no comportamento de formigas para resolver problemas de otimização.

O objetivo deste trabalho é o de modelar o PPSSP como um problema de programação linear inteira e utilizar resolvidores de PLI, a fim de comparar os resultados desses resolvidores com o resultado da heurística GRASP apresentada no artigo [13]. Esta pesquisa deve explorar a qualidade das soluções de PPSSP geradas pelos resolvidores de problema PLI e pela heurística GRASP e tentar identificar qual é a melhor estratégia a ser adotada para certas instâncias do problema.

## 2 Revisão de Literatura

Nesta sessão, nós abordaremos a revisão de literatura sobre o problema principal desta pesquisa, o PPSSP, além de esclarecer as abordagens que serão estudadas, para resolver este problema. Ademais, nós iremos explicar sobre os métodos e algoritmos que serão utilizados para abordar este problema e detalhar sobre a implementação dos mesmos.

### 2.1 Conceito de PPSSP

O problema PPSSP [5, 13] é um desafio comum enfrentado por empresas e organizações que precisam decidir quais projetos devem ser incluídos em seu portfólio e em que ordem eles devem ser executados [5]. Em outras palavras, de maneira informal, encontrar uma solução para o PPSSP é como montar um quebra-cabeça com várias opções de projetos, cada um com seu próprio conjunto de requisitos, prazos e custos, com o objetivo de escolher e agendar os projetos de maneira a mais adequada para maximizar o valor total do portfólio e utilizar recursos de forma eficiente em um período determinado de tempo. Por exemplo, considere uma versão simplificada do problema encontrado em uma empresa geradora de energia elétrica [13], o qual precisa escolher quais projetos de manutenção deve realizar em seus equipamentos e usinas dentro de um horizonte de planejamento de 5 anos (60 meses), levando em consideração seus objetivos de produção de energia e restrições, como, por exemplo, o limite de recursos disponíveis, a necessidade de seguir as regulamentações governamentais, entre outras. Os especialistas da empresa levantam pontos de atenção que descrevem o potencial risco de haver falhas nos equipamentos da geração de energia. Considere que neste exemplo de uma versão simplificada do problema, existem apenas dois pontos de atenção,  $w_1$  e  $w_2$ , tais que o risco associado a  $w_1$  é 30 e o risco associado a  $w_2$  é 70. A empresa possui apenas 6 projetos de manutenção que ao ser finalizados resolvem os pontos de atenção anteriores e, logo, devem ser agendados para garantir o bom funcionamento da operação no período de 5 anos. Cada ponto de atenção é resolvido por um subconjunto dos projetos definidos no problema e observe também que um projeto pode contribuir para resolver mais de um ponto de atenção. Além disso, o risco associado ao ponto de atenção é dividido igualmente entre os projetos que devem resolvê-lo. Neste exemplo em particular, considere que os projetos  $G_{w_1} = \{p_1, p_4, p_5, p_6\}$  contribuem para resolver o risco associado ao ponto de atenção  $w_1$  e que os projetos  $G_{w_2} = \{p_2, p_3, p_4\}$  contribuem para resolver o risco de  $w_2$ ; logo cada projeto de  $G_{w_1}$  irá resolver 7,5 de risco do total de 30 associado ao ponto

de atenção  $w_1$  [13]. Observe, no entanto, que um projeto que contribui para resolver mais de um ponto de atenção acumula os valores de risco de cada ponto de atenção o qual resolve. Por exemplo, o projeto  $p_4$  contribui para resolver 7,5 de risco associado ao ponto de atenção  $w_1$  e 23,33 de risco associado ao ponto de atenção  $w_2$ , então o projeto  $p_4$  contribui para resolver um valor 30,83 de risco para todos os pontos de atenção desta instância. Além disso, no problema real encontrado no artigo [13], os custos de cada projeto são classificados em CAPEX (custo de capital) e OPEX (custo operacional), entretanto, para facilitar a compreensão do leitor, vamos assumir que todos os projetos deste exemplo possuem custos classificados como CAPEX. Cada projeto contará com uma lista de custos mensais, enquanto os recursos disponíveis pela empresa para cobrir os cursos são organizados em uma lista anual de recursos de cada categoria (CAPEX ou OPEX). A distinção entre listas de custos mensais e de recursos anuais é relevante, pois um projeto pode, por exemplo, ser agendado para iniciar em um ano e ter a sua execução finalizada apenas no ano seguinte, o que resulta na utilização parcial de recursos de ambos os anos. Neste exemplo em particular, nós devemos assumir que temos um montante de R\$135.000 em CAPEX por ano do horizonte de planejamento.

