



Guilherme Angelo Leite

**Desenvolvimento de sistema de agendamento
de serviços de manutenção de plataformas com
alocação de funcionários**

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Engenharia Elétrica do Departamento de Engenharia Elétrica da PUC-Rio.

Orientador: Profa. Marley Maria Bernardes Rebuzzi Vellasco

Rio de Janeiro
Setembro de 2019



Guilherme Angelo Leite

**Desenvolvimento de sistema de agendamento
de serviços de manutenção de plataformas com
alocação de funcionários**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Engenharia Elétrica do Departamento de Engenharia Elétrica da PUC-Rio. Aprovada pela Comissão Examinadora abaixo.

Profa. Marley Maria Bernardes Rebuzzi Vellasco

Orientador

Departamento de Engenharia Elétrica – PUC-Rio

Prof. Ricardo Tanscheit

Departamento de Engenharia Elétrica – PUC-RIO

Dra. Manoela Rabello Kohler

Departamento de Engenharia Elétrica – PUC-RIO

Prof. Douglas Mota Dias

Universidade Estadual do Rio de Janeiro – UERJ

Rio de Janeiro, 16 de Setembro de 2019

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Guilherme Angelo Leite

Graduou-se em Engenharia Eletrônica pelo Instituto Militar de Engenharia (IME) em 2014. Em 2015 entrou na iniciativa privada, onde contribui com o desenvolvimento de projetos de pesquisa e desenvolvimento na área de monitoramento de dutos flexíveis.

Ficha Catalográfica

Leite, Guilherme Angelo

Desenvolvimento de sistema de agendamento de serviços de manutenção de plataformas com alocação de funcionários / Guilherme Angelo Leite; orientador: Marley Maria Bernardes Rebuzzi Vellasco . – Rio de Janeiro: PUC-Rio, Departamento de Engenharia Elétrica , 2019.

v., 69 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica .

Inclui bibliografia

1. Engenharia Elétrica – Teses. 2. Modelos de Apoio a Decisão – Teses. 3. Otimização por Enxame de Partículas;. 4. Otimização;. 5. Problema de Ordem;. 6. Manutenção Embarcada;. 7. PSO Combinatorial.. I. Vellasco , Marley Maria Bernardes Rebuzzi. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica . III. Título.

CDD: 621.3

Agradecimentos

À minha orientadora, Professora Marley Vellasco, pelo estímulo e paciência durante a realização deste trabalho.

À minha família e amigos, pelo apoio e incentivo a iniciar e continuar no curso mesmo frente a adversidades, em especial à Camila por me motivar ao longo desse árduo caminho.

À PUC-Rio, pelo auxílio e estrutura concedidos para minha formação.

Aos demais professores do Departamento de Engenharia Elétrica, particularmente, aqueles com os quais tive o prazer de conviver durante os cursos do Programa de Pós-graduação, pela dedicação e competência.

Aos funcionários do Departamento de Engenharia Elétrica, pela gentileza e atenção.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001

Resumo

Leite, Guilherme Angelo; Vellasco, Marley Maria Bernardes Rebuzzi. **Desenvolvimento de sistema de agendamento de serviços de manutenção de plataformas com alocação de funcionários**. Rio de Janeiro, 2019. 69p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Com o objetivo de desenvolver um sistema de apoio à decisão na área de manutenção embarcada, este trabalho apresenta um modelo para problemas de ordem com restrições: CPSO+. Este modelo é a combinação de dois modelos da literatura, o PSO+, que apresenta bons resultados em problemas com restrições, e o CPSO, que introduz as modificações necessárias para aplicar o PSO em problemas de ordem. O modelo proposto foi adaptado para resolver o complexo problema de definir a melhor sequência de atividades embarcadas e funcionários alocados, de forma a maximizar o lucro da prestadora de serviço no período de três meses respeitando todas as restrições de prazo de conclusão dos serviços e restrições específicas do segmento *offshore*. Para avaliar o desempenho deste novo modelo na resolução do problema proposto, duas variantes do CPSO+ foram avaliadas frente ao modelo da literatura, CPSO, em seis casos de simulação propostos. Conclui-se pelos resultados das simulações que o modelo CPSO+ com inicialização reduzida destaca-se dos demais avaliados por apresentar um tempo de execução moderado e com soluções melhores que as dos demais.

Palavras-chave

Otimização por Enxame de Partículas; Otimização; Problema de Ordem; Manutenção Embarcada; PSO Combinatorial.

Abstract

Leite, Guilherme Angelo; Vellasco , Marley Maria Bernardes Rebuzzi (Advisor). **Development of offshore maintenance service scheduling system with workers allocation..** Rio de Janeiro, 2019. 69p. Dissertação de mestrado – Departamento de Engenharia Elétrica , Pontifícia Universidade Católica do Rio de Janeiro.

In order to develop an offshore maintenance support system, this work presents a model for constrained combinatorial problems: CPSO+. This model is a combination of two models, the PSO+, which presented good results in problems with constraints, and the CPSO, which is an adaptation of PSO for application in combinatorial problems. The proposed model has been adapted to solve the complex problem of defining the best sequence of offshore activities and allocated staff so as to maximize service provider profitability within three months while respecting all service completion time constraints and specific offshore work constraints. To evaluate the performance of this new model in solving the proposed problem, two CPSO+ variants were evaluated against the literature model, CPSO, in six proposed simulation cases. It is concluded from the results of the simulations that the CPSO+ model with reduced initialization outperforms other evaluated models with respect to execution time and solutions to given problem.

Keywords

Particle Swarm Optimization; Optimization; Combinatorial Problem; Offshore Maintenance; Combinatorial PSO.

Sumário

1	Introdução	11
1.1	Motivação	11
1.2	Objetivo	12
1.3	Organização da Dissertação	13
2	Planejamento Flexível de Serviços de Manutenção Embarcada	14
2.1	Problema de Planejamento Flexível	14
2.1.1	NP-Difícil	15
2.1.2	Meta-Heurística	15
2.1.3	Planejamento Ativo	16
2.1.4	Problemas de ordem	17
2.1.5	Problema de Planejamento Flexível de Atividades	18
2.2	Serviços de Manutenção Embarcados e Suas Restrições	19
2.2.1	Restrições de Trabalhos Embarcados	20
2.2.2	Metas do planejamento de serviços de manutenção embarcados	21
3	Otimização em Problemas de Planejamento	23
3.1	Otimização	23
3.1.1	Função objetivo	24
3.1.2	Espaço de busca	25
3.1.3	Restrições	25
3.2	Técnicas para abordar restrições	26
3.2.1	Penalização	26
3.2.2	Multiobjetivo	27
3.2.3	Gerador de Soluções Válidas	27
3.3	Particle Swarm Optimization (PSO)	27
3.3.1	PSO Clássico	28
3.3.2	PSO com Inércia	29
3.3.3	CPSO	29
3.3.4	PSO+	30
3.3.5	Topologias de PSO	31
3.3.5.1	Vizinhança dinâmica	32
3.3.5.2	Vizinhança Estática	33
4	CPSO+ para Planejamento de Serviços Embarcados	35
4.1	Atualização de posição no CPSO+	36
4.2	CPSO+ com inicialização modificada	37
4.2.1	Modificação da inicialização dos enxames de referência e apoio	37
4.3	Funções Objetivo	39
4.3.1	Função objetivo CPSO+	39
4.3.2	função objetivo CPSO penalização	40
4.4	Premissas do modelo matemático	40
4.5	Modelagem	41
4.5.1	Representação da ordem das tarefas	42

4.5.2	Representação do recurso selecionado	43
4.5.3	Decodificação do vetor de ordem em planejamento ativo	44
4.5.4	Outras Restrições	45
5	Estudo de Caso	48
5.1	Introdução	48
5.2	Base de Dados	49
5.3	Casos de simulação	50
5.3.1	Simulação 01: inicialização da partícula	50
5.3.2	Simulação 02: Tempo de montagem do cronograma	51
5.3.3	Simulações para comparação entre modelos	53
5.4	Resultados	56
5.4.1	Simulação 03: espaço de busca 100% inválido	56
5.4.2	Simulação 04: espaço de busca 100% válido	58
5.4.3	Simulação 05: espaço de busca predominantemente válido	59
5.4.4	Simulação 06: espaço de busca predominantemente inválido	60
5.5	Análise comparativa dos resultados	61
6	Conclusão e Trabalhos Futuros	64
6.1	Conclusões	64
6.2	Trabalhos Futuros	64
	Referências	66

Lista de figuras

Figura 2.1	Fluxograma de construção do gráfico de Gantt	16
Figura 2.2	Fluxograma de construção do gráfico de Gantt modificado	17
Figura 2.3	Exemplo de representação por prioridade	17
Figura 2.4	Exemplo de representação por combinação	18
Figura 2.5	Exemplo de representação de seleção de recurso	19
Figura 3.1	Representação gráfica do ótimo em região de fronteira.	24
Figura 3.2	Esquemático com o fluxo do PSO	28
Figura 3.3	Ilustração da representação do espaço de busca do PSO Combinatorial	29
Figura 3.4	Exemplo de atualização do PSO Combinatorial	30
Figura 3.5	Exemplo de topologia dinâmica: a) mudança de estrutura; b) mudança de vizinhos	32
Figura 3.6	Exemplo de topologias estáticas; a) topologia toda interconectada; b) topologia em anel; c) topologia agrupamento aleatório	33
Figura 3.7	Ilustração da topologia aleatória coordenada	34
Figura 4.1	Elementos do CPSO+	35
Figura 4.2	Exemplo de atualização do CPSO+	36
Figura 4.3	Fluxograma do decodificador do vetor de ordem	46
Figura 4.4	Fluxograma de montagem do cronograma	47
Figura 5.1	Boxplot das simulações 100% inválidos	57
Figura 5.2	Boxplot das simulações 100% válido	58
Figura 5.3	Boxplot da simulação predominantemente valida	60
Figura 5.4	Boxplot da simulação predominantemente invalida	61

Lista de tabelas

Tabela 5.1	Casos de simulação do processo de geração de partículas de ordem	50
Tabela 5.2	Tempo das simulações de geração de partículas de ordem	51
Tabela 5.3	Resultados da diversidade da primeira posição modelo original	51
Tabela 5.4	Resultados da diversidade da primeira posição modelo novo	51
Tabela 5.5	Casos de simulação da criação de cronograma	52
Tabela 5.6	Tempo médio da criação de cronograma	52
Tabela 5.7	Descrição das atividades do bloco andaime	55
Tabela 5.8	Descrição das atividades do bloco solda	55
Tabela 5.9	Descrição das atividades do bloco pintura	55
Tabela 5.10	Descrição dos serviços simulados	56
Tabela 5.11	Resumo dos parâmetros fixos das simulações	56
Tabela 5.12	Resultados das simulações 100% inválido	57
Tabela 5.13	Resultados das simulações 100% válido	58
Tabela 5.14	Resultados das simulações predominantemente valida	59
Tabela 5.15	Resultados das simulações predominantemente invalida	60
Tabela 5.16	Resultados do tempo de inicialização dos casos analisados	61
Tabela 5.17	Resultados do tempo de busca dos casos analisados	62
Tabela 5.18	Resultados médios dos casos analisados	62

1

Introdução

1.1

Motivação

No Brasil, um dos segmentos mais relevantes para a economia é o setor de óleo e gás. Este setor, que já chegou a representar 13% do PIB nacional [32], enfrenta um grande desafio: atualizar seus arcaicos processos por técnicas modernas e atuais a fim de otimizar o uso dos recursos disponíveis utilizando técnicas de análise de dados [31]. Dentre as diversas subdivisões do setor, a principal área de apoio à extração do petróleo é a de manutenção embarcada, responsável por garantir que todos os equipamentos e instalações estejam em condições para operar corretamente. Nesta área, o processo mais desatualizado e, portanto, com maior impacto, é o problema de planejar o embarque de funcionários para a manutenção de plataformas e embarcações que atualmente carece de ferramentas específicas para suas restrições.

O problema em questão é classificado na literatura como Problema de Planejamento Flexível, no qual se entende por Planejamento um problema de otimização de alocação de máquinas e recursos a tarefas, segundo restrições conhecidas [26]. Atualmente na área de manutenção embarcada, muitas empresas ainda operam sem o apoio de softwares, otimizando a alocação dos funcionários por meio de especialistas. Outras utilizam-se de softwares genéricos de planejamento que não possuem uma consideração específica para as restrições do setor, gerando soluções não válidas que devem então ser ajustadas por especialistas para soluções aplicáveis. Apesar de conseguirem operar por meio destes métodos, o processo pode, em casos mais complexos, demandar até uma semana de trabalho e está muito vulnerável a erros humanos. Dessa forma, no processo atual, não apenas a qualidade da solução encontrada pode ser ruim como algumas restrições podem ser avaliadas de forma errada e levar a uma solução inválida.

Devido à falta de referência na literatura de técnicas abordando especificamente as restrições da área, tomou-se como base problemas da literatura com restrições similares em outros segmentos da indústria. Destaca-se entre eles o problema de processamento de dados em vários processadores [27], a es-

colha de tráfego de dados em uma rede com limitação de banda [3] e trabalhos abordando a modelagem de problemas de planejamento [25], [17], [14]. Nestes trabalhos, observa-se que técnicas heurísticas como o PSO têm apresentado um bom desempenho tanto em tempo de processamento quanto em qualidade do ótimo encontrado para problemas de grande dimensionalidade. Contudo, a rápida evolução dos modelos não ocorre igualmente em todos os ramos de aplicação. Por isso, variantes mais eficientes do PSO, como o PSO+ [22], ainda não possuem uma descrição detalhada para aplicações em problemas de planejamento e precisam de pequenas adaptações para que este possa lidar com problemas de planejamento.

Este trabalho busca, então, apoiar a importante indústria de óleo e gás no específico problema de planejamento de manutenções embarcadas através de uma técnica de otimização baseada no PSO+ e no CPSO [39] considerando as restrições específicas da área.

1.2 Objetivo

O objetivo deste trabalho é desenvolver um modelo de otimização para problemas de planejamento flexíveis com restrições específicas da área de manutenção embarcada.

Para isso, este trabalho baseia-se em um modelo de tempo de processamento reduzido e bons resultados em problemas com restrições, o PSO+. Contudo, para a aplicação deste modelo no problema de planejamento, é necessário modificá-lo para lidar com problemas de ordem. Para essa modificação, inspirou-se no modelo CPSO, que apresentou bons resultados para problemas de ordem sem restrição. Além disso, inclui-se no modelo adaptado, CPSO+, etapas para avaliar as restrições específicas do setor:

- Tempo máximo para um funcionário permanecer embarcado;
- Tempo de descanso regulamentado;
- Limite de funcionários a bordo;
- Atividades pré requisito para embarque;
- Prazo para conclusão dos serviços.

1.3

Organização da Dissertação

O resto desta dissertação está organizada em 5 capítulos.

O Capítulo 2 aborda os principais conceitos relacionados ao planejamento flexível e ao trabalho embarcado, descrevendo suas principais restrições e objetivos.

Em seguida, no Capítulo 3, apresenta-se uma breve descrição dos algoritmos e métodos já existentes na literatura que serviram de base para as soluções implementadas neste trabalho para resolver o problema de agendamento de serviços de manutenção de plataformas com alocação de funcionários.

O Capítulo 4 descreve quais os principais desafios dos modelos encontrados na literatura para sua aplicação no problema em análise e as modificações propostas neste trabalho para encontrar o melhor agendamento segundo as condições específicas.

O Capítulo 5 apresenta as simulações realizadas, os resultados obtidos e uma breve comparação entre as diferentes variantes do modelo CPSO+ avaliadas.

O Capítulo 6 apresenta as conclusões deste trabalho e as sugestões para trabalhos futuros.

2

Planejamento Flexível de Serviços de Manutenção Embarcada

Este capítulo aborda a definição e principais características de um planejamento flexível. Essas definições servem de base para o módulo do método proposto que converte a sequência das atividades em um planejamento válido.

Além disso, este capítulo detalha as restrições e as principais metas dos serviços de manutenção embarcada. As restrições deste serviço auxiliaram na construção das restrições avaliadas pelo modelo desenvolvido enquanto as metas ajudam a esclarecer a função objetivo construída para o problema abordado.

2.1

Problema de Planejamento Flexível

Na literatura, entende-se por Planejamento um problema de otimização de alocação de recursos a tarefas segundo restrições conhecidas [26]. Em sua forma mais simples, o problema de Planejamento pode ser decomposto em duas etapas: a definição da melhor sequência de atividades e a transformação desta sequência em um cronograma. A primeira etapa é nomeada na literatura como problema de Ordem, enquanto a segunda de Planejamento Ativo.

Entretanto, o problema de Planejamento pode exigir além da melhor sequência outras considerações. As variantes mais relevantes para este trabalho são: alocação de recursos em atividades, conhecido como Planejamento Flexível de Projetos [34, 18], e compartilhamento de recursos em Planejamento de Projetos Paralelos [10, 11]

É importante ressaltar também que os Problemas de Planejamento são problemas do tipo NP-difícil, como descrito na Seção 2.1.1. Por isso, quando possuem um espaço de busca de alta dimensão e muitas restrições, são custosos de serem resolvidos por técnicas determinísticas como Programação linear [15] e aconselha-se o uso de técnicas Meta-Heurísticas, como descrito na Seção 2.1.2.

