



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO ACADÊMICO DO AGRESTE
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

LEONARDO JOSÉ CAVALCANTE VASCONCELOS

**UMA HEURÍSTICA BASEADA EM PARETO LOCAL SEARCH PARA UM
PROBLEMA DE SEQUENCIAMENTO DE TAREFAS EM AMBIENTE FLOW
SHOP COM CONSIDERAÇÕES AMBIENTAIS**

Caruaru

2023

LEONARDO JOSÉ CAVALCANTE VASCONCELOS

**UMA HEURÍSTICA BASEADA EM PARETO LOCAL SEARCH PARA UM
PROBLEMA DE SEQUENCIAMENTO DE TAREFAS EM AMBIENTE FLOW
SHOP COM CONSIDERAÇÕES AMBIENTAIS**

Dissertação apresentada ao Programa de Pós-graduação em Engenharia de Produção da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de mestre em Engenharia de Produção. Área de Concentração: Otimização e Gestão da Produção.

Orientador: Prof. Dr. Luciano Carlos Azevedo da Costa

Coorientador: Prof. Dr. Raphael Harry Frederico Ribeiro Kramer

Caruaru

2023

Catálogo na fonte:
Bibliotecária – Nasaré Oliveira - CRB/4 - 2309

V237h Vasconcelos, Leonardo José Cavalcante.
Uma heurística baseada em Pareto local search para um problema de sequenciamento de tarefas em ambiente *flow shop* com considerações ambientais. / Leonardo José Cavalcante Vasconcelos. – 2023.
74 f.; il.: 30 cm.

Orientador: Luciano Carlos Azevedo da Costa.
Coorientador: Raphael Harry Frederico Ribeiro Kramer
Dissertação (Mestrado) – Universidade Federal de Pernambuco, CAA, Programa de Pós- Graduação em Engenharia de Produção, 2023.
Inclui Referências.

1. Otimização multiobjetivo. 2. Programação matemática. 3. Algoritmo computacional. I. Costa, Luciano Carlos Azevedo da (Orientador). II. Kramer, Raphael Harry Frederico Ribeiro (Coorientador). III. Título.

CDD 658.5 (23. ed.) UFPE (CAA 2023-093)

LEONARDO JOSÉ CAVALCANTE VASCONCELOS

**UMA HEURÍSTICA BASEADA EM PARETO LOCAL SEARCH PARA UM
PROBLEMA DE SEQUENCIAMENTO DE TAREFAS EM AMBIENTE FLOW SHOP
COM CONSIDERAÇÕES AMBIENTAIS**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção da Universidade Federal de Pernambuco, Centro Acadêmico do Agreste, como requisito parcial para obtenção do título de Mestre em Engenharia de Produção. Área de concentração: Otimização e Gestão da Produção.

Aprovada em: 24/11/2023.

BANCA EXAMINADORA

Prof. Dr. Luciano Carlos Azevedo da Costa (Orientador)
Universidade Federal da Paraíba - UFPB

Prof. Dr. Walton Pereira Coutinho (Examinador Interno)
Universidade Federal de Pernambuco - UFPE

Prof. Dr. Yuri Laio Teixeira Veras Silva (Examinador Externo)
Universidade Federal de Campina Grande – UFCG

Dedico este trabalho a minha mãe, Janicleide Cavalcante, por todo exemplo de força e superação para conclusão de mais uma etapa na minha jornada acadêmica.

AGRADECIMENTOS

Primeiramente, gostaria de agradecer a Deus por ter me dado força, paciência e motivação para concluir mais uma etapa de minha formação.

Aos meus pais, Janicleide Cavalcante e Manoel Vasconcelos, pelo imenso apoio ao longos dos anos e por sempre me incentivarem a estudar. Obrigado por todo carinho.

À minha noiva e futura esposa Eloyse Viana, que me acompanha desde o momento que fui aprovado na seleção do mestrado. Agradeço por toda compreensão nos momentos em que eu necessitava me dedicar na pesquisa pelos conselhos em momentos difíceis.