A Tabela 1 a seguir informa o tamanho do horizonte de planejamento (HP), a lista de projetos na instância de entrada do problema, a duração em meses de cada projeto, o montante de risco que cada projeto pode contribuir para resolver em todos os pontos de atenção, o custo de cada projeto e o custo total de cada projeto.

Projetos	Duração (meses)	Risco	Custo mensal	Custo total
$p_1$	9	7,5	14.444,44	130.000,00
$p_2$	11	23,33	11.818,81	130.000,00
$p_3$	10	23,33	13.000,00	130.000,00
$p_4$	16	30,83	9.375,00	150.000,00
$p_5$	4	7,5	32.500,00	130.000,00
$p_6$	10	7,5	13.000,00	130.000,00

Tabela 1: Exemplo de lista de projetos

As restrições são as limitações que devem ser satisfeitas para que a solução seja considerada viável. No exemplo que estamos apresentando nessa proposta, as restrições se assemelham com as restrições demonstradas no artigo [13], que por sua vez, são apresentadas como:

1. Agendamento consistente: Um projeto não pode ser agendado para iniciar em dois meses diferentes.
2. Restrição de recursos: O custo total dos projetos executados em um certo ano não deve exceder o orçamento anual
3. Prazo limite de resolução de um ponto de atenção: Um ponto de atenção de alto risco é classificado como intolerável. Isso significa que todos os projetos que controlam um ponto de atenção intolerável devem ser completados antes do seu prazo limite.
4. Restrição de não simultaneidade: Alguns projetos não podem ser executados de maneira simultânea. Por exemplo, suponha que os projetos  $p_2$  e  $p_3$  são projetos de manutenção que atuam em geradores distintos de uma mesma usina. Além disso, a empresa possui 4 geradores apenas nesta usina e só pode desligar um por vez para produzir o mínimo de energia regulamentado pelo governo. Logo os projetos  $p_2$  e  $p_3$  não podem ser executados simultaneamente, pois, caso contrário, ao menos 2 geradores seriam interrompidos.
5. Restrição de interdependência: Alguns projetos, necessitam de outros estarem prontos antes de serem iniciados. Por exemplo, o projeto  $p_4$ , o qual só poderá ser agendado e iniciado após a conclusão dos projetos  $p_2$  e  $p_3$ , pois necessita da conclusão dos riscos atribuído aos geradores.

O conjunto dos projetos associados aos meses de início de sua execução ao longo de um horizonte de planejamento é chamado de *portfólio de projetos* e este é *válido* quando as restrições do problema não são violadas.

A empresa deve escolher um portfólio de projetos que agende o máximo de projetos o mais cedo possível, levando em consideração o horizonte de planejamento de 60 meses. Na Seção 2.1.1 nós iremos apresentar uma função matemática para a versão aqui estudada do PPSSP que toma um portfólio válido como entrada e retorna um valor que deve ser maximizado e que captura o critério anterior de agendar projetos o mais cedo possível com o intuito de resolver o quanto antes os riscos associados aos pontos de atenção.