2.1.1

NP-Difícil

Essa categoria de problemas matemáticos é caracterizada por não possuir solução em tempo polinomial. Dessa forma, a dificuldade de solucionar esses problemas por métodos exaustivos cresce com o número de parâmetros de forma desproporcional. Esse crescimento do tempo de solução exata torna inviável a aplicação destes métodos em problemas com um espaço de busca com alta dimensão e muitas regras [13].

Para essa categoria de problemas, metodologias que exploram o espaço de busca de forma semi-aleatória e que usufruem do conhecimento adquirido para melhor se direcionarem, também denominadas metodologias Meta-Heurísticas, apresentam melhores resultados [38].

2.1.2

Meta-Heurística

Heurísticas são métodos definidos como estratégias que ignoram parte da informação do problema com o objetivo de tornar a busca mais fácil e rápida. As chamadas Meta-Heurísticas são metodologias que, apesar de não determinísticas e de ignorarem parte da informação, utilizam o conhecimento adquirido em passos anteriores para guiar racionalmente as próximas iterações. Dessa forma, as Meta-Heurísticas não garantem matematicamente o atingimento do ótimo global, mas se aproximam deste através de estratégias de convergência [4].

Algumas das meta-heurísticas mais conhecidas são:

- Algoritmo Genético (GA) [37],
- Evolução Diferencial [36],
- Colônia de Formigas [33],
- Colônia de Abelhas [40], e
- Enxame de Partículas (PSO) [23]

Vale ressaltar que, frente às diferentes técnicas, os Algoritmos Genéticos e o Enxame de Partículas apresentam bons resultados de convergência para o problema de planejamento e são a base de alguns estudos recentes [2][19][6][20].

Comparando estes dois modelos muito utilizados para problemas de planejamento, encontra-se na literatura o PSO+ [22] que supera as modernas variantes do GA para *benchmarks* de problemas com restrições. Outro ponto a favor dos modelos PSO frente aos modelos GA é o tempo de processamento. Apesar dos mínimos encontrados serem competitivos com os do PSO, os

algoritmos genéticos dependem de um tempo de processamento mais elevado [39]. Portanto, neste trabalho, optou-se por seguir pelas técnicas de PSO.

2.1.3 Planejamento Ativo

Por definição [16], um planejamento ativo é um sub-conjunto dos possíveis gráficos de Gantt com a propriedade de que nenhuma operação pode ser realizada para que as tarefas se iniciem antes e continuem viáveis. Isso significa que, dentre todas as possíveis formas de se alocar as atividades respeitando os critérios de precedência e uma sequência pré definida entre as atividades, o planejamento ativo é aquele com menor tempo total de execução planejada. Além disso, prova-se em [16] que todo ponto ótimo de um problema de planejamento com objetivo de menor tempo de conclusão é um planejamento ativo ou possui um equivalente no conjunto com mesma avaliação. Dessa forma, pode-se reduzir o espaço de busca do problema de planejamento para um problema de ordem das atividades através de transformações que levem a Planejamentos Ativos sem excluir o ponto ótimo do problema original.

Para decodificar uma sequência de atividades em um planejamento ativo, segue-se o fluxograma da Figura 2.1: aloca a primeira tarefa no instante posterior ao primeiro tempo em que o recurso, R_i , está livre e as tarefas precedentes foram concluídas; atualiza o instante em que o recurso está livre para o término da tarefa; segue para a alocação da próxima tarefa.

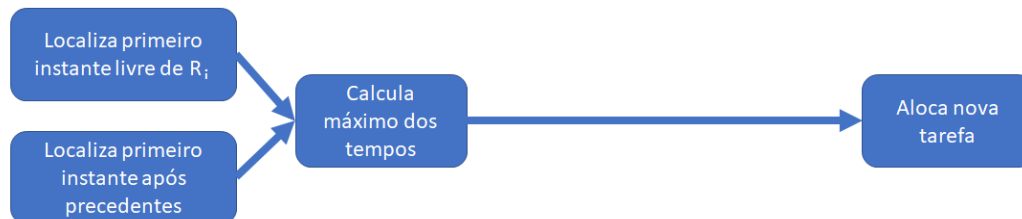


Figura 2.1: Fluxograma de construção do gráfico de Gantt

Dessa forma, escolhida a ordem de execução das tarefas, existe apenas um planejamento ativo possível de ser montado. Além disso, o ponto ótimo de um problema de planejamento é um planejamento ativo, ou seja, pode ser representado por uma ordem ótima de execução das tarefas. Dessa forma, a melhor ordem de execução das tarefas também representa o melhor planejamento.

Contudo, para o problema proposto neste trabalho, outras restrições se fazem necessárias para qualificar um planejamento como válido, modificando o fluxograma para o apresentado na Figura 2.2. Estas modificações são mais detalhadas na Seção 4.5.3.

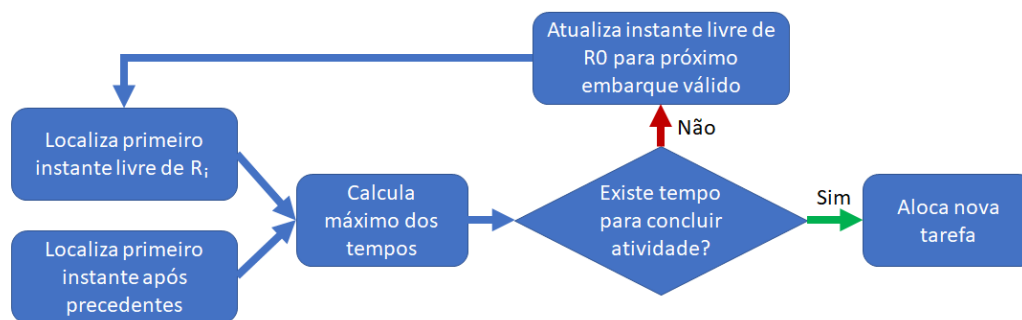


Figura 2.2: Fluxograma de construção do gráfico de Gantt modificado

2.1.4

Problemas de ordem

Como mencionado na Seção 2.1.3, os problemas de planejamento são convertidos em problemas de ordem quando definida a forma de construção do planejamento ativo. O espaço de busca de problemas de ordem pode ser modelado de várias formas, contudo as formas encontradas na literatura [39] como mais eficientes são:

- **Por prioridade:** Este tipo de problema pode ser modelado em um vetor no qual cada posição representa uma atividade e cada valor no vetor sua prioridade, como representado na Figura 2.3. As atividades de maior prioridade devem ser encaixadas no planejamento ativo antes das de menor prioridade.

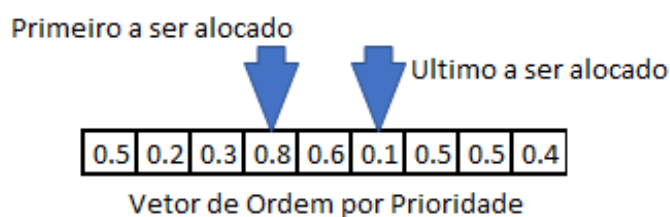


Figura 2.3: Exemplo de representação por prioridade

- **Por combinação:** Este tipo de problema pode ser modelado em um vetor no qual cada posição representa uma prioridade: $V(0)$ é executado antes de $V(1)$, e cada valor no vetor uma atividade, como representado na Figura 2.4. Dessa forma, se a posição $V(0)$ possui o valor 5, então a atividade 5 deve ser a primeira a ser alocada. É importante ressaltar que, nesse tipo de representação, as atividades são números inteiros que não se repetem no vetor.

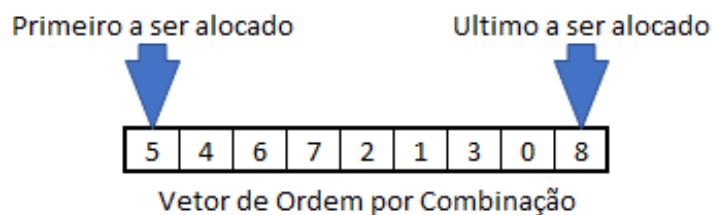


Figura 2.4: Exemplo de representação por combinação

Estudos recentes com Algoritmos Genéticos e PSO demonstraram que, em um conjunto de problemas referência, a modelagem por combinação apresenta resultados ligeiramente melhores que a por prioridade [39]. Portanto, neste trabalho é utilizado nos modelos propostos a modelagem por combinação.

2.1.5

Problema de Planejamento Flexível de Atividades

O problema de planejamento flexível de atividades é uma variante do problema de planejamento em que as atividades podem ser executadas por mais de um recurso [7]. Dessa forma, além da sequência de execução das tarefas, é importante selecionar os recursos disponíveis para paralelizar as atividades o máximo possível.

É importante ressaltar que recursos podem ser tanto matérias primas necessárias para um processo como funcionários capacitados para executá-los. Neste segundo caso, nem todo recurso pode ser alocado a qualquer atividade, sendo necessário utilizar uma matriz para relacionar os recursos aptos para cada atividade.

Como mencionado anteriormente na Seção 2.1.3, problemas de planejamento podem ser convertidos em problemas de ordem. Entretanto, para incluir a seleção de recursos na representação, mais informação deve ser codificada no vetor. Essa parte do espaço de busca pode ser modelada como um vetor de seleção de funcionários, representado na Figura 2.5. Neste modelo de decodificação, a posição N no vetor de Funcionário_Alocado representa a atividade N na qual o recurso (funcionário) será alocado. Por outro lado, o valor no vetor, *index*, representa qual a coluna na *Tabela de Decodificação* será utilizado. A combinação de um *index* e uma posição N é então decodificada em um funcionário específico. Diferentes combinações de *index* e posição podem resultar no mesmo funcionário se este for capacitado para executar as diferentes atividades. Por exemplo, na posição $N = 3$ do Vetor de Funcionário_Alocado temos o *index* = 2, o qual, na Tabela de decodificação, na Atividade A3, o

$index = 2$ representa o funcionário F8, como indicado na tabela Funcionários decodificados.

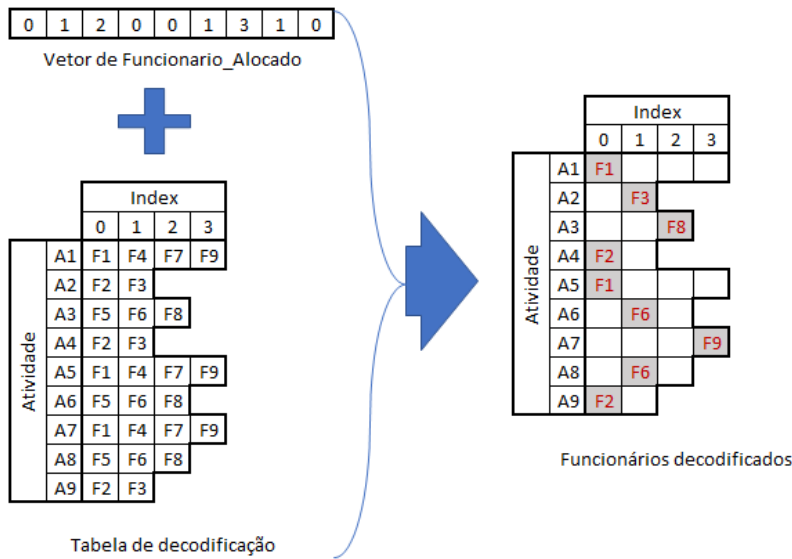


Figura 2.5: Exemplo de representação de seleção de recurso

Além disso, é comum neste tipo de problema que, além do tradicional objetivo de reduzir o tempo total das atividades planejadas, seja utilizado um modelo de função objetivo que também busque reduzir o número de recursos utilizados simultaneamente. Portanto, neste trabalho, a função objetivo descrita na Seção 4.3 considera um custo diferenciado para os funcionários que não foram selecionados para nenhuma atividade de forma a guiar o modelo para uma solução com menos funcionários alocados.

2.2 Serviços de Manutenção Embarcados e Suas Restrições

Devido à sua distância da costa, as plataformas e navios necessitam de uma diversidade de serviços embarcados para poderem se manter em operação, tais como alimentação, alojamento, energia elétrica, transporte para a costa, meios de carga e descarga, telecomunicações, serviços médicos, manutenção de equipamentos, equipamentos de segurança e emergenciais [12]. Estes serviços, apesar de similares a serviços realizados em terra, são definidas por lei como serviços embarcados, devido à peculiaridade do isolamento no mar.

No Brasil, os serviços embarcados se concentram em torno do setor de óleo e gás, responsável por cerca de 10% do PIB do país em 2010 [1]. Devido ao grande potencial do pré-sal, região marítima mais afastada da costa, novas plataformas vêm sendo construídas, de forma a expandir a exploração de petróleo no Brasil. Contudo, estas plataformas são localizadas cada vez mais

distantes da costa, tornando mais crítica a questão da logística das mesmas, além dos diversos serviços necessários para seu funcionamento. Devido ao grande número de plataformas, diversas empresas se especializaram em suprir os serviços essenciais, entre esses o serviço de manutenção embarcado.

Entretanto, diferente dos serviços em terra, o planejamento de atividades embarcadas está sujeito às peculiares condições das plataformas. Essa fábrica densamente povoada em alto mar possui um acesso controlado em datas agendadas, exige treinamentos específicos de segurança, vistorias médicas periódicas entre outras restrições [28]. Por conta disso, o planejamento de atividades embarcadas é menos flexível que o de atividades em terra, tanto em início e término de atividades quanto na escolha de funcionários capacitados.

Atualmente, o mercado dispõe de poucas ferramentas para auxiliar as empresas nesse setor. Algumas das ferramentas encontradas não consideram adequadamente o tempo de descanso dos funcionários, nem as variações específicas do trabalho embarcado. Outras, apesar de considerarem as restrições da área, utilizam técnicas desatualizadas, com alto custo computacional para otimizar o agendamento. Dessa forma, muitas empresas não investem em um software de apoio ao planejamento e operam com base em especialistas, pois avaliam que o desempenho destes é melhor que de uma técnica não específica para o setor, mesmo sob o risco de erros humanos e soluções sub-ótimas.

2.2.1

Restrições de Trabalhos Embarcados

Como mencionado anteriormente, o setor de serviços embarcados possui restrições específicas da área. Muitas dessas restrições são normatizadas na lei 5.811/72 [24], uniformizando o problema de planejamento da área para as diversas empresas do segmento. Os principais pontos dessa lei relevantes para este trabalho são:

- Revezamento em turnos de 12 horas;
- Vedação ao trabalho em período superior a 15 (quinze) dias consecutivos;
- 24 horas de repouso para cada turno trabalhado (15 dias).

Além disso, devido à restrição de espaço nas embarcações, existe um limite de pessoas que podem ser alocadas dentro de um determinado período de tempo. Essa capacidade é dividida entre a tripulação essencial para o funcionamento da embarcação e as diferentes empresas de serviços terceirizados, como a manutenção. Contudo, diferentes habilidades são necessárias em um mesmo serviço, como caldeiraria, solda, pintura e montagem de andaimes. Portanto,

uma empresa de serviços de manutenção de embarcações necessita, frequentemente, trocar a equipe a bordo em uma sincronia planejada para executar as diferentes atividades de um serviço.

Outro ponto relevante é o acesso limitado à embarcação, restringindo os períodos de embarque e desembarque de funcionários. Dessa forma, mesmo com um planejamento perfeito é possível que haja ociosidade de funcionários a bordo, caso as atividades finalizem antes da data de troca das equipes.

Outro fator relevante são os certificados de trabalho. Somente são autorizados a trabalhar embarcados funcionários com certificados específicos e atestado médico em dia. Portanto, periodicamente os funcionários devem realizar treinamentos e inspeções de saúde, de forma a terem permissão para continuar a embarcar. Um atraso em qualquer um desses certificados pode impossibilitar o funcionário de participar de uma tarefa prevista, atrasando o planejamento.

Portanto, cabe à equipe de planejamento: dividir os serviços em atividades que durem no máximo um turno (12 h); selecionar as equipes com as habilidades necessárias; e definir os períodos de embarque, respeitando o período de repouso e o limite de pessoas a bordo reservado à empresa, assim como organizar os treinamentos e agendar os exames periódicos dos funcionários.

2.2.2

Metas do planejamento de serviços de manutenção embarcados

A principal meta da equipe de planejamento de serviços embarcados é maximizar o lucro da empresa no período de um mês, dentro das restrições da área. Naturalmente, o lucro está correlacionado com os serviços concluídos e, desta forma, maximizar o lucro pode ser interpretado como minimizar o tempo de conclusão da carteira de serviços. Contudo, o problema se torna mais complexo ao se analisarem outros fatores relevantes para a empresa, tais como: o prazo de conclusão desejado pelo cliente, cota de vagas a bordo e serviços emergenciais.