Ao meu orientador, professor Dr. Luciano Costa, por toda compreensão ao longo da orientação e dedicação na pesquisa e na docência.

Ao meu coorientador, professor Dr. Raphael Kramer pela disponibilidade em contribuir com a pesquisa e repassar o conhecimento e experiência.

Aos professores Dr. Luciano Costa, Dr. Walton Coutinho e Dr. Lúcio Câmara e Silva por me apresentarem as áreas de Pesquisa Operacional e Otimização Combinatória.

À Rupert Aeronaves, especialmente a Ridley Rupert Jones, pela flexibilidade da jornada de trabalho que me permitiram concluir as disciplinas do programa.

Ao meu colega Josenildo Ferreira do grupo de pesquisa em Engenharia Econômica e Otimização Combinatória (GEEOC) pelas discussões de implementações de medidas de desempenho.

Aos demais professores do programa e aos colegas de turma de mestrado pelas conversas acadêmicas e momentos de descontração, que sem dúvidas foram fundamentais para a construção deste momento.

Aos demais que direta ou indiretamente contribuíram para o desenvolvimento desta pesquisa.

RESUMO

Esta dissertação trata do problema Green permutation flow shop de m máquinas com tempos de *setup* dependentes da sequência (GPFSP TSDS) biobjetivo. Os dois objetivos considerados no problema, conflitantes entre si, referem-se a minimização do *makespan*, como indicador do nível de serviço, e a minimização do consumo total de energia, como indicador ambiental, que é função da velocidade das máquinas. Uma abordagem multiobjetivo baseada no método *Two-Phase Pareto Local Search* (2PPLS) é utilizada na geração do conjunto de soluções do problema. Durante a primeira fase, um conjunto inicial é gerado a partir de uma metaheurística com NEHT-RB e busca local *Variable Neighborhood Descend* (VND). Também foi proposto um algoritmo de economia de energia. Em seguida, as soluções presentes neste conjunto são exploradas por meio de um *Pareto Local Search* (PLS) adaptado. Os resultados computacionais mostram que a abordagem utilizada em duas fases é promissora para obtenção de melhores fronteiras em comparação com apenas uma fase. Os indicadores Hipervolume (H), Medida R (R), Número de soluções não dominadas (NDS) obtidas e tempo de execução, foram utilizados na avaliação dos conjuntos de soluções geradas.

Palavras-chave: green permutation flowshop; otimização multiobjetivo; two-phase pareto local search.

ABSTRACT

This dissertation deals with the biobjective m machine permutation flowshop problem with sequence dependent setup times (GPFSP TSDS). The two objectives considered in the problem, conflicting with each other, refer to the minimization of *makespan*, as an indicator of the level of service, and the minimization of total energy consumption, as an environmental indicator, which is a function of the speed of the machines. A multi-objective approach based on the 2PPLS method is used to generate the set of solutions to the problem. During the first phase, an initial set is generated from a metaheuristic with NEHT-RB and local search VND. An energy saving algorithm was also proposed. Then, the solutions present in this set are explored through an adapted PLS. The computational results show that the approach used in two phases is promising for obtaining better boundaries compared to just one phase. The indicators Hypervolume (H), R Measure (R), number of non dominated solutions (NDS) obtained and execution time were used to evaluate the sets of solutions generated.

Keywords: green permutation flowShop; multi-objective optimization; two-phase pareto local search.