Um portfólio de projetos que maximize o valor de retorno é considerado uma solução ótima para o problema, porém, encontrar uma solução ótima para o PPSSP é uma tarefa complexa, pois o PPSSP é um problema NP-difícil [5], portanto, não existe um algoritmo conhecido capaz de resolvê-lo em tempo polinomial [6]. Todavia, existem técnicas para tentarmos encontrar soluções ótimas ou de qualidade razoável. Por exemplo, nós podemos utilizar meta-heurísticas como Busca Tabu [4], Têmpera Simulada [3], Algoritmos Genéticos [12], GRASP [13], dentre outros. Além disso, neste trabalho

nós iremos utilizar a estratégia de modelar o PPSSP como um problema de programação linear inteira e avaliar se programas resolvidores de PLI conseguem encontrar soluções razoáveis, ou mesmo ótimas, para uma versão relaxada do problema.

### 2.1.1 Formalização do problema

A princípio, para a formalização do problema, vamos estudar a função objetivo, a qual se trata de uma expressão matemática que define a medida de desempenho desejada e é utilizada como base para a tomada de decisão [6]. A função objetivo é um elemento fundamental na resolução de problemas de otimização e é uma fórmula algébrica composta por variáveis e parâmetros relacionados ao critério que se deseja otimizar. O domínio da função objetivo é definido pelo conjunto de soluções viáveis, em nosso caso, este é o conjunto de portfólios que satisfazem as restrições do problema PPSSP. As variáveis representam a escolha das soluções viáveis possíveis e os parâmetros são valores conhecidos no momento da formulação do problema.

Considere a versão particular do problema PPSSP [13] que deveremos estudar neste trabalho. Podemos formalizar o problema usando a notação Tabela 2.1.1.

$I$	Conjunto de projetos em uma instância de PPSSP
$P$	Portfólio de projetos
$I_P$	Conjunto dos projetos em um portfólio $P$
$W$	Conjunto dos pontos de atenção
$w$	Ponto de atenção
$G_w$	Conjunto dos projetos que resolvem um ponto de atenção $w$
$R$	Risco
$R_w$	Risco associado ao ponto de atenção $w$
$[n]$	Intervalo de 1, 2, ..., $n$
$p$	Índice do projeto
$m_p$	Mês de início de um projeto $p$
$d_p$	Duração em meses de um projeto $p$
$T$	Horizonte de planejamento em meses

Tabela de Notações

Uma função objetivo para este problema deve capturar a ideia de medir o valor do risco resolvido o mais cedo possível, em outros termos, queremos encontrar o valor do risco multiplicado pelo intervalo de tempo no qual este risco está resolvido ao longo do horizonte de planejamento para um dado portfólio de projetos.

Nossa função objetivo neste exemplo, foi inspirada na mesma função apresentada no problema [13], a qual foi desenvolvida pensando na redução de riscos, sabendo que respectivamente  $p$  simboliza o índice do projeto, podemos assumir que:

$$\max(2T \sum_{w \in W} R_w - \sum_{w \in W} R_w (2T - \max_{p \in G_w} (m_p + d_p + 1))) \quad (1)$$

Sujeito as restrições do problemas, as quais também podem ser formalizadas. Todavia, para fins de consulta da formalização de restrições do problema, basta seguir as

### 2.1.2 Programação Linear Inteira

O problema de programação linear é o problema de minimizar ou maximizar uma função linear sujeita a um conjunto finito de restrições lineares [6]. Por outro lado, a programação linear inteira (PLI) é uma extensão da programação linear, em que algumas ou todas as variáveis devem ser inteiras [2, 14].

Neste trabalho deveremos estudar uma maneira de modelar o problema PPSSP como um PLI e realizar experimentos computacionais para encontrar a qualidade das soluções por resolvedores.

#### 2.1.2.1 Relaxação da PLI

A relaxação da PLI é um processo em que as variáveis deixam de assumir valores inteiros no problema, ou seja, as variáveis podem receber valores reais. Isso permite que os algoritmos resolvedores de programação linear sejam utilizados para encontrar uma solução ótima ou aproximada. Em seguida, essa solução encontrada pela relaxação é convertida em uma solução inteira e é aproximada utilizando meios de arredondamento.