Para agravar ainda mais a situação, muitas vezes os serviços mais lucrativos para a prestadora de serviço não são os mais urgentes para o cliente. Serviços mais longos implicam em mais horas de trabalho e, conseqüentemente, mais retorno financeiro para as empresas. Entretanto, atividades de menor duração possuem um impacto mais imediato no dia a dia das operações e, conseqüentemente, mais visibilidade de valor para o cliente. Assim, muitas vezes são impostos alguns prazos para a conclusão de serviços que não necessariamente trarão mais lucro no curto prazo, mas que são importantes para manter boas relações com o cliente, garantindo um fluxo de serviços ao

longo do tempo e, assim, o lucro futuro.

Outro ponto relevante é a cota de vagas a bordo da empresa. Por padrão do mercado, o número de vagas a bordo é dividido entre os diferentes prestadores de serviços, visando a maximizar os resultados do período. Portanto, não ocupar as vagas dedicadas à sua empresa representa um desperdício para o cliente. Deste modo, um dos objetivos no planejamento é equalizar o número de funcionários a bordo no período do serviço, visando a manter sua cota.

Além disso, deve-se deixar uma equipe de contingência para eventuais emergências. É de interesse do cliente manter a plataforma operando constantemente, mas, para isso, alguns defeitos devem ser corrigidos em caráter emergencial. Esses serviços ganham prioridade sobre o resto e afetam o planejamento do período.

Atualmente, o planejamento não é automatizado e especialistas empenham um considerável tempo para organizar as atividades sem uma ferramenta específica para o setor. Esse trabalho visa a suprir, sem deixar de atender às restrições da área, a necessidade do replanejamento rápido das atividades frente a novas prioridades como serviços emergenciais.

3

Otimização em Problemas de Planejamento

Neste capítulo são descritos os principais conceitos relacionados à otimização para problemas de planejamento os quais serviram de base para o melhor entendimento do modelo proposto neste trabalho, o CPSO+. Além disso, detalham-se também neste capítulo os métodos de otimização utilizados para a construção do método proposto neste estudo.

3.1

Otimização

Um problema de otimização é definido como o problema matemático de encontrar o ponto ótimo, $\mathbf{x} \in X$, que leva ao máximo ou ao mínimo global de uma função objetivo, $f(\mathbf{x})$, em um espaço de busca, $X \subset \mathbb{R}^n$, segundo restrições, $g(\mathbf{x}) < 0$, $h(\mathbf{x}) \leq 0$ [8].

Este tipo de problema pode ser escrito na forma:

$$\begin{aligned} \min f(\mathbf{x}) : \\ \mathbf{x} \in X \mid X \subset \mathbb{R}^n, \\ g(\mathbf{x}) < 0, \\ h(\mathbf{x}) \leq 0 \end{aligned}$$

Na qual:

- X é o domínio das funções f , g e h e também nomeado de espaço de busca;
- $f(x)$ é a função objetivo;
- $g(x) < 0$ e $h(x) \leq 0$ são as de restrições;
- Os elementos x que solucionam o problema são denominados de Ponto ótimo.

É importante ressaltar que, sem perda de generalidade, o problema de determinar o mínimo de $f(\mathbf{x})$ é equivalente ao problema de encontrar o máximo da função $F(\mathbf{x}) = -f(\mathbf{x})$ [35]. Além disso, restrições do tipo $H(x) = 0$ podem ser reescritas sem perda de generalidade como $H(x) \leq 0$; $-H(x) \leq 0$, ou seja,

são um caso particular da forma $h(x) \leq 0$. Portanto, neste trabalho, adota-se a minimização da função objetivo como sinônimo de problema de otimização e apenas restrições de desigualdade.

3.1.1

Função objetivo

A função objetivo é a função a ser minimizada segundo um espaço de busca, $\mathbf{x} \in X$, limitado por restrições, $g(\mathbf{x}) < 0$, $h(\mathbf{x}) \leq 0$.

Vale ressaltar que, em alguns casos, como em funções objetivo lineares, o ponto ótimo encontra-se na fronteira do espaço válido, como no problema representado pela Figura 3.1:

$$\min -x_1 - x_2$$

Onde:

$$x_1 + 2 * x_2 \leq 1$$

$$2 * x_1 + x_2 \leq 1$$

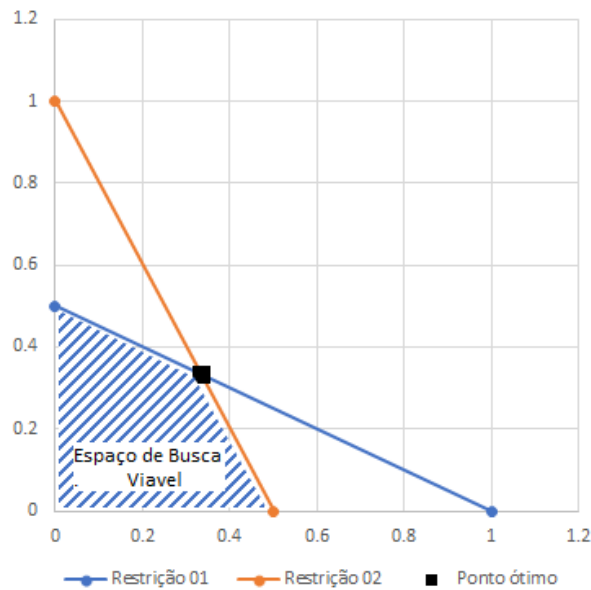


Figura 3.1: Representação gráfica do ótimo em região de fronteira.

A localização do ótimo em relação as restrições é um aspecto explorado por alguns métodos, como a programação linear. Por outro lado, pode se tornar um motivo de perda de eficiência em modelos que não possuem mecanismos específicos para explorar as fronteiras, como o *Particle Swarm Optimization* (PSO) clássico. Além disso, quando existem descontinuidades nas regiões de espaço válido ou mínimos locais próximos ao mínimo global é ainda mais

complicado para os modelos identificarem o ótimo global. Portanto, a escolha do método de otimização deve ser condizente, entre outros pontos, com as peculiaridades da modelagem matemática da função objetivo do problema [35].

3.1.2

Espaço de busca

O Espaço de busca é um espaço n dimensional, normalmente real, formado por todas as variáveis do problema, ou seja, a união dos domínios das funções $f(\mathbf{x})$, $g(\mathbf{x})$ e $h(\mathbf{x})$.

Contudo, para o problema de otimização, apenas um subconjunto deste espaço é considerado uma solução válida ao problema. Esse subconjunto, limitado pelas restrições do problema é denominado espaço de busca viável [5]. Além disso, é importante ressaltar que o espaço de busca viável pode possuir descontinuidades e limites não lineares, tornando a busca do ponto ótimo uma tarefa extremamente complexa.

Apesar de existir uma forma padrão para se representar a função objetivo e as restrições, o espaço de busca pode ser modelado de diferentes formas. As diferentes modelagens do espaço de busca influenciam significativamente no desempenho dos métodos de otimização e de forma diferente para cada tipo de problema [39]. Portanto, a escolha de como representar o espaço de busca está fortemente relacionada ao problema a ser resolvido e ao modelo de otimização.

3.1.3

Restrições

Restrições de um problema de otimização é o conjunto de inequações na forma $g(\mathbf{x}) < 0$ ou $h(\mathbf{x}) \leq 0$ que limitam o espaço de busca válido.

Alguns métodos de otimização exata foram desenvolvidos para casos particulares de restrições, como o *simplex* para restrições lineares, aproveitando de peculiaridades do espaço válido formado para encontrar o ponto ótimo. Contudo, no caso genérico com restrições não lineares e possíveis descontinuidades no espaço de busca válido, algoritmos de otimização exatos perdem em desempenho para métodos heurísticos ou meta-heurísticos como Algoritmos genéticos e PSO, conforme descrito na Seção 2.1.2.

Outro fator importante quando analisa-se as restrições de um problema é a existência de restrições de igualdade. Alguns métodos podem ser mais impactados por restrições de igualdade devido ao seu mecanismo de busca, como o PSO. Para lidar com isso, pode-se substituir as restrições de igualdade, $h(x) = 0$, por restrições na forma $\{H(x) < \epsilon; H(x) > -\epsilon\}$, onde ϵ é o erro aceitável para o problema real [22].

3.2

Técnicas para abordar restrições

Como mencionado anteriormente, alguns métodos de otimização foram desenvolvidos especificamente para um grupo particular de restrições. Contudo, métodos capazes de resolver um grupo genérico de restrições devem utilizar uma estratégia para explorar o espaço de busca válido sem ter uma técnica para definir claramente suas fronteiras. As estratégias para abordar as restrições encontradas na literatura são:

- Penalização
- Multiobjetivo
- Gerador de soluções válidas

3.2.1

Penalização

Os métodos de otimização que utilizam a estratégia de Penalização alteram a função objetivo de forma a direcionar a busca para os espaço válido. Isso é feito por uma função penalização que não altera o valor da função objetivo em pontos válidos, mas incrementa seu valor quando alguma restrição é violada.

Essa penalização pode ser multiplicativa, como na equação abaixo:

$$\begin{cases} \min f(x) & \text{se espaço viável} \\ \min g(R) * f(x) & \text{se Restrições } R \text{ forem violadas} \end{cases}$$

Contudo, o caso mais comum é a utilização de uma função penalização aditiva, como:

$$\begin{cases} \min f(x) & \text{se espaço viável} \\ \min f(x) + g(R) & \text{se Restrições } R \text{ forem violadas} \end{cases}$$

Além disso, a função penalização pode ou não variar de valor com o número de restrições violadas [9].

Essa estratégia apresenta bons resultados em problemas com espaço de busca contínuo, mesmo que não lineares. Entretanto, em caso de descontinuidades do espaço de busca, ou mesmo de um ótimo na fronteira, as penalizações podem evitar uma completa exploração do espaço e levar a uma convergência prematura em um mínimo local.

3.2.2

Multiobjetivo

Outra forma de abordar as restrições é por sua conversão em objetivos de otimização. Cada restrição da forma $g(h) < 0$ ou $h(x) \leq 0$ torna-se uma função objetivo de minimização. Assim, converte-se o espaço de busca complexo, descontínuo e com restrições em um simples espaço de busca sem restrições de maior dimensão.

Entretanto, para problemas com muitas restrições, o crescimento da dimensionalidade do espaço de busca torna o processo de otimização mais demorado e menos eficiente frente a técnicas de penalização ou de gerador de soluções válidas.

3.2.3

Gerador de Soluções Válidas

Em alguns tipos de problema, é possível utilizar o conhecimento prévio da estrutura do problema para desenvolver um Gerador de Soluções Válidas ou um mecanismo de correção de soluções inválidas.

No campo dos algoritmos genéticos, algumas soluções nesse caminho têm mais atenção na literatura, como o GENOCOP [29], para restrições lineares, e o GENOCOP III [30], para restrições não lineares. Com inspiração no GENOCOP III, técnicas similares vêm sendo também desenvolvidas para o PSO, como o PSO+ [22].

3.3

Particle Swarm Optimization (PSO)

A técnica conhecida como Particle Swarm Optimization (PSO), ou Enxame de Partículas, inspira-se na natureza, na movimentação de aves em bando, para desenvolver uma meta-heurística de otimização.

Neste modelo, o espaço de busca é convertido em um espaço n -dimensional onde partículas de dimensão nula se movimentam. Cada partícula é representada por seu vetor posição e possui as informações de velocidade, melhor posição que o bando já encontrou e melhor posição que esta partícula já encontrou.

A metodologia inicia-se pela criação de diversas partículas aleatoriamente; calcula-se então a função objetivo de cada uma e utilizam-se estas informações para movimentar as partículas no espaço, como ilustrado na Figura 3.2.

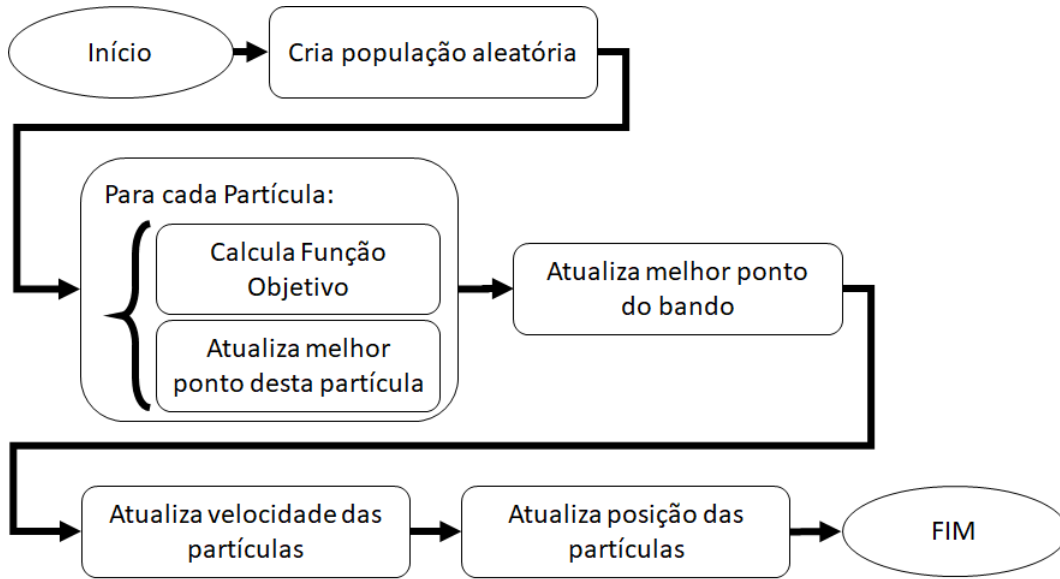


Figura 3.2: Esquemático com o fluxo do PSO

O grande diferencial das diversas variantes do PSO clássico, como o PSO+ e o PSO com inércia, é como a posição das partículas é atualizada em função das informações que ela armazena.

3.3.1 PSO Clássico

O PSO clássico, desenvolvido por Kennedy e Eberhart [23], foi o primeiro modelo desta categoria e se baseia no deslocamento de partículas em direção ao melhor global. Para isso, a estratégia original segue o fluxograma da Figura 3.2, em que cada partícula é avaliada e o melhor de cada partícula e o melhor global são salvos e influenciam na velocidade da partícula segundo as equações:

$$\mathbf{V}_i(t+1) = \mathbf{V}_i(t) + C_1 * r_1 * (\mathbf{gbest} - \mathbf{X}_i(t)) + C_2 * r_2 * (\mathbf{pbest}_i - \mathbf{X}_i(t))$$

$$\mathbf{X}_i(t+1) = \mathbf{X}_i(t) + \mathbf{V}_i(t+1)$$

em que:

- $\mathbf{V}_i(t)$ é a velocidade da partícula i na iteração t ;
- $\mathbf{X}_i(t)$ é a posição da partícula i na iteração t ;
- \mathbf{gbest} é a posição com a melhor avaliação que o bando passou;
- \mathbf{pbest}_i é a posição com a melhor avaliação que a partícula i passou;
- r_1 e r_2 são variáveis aleatórias com valor entre 0 e 1;
- C_1 e C_2 são constantes de ajuste de aceleração.

3.3.2

PSO com Inércia

Uma das desvantagens do PSO clássico é que, devido à aleatoriedade da inicialização, ele pode convergir para um mínimo local sem explorar devidamente o espaço de busca. Para contornar esse problema, o PSO com inércia foi desenvolvido [21].

A inércia é um parâmetro extra, $w(t)$, que multiplica a velocidade anterior, $V(t)$, na equação de atualização de velocidade. Este parâmetro decresce com as iterações para valorizar a exploração nas primeiras iterações e a convergência nas últimas iterações, evitando assim a convergência prematura.

Existem diferentes formas de alterar a inércia com as iterações, como linear, exponencial ou funções mais complexas. Contudo, neste trabalho, será considerada uma inércia linear decrescente.

3.3.3

CPSO

Para aplicar a técnica PSO a problemas de ordem, faz-se necessário uma representação especial do problema em vetor de estados, como ilustrado na Figura 3.3 e explicado nas Seções 2.1.3, 2.1.4 e 2.1.5.

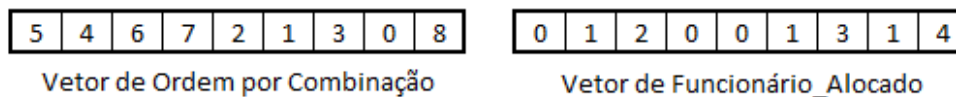


Figura 3.3: Ilustração da representação do espaço de busca do PSO Combinatorial

Originalmente, a metodologia PSO tem como base a movimentação de partículas no espaço n dimensional contínuo e independente. Contudo, na representação por combinação, cada posição é dependente da outra e não pode assumir qualquer valor real. Portanto, um fluxo alternativo deve ser utilizado para atualizar essa parte do vetor posição de cada partícula.

Segundo descrito em [39], não é necessário alterar a forma de cálculo da velocidade, mas sim a forma de execução da operação $x(t+1) = x(t) + v(t+1)$. Essa operação deve ser substituída por um processo de 5 passos.