LISTA DE FIGURAS

Figura 1 – Hipervolume da Fronteira de Pareto	27
Figura 2 – Diagrama de Gantt de 3 tarefas em 3 máquinas	33
Figura 3 – Representação dos gaps	39
Figura 4 – Movimentos para gerar vizinhança	47
Figura 5 – Fronteiras obtidas para NEHT-RB + SWAP e NEHT-RB + SWAP + ECON. DE ENERGIA	52
Figura 6 – Fronteiras obtidas para NEHT-RB + SWAP + ECON. DE ENERGIA e NEHT-RB + VND + ECON. DE ENERGIA	54
Figura 7 – Fronteiras obtidas para NEHT-RB + SWAP e NEHT-RB + SWAP + 2 ^a FASE	55
Figura 8 – Fronteiras obtidas para NEHT-RB + SWAP + 2 ^a FASE e NEHT-RB + SWAP + ECON. DE ENERGIA ADAPTADO + 2 ^a FASE	57
Figura 9 – Fronteiras obtidas para NEHT-RB + VND + 2 ^a FASE e NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO + 2 ^a FASE	58
Figura 10 – Resumo das Fronteiras obtidas com as diferentes configurações	59

LISTA DE QUADROS

Quadro 1 – Indicadores de Desempenho	51
------------------------------------------------	----

LISTA DE TABELAS

Tabela 1 – Resumo das principais referências utilizadas da revisão da literatura	32
Tabela 2 – Resultados comparativos entre NEHT-RB + SWAP e NEHT-RB + SWAP + ECON. DE ENERGIA	52
Tabela 3 – Resultados comparativos entre NEHT-RB + SWAP + ECON. DE ENER- GIA e NEHT-RB + VND + ECON. DE ENERGIA	53
Tabela 4 – Resultados comparativos entre NEHT-RB + SWAP e NEHT-RB + SWAP + 2ª FASE	55
Tabela 5 – Resultados comparativos entre NEHT-RB + SWAP + 2ª FASE e NEHT- RB + SWAP + ECON. DE ENERGIA ADAPTADO + 2ª FASE	56
Tabela 6 – Resultados comparativos entre NEHT-RB + VND + 2ª FASE e NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO + 2ª FASE	58

LISTA DE SIGLAS

2PPLS	<i>Two-Phase Pareto Local Search</i>
CO ₂	Dióxido de Carbono
FP	Fronteira de Pareto
FTP	Função de Tchebycheff Ponderada
FTPs	Funções de Tchebycheff Ponderadas
GPFSTSDS	Green permutation flow shop de m máquinas com tempos de <i>setup</i> dependentes da sequência
MILP	<i>Mixed Integer Linear Programming</i>
NDS	Número de soluções não dominadas
PCP	Planejamento e Controle de Produção
PFSTSDS	Permutation flow shop de m máquinas com tempos de <i>setup</i> dependentes da sequência
PLS	<i>Pareto Local Search</i>
POM	Problema de Otimização Multiobjetivo
POMs	Problemas de Otimização Multiobjetivo
TEC	Consumo Total de Energia
VND	<i>Variable Neighborhood Descend</i>

LISTA DE SÍMBOLOS

j, k	Índices para tarefas $j, k = 1, 2, \dots, n$
i	Índice para máquinas $i = 1, 2, \dots, m$
l	Índice para níveis de velocidade de processamento $l = 0, 1, 2$ (velocidade lenta, normal e rápida respectivamente)
n	Número de tarefas
m	Número de máquinas
sp_l	Fator de velocidade de processamento l
p_{ij}	Tempo de processamento da tarefa j na máquina i
st_{ijk}	Tempo de <i>setup</i> para a mudança da tarefa j para a tarefa k na máquina i
λ_l	Fator de conversão para a velocidade de processamento l
ϑ_l	Fator de conversão para o tempo ocioso na máquina i
π_i	Potência da máquina i
M	Um número grande
s_{ij}	Tempo inicial da tarefa j na máquina i
o_{ij}	Tempo de <i>setup</i> para a tarefa j na máquina i , $i \neq 1$, se a tarefa j é a primeira tarefa na sequência
c_{ij}	Tempo de conclusão da tarefa j na máquina i
θ_i	Tempo ocioso na máquina i
∂_j	1 se a tarefa j é a primeira tarefa, 0 se contrário
x_{jk}	1 se a tarefa j precede a tarefa k , 0 se contrário ($j \neq k$)
y_{ijl}	1 se a tarefa j é processada no nível de velocidade l na máquina i , 0 se contrário