A técnica de relaxação para encontrar soluções de problemas PLI, pode encontrar soluções ótimas ou aproximadas. Contudo, vale ressaltar que a solução encontrada ou seu arredondamento pode não ser ótima em relação ao problema PLI original [6]. De fato, podem haver instâncias do problema para as quais a solução encontrada usando a relaxação pode ter uma qualidade muito inferior ao de soluções ótimas.

### 2.1.3 Método GRASP

O GRASP (*Generalized Randomized Adaptive Search Procedures*) é uma heurística que é utilizada para resolver problemas de otimização combinatória [8,

9, 13]. A heurística GRASP é baseada em duas etapas: a etapa de construção de uma solução inicial e a etapa de uma busca local.

A etapa de solução inicial é realizada de maneira aleatória e gulosa, mas com a utilização de critérios de seleção baseados em uma função de seleção, a qual é utilizada para escolher os elementos que serão adicionados à solução inicial.

A etapa de busca local é realizada a partir da solução inicial, explorando vizinhanças de soluções correntes para encontrar ou construir soluções melhores.

## **2.2 Implementação do processo para a otimização do PPSSP utilizando GLPK**

Nós utilizaremos o resolvidor GLPK (*GNU Linear Programming Kit*) [?] para resolver o problema PPSSP. O GLPK é uma biblioteca de software livre e de código aberto que soluciona problemas de programação linear. A linguagem de programação do resolvidor GLPK é a linguagem C.

O GLPK utiliza o método simplex e o método primal-dual para problemas lineares, bem como, o algoritmo *branch-and-bound* com cortes de Gomory para problemas inteiro-mistos [16]. O resolvidor GLPK é adotado como ferramenta para resolução de problemas de otimização em várias indústrias e na academia, como, por exemplo, [exemplos...]. Devido a sua ampla utilização e facilidade de uso, nós decidimos utilizar o GLPK neste trabalho.

## **3 Objetivos Gerais**

O objetivo principal deste trabalho é realizar um estudo de uma versão de PPSSP e estudo da técnica de programação linear inteira. Nós iremos modelar o PPSSP como um problema PLI, utilizar resolvidores e trabalhar em soluções de pequenas instâncias. Além disso, iremos fazer uma relaxação PL do problema e a implementação da heurística GRASP para comparar os resultados das soluções obtidas.

### **3.1 Objetivos específicos**

1. Estudar programação linear inteira
2. Estudar e modelar o PPSSP.



3. Utilizar resolvedores de programação linear em experimentos computacionais. Obter resultados de soluções ótimas de instâncias pequenas através dos resolvedores.
4. Estudar e comparar os resultados obtidos contra as melhores soluções encontradas pelo GRASP.
5. Estudar uma relaxação linear do problema.
6. Analisar e determinar os melhores resultados dentre os métodos utilizados.

## 4 Cronograma de atividades

1. Estudar o Problema de Seleção e Agendamento de Portfólio de Projetos (PPSSP).
2. Revisar meta heurísticas.
3. Estudar programação linear inteira (PLI).
4. Estudar linguagem C para aplicação dos resolvedores.
5. Estudar a utilização de resolvedores de problemas de PLI.
6. Modelar o problema PPSSP como um problema PLI.
7. Utilizar os resolvedores para encontrar soluções para o modelo PLI.
8. Realizar os experimentos e análises dos resultados.
9. Escrever relatórios e artigos acadêmicos.

## 5 Metodologia

Para conseguir atingir os objetivos deste trabalho, serão realizadas atividades de pesquisa, análise e desenvolvimento.