O primeiro passo é saturar a velocidade anterior entre $-N$ e N , onde N é o número de posições do vetor. Em seguida, deve-se normalizar o módulo do vetor para valores entre 0 e 1 dividindo este por N . Utilizando o novo vetor velocidade normalizado como um vetor de probabilidade, sorteiam-se as posições que irão ser testadas para troca.

Uma posição sorteada é então comparada com a posição de uma partícula referência. A atividade na posição escolhida da partícula de referência é a candidata à troca com a atividade selecionada da partícula original. As atividades selecionadas nas duas partículas devem então ser analisadas para avaliar se a troca é possível quanto à precedência das demais atividades. Se for uma troca viável, esta é executada e as posições são marcadas para que não ocorram duas trocas com a mesma posição. O vetor resultante é considerado a nova posição da partícula. Essa nova operação é ilustrada na Figura 3.4.

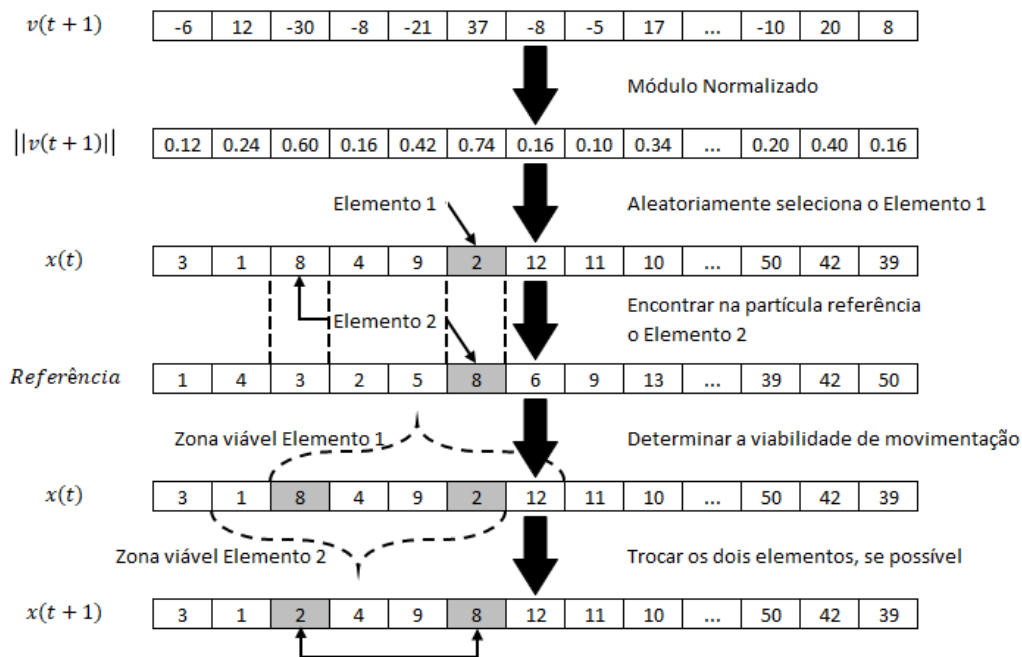


Figura 3.4: Exemplo de atualização do PSO Combinatorial

3.3.4 PSO+

O PSO+ [22], é um aprimoramento do PSO clássico que inclui:

- Gerador de solução válida, descrito na Seção 3.2.3;
- Inércia, descrito na Seção 3.3.2;
- Topologia estática aleatória coordenada, descrito na Seção 3.3.5

A metodologia PSO+ utiliza, além das técnicas mencionadas anteriormente, três conjuntos distintos como base para a busca do ótimo:

- Conjunto de Referência

- Conjunto de Fronteira
- Conjunto de Pontos de Apoio

O conjunto de referência armazena as melhores soluções encontradas e o de fronteira explora o espaço de busca, podendo ser inválido. É importante ressaltar também que o conjunto de fronteira segue uma topologia aleatória coordenada, descrita na Seção 3.3.5, para minimizar a possibilidade de estagnação em um mínimo local. Além disso, é definido que uma partícula de fronteira inválida é avaliada por meio de uma partícula temporária válida criada pela combinação linear da partícula de fronteira e uma partícula de referência, segundo equação:

$$\mathbf{x}_{corrigido} = R * (\mathbf{x}_{Fronteira} - \mathbf{x}_{Ref}) + \mathbf{x}_{Ref} , R \in [0, 1]$$

Este mecanismo de avaliação é um gerador de solução válida, como descrito na Seção 3.2.3. Após a avaliação da partícula inválida, se a temporária possuir avaliação melhor que a partícula de referência, esta é substituída pela temporária como uma melhor solução de otimização.

Outra característica do PSO+ é o conjunto de pontos de apoio que servem para redirecionar as partículas de fronteira. Os pontos de apoio são pontos inválidos iniciados aleatoriamente para redirecionar os pontos válidos para buscas nas fronteiras do conjunto viável. Sempre que uma partícula viável tiver sua posição atualizada, existe uma chance desta ser redirecionada para um ponto de apoio aleatório segundo as equações:

$$V_{max} = F_{apoio} - x(t)$$

$$V_{temp} = v(t) + C_1 * (F_{apoio} - x(t))$$

$$C_1 \in [0, 1]$$

$$v(t+1) = \min(V_{temp}, V_{max})$$

$$x(t+1) = x(t) + v(t+1)$$

3.3.5 Topologias de PSO

A topologia de PSO refere-se a como as partículas comunicam o ótimo entre si. No modelo clássico, todas as partículas compartilham a informação com todas as outras através do melhor global, *gbest*. Contudo, isso pode limitar a exploração do espaço de busca a um entorno do primeiro ótimo local encontrado.

Para contornar isto, pode-se atrasar a propagação da informação por meio de uma topologia de repasse da informação. Esse atraso é observado na modificação das equações de atualização da velocidade em que o melhor global é substituído pelo melhor ponto encontrado pelos vizinhos da partícula.

As topologias se dividem entre vizinhanças estáticas, nas quais as ligações que determinam o fluxo de informação entre partículas se mantêm constantes ao longo de todas as iterações, e vizinhança dinâmica, nas quais as ligações são modificadas ao longo das iterações.

3.3.5.1

Vizinhança dinâmica

As topologias dinâmicas diferenciam-se das estáticas por alterar as conexões entre partículas durante as iterações. Apesar da implementação ser mais complexa, evita-se a ocasional tendência de convergir prematuramente para mínimos locais características das topologias estáticas.

A mudança de vizinhança ao longo das iterações visa a modificar a estrutura de comunicação do ótimo global de uma topologia mais focada em exploração global nas primeiras iterações para uma topologia que melhor explora a vizinhança local nas últimas iterações. Exemplos dessas transformações de vizinhança que o modelo dinâmico pode assumir são apresentadas na Figura 3.5.

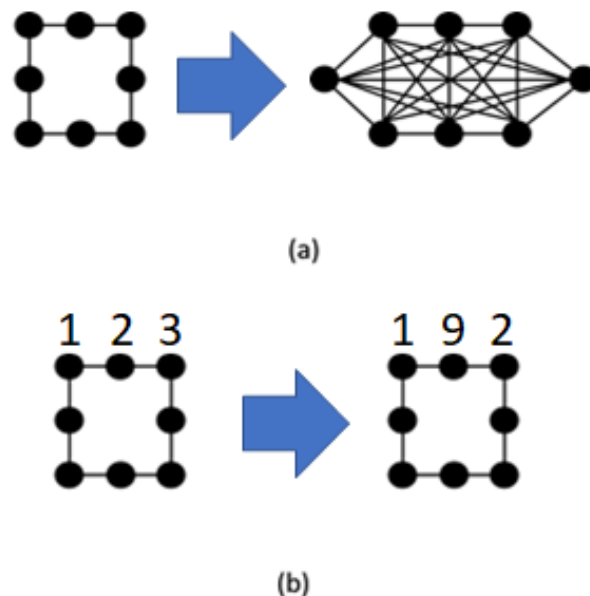


Figura 3.5: Exemplo de topologia dinâmica: a) mudança de estrutura; b) mudança de vizinhos

3.3.5.2

Vizinhança Estática

As topologias estáticas são assim denominadas por não se alterarem ao longo das iterações. Isso facilita a implementação e a análise dos resultados obtidos. Contudo, pode criar um viés a depender de como as partículas que estão conectadas foram iniciadas.

As formas mais comuns de vizinhança estática são: topologia toda interconectada, utilizada no PSO clássico; topologia em anel, em que cada partícula possui apenas 2 vizinhos; e topologia agrupamento aleatório, em que as conexões são feitas de modo aleatório, mas garantindo que nenhuma partícula fique desconectada do bando. Um exemplo gráfico dessas topologias é apresentado na Figura 3.6.

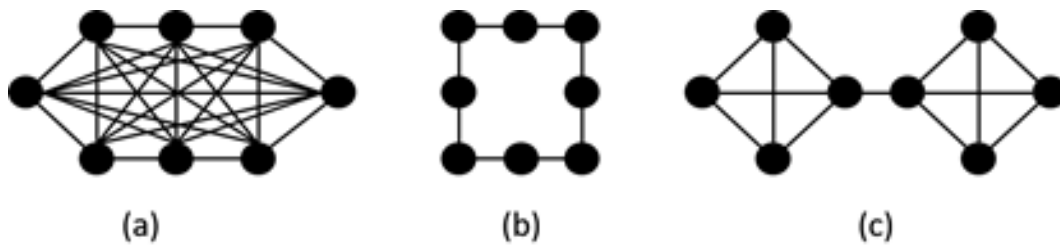


Figura 3.6: Exemplo de topologias estáticas; a) topologia toda interconectada; b) topologia em anel; c) topologia agrupamento aleatório

Contudo, algumas técnicas, como o PSO+, fazem uso de topologias que são uma combinação das técnicas básicas, como por exemplo a topologia aleatória coordenada. Essa é uma mistura de uma topologia em anel com elemento central, todos conectados, e agrupamento aleatório. As partículas são separadas aleatoriamente, como no agrupamento aleatório, em anéis de 5 elementos (4 externos e 1 central), como na topologia em anel, e as partículas centrais de cada anel constituem uma segunda camada que se comunica totalmente interconectada entre si. Essa configuração é ilustrada na Figura 3.7.

Os modelos de referência, CPSO e PSO+, utilizam uma topologia toda interconectada e aleatória coordenada respectivamente. Neste trabalho optou-se por utilizar a topologia aleatória coordenada nos modelos propostos, por apresentarem bons resultados na referência [22].

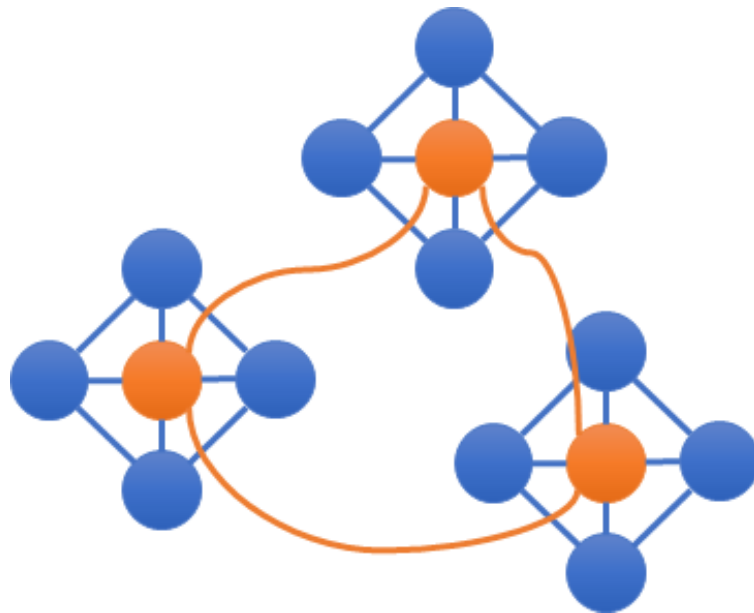


Figura 3.7: Ilustração da topologia aleatória coordenada

Este capítulo descreve as modificações realizadas nos modelos encontrados na literatura, PSO+ e CPSO, para compor o modelo CPSO+ proposto. Os principais elementos apropriados do PSO+ foram: Múltiplos Enxames; Pontos de Apoio e Topologia de Vizinhaça. Enquanto os elementos baseados no CPSO são: Atualização de posição para problemas de ordem e Conversão de vetor de ordem em um Planejamento Ativo. O diagrama que resume os principais elementos do CPSO+ é ilustrado na Figura 4.1.

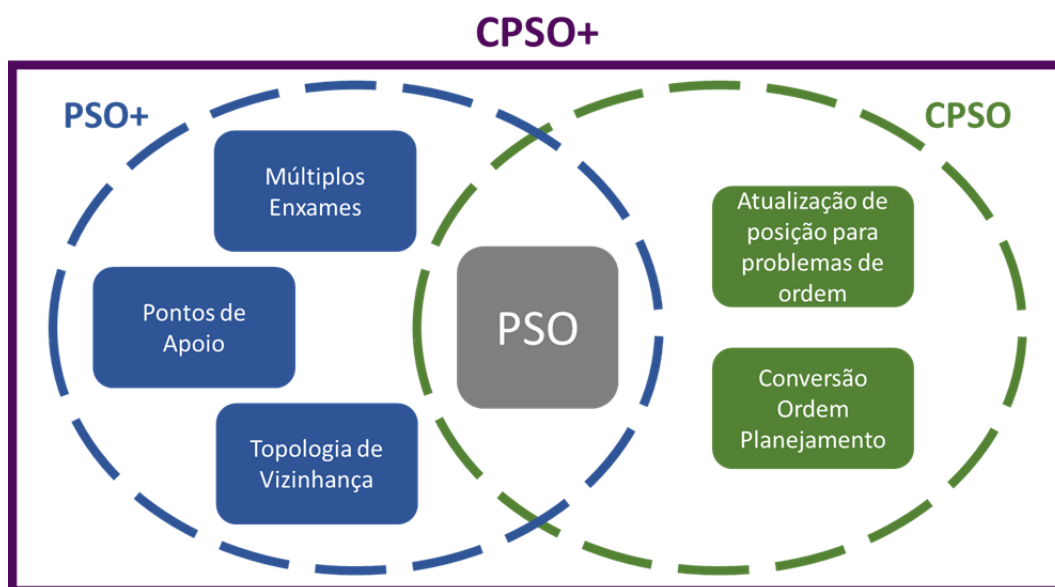


Figura 4.1: Elementos do CPSO+

Além disso, este capítulo também aborda os principais pontos de modelagem do Planejamento de Serviços Embarcados em um problema de ordem, detalhando sua função objetivo e restrições utilizadas.

Outro ponto relevante deste capítulo é a modificação da inicialização do modelo PSO+. Identificou-se que, para o problema de planejamento abordado, o tempo de processamento investido na inicialização pode ser elevado. Entretanto, a longa inicialização do modelo PSO+ é necessária para o método iniciar seus enxames de apoio e poder navegar pelo espaço repleto de restrições. Por-

tanto, reduzir este tempo pode levar a uma queda no desempenho do modelo proposto.

Para melhor avaliar os benefícios de uma redução no tempo de inicialização, propõe-se neste Capítulo uma variante com inicialização modificada a ser avaliada no Capítulo 5 frente ao CPSO+ com a inicialização original do PSO+ e ao CPSO.

4.1

Atualização de posição no CPSO+

Como descrito na Seção 3.3.4, o modelo PSO+ apresentou resultados superiores frente aos modelos mais recentes de GA em problemas referência de otimização com restrições [22] e por isso foi escolhido como base para o modelo proposto neste trabalho. Contudo, o modelo descrito no trabalho original não possui as características necessárias para tratar um problema de ordem, como descrito na Seção 3.3.3. Portanto, neste trabalho propõe-se adaptar a função de atualização da posição das partículas do PSO+ na criação de um novo modelo, denominado Combinatorial PSO+ (CPSO+).

A operação de atualização de posição das partículas, $x(t+1) = x(t) + v(t+1)$, deve ser substituída por um processo de cinco passos descrito na Seção 3.3.3. Para efeito de clareza, a Figura 3.4 é repetida a seguir, na Figura 4.2.