C_{max}	<i>Makespan</i> , o tempo de conclusão da última tarefa denominada n na última máquina denominada m
TEC	Consumo total de energia

SUMÁRIO

1	INTRODUÇÃO	16
1.1	DEFINIÇÃO DO TEMA	16
1.2	JUSTIFICATIVA	17
1.3	OBJETIVOS	18
1.3.1	Objetivo geral	18
1.3.2	Objetivos específicos	18
1.4	ESTRUTURA DO TRABALHO	19
2	FUNDAMENTAÇÃO TEÓRICA	20
2.1	OTIMIZAÇÃO MULTIOBJETIVO	20
2.1.1	Métodos para resolução de problemas de otimização multiobjetivo	21
2.1.1.1	<i>Métodos clássicos</i>	22
2.1.1.2	<i>Técnicas aproximativas</i>	23
2.1.2	Avaliação do desempenho dos algoritmos multiobjetivo	24
2.1.2.1	<i>Hipervolume</i>	26
2.1.2.2	<i>Medida R</i>	27
2.2	REVISÃO DA LITERATURA	29
3	METODOLOGIA	33
3.1	DESCRIÇÃO DO PROBLEMA	33
3.2	MÉTODOS DE RAMEZANIAN	36
3.2.1	Comentários sobre o algoritmo de economia de energia proposto por Ramezani	41
3.3	DESCRIÇÃO DO MÉTODO	43
3.3.1	Primeira fase	43
3.3.2	Segunda fase	47
4	RESULTADOS E DISCUSSÕES	50
4.1	ANÁLISE DO ALGORITMO DE ECONOMIA DE ENERGIA ADAPTADO	51
4.2	ANÁLISE DO IMPACTO DA BUSCA LOCAL PROPOSTA	53
4.3	ANÁLISE DO IMPACTO DA 2ª FASE PARA UMA 1ª FASE OBTIDA POR NEHT-RB + SWAP	54

4.4	ANÁLISE DO IMPACTO DA 2ª FASE PARA A 1ª FASE NEHT-RB + VND E PARA A 1ª FASE NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO	57
4.5	RESUMO DAS CONFIGURAÇÕES	59
5	CONCLUSÕES	60
	REFERÊNCIAS	62

1 INTRODUÇÃO

1.1 DEFINIÇÃO DO TEMA

No Planejamento e Controle de Produção (PCP), as principais decisões envolvidas no nível operacional são a designação de tarefas (*jobs*) a máquinas e a programação (*scheduling*) da ordem de execução das tarefas em cada máquina. Estas decisões têm por objetivo, determinar a sequência de processamento das tarefas e o instante de início e término dos seus processamentos (ARENALES et al., 2007).

Os sistemas de produção podem ser caracterizados por vários fatores: o número de recursos ou máquinas, suas características e configuração, o nível de automação, o tipo de sistema de manuseio de materiais, etc. As diferenças em todas essas características dão origem a um grande número de diferentes modelos de planejamento e programação (PINEDO, 2005).

A combinação destas características fez com que fosse necessário o desenvolvimento de sistemas flexíveis de manufatura. Estes sistemas funcionam de forma integrada, controlados por computadores, dispositivos automatizados de manuseio de materiais e máquinas de controle numérico. A meta destes sistemas é atingir a eficiência da produção automatizada de grande escala (ARENALES et al., 2007).

Em geral, na literatura, para a avaliação da qualidade de um programa de produção, as principais medidas de desempenho (objetivos) estão relacionadas a aspectos produtivos, tais como, o *makespan*, atraso máximo e número de tarefas atrasadas. Porém, a crescente preocupação com os impactos ambientais e aumento do consumo de energia, têm motivado as empresas a se tornarem mais sustentáveis (verdes) ao longo dos anos (RAMEZANIAN; VALI-SIAR; JALALIAN, 2019).