A princípio, será feita uma revisão de literatura sobre um problema de otimização que consiste na seleção e agendamento de portfólio de projetos. Além disso, do mesmo modo estudaremos a heurística GRASP, por meio de artigos acadêmicos e trabalhos de conclusão de curso. Também nós estudaremos a programação linear inteira e programação em C por meio de apostilas, livros

	2023					2024					
	ago	set	out	nov	dez	jan	fev	mar	abr	jun	jul
(1)	x	x	x								
(2)		x	x								
(3)			x	x							
(4)				x	x						
(5)					x	x					
(6)					x	x					
(7)						x	x	x	x	x	
(8)	x	x	x	x	x	x	x	x	x	x	x

Tabela 2: Cronograma de atividades

e artigos científicos a fim de resolver o problema de otimização estudado neste trabalho. O software resolvidor de programação linear inteira adotado neste trabalho, será o GLPK (GNU Linear Programming Kit).

Para o desenvolvimento desta iniciação científica, serão feitas reuniões semanais com o orientador, a fim de acompanhar a progressão do trabalho, discutir sobre tópicos e metas, além de resolução de dúvidas. Ademais, nós faremos em reuniões o acompanhamento da revisão de literatura, revisão dos textos que serão produzidos e a análise de resultados de experimentos e testes.

## 6 Resultados Esperados

A produção esperada deste trabalho, além de amplificar o conhecimento acerca do problema de seleção e agendamento de portfólio, serão:

1. Obter resultados ótimos em instâncias pequenas.
2. Concluir e expor sobre os métodos utilizados .

## Referências

- [1]
- [2] Rui Alves and Catarina Delgado. Programação linear inteira. 1997.
- [3] Dimitris Bertsimas and John Tsitsiklis. Simulated annealing. *Statistical science*, 8(1):10–15, 1993.

- [4] Aurora da Conceição João Cameirão. Híbridização de algoritmos genéticos e procura tabu para o problema de job-shop scheduling. 2009.
- [5] Ana F Carazo, Trinidad Gómez, Julián Molina, Alfredo G Hernández-Díaz, Flor M Guerrero, and Rafael Caballero. Solving a comprehensive model for multiobjective project portfolio selection. *Computers & operations research*, 37(4):630–639, 2010.
- [6] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edition, 2009.
- [7] Jorge José de Magalhães Mendes, José Fernando Gonçalves, and Mauricio G. C. Resende. A random key based genetic algorithm for the resource constrained project scheduling problem. *Computers & OR*, 36(1):92–109, 2009.
- [8] Thomas A Feo and Mauricio GC Resende. Greedy randomized adaptive search procedures. *Journal of global optimization*, 6(2):109–133, 1995.
- [9] Paola Festa and Mauricio GC Resende. GRASP: An annotated bibliography. In *Essays and surveys in metaheuristics*, pages 325–367. Springer, 2002.
- [10] José Fernando Gonçalves and Mauricio GC Resende. An evolutionary algorithm for manufacturing cell formation. *Computers & industrial engineering*, 47(2-3):247–273, 2004.
- [11] Kyle Harrison, Saber Elsayed, Terence Weir, Ivan Garanovich, Richard Taylor, and Ruhul Sarker. An exploration of meta-heuristic approaches for the project portfolio selection and scheduling problem in a defence context. 12 2020.
- [12] K-M Lee, Takeshi Yamakawa, and Keon-Myung Lee. A genetic algorithm for general machine scheduling problems. In *1998 Second International Conference. Knowledge-Based Intelligent Electronic Systems. Proceedings KES’98 (Cat. No. 98EX111)*, volume 2, pages 60–66. IEEE, 1998.
- [13] Cleber Mira, Paulo R. Viadanna, Maria Ângélica Souza, Arnaldo Moura, João Meidanis, Gabriel A. Costa Lima, and Renato P. Bossolan. Project scheduling optimization in electrical power utilities. pages 1–26, 2015.
- [14] Darci Prado. *Programação linear*, volume 1. Falconi Editora, 2016.

- [15] Bruno Yoichi Tanaka. Aplicação dos métodos grasp e brkga para um problema de otimização de portfólios. page 54. Universidade Estadual de Mato Grosso do Sul, 11 2018.
- [16] Daniela Arai yamanaka. Um estudo sobre o problema de portfolio de ações com custos fixos de transação. page 31. Instituto de Ciências Matemáticas e de Computação, 10 2008.