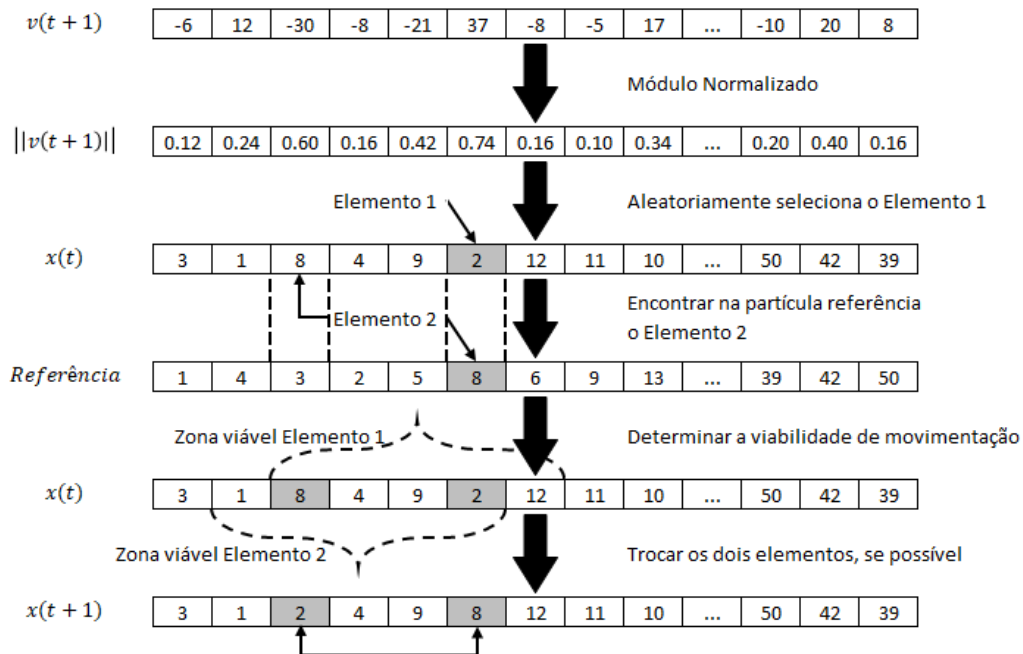


Figura 4.2: Exemplo de atualização do CPSO+

Portanto, no CPSO+, as principais operações do PSO+ são modificadas:

- **Atualização em vizinhança:** A partícula referência é sorteada a cada posição entre a melhor da vizinhança e a melhor desta partícula.

$$v(t+1) = v(t) * w(t) + C_1 * r_1 * (P_{vizinho} - x(t)) + C_2 * r_2 * (P_{passado} - x(t))$$

$$x(t+1) = x(t) + v(t+1)$$

- **Atualização em direção ao Ponto de Apoio:** A partícula referência é o ponto de apoio.

$$v(t+1) = v(t) * w(t) + C_1 * r_1 * (P_{apoio} - x(t))$$

$$x(t+1) = x(t) + v(t+1)$$

- **Atualização para corrigir partícula:** A partícula referência é a partícula válida escolhida para guiar a partícula inválida.

$$v(t+1) = r_1 * (P_{valida} - x(t))$$

$$x(t+1) = x(t) + v(t+1)$$

4.2

CPSO+ com inicialização modificada

Outro tópico importante a ser abordado é a inicialização dos enxames do CPSO+. Por possuir três enxames distintos, sendo dois deles exclusivos de uma região do espaço de busca, esse processo pode consumir uma grande parcela do tempo de processamento.

Além disso, o modelo PSO+ original não aborda casos em que o espaço de busca é completamente válido ou completamente inválido. Estes casos particulares geram problemas na implementação do modelo para casos práticos onde a validade do espaço de busca é conhecida a posteriori. Pequenos ajustes são propostos nessa seção para minimizar esses impactos.

4.2.1

Modificação da inicialização dos enxames de referência e apoio

O modelo proposto em [22] para o PSO modificado, o PSO+, considera a utilização de três tipos de enxames:

- Busca – encontram a solução do problema se movendo no espaço de busca, válido e inválido;
- Apoio – todas inválidas e utilizadas para explorar melhor as fronteiras do problema;

- Referência – todas válidas e utilizadas para cálculo de valor das partículas de busca inválidas.

Contudo, não foi encontrado na referência nenhum processo para problemas com todo o espaço de busca válido ou todo o espaço de busca inválido, cenários usuais em aplicações reais. No caso de um problema completamente válido, o modelo não conseguiria definir um grupo de apoio. Como este é utilizado apenas ocasionalmente para diversificar a busca, pode-se operar ignorando esse passo. Contudo, segundo o processo original, na fase de inicialização do modelo, o programa deveria tentar gerar partículas inválidas por 100 rodadas antes de desistir de preencher esse grupo, gerando assim um enorme desperdício de tempo e poder de processamento.

De forma similar, em um problema completamente inválido, o modelo não conseguiria gerar o grupo de referência. Além de gastar muito tempo na inicialização buscando soluções não existentes, não conseguiria avaliar as partículas de busca válidas.

Como a única restrição que define a validade do espaço de busca é o prazo de conclusão dos serviços, prazos muito longos ocasionam espaços de busca predominantemente válidos. Por outro lado, prazos curtos e concorrentes restringem consideravelmente o espaço de busca, podendo inclusive torná-lo completamente inválido.

Para contornar a possível demora para iniciar os enxames auxiliares, adotou-se uma nova estratégia de inicialização: Limitou-se a busca inicial em $2 * N$ partículas aleatórias. As N primeiras, válidas ou não, formam o grupo de busca. Até N partículas válidas de melhor custo formam o grupo de referência. Até N partículas inválidas (selecionadas aleatoriamente) formam o grupo de apoio. Por fim, utiliza-se o modelo de penalização caso não haja partículas de referência.

Dessa forma, mesmo não construindo um exame auxiliar completo na inicialização, o programa seguirá executando o modelo com os parâmetros disponíveis. Contudo, durante o processo de busca, as partículas de busca válidas e inválidas são utilizadas para completar os exames auxiliares. Isso garante que mesmo em inicializações com um exame vazio, ao longo do processo o modelo se beneficiará das vantagens de dispor de exames auxiliares, caso esses sejam preenchidos a posteriori. Este modelo proposto com inicialização modificada será tratado neste trabalho como $CPSO+_m$. Vale ressaltar que todas as demais considerações feitas ao modelo com a inicialização original do PSO+ se aplicam ao $CPSO+_m$, modificado apenas em sua inicialização.

4.3

Funções Objetivo

Um dos pontos chave para aplicar o modelo CPSO+ ao problema de Planejamento Flexível Embarcado é a definição da sua função objetivo. Essa função deve ser definida de forma a guiar o modelo na direção do ótimo desejado: Mais serviços completos com o mínimo de funcionários a bordo.

Além disso, em algumas avaliações do modelo CPSO+, é necessário considerar uma configuração de função objetivo por penalização. Esta configuração diferenciada, por penalização, também foi utilizada no modelo de comparação, o CPSO.

4.3.1

Função objetivo CPSO+

A função objetivo deve traduzir os principais pontos de interesse do usuário em formulas matemáticas. No problema proposto, otimização do agendamento de serviços embarcados, os principais objetivos da alocação são:

- Terminar as atividades com o menor número de funcionários;
- Reduzir o tempo embarcado dos funcionários;
- Concluir o maior número de serviços e não de atividades.

Vale ressaltar também que utilizou-se um método para agrupar múltiplos objetivos em um único através de pesos. Para isso, a função objetivo utilizadas neste trabalho foi definida como:

$$MAXIMIZAR : 100 \sum hh_A \times a_i - \sum d_{NE}[f_j] - 2 \sum d_E[f_j]$$

Onde:

- a_i - booleano das atividades. Representam a conclusão de um serviço e assume 1 quando concluída dentro do prazo;
- hh_A - tempo mínimo de horas para se terminar o serviço;
- f_j - funcionário j . Apenas os alocados em alguma atividade durante o período de simulação;
- $d_{NE}[f_j]$ - Número de dias que o funcionário j não foi alocado em um embarque;
- $d_E[f_j]$ - Número de dias que o funcionário j foi alocado em um embarque.

O peso do trabalho embarcado é o dobro do peso do trabalho não embarcado para refletir a proporção salarial dos funcionários nessas condições. Por outro lado, o peso da conclusão do serviço, 100, foi determinado por testes preliminares com o modelo.

4.3.2

função objetivo CPSO penalização

Quando não houver partículas de referência, o modelo necessita de uma outra forma de avaliação das partículas inválidas para então seguir rumo a um ótimo. A solução proposta neste trabalho é utilizar o método da penalização.

Em problemas de maximização, como o aqui proposto, define-se um número negativo grande para cada restrição não atendida. Para guiar as soluções em direção a um ótimo, é interessante também diferenciar as soluções inválidas com o mesmo número de restrições não atendidas. Portanto, adiciona-se a esse número negativo o custo da solução com base nas outras atividades concluídas e no número de funcionários ativos, como representado abaixo:

$$100 \sum a_i \times hh_A - \sum d_{NE}[f_j] - 2 \sum d_E[f_j] - M \times r$$

Onde:

- M - soma de todos os hh_A das atividades simuladas;
- r - número de restrições não atendidas.

4.4

Premissas do modelo matemático

Apesar de tentar se aproximar ao máximo do caso real, o modelo matemático apresentado neste trabalho para representar operações de manutenção embarcadas leva em consideração algumas premissas sobre a execução das atividades:

Premissa 1: Atividades não são interrompidas

Uma vez iniciada uma atividade, um agente deve seguir com sua execução até a conclusão sem interrupção. Além disso um agente só pode executar uma atividade por vez;

Premissa 2: Todos os parâmetros do modelo são conhecidos

Para calcular a melhor alocação de cada agente em cada atividade, informações sobre a disponibilidade de cada agente, especialização, tempo entre atividades realizadas por um mesmo operário e proporção entre tempo embarcado e tempo de descanso devem ser conhecidas ou estimadas para o agendamento.

Premissa 3: Os componentes do problema são estáticos

Como o agendamento é realizado antes da execução das atividades, supõe-se que os agentes não irão adquirir novas habilidades nem perder habilidades atuais, que manterão na empresa até a realização das atividades e que as atividades permanecerão relevantes até o período de seu agendamento. Caso algum destes fatores se altere ao longo da execução da agenda, esta deve ser recalculada para um novo ótimo.

Premissa 4: Cada atividade deve ser executada por exatamente um agente

Para os diversos modelos estudados, há restrições para encontrar o ponto ótimo caso sejam incluídas atividades que exijam mais de um recurso para sua conclusão. Estas atividades devem ser inseridas no modelo como uma atividade de recurso único, que representa um grupo pré definido de agentes capazes de executar a atividade.

Premissa 5: Atividades não dependem do agente

A duração de execução de cada atividade não depende do agente ou grupo de agentes selecionado para executar a mesma. Essa premissa foi incluída devido a não praticidade de se estimar a eficiência de cada funcionário para diferentes tarefas para o setor em análise.

Vale ressaltar que, apesar de restringirem a forma de implementar o caso real em um modelo matemático, nenhuma das premissas reduz o problema original ou inviabiliza sua implementação prática.

4.5

Modelagem

Para otimizar um problema por meio da técnica de CPSO+, um dos fatores mais importantes para o desempenho é a representação do problema

em um vetor de estados. Para um problema com diversas atividades e recursos, duas formas de representação fazem-se necessárias: uma para a ordem das tarefas e outra para selecionar o recurso associado.

4.5.1

Representação da ordem das tarefas

Como mencionado anteriormente, resultados recentes demonstram que entre as duas opções de representação de ordem das tarefas, por ordem e por prioridade, a primeira apresenta melhores resultados, como discutido na Seção 4.1. Essa representação consiste em um vetor com m posições, onde m é o número de tarefas. Cada posição deve ser então preenchida com um número inteiro entre 1 e m , representando a sequência a executar as tarefas.

Outra consideração importante são as atividades que não são realizadas embarcadas, como exames médicos e treinamentos. Estas atividades são independentes das atividades embarcadas e, por isso, poderiam ser alocadas em qualquer posição do vetor. Para diferenciar essas atividades das demais, além dos vetores de duração e habilidade necessária, cada atividade tem um valor associado para indicar se é embarcada ou não embarcada.

É também importante ressaltar o processo de inicialização do segmento de ordem da representação. Por natureza da solução proposta, a partícula só se movimenta para soluções que atendem às restrições de antecedência das atividades. Além disso, o modelo pressupõe que o primeiro ponto já atende a todas essas restrições e que só é necessário avaliar as mudanças entre dois pontos. Portanto, o mecanismo para gerar diferentes pontos iniciais é um ponto crucial do modelo.

O método encontrado na literatura [39] para iniciar as partículas em uma posição válida mas de forma a abranger uma parcela significativa do espaço de busca é:

- Partir de uma sequência válida conhecida (*seed*);
- Realizar cem movimentos aleatórios de ordem na sequência primordial.

Como os movimentos de ordem não violam a validade do vetor, a solução gerada é uma solução que atende aos critérios de antecedência e é aleatoriamente diferente da sequência primordial.

Contudo, na prática, essas cem operações não apenas consomem um tempo de processamento verificando a viabilidade das trocas de prioridade das atividades, como não geram soluções muito diversas. Para acelerar o processo e reduzir o processamento, optou-se por utilizar um método heurístico para corrigir posições não permitidas.

Primeiro gera-se um vetor aleatório da permutação da sequência de atividades. Depois, garantindo por construção do problema que todas as dependências da atividade N possuem índice menor que ela, deve-se validar sequencialmente se as atividades estão em prioridades válidas.

As atividades que não estão em uma posição válida necessitam ser realocadas. Se elas fossem realocadas imediatamente após a atividade precedente, perder-se-ia grande parte da aleatoriedade do vetor. Por isso optou-se por sortear uma nova posição entre as posições válidas (após todos os precedentes).

No Capítulo 5, alguns testes são realizados para comprovar que o modelo proposto não só garante pontos de partida mais diversos como necessita de menos tempo de processamento.

4.5.2

Representação do recurso selecionado

Como mencionado anteriormente na Seção 2.1.5, para problemas com múltiplos recursos deve-se incluir no vetor posição um segmento extra para seleção do recurso utilizado na atividade. As principais habilidades dos serviços de manutenção embarcada a serem abordadas nesse trabalho são:

- Montadores de Andaime;
- Supervisor de montagem de andaime;
- Caldeireiros;
- Ajudante;
- Inspetor de Solda;
- Supervisor geral;
- Pintor;
- Inspetor de pintura.

As atividades não embarcadas são direcionadas exclusivamente a um indivíduo e por isso não necessitam ter correspondente no vetor posição. Contudo, as atividades embarcadas possuem mais de uma possibilidade de recurso, possibilitando otimizações. Portanto, o vetor seleção de recurso só representa os recursos embarcados e as posições deste estão diretamente associadas à representação numérica das atividades. Ou seja, a atividade representada pelo número 1 será executada pelo recurso selecionado pela posição 1 do vetor seleção de recurso.

Além disso, os recursos são representados por números inteiros entre 0 e N_i , onde N_i é o número de possíveis recursos para a atividade i . Para cada

atividade, um vetor decodificação é criado para associar o número do recurso no vetor ao número do recurso na tabela geral de recursos.

Outro ponto relevante é a criação de grupos de recursos. Como mencionado nas premissas, Seção 4.4, uma atividade só pode ter um recurso associado a ela, mas algumas necessitam de mais de uma habilidade para serem concluídas. Portanto, criaram-se grupos de recursos associados aos recursos reais para contornar essa limitação. Esses grupos são formados segundo os dados de um histórico real de serviços embarcados, no qual se identificou que os principais grupos de trabalho são:

- Montadores de Andaime - normalmente trabalham em um grupo de 4 funcionários;
- Caldeireiro e Ajudante - Apesar de nem sempre trabalharem juntos, o agrupamento representa bem a realidade em que ambos estão embarcados no mesmo período;
- Pintores - Trabalham em duplas ou quartetos.

As demais funções identificadas, como Supervisão e Inspeção dos serviços, atuam normalmente sozinhos e por isso não foram agrupados na representação desse problema.

4.5.3

Decodificação do vetor de ordem em planejamento ativo

Como discutido na Seção 2.1.3, pode-se representar um problema de planejamento como um problema de ordem segundo um decodificador de planejamento ativo. Esse decodificador é um processo para alocar as atividades no vetor de ordem segundo as restrições temporais do problema.

No problema em estudo, as restrições são:

- Uma atividade embarcada deve ser iniciada após um embarque;
- As atividades embarcadas devem ser concluídas antes do fim do período válido de embarque, 15 dias;
- O período de descanso de um funcionário embarcado é igual ao período embarcado;
- Cada período de tempo, duração de um embarque, possui um limite máximo de pessoas a bordo;
- Atividades não embarcadas devem ser executadas em um período em que o funcionário não está embarcado;
- Um funcionário não pode executar duas atividades ao mesmo tempo.

Como muitas das restrições do problema estão relacionadas a dias, optou-se por uma representação diferenciada do tempo. Os dias de simulação foram representados de forma discreta entre 0 e D , enquanto as horas em um dia são representadas por números reais entre 0 e HH , o período máximo de trabalho em 1 dia. Essa representação simplifica as comparações em dias, além de flexibilizar a estimativa de duração das atividades para tempos contínuos.

Entretanto, como não há mecanismos para quebrar as atividades que passam do tempo diário de trabalho, essa abordagem limita a duração máxima das atividades, pois apenas as que possam ser executadas em HH horas podem ser alocadas em determinado dia. Para contornar este problema, as atividades originais do histórico passam por um processo de desconstrução em blocos sequenciais de tempo máximo de duas horas. Como o problema proposto apresenta atividades com previsão de horas inteiras, esse processo discretiza as atividades em blocos de duas horas e um bloco final de uma hora, se as horas estimadas forem um número ímpar. Escolheu-se trabalhar com uma discretização de duas horas para reduzir o espaço de busca frente a separação por hora, mas trabalhos futuros podem avaliar os ganhos de outras discretizações.