Com relação as empresas, as indústrias manufatureiras são as maiores consumidoras de energia do mundo, respondendo por cerca de metade do consumo global de energia (MOUZON; YILDIRIM; TWOMEY, 2007) e um terço das emissões totais de dióxido de carbono (KONSTANTINAVICIUTE; BOBINAITE, 2015). Neste contexto, nos últimos anos, muitas empresas têm buscado adotar medidas para reduzir o impacto de suas operações no meio ambiente. Uma das formas para reduzir o consumo de energia é controlar a velocidade das atividades de produção durante o processo de fabricação. Mansouri, Aktas e Besikci (2016), Mansouri e Aktas (2016) e Ramezian, Vali-Siar e Jalalian (2019) utilizam esta abordagem.

Normalmente, quanto maior a velocidade de trabalho em uma máquina, maior será o seu

consumo de energia. Desta forma, uma medida para redução do consumo de energia seria a redução da velocidade das máquinas. No entanto, esta medida, poderia levar o tempo de conclusão (*makespan*) da operação a ser aumentado. Desta forma, os objetivos são conflitantes e a otimização de uma medida deve levar em consideração o *trade-off* caracterizado pela otimização dos múltiplos objetivos, em que a melhoria de um critério causa a deterioração do outro critério. Portanto, uma abordagem multiobjetivo é aplicável e o desenvolvimento de métodos específicos pode ser necessário.

O problema estudado nesta dissertação é o GPFSPSTSDS biobjetivo, o qual pode ser denotado, considerando a notação de três campos de Graham ($\alpha/\beta/\gamma$) (GRAHAM et al., 1979), como $Fm/STsd/C_{max}, TEC$. Isso significa que n jobs (tarefas) devem ser processados em m máquinas sequencialmente, com tempos de *setup* dependentes da sequência e tem por objetivo a minimização do makespan (C_{max}) e do Consumo Total de Energia (TEC) – detalhado na Seção 3.1. O GPFSPSTSDS é NP-difícil, pois ele estende o $F2/STsd/C_{max}$ que é NP-difícil (GUPTA; DARROW, 1986). Devido à complexidade combinatória e restrições de tempo, os métodos heurísticos são viáveis para resolver problemas de grande porte. Neste contexto, o presente trabalho propõe uma metaheurística multiobjetivo para lidar com o GPFSPSTSDS.

1.2 JUSTIFICATIVA

A escassez de recursos naturais, o aumento do consumo global de energia e as grandes variações em seu custo são boas razões para que as indústrias manufatureiras se tornem mais sustentáveis (verdes). As indústrias são as principais consumidoras de energia, utilizando cerca de metade da energia do mundo (MOUZON; YILDIRIM; TWOMEY, 2007) e sendo responsáveis por um terço das emissões totais de Dióxido de Carbono (CO₂) (KONSTANTINAVICIUTE; BOBINAITE, 2015). Desta forma, a pressão aumenta sobre as indústrias para melhorar seu consumo de energia e reduzir seu impacto prejudicial ao meio ambiente. Além disso, o crescimento da população mundial e o aumento das demandas vêm alterando os padrões de consumo de energia (FENG et al., 2016). Isto que significa que as indústrias precisam de mais energia para responder às novas demandas. Portanto, é necessário o uso adequado de energia e a programação verde de chão de fábrica está se tornando cada vez mais necessária para uma fabricação sustentável.

Uma abordagem para obter esta programação sustentável é utilizando formas eficazes de reduzir seu consumo de energia elétrica. Exemplos de tais formas incluem desligar seletivamente

as máquinas durante o tempo ocioso quando viável ou operá-las em velocidades permitidas pelas metas de nível de serviço definidas (GAHM et al., 2016).