Para lidar com essas restrições, o decodificador segue um fluxo lógico de passos. Primeiro, determina qual a atividade a ser alocada pelo valor na próxima posição do vetor de ordem. Depois, determina qual o recurso a ser utilizado nessa atividade pelo vetor de seleção de recurso na posição da atividade. Em seguida, encontra o primeiro dia livre do recurso e o último instante de execução das atividades marcadas como requisito para execução da selecionada. Com esses dados, o decodificador pode determinar que o primeiro dia possível é o máximo das duas datas encontradas.

Determinado o primeiro dia a ser avaliado pelo decodificador, este então inicia a análise das restrições. Verifica se o recurso está embarcado e se isso é condizente com os requisitos da atividade. Verifica se a atividade termina antes do fim do embarque e, caso não termine, atualiza o primeiro dia livre do recurso selecionado para a próxima data calculada como válida, respeitando tempo de descanso e o limite de pessoas a bordo, e reinicia a busca. Esses passos podem ser visualizadas no fluxograma da Figura 4.3.

4.5.4

Outras Restrições

Apesar de o decodificador de ordem alocar as atividades de forma a não violar nenhuma das restrições temporais, outras restrições devem ser consideradas nesse problema. A principal restrição a ser analisada e corrigida

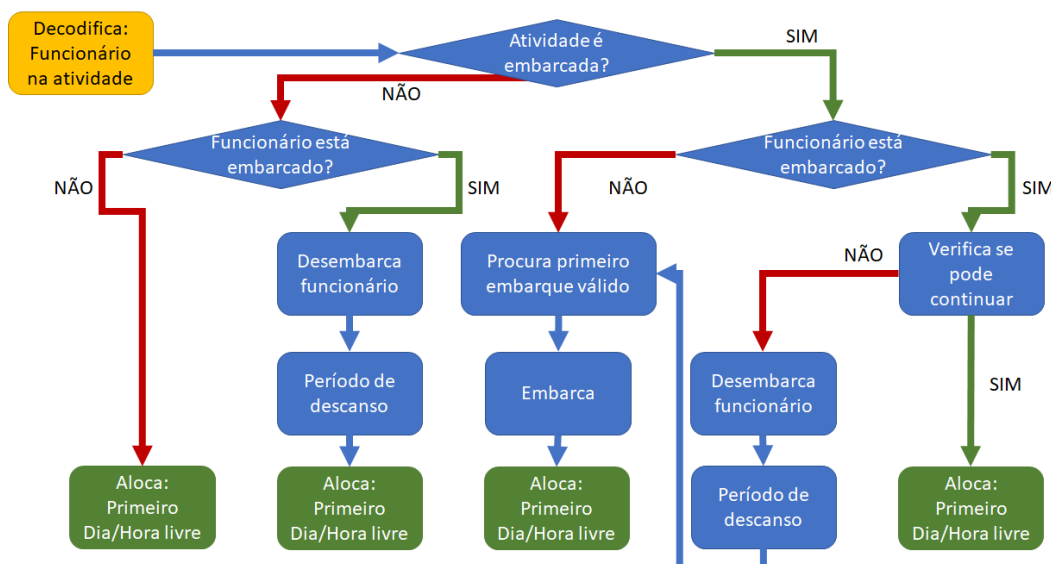


Figura 4.3: Fluxograma do decodificador do vetor de ordem

pelo CPSO+ são as datas limite.

Uma vez decodificado o vetor de ordem, as atividades alocadas no tempo devem respeitar o prazo estabelecido em projeto. Apesar de identificar as atividades que violam essa regra, o decodificador não é capaz de corrigir esse planejamento sem modificar o vetor de ordem. Portanto, essa restrição deve ser resolvida pelas técnicas de correção de solução do CPSO+.

Desta forma, em meio ao processo de solução do problema, o CPSO+ segue o fluxograma da Figura 4.4. Primeiro o vetor de funcionário alocado é decodificado para determinar quais são os funcionários escolhidos para cada atividade. Em seguida, a sequência de atividades e os funcionários selecionados para cada atividade são convertidos em cronograma pelo decodificador apresentado na Seção 4.5.3. Este cronograma é então avaliado para determinar o atendimento aos prazos previstos. Caso não atenda a um ou mais prazos, esta partícula é considerada inválida.

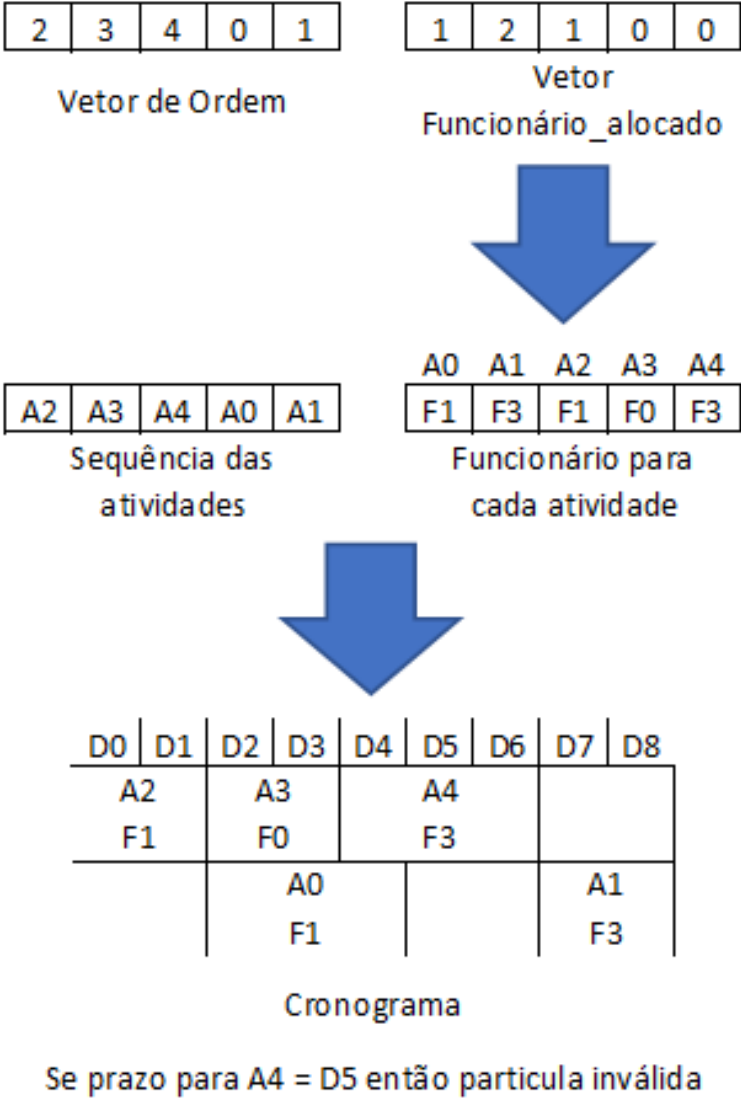


Figura 4.4: Fluxograma de montagem do cronograma

5

Estudo de Caso

Para avaliar o desempenho do modelo proposto neste trabalho, CPSO+, e sua variante, CPSO+ com inicialização modificada, $CPSO+_m$, frente ao modelo da literatura, CPSO, propõem-se seis simulações do Problema de Planejamento de Serviços de Manutenção Embarcada.

- Duas simulações para avaliar as etapas do modelo que mais consomem tempo de processamento;
- Quatro para avaliar a qualidade dos mínimos encontrados e o tempo total de processamento.

5.1

Introdução

Os modelos propostos neste trabalho para solucionar o problema em estudo são:

- CPSO+;
- $CPSO+_m$.

O modelo da literatura, CPSO por penalização, foi utilizado como referência de tempo de processamento, pois possui os menores processos de inicialização e os mecanismos mais simples de busca. Por outro lado, o CPSO+ foi selecionado como a referência de qualidade de busca, ou seja, resultados mais próximos do ótimo global e menos dispersos em diferentes repetições.

As principais modificações a serem avaliadas no $CPSO+_m$ são:

- Não inicializar por completo os enxames de apoio e de referência;
- Incrementar esses enxames com partículas descobertas ao longo do processo;
- Utilizar o modelo de penalização enquanto não encontrar partículas válidas.

Todos os modelos CPSO mencionados utilizaram o processo de criação de partículas de ordem válidas proposto na Seção 4.5.1. Portanto, para avaliar melhor os ganhos daquele processo, este será avaliado isoladamente dos modelos.

Além disso, um estudo de caso será dedicado a avaliar o tempo de processamento da função de criação de cronogramas válidos. Apesar de seguir os princípios encontrados na literatura, essa ferramenta foi adaptada para considerar as restrições específicas de problemas de agendamento de atividades embarcadas e, por isso, seu desempenho deve ser analisado.

5.2

Base de Dados

Para realizar simulações mais fieis ao caso real, utilizou-se uma base de dados históricos com os serviços realizados por uma empresa do segmento nos últimos 6 meses e a escolha de funcionários para cada atividade.

Observou-se na base de dados que os serviços são compostos por quatro blocos principais de atividade:

- Montagem de andaime
- Solda e Caldeiraria
- Pintura
- Desmontagem de andaime

Cada serviço encontrado na base é uma combinação desses blocos, repetidos ou não, com algumas variações em suas durações para adequá-los às necessidades de campo. Contudo, sempre a mesma sequência de habilidades são necessárias para executar cada etapa do bloco.

A montagem e a desmontagem de andaime seguem o padrão: liberação da operadora para executar o serviço, realizada pelo supervisor de montagem de andaime; 4 montadores de andaime são então acionados para limpar a área e montar os andaimes; o supervisor confirma a conclusão bem sucedida da montagem; e, por fim, fecha junto à operadora a conclusão desta etapa.

De modo similar, a equipe de solda e caldeiraria segue o mesmo fluxo de trabalho: liberação de serviço pelo supervisor de solda; limpeza e execução da solda por um caldeireiro e um ajudante; Inspeção da qualidade do serviço, por um inspetor de solda com equipamentos para teste de solda; e, então, o fechamento da atividade junto à operadora

Por outro lado, a equipe de pintura possui uma peculiaridade a mais. Após as etapas de liberação pelo inspetor, a equipe segue quatro ciclos

das atividades: execução de demão pelos pintores, inspeção da aplicação pelo inspetor e um intervalo necessário para a tinta secar. Esse intervalo foi modelado neste trabalho como mais uma atividade no bloco de pintura. Contudo, este pode ser um ponto de melhora no modelo proposto, uma vez que é factível que o programa, erroneamente, separe este tempo de espera entre dois dias diferentes ou mesmo em embarques distintos.

5.3

Casos de simulação

5.3.1

Simulação 01: inicialização da partícula

Para avaliar a heurística proposta neste trabalho para inicializar um partícula de ordem atendendo às restrições de precedência das atividades, algumas simulações foram realizadas apenas com essa etapa do processo.

As métricas utilizadas para comparar os resultados foram: tempo de execução dos programas e diversidade da solução gerada. O tempo de execução foi avaliado para um código em Python 3, executado em um laptop com sistema operacional Windows 10, um processador core i7 e uma placa gráfica *NVIDIA GeForce GTX 1050 Ti 4GB*. A diversidade da solução foi medida como a frequência, nas diversas repetições do teste, que uma atividade sem dependências aparece na primeira posição do vetor. Esta métrica de diversidade avalia se todas as atividades que podem ocupar a primeira posição aparecem de forma homogênea em uma inicialização aleatória de um enxame. Caso isso não ocorra, o processo de inicialização não é avaliado como capaz de gerar soluções realmente diversas.

O principal parâmetro que impacta na diversidade das partículas geradas e no tempo de processamento desta etapa é o número de atividades dependentes no vetor de ordem.

Portanto, para avaliar o desempenho dos dois processos, simulou-se a geração de 20 partículas de diferentes tamanhos e com o mesmo número de atividades independentes, como descrito na Tabela 5.1.

Tabela 5.1: Casos de simulação do processo de geração de partículas de ordem

Número de atividades	Número de repetições	Atividades independentes
10	20	[0,2,4,6,8]
50	20	[0,10,20,30,40]
100	20	[0,20,40,60,80]
250	20	[0,50,100,150,200]
500	20	[0,100,200,300,400]

Os resultados obtidos nessas simulações, como observado nas Tabelas 5.2; 5.3 e 5.4, demonstram que o processo proposto é cerca de 170 vezes mais veloz que o modelo original e muito mais diverso em casos com muitas atividades dependentes.

Tabela 5.2: Tempo das simulações de geração de partículas de ordem

Número de atividades	Tempo médio modelo proposto	Tempo médio modelo original
10	3 ms	0.5 s
50	16 ms	2.8 s
100	31 ms	5.6 s
250	94 ms	15.7 s
500	208 ms	32.9 s

Tabela 5.3: Resultados da diversidade da primeira posição modelo original

Número de atividades	frequência				
	0	1	2	3	4
10	3	5	7	2	3
50	20	0	0	0	0
100	20	0	0	0	0
250	20	0	0	0	0
500	20	0	0	0	0

Tabela 5.4: Resultados da diversidade da primeira posição modelo novo

Número de atividades	frequência				
	0	1	2	3	4
10	3	2	5	6	4
50	6	4	4	3	3
100	3	5	3	2	7
250	2	5	3	2	8
500	3	2	5	2	8

5.3.2

Simulação 02: Tempo de montagem do cronograma

Em outros trabalhos encontrados na literatura, a etapa de montar o cronograma a partir do vetor de ordem costuma ser brevemente descrita e negligenciada. Isso ocorre pois, na maior parte dos casos, as restrições são simples e não necessitam de lógicas muito complexas para a construção do cronograma. Contudo, o trabalho embarcado possui restrições não usuais na literatura, como limite de pessoas no ambiente de trabalho e de tempo de descanso após um período de trabalho.

Essas restrições necessitam de um fluxo de decisão um pouco mais complexo que o fluxo padrão. Essas decisões e as várias variáveis de apoio que devem ser consultadas para tomá-las acabam ocasionando um tempo de processamento superior ao esperado. Como os modelos PSO+ realizam mais operações de cálculo de custo do que o PSO convencional, um tempo significativo no processo de montagem de cronograma pode impactar no tempo total de execução.

Para melhor avaliar o impacto desse processo no modelo, simulou-se a criação de 20 cronogramas para 5 casos distintos de atividades, como descrito na Tabela 5.5. Os resultados das simulações são mostrados na Tabela 5.6.

Tabela 5.5: Casos de simulação da criação de cronograma

Número de atividades	Número de repetições	Dias simulados
10	20	35
50	20	35
100	20	35
250	20	35
500	20	35

Tabela 5.6: Tempo médio da criação de cronograma

Número de atividades	Tempo médio de execução
10	6 ms
50	31 ms
100	66 ms
250	185 ms
500	401 ms

Observando os resultados obtidos, verifica-se um crescimento exponencial do tempo de execução do processo de criação do cronograma com o aumento das atividades. Nos casos propostos neste trabalho, estima-se cerca de 250 atividades por simulação, com enxames de 100 partículas e uma média de 20 épocas até convergir para um máximo. Segundo os resultados obtidos e as considerações feitas, o tempo médio estimado apenas com a montagem de cronograma seria de:

$$185ms \text{ (tempo para 250 atividades)} * 100 \text{ partículas} * 20 \text{ épocas} = 370s \text{ (6 min)}$$

Contudo, o modelo CPSO+ proposto também deve avaliar partículas inválidas por meio de algumas partículas temporárias, incrementando o tempo de processamento. Ou seja, quanto maior o número de partículas inválidas no enxame de busca do modelo CPSO+, maior o tempo de processamento do

modelo. Para abrandar essa questão, utilizou-se o processamento paralelo nas etapas de movimentação e avaliação das partículas.

Portanto, ainda que abrandado com o processamento paralelo, o elevado tempo de montagem do cronograma eleva o tempo necessário para cada avaliação. Por isso, o modelo CPSO+, que necessita de várias avaliações para tentar corrigir as partículas inválidas, tende a consumir mais tempo quando o espaço de busca é predominantemente inválido.

5.3.3

Simulações para comparação entre modelos

Para comparar os modelos encontrados na literatura, CPSO por penalização, com o modelo modificado proposto neste trabalho, CPSO+ & $CPSO+_m$, propõem-se quatro condições:

- espaço de busca 100% inválido
- espaço de busca 100% válido
- espaço de busca predominantemente válido
- espaço de busca predominantemente inválido

Essas condições foram escolhidas para representar todos os possíveis cenários de uso dos modelos no problema proposto. Vale ressaltar também que a viabilidade do espaço de busca nestes modelos é determinado exclusivamente pelo prazo de conclusão das atividades. Portanto, o caso 1 representa atividades com prazos impossíveis de serem cumpridos, mas só descobertos após a confecção do cronograma, devido à alta complexidade de avaliar estes a priori. Por outro lado, o caso 2 representa as solicitações de serviço sem prazos determinados, por possuírem menor impacto na operação. Os demais casos, então, representam os serviços mais usuais, nos quais os prazos não são impossíveis, apesar de restritivos.

Esses casos foram traduzidas na simulação como diferentes prazos para um dos serviços simulados. Um prazo impossível é o único parâmetro que garante a inviabilidade de todo o espaço de busca. Para a completa viabilidade do espaço de busca, removeram-se todos os prazos. Calculando a duração esperada para a conclusão mais rápida da atividade, um prazo 1.5 vezes maior foi considerado um espaço de busca predominantemente inválido. Por fim, um prazo próximo ao fim dos dias simulados foi utilizado para tornar o espaço de busca mais válido que inválido.

Para avaliar apenas o impacto das condições propostas, todos os demais parâmetros devem ser idênticos nas simulações. Esse parâmetros são:

- Número de funcionários disponíveis para alocação;
- Limite de pessoas a bordo;
- Número e sequência de atividades;
- Sequência de dias de embarque possíveis.

Normalmente, as empresas de manutenção *offshore* trabalham com grupos de funcionários por plataforma. Estes grupos são compostos por, ao menos, três funcionários com as habilidades principais para os serviços: Caldeiraria; Pintura; Montador de andaime. Dois desses revezam-se entre embarques, quinze dias em terra, quinze na plataforma, o terceiro é necessário para substituições em casos de treinamento e licença médica. Contudo, para facilitar a exploração do espaço de busca, serão considerado neste trabalho seis funcionários com cada habilidade. Com isso, apesar de reduzir o número de funcionários escolhidos durante a otimização, o programa tem mais liberdade de escolhas iniciais, evitando que os únicos pontos válidos sejam a solução ótima.

O limite de pessoas a bordo no caso real oscila de empresa para empresa e ao longo do tempo. O valor sugerido pela empresa fornecedora dos dados é de 20 pessoas a bordo por embarque. Entretanto, considerando que os funcionários do modelo representam grupos de mais de um indivíduo, o limite utilizado nas simulações foi de 4. Esse limite permite aproximar a simulação das restrições reais e desafia o modelo a coordenar melhor o recurso já embarcado.

Seguindo o padrão descrito pela empresa fornecedora dos dados históricos, os embarques seguiram a mesma sequência da realidade. A cada sete dias, um embarque é autorizado a levar os funcionários da empresa de manutenção para a plataforma e voltar com os funcionários embarcados agendados para descer. Na simulação, os dias de embarque e desembarque não são considerados válidos para alocação de atividades.

Outro ponto relevante é que não foram considerados dois turnos de 12 horas por dia, mas apenas 1 turno de 10 horas. Primeiro, por julgar que os dois períodos não serão utilizados para a mesma atividade mas para atuar em serviços distintos. Segundo, para considerar o tempo de almoço, jantar e possíveis atrasos na passagem de uma atividade para outra, reduziu-se o tempo de 12 para 10 horas.

Como mencionado anteriormente na Seção 4.5.3, as atividades que compõem os serviços podem ser agrupadas nos blocos: Montagem e Desmontagem de andaime; Solda e Caldeiraria e Pintura. Neste trabalho, estes blocos foram codificados como descrito nas Tabelas 5.7, 5.8, 5.9.

Esses blocos foram então combinados para criar os serviços na simulação. Após combinadas, as atividades originais foram então quebradas e separadas

Tabela 5.7: Descrição das atividades do bloco andaime

	Tempo [horas]	Funcionário
Liberação de serviço	0.5	Supervisor de andaime
Montagem	15	Montador de andaime
Limpeza	1	Montador de andaime
Fechamento do serviço	0.5	Supervisor de andaime

Tabela 5.8: Descrição das atividades do bloco solda

	Tempo [horas]	Funcionário
Movimentação de equipamento	2	Caldeireiro
Liberação de serviço	0.5	Supervisor
Solda	28	Caldeireiro
Inspeção da solda	3	Inspetor de solda
Limpeza	2	Caldeireiro
Fechamento do serviço	0.5	Supervisor

Tabela 5.9: Descrição das atividades do bloco pintura

	Tempo [horas]	Funcionário
Liberação de serviço	0.5	Supervisor de pintura
Movimentação de equipamento	2	Pintor
Primeira demão	6	Pintor
Inspeção	1	Supervisor de pintura
Espera secar	6	Pintor
Segunda demão	8	Pintor
Inspeção	1	Supervisor de pintura
Espera secar	6	Pintor
Terceira demão	8	Pintor
Inspeção	1	Supervisor de pintura
Espera secar	6	Pintor
Tinta final de proteção	8	Pintor
Inspeção	1	Supervisor de pintura
Limpeza	2	Pintor
Fechamento do serviço	0.5	Supervisor de pintura

em sub-atividades de duração máxima de 2 horas. Os quatro serviços da simulação, com seus respectivos blocos e número de sub-atividades a serem alocadas, são descritos na Tabela 5.10.

Cada modelo, CPSO; CPSO+; $CPSO+_m$, foi testado em cada um dos quatro casos de simulação proposto por 20 repetições nas mesmas condições descritas nesta Seção com o intuito de comparar a dispersão dos resultados, o tempo de inicialização, o tempo de busca, e o número de épocas até convergir. Os parâmetros fixos da simulação foram resumidos na Tabela 5.11.

Tabela 5.10: Descrição dos serviços simulados

	Blocos	Número de subatividades
Serviço 01	Andaime, Solda, Andaime	43
Serviço 02	Solda, Solda	42
Serviço 03	Solda, Pintura	53
Serviço 04	Andaime, Solda, Pintura, Andaime	75

Tabela 5.11: Resumo dos parâmetros fixos das simulações

Parâmetro	Valor
Número de funcionários	42
Limite de pessoas abordo	4
Número de atividades	213
Dias de embarque	0 ; 6 ; 13 ; 20 ; 27 ; 34 ; 41 ; 48 ; 55 ; 62 ; 69 ; 76 ; 83
Tempo simulado	93 dias

5.4

Resultados

Após 20 repetições de cada simulação em estudo, as principais métricas para comparar os modelos foram agrupadas:

- Tempo médio de inicialização;
- Tempo médio de busca;
- Tempo médio de execução;
- Número de épocas para convergir;
- Melhor Valor encontrado;

O modelo a ser considerado como mais adequado ao problema proposto é aquele com valor da função objetivo maior, menos disperso e com tempo de execução considerado viável para implementação na indústria.

5.4.1

Simulação 03: espaço de busca 100% inválido

Como mencionado anteriormente na Seção 5.3.3, utilizou-se um prazo impossível para o serviço 2, 2 dias, para restringir o espaço de busca para uma completa invalidez. Os resultados agregados das 20 repetições são mostradas na Tabela 5.12 e no boxplot da Figura 5.1.

O modelo CPSO por penalização não é afetado pelas condições do espaço de busca, apresentando desempenho médio. Contudo, os modelos CPSO+

e $CPSO+_m$ necessitam de mais avaliações de partículas na inicialização, elevando a média deste tempo.

Tabela 5.12: Resultados das simulações 100% inválido

Parâmetro	CPSO	CPSO+	$CPSO+_m$
Tempo médio de inicialização	18 s	1 min 14 s	36 s
Tempo médio de busca	2 min 10 s	3 min 36 s	2 min 45 s
Tempo médio de execução	2 min 28 s	4 min 50 s	3 min 21 s
Número médio de épocas	12	11.7	11.3

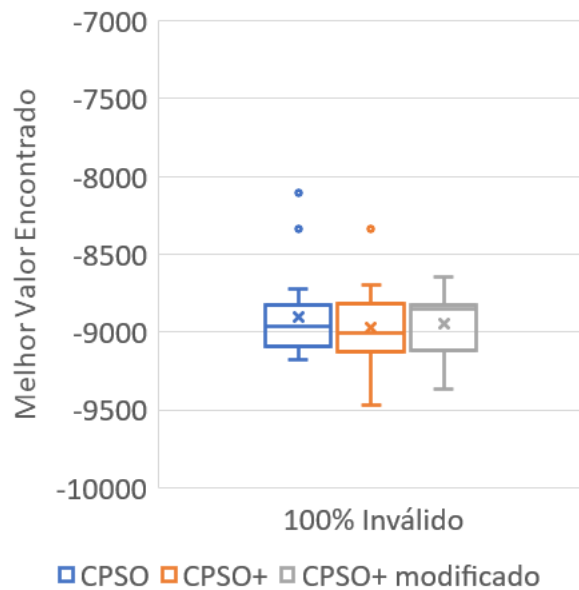


Figura 5.1: Boxplot das simulações 100% inválidos

Pelos dados da Tabela 5.12, pode-se observar que os tempos de inicialização dos modelos são consideravelmente estáveis e distintos entre si. Como no caso proposto todos os pontos são inválidos, os modelos CPSO+ e $CPSO+_m$ investem mais tempo tentando localizar pontos válidos, etapa que o modelo CPSO não realiza nesta fase do código. Contudo, as modificações propostas foram capazes de reduzir o tempo de inicialização do modelo $CPSO+_m$.

Outro ponto importante de ser ressaltado é o tempo médio de cada época. Como o mecanismo de busca com todas as partículas inválidas é idêntico para os três modelos, o número médio de épocas possui médias próximas. Contudo, o tempo de busca é também influenciado por outras etapas do código que só são realizadas pelos modelos CPSO+ e $CPSO+_m$, tornando estes mais lentos que o CPSO.

Além disso, um espaço de busca completamente inválido é também a razão dos resultados similares entre os modelos. Como todos estão utilizando o mesmo mecanismo de busca para resolver este caso de simulação, todos

acabam convergindo, em média, para o mesmo ponto ótimo. Dessa forma, não há ganhos em utilizar os modelos CPSO+ para casos 100% inválidos.

5.4.2
Simulação 04: espaço de busca 100% válido

Para simular um espaço de busca completamente válido, proposto na Seção 5.3.3, este caso de estudo não possui restrições de prazo. Desta forma, como mencionado na Seção 4.5.4, a única restrição que pode inviabilizar o cronograma foi suprimida. Os resultados agregados das 20 repetições foram dispostos na Tabela 5.13 e no boxplot da Figura 5.2.

Assim como no caso 100% inválido, a inicialização do modelo CPSO, que é indiferente ao espaço de busca, apresenta seu comportamento típico. Por outro lado, os modelos CPSO+ e $CPSO+_m$ buscam pontos inválidos para utilizar nas suas mecânicas. Essa busca pelo impossível consome um tempo significativo no modelo CPSO+ e um tempo menor no $CPSO+_m$.

Tabela 5.13: Resultados das simulações 100% válido

Parâmetro	CPSO	CPSO+	$CPSO+_m$
Tempo médio de inicialização	17 s	1 min 7 s	29 s
Tempo médio de busca	1 min 17 s	2 min 32 s	1 min 55 s
Tempo médio de execução	1 min 36 s	3 min 39 s	2 min 14 s
Número médio de épocas	19.4	14.8	14.9

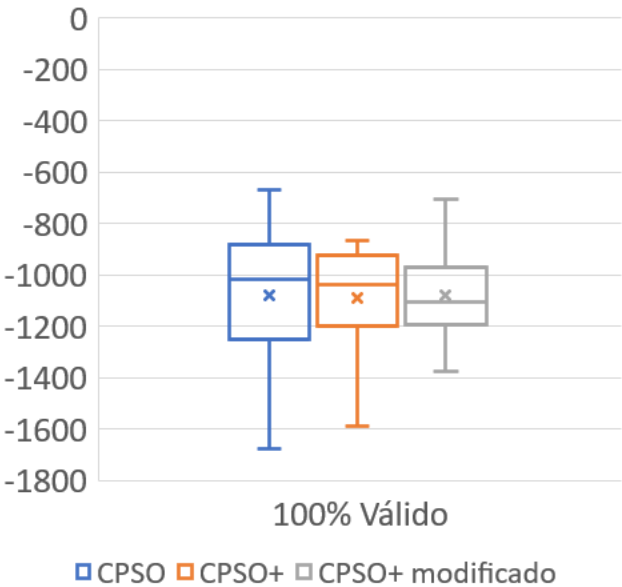


Figura 5.2: Boxplot das simulações 100% válido

De forma similar ao caso de estudo 100% inválido, os modelos, por falta de partículas inválidas, utilizam o mesmo mecanismo de busca. Portanto, como

esperado, os resultados médios dos valores ótimos encontrados é bem similar.

Além disso, pode-se observar que o tempo de busca do modelo CPSO é inferior ao dos demais modelos. Apesar de utilizarem o mesmo mecanismo de busca e, portanto, levarem o mesmo número médio de épocas para convergir, os modelos CPSO+ e $CPSO+_m$ passam por etapas onde tentam encontrar novos pontos para seus enxames auxiliares, consumindo um pouco mais de tempo na busca.

5.4.3

Simulação 05: espaço de busca predominantemente válido

No problema em estudo, um caso com espaço de busca predominantemente válido pode ser criado ao aproximar o prazo de conclusão da atividade do tempo limite de simulação. Assim, escolheu-se um prazo de 70 dias para o serviço 2 neste caso de simulação definido na Seção 5.3.3. Os resultados agregados das 20 repetições foram dispostos na Tabela 5.14 e no boxplot da Figura 5.3.

Devido à dificuldade de encontrar um ponto inválido, o modelo CPSO+ necessita de muito tempo para inicializar frente ao modelo CPSO, que apenas inicializa o enxame de busca. Devido às alterações propostas, o modelo $CPSO+_m$ não consome tanto tempo de inicialização buscando pontos para os enxames auxiliares e dilui essa tarefa na etapa de busca do algoritmo.

Tabela 5.14: Resultados das simulações predominantemente válida

Parâmetro	CPSO	CPSO+	$CPSO+_m$
Tempo médio de inicialização	18 s	1 min 13 s	36 s
Tempo médio de busca	2 min 24 s	7 min 31 s	9 min 40 s
Tempo médio de execução	2 min 42 s	8 min 44 s	10 min 16 s
Número médio de épocas	13.5	15.3	16.8

Nesta simulação, pode-se observar que, apesar de predominantemente válido, os modelos CPSO+ e $CPSO+_m$ foram capazes de encontrar pontos inválidos para suas estratégias alternativas de movimento. Apesar de diluir a busca por pontos inválidos ao longo da busca, o modelo $CPSO+_m$ foi capaz de encontrar ótimos próximos ao modelo CPSO+. Em contrapartida, o modelo CPSO, com seu mecanismo mais simples de busca, apresentou resultados abaixo dos proporcionados pelos demais modelos nesta simulação.

Contudo, os melhores resultados obtidos pelos modelos CPSO+ e $CPSO+_m$ possuem um custo computacional associado. Como parte da estratégia de busca envolve avaliar uma partícula inválida por meio de uma válida, mais avaliações são necessárias e com isto mais tempo de busca. Além disso,

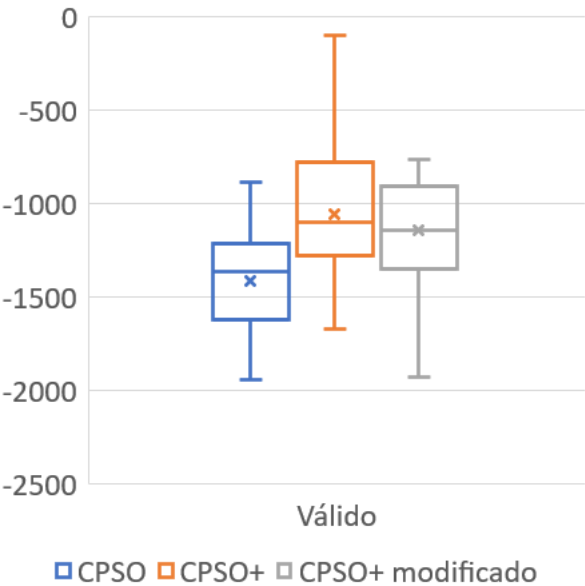


Figura 5.3: Boxplot da simulação predominantemente valida

por não investir na inicialização dos enxames auxiliares, o modelo $CPSO+_m$ consome um tempo maior na busca, degradando seu tempo total.

5.4.4
Simulação 06: espaço de busca predominantemente inválido

Para formar um espaço de busca predominantemente inválido, definiu-se um prazo muito próximo ao início das atividades, 15 dias, para concluir o serviço 2 descrito na Seção 5.3.3. Desta forma, a única solução viável é dedicar o primeiro embarque a este serviço e utilizar o menor número de funcionários possível, restringindo o número de soluções válidas. Os resultados agregados das 20 repetições constam na Tabela 5.15 e no boxplot da Figura 5.4.

Assim como no caso predominantemente válido, os modelos CPSO+ e $CPSO+_m$ s enfrentam o desafio de iniciar seus enxames auxiliares com pontos válidos e inválidos, investindo mais tempo na inicialização. Apesar disso, o $CPSO+_m$, que relaxa as necessidades de criar o enxame na inicialização, consome menos tempo que o modelo CPSO+. Por outro lado, o modelo CPSO inicializa com seu tempo padrão, pois é indiferente às condições do espaço de busca.

Tabela 5.15: Resultados das simulações predominantemente invalida

Parâmetro	CPSO	CPSO+	$CPSO+_m$
Tempo médio de inicialização	19 s	1 min 14 s	36 s
Tempo médio de busca	2 min 32 s	3 min 11 s	3 min 25 s
Tempo médio de execução	2 min 51 s	4 min 25 s	4 min
Número médio de épocas	14.3	13.1	12.3

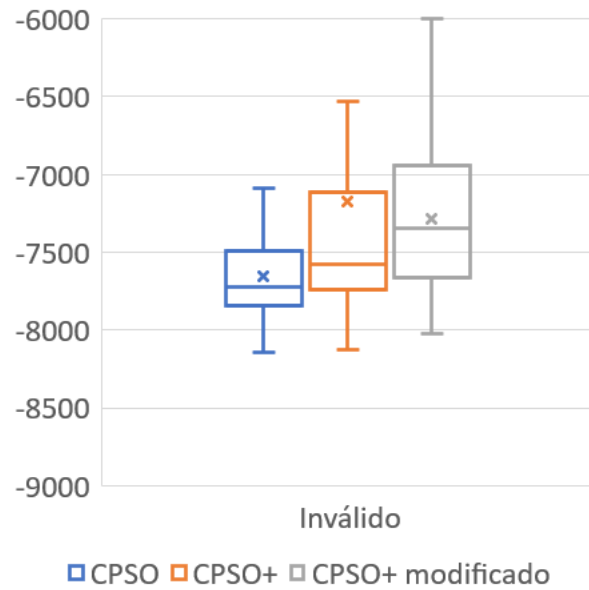


Figura 5.4: Boxplot da simulação predominantemente inválida

Por possuir pontos válidos e inválidos, os modelos CPSO+ e $CPSO+_m$ realizam uma exploração diferenciada do espaço de busca. Apesar de iniciar com um enxame de referência menor que o CPSO+, essa diferença na partida não afeta os resultados encontrados. Contudo, é um mecanismo melhor que a simples penalização utilizada pelo CPSO, o que ocasiona um pior resultado para o último.

Apesar dos bons resultados, os modelos CPSO+ e $CPSO+_m$ realizaram mais avaliações na sua busca. Essas avaliações extra em comparação ao CPSO, como demonstrado na Seção 5.3.2, acarretam em acréscimo no tempo de busca e, conseqüentemente, um tempo total de execução maior.

5.5

Análise comparativa dos resultados

Como apresentado na Seção 5.4, o modelo CPSO, por sua constância nos diversos casos de simulação, pode ser utilizado como referência nos resultados obtidos pelos modelos CPSO+ e $CPSO+_m$.

Tabela 5.16: Resultados do tempo de inicialização dos casos analisados

Tempo médio de inicialização	CPSO	CPSO+	$CPSO+_m$
Simulação 03	18 s	1 min 14 s	36 s
Simulação 04	17 s	1 min 7 s	29 s
Simulação 05	18 s	1 min 13 s	36 s
Simulação 06	19 s	1 min 14 s	36 s

Como descrito na Tabela 5.16, o modelo com menor tempo de inicialização entre todos os apresentados é o CPSO, por possuir menos etapas que os demais. Ainda na mesma lógica, o modelo $CPSO+_m$ apresenta um tempo médio de inicialização de aproximadamente duas vezes o tempo médio de inicialização do CPSO, sem oscilar significativamente com o espaço de busca. Por fim, o maior tempo de inicialização é do modelo com mais avaliações nesta etapa, o CPSO+.

Tabela 5.17: Resultados do tempo de busca dos casos analisados

Tempo médio de busca	CPSO	CPSO+	$CPSO+_m$
Simulação 03	2 min 10 s	3 min 36 s	2 min 45 s
Simulação 04	1 min 17 s	2 min 32 s	1 min 55 s
Simulação 05	2 min 24 s	7 min 31 s	9 min 40 s
Simulação 06	2 min 32 s	3 min 11 s	3 min 25 s

Além disso, observa-se na Tabela 5.17 que os modelos mais lentos na inicialização apresentam também tempo de busca superior ao do modelo CPSO. Isto ocorre devido ao maior número de avaliações de partículas nas mecânicas de busca envolvendo os enxames auxiliares. Esse tempo se torna mais expressivo conforme fica mais simples encontrar pontos válidos e inválidos, situação em que estes modelos podem realizar mais comparações com partículas de apoio e de referência. Vale ressaltar também que o modelo $CPSO+_m$, devido a sua inicialização reduzida, pode executar mais avaliações que o CPSO+ nestes casos, ocasionando um tempo de busca superior aos demais modelos, como na Simulação 05.

Tabela 5.18: Resultados médios dos casos analisados

Resultado médio das simulações	CPSO	CPSO+	$CPSO+_m$
Simulação 03	-8900	-8980	-8945
Simulação 04	-1080	-1095	-1080
Simulação 05	-1415	-1055	-1140
Simulação 06	-7650	-7175	-7285

Entretanto, a inicialização e a busca mais demoradas dos modelos CPSO+ e $CPSO+_m$ são contra balanceadas pelos resultados para o ótimo do problema. Estes modelos possuem um resultado médio superior ou igual ao modelo CPSO por penalização, como observado na Tabela 5.18.

Como não é possível identificar a priori a validade do espaço de busca nos problemas reais, é necessário ponderar entre resultados possivelmente superiores e o tempo de execução dos modelos. Devido às modificações propostas, o modelo $CPSO+_m$ apresenta, com tempos de execução moderados, resultados

similares ao modelo CPSO+. Por isso, recomenda-se a aplicação do $CPSO+_m$ no problema em estudo.

Ao comparar o modelo CPSO e o modelo $CPSO+_m$, observa-se que, para uma simulação de até 250 atividades, o modelo CPSO+ deste trabalho apresenta tempo total de execução aceitável para a aplicação com melhores resultados. Ou seja, recomenda-se o uso do modelo $CPSO+_m$ para os casos típicos do problema estudado neste trabalho.

6

Conclusão e Trabalhos Futuros

6.1

Conclusões

Com o objetivo de desenvolver um sistema de apoio à decisão na área de planejamento de manutenções embarcadas, este trabalho apresentou um novo modelo de Otimização Combinatorial por Enxame de Partículas: o CPSO+. O modelo proposto é uma adaptação do PSO+ com o CPSO para resolver o complexo problema de definir a melhor sequência de atividades embarcadas e funcionários alocados para maximizar o lucro da prestadora de serviço no período de três meses, respeitando todas as restrições de prazo de conclusão dos serviços e restrições específicas do segmento *offshore*.

Para avaliar o desempenho do CPSO+ como o de uma variante com inicialização reduzida, $CPSO+_m$, utilizou-se o CPSO como referência de comparação. Estes 3 modelos foram avaliados em 4 casos de simulação variando entre si apenas em um critério: o prazo de conclusão de um dos serviços. As simulações propostas foram: a) Um serviço com prazo impossível; b) Nenhum dos serviços com prazo de conclusão; c) Um serviço com um prazo longo para conclusão e d) Um serviço com um prazo curto para sua conclusão.

Todos os modelos avaliados foram capazes de resolver satisfatoriamente todas as simulações sugeridas, apresentando diferentes tempos de execução e valor ótimo encontrado. O modelo $CPSO+_m$ destaca-se dos demais por apresentar um tempo médio de execução moderado, 4 min 25 seg, e bons resultados para o ótimo encontrado.

6.2

Trabalhos Futuros

Apesar de explorar diversos aspectos do problema proposto, este trabalho não modela perfeitamente a situação real. Para melhor adequar o modelo ao problema e possivelmente encontrar melhores resultados, futuros trabalhos podem abordar:

- Modelagens que considerem mais de um funcionário por atividade;

- Processos diferentes para a transformação do vetor de ordem em um cronograma válido;
- Avaliação de outros modelos de otimização para problemas de ordem;
- Inclusão de uma avaliação multiobjetivo para melhor avaliar os objetivos propostos: maximizar conclusão de serviços e minimizar o custo com funcionários.
- Inclusão de uma avaliação multiobjetivo para maximizar o lucro da empresa enquanto se maximiza o volume médio de funcionários embarcados.

Referências

- [1] ADAMS, R.. **As condições de trabalho no setor petrolífero offshore: Uma revisão de literatura.** XII CNEG & III INOVARSE, 2016.
- [2] ARRAM, A.; AYOB, M.. **A novel multi-parent order crossover in genetic algorithm for combinatorial optimization problems.** Computers & Industrial Engineering, 133:267 – 274, 2019.
- [3] BALAPRAKASH, P.; MOROZOV, V.; KETTIMUTHU, R.; KUMARAN, K. ; FOSTER, I.. **Improving data transfer throughput with direct search optimization.** Em: 2016 45TH INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING (ICPP), p. 248–257, Aug 2016.
- [4] BALERA, J. M.; DE SANTIAGO JÚNIOR, V. A.. **A systematic mapping addressing hyper-heuristics within search-based software testing.** Information and Software Technology, 114:176 – 189, 2019.
- [5] BI, Z.. **Chapter 1 - overview of finite element analysis.** Em: Bi, Z., editor, FINITE ELEMENT ANALYSIS APPLICATIONS, p. 1 – 29. Academic Press, 2018.
- [6] CHE, H.; WANG, J.. **A collaborative neurodynamic approach to global and combinatorial optimization.** Neural Networks, 114:15 – 27, 2019.
- [7] DAI, M.; TANG, D.; GIRET, A. ; SALIDO, M. A.. **Multi-objective optimization for energy-efficient flexible job shop scheduling problem with transportation constraints.** Robotics and Computer-Integrated Manufacturing, 59:143 – 157, 2019.
- [8] DANTZIG, G. B.. **The nature of mathematical programming.** Mathematical Programming Glossary, INFORMS Computing Society, 2014.
- [9] EBBESEN, S.; KIWITZ, P. ; GUZZELLA, L.. **A generic particle swarm optimization matlab function.** Em: 2012 AMERICAN CONTROL CONFERENCE (ACC), p. 1519–1524, June 2012.

- [10] FANJUL-PEYRO, L.; RUIZ, R. ; PEREA, F.. Reformulations and an exact algorithm for unrelated parallel machine scheduling problems with setup times. *Computers & Operations Research*, 2018.
- [11] FLESZAR, K.; HINDI, K. S.. Algorithms for the unrelated parallel machine scheduling problem with a resource constraint. *European Journal of Operational Research*, 271(3):839 – 848, 2018.
- [12] FREITAS, C. M. D.; SOUZA, C. A. V. D.; MACHADO, J. M. H. ; PORTO, M. F. D. S.. Acidentes de trabalho em plataformas de petróleo da bacia de campos. *Caderno de Saúde Pública*, 17(1):117–130, 2001.
- [13] GAREY, M. R.. *Computers and intractability : A guide to the theory of np-completeness*, freeman. Fundamental, 1997.
- [14] GAJARSKÝ, J.; HLINĚNÝ, P.; KOUTECKÝ, M. ; ONN, S.. Parameterized shifted combinatorial optimization. *Journal of Computer and System Sciences*, 99:53 – 71, 2019.
- [15] GARAIX, T.; GONDRAN, M.; LACOMME, P.; MURA, E. ; TCHERNEV, N.. Workforce scheduling linear programming formulation. *IFAC-PapersOnLine*, 51(11):264 – 269, 2018. 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018.
- [16] GIFFLER, B.; THOMPSON, G. L.. Algorithms for solving production-scheduling problems. *Operations Research*, 8(4):487–503, 1960.
- [17] GOERIGK, M.; MAHER, S. J.. Generating hard instances for robust combinatorial optimization. *European Journal of Operational Research*, 2019.
- [18] GONG, G.; DENG, Q.; GONG, X.; LIU, W. ; REN, Q.. A new double flexible job-shop scheduling problem integrating processing time, green production, and human factor indicators. *Journal of Cleaner Production*, 174:560 – 576, 2018.
- [19] GUIJARRO, F.; MARTÍNEZ-GÓMEZ, M. ; VISBAL-CADAVID, D.. A combined genetic algorithm and inverse data envelopment analysis model for target setting in mergers. *Expert Systems with Applications: X*, p. 100012, 2019.
- [20] HAMDI, M.; ZAIED, M.. Resource allocation based on hybrid genetic algorithm and particle swarm optimization for d2d multicast communications. *Applied Soft Computing*, 83:105605, 2019.

- [21] IMRAN, M.; HASHIM, R. ; KHALID, N. E. A.. **An overview of particle swarm optimization variants**. Procedia Engineering, 53:491 – 496, 2013. Malaysian Technical Universities Conference on Engineering & Technology 2012, MUCET 2012.
- [22] KOHLER, M. R.. **Pso+: Algoritmo com base em enxame de partículas para problemas com restrições lineares e não lineares**. Tese de doutorado, PUC-RIO, 2017.
- [23] KENNEDY, J.; EBERHART, R.. **Particle swarm optimization**. Em: PROCEEDINGS OF ICNN'95 - INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, volume 4, p. 1942–1948 vol.4, Nov 1995.
- [24] **Lei nº 5.811, de 11 de outubro, 1972**. Dispõe sobre o regime de trabalho dos empregados nas atividades de exploração, perfuração, produção e refinação de petróleo, industrialização do xisto, indústria petroquímica e transporte de petróleo e seus derivados por meio de dutos.
- [25] LENDL, S.; ĆUSTIĆ, A. ; PUNNEN, A. P.. **Combinatorial optimization with interaction costs: Complexity and solvable cases**. Discrete Optimization, 33:101 – 117, 2019.
- [26] LU, H.; SHI, J.; FEI, Z.; ZHOU, Q. ; MAO, K.. **Analysis of the similarities and differences of job-based scheduling problems**. European Journal of Operational Research, 270(3):809 – 825, 2018.
- [27] TOMPKINS, M. F.. **Optimization techniques for task allocation and scheduling in distributed multi-agent operations**. Dissertação de mestrado, Massachusetts Institute of Technology, 2003.
- [28] MARTINS, S. M. R.. **O trabalho offshore: Um estudo sobre as repercussões do confinamento nos trabalhadores das plataformas de petróleo na bacia de campos, rj**. Dissertação de mestrado, UNIVERSIDADE ESTADUAL DO NORTE FLUMINENSE, 2006.
- [29] MICHALEWICZ, Z.; JANIKOW, C. Z.. **Genocop: A genetic algorithm for numerical optimization problems with linear constraints**. Commun. ACM, 39(12es), Dec. 1996.
- [30] MICHALEWICZ, Z.; NAZHIYATH, G.. **Genocop iii: a co-evolutionary algorithm for numerical optimization problems with nonlinear constraints**. Em: PROCEEDINGS OF 1995 IEEE INTERNATIONAL CONFERENCE ON EVOLUTIONARY COMPUTATION, volume 2, p. 647–651 vol.2, Nov 1995.

- [31] MIRAGLIOTTA, G.; SIANESI, A.; CONVERTINI, E. ; DISTANTE, R.. **Data driven management in industry 4.0: a method to measure data productivity.** IFAC-PapersOnLine, 51(11):19 – 24, 2018. 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018.
- [32] PROMINP. **Participação do setor de petróleo e gás chega a 13% do pib brasileiro**, 2013.
- [33] SELÇUK, T.; ALKAN, A.. **Detection of microaneurysms using ant colony algorithm in the early diagnosis of diabetic retinopathy.** Medical Hypotheses, 129:109242, 2019.
- [34] SHEN, L.; DAUZÈRE-PÉRÈS, S. ; NEUFELD, J. S.. **Solving the flexible job shop scheduling problem with sequence-dependent setup times.** European Journal of Operational Research, 265(2):503 – 516, 2018.
- [35] SIOURIS, S.; BLAKEY, S.. **Fitness functions for evolutionary optimization of rate parameters in chemically reacting systems.** Chemical Engineering Science, 2018.
- [36] VARGAS, D.; LEMONGE, A.; BARBOSA, H. ; BERNARDINO, H.. **Um algoritmo baseado em evolução diferencial para problemas de otimização estrutural multiobjetivo com restrições.** Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería, 32(2):91 – 99, 2016.
- [37] YOO, S.; HARMAN, M.. **Pareto efficient multi-objective test case selection.** Em: PROCEEDINGS OF THE 2007 INTERNATIONAL SYMPOSIUM ON SOFTWARE TESTING AND ANALYSIS, ISSTA '07, p. 140–150, New York, NY, USA, 2007. ACM.
- [38] ZHANG, A.; CHEN, Y.; CHEN, L. ; CHEN, G.. **On the np-hardness of scheduling with time restrictions.** Discrete Optimization, 28:54 – 62, 2018.
- [39] ZHANG, H.; LI, X.; LI, H. ; HUANG, F.. **Particle swarm optimization-based schemes for resource-constrained project scheduling.** Automation in Construction, 14(3):393 – 404, 2005. International Conference for Construction Information Technology 2004.
- [40] ZHANG, M.; TAN, Y.; ZHU, J.; CHEN, Y. ; LIU, H.. **Modeling and simulation of improved artificial bee colony algorithm with data-driven optimization.** Simulation Modelling Practice and Theory, 93:305 – 321, 2019. Modeling and Simulation of Cloud Computing and Big Data.