Existem poucos trabalhos de Permutation flow shop de m máquinas com tempos de *setup* dependentes da sequência (PFSPTSDS) considerando o consumo de energia. Apenas os trabalhos de Lu et al. (2017) e Jiang e Wang (2019) (ambos utilizando a abordagem *on/off* - ligar e desligar as máquinas) e Ramezani, Vali-Siar e Jalalian (2019) consideraram o aspecto ambiental no problema. Ramezani, Vali-Siar e Jalalian (2019) propõe para o GPFSPSTSDS, uma heurística com o intuito de economia de energia sem prejuízos ao tempo de conclusão (*makespan*) da operação. Desta forma, observa-se a necessidade de discussão sobre GPFSPSTSDS, que é abordada nesta dissertação. Este trabalho revisita o trabalho de Ramezani, Vali-Siar e Jalalian (2019), adicionando novas estruturas de vizinhança na busca local, propondo um novo algoritmo adaptado para a redução do consumo de energia e propondo uma abordagem multiobjetivo baseada em 2PPLS.

Nesta pesquisa, é estudado um problema de programação de produção (*scheduling*) do tipo GPFSPSTSDS. Dois objetivos são considerados: a minimização do *makespan* (tempo que decorre do início ao fim do trabalho), como indicador do nível de serviço, e a minimização do TEC, como indicador ambiental. Devido ao *trade-off* existente entre os objetivos de natureza conflitantes, os tomadores de decisão podem considerar diferentes soluções que favoreçam a minimização de um ou outro objetivo por vez. Estas soluções podem ser geradas por meio de um método multiobjetivo. De posse destas soluções, o decisor pode escolher a que seja mais adequada a sua realidade e ao seu processo de fabricação.

1.3 OBJETIVOS

1.3.1 Objetivo geral

Desenvolver um algoritmo heurístico para solucionar o GPFSPSTSDS biobjetivo.

1.3.2 Objetivos específicos

- Definir um procedimento construtivo para a geração de soluções iniciais;
- Aplicar buscas locais com estruturas de vizinhança existentes na literatura;

- Propor uma busca local para minimizar o objetivo relacionado ao consumo total de energia;
- Aplicar o 2PPLS;
- Testar o algoritmo proposto em diferentes instâncias constantes na literatura;
- Analisar os resultados obtidos nas duas fases com métricas para problemas multiobjetivo.

1.4 ESTRUTURA DO TRABALHO

O restante do trabalho é estruturado conforme segue: O Capítulo 2 lista e discute alguns dos principais conceitos relacionados a temática da Otimização Multiobjetivo e trabalhos relacionados à temática do Problema de *Scheduling* resolvido no contexto ambiental. A descrição do problema e do método proposto na resolução do problema são apresentados no Capítulo 3. O Capítulo 4 apresenta e discute os resultados obtidos. Por fim, o Capítulo 5 conclui o trabalho e apresenta perspectivas para futuras pesquisas.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 OTIMIZAÇÃO MULTIOBJETIVO

No cotidiano, os problemas de otimização consideram mais de um objetivo e estes por sua vez possuem funções objetivo que podem ser conflitantes entre si (FERREIRA, 2020). Farias (2008) e Bastos (2017) apresentam como exemplo de conflito de objetivos a compra de um produto (como um carro ou um computador) onde a aquisição ótima seja ter um custo mínimo e máximo desempenho. Nestes casos, a otimização de um objetivo pode acarretar na piora do outro.

Estes problemas são definidos como Problemas de Otimização Multiobjetivo (POMs) e possuem um conjunto de funções objetivo a serem minimizadas ou maximizadas e restrições que devem ser respeitadas (BASTOS, 2017).

Conforme apresentado em Ehrgott (2005), o Problema de Otimização Multiobjetivo (POM) pode ser definido em um espaço de soluções X e visa otimizar simultaneamente dois ou mais objetivos conflitantes. Um POM, pode ser de minimização ou maximização (2.1) de M objetivos e pode ser representado da seguinte forma (DEB, 2011):

$$\min \text{ ou } \max f_m(x), \quad m = 1, \dots, M \quad (2.1)$$

$$g_j(x) \geq 0, \quad j = 1, \dots, J \quad (2.2)$$

$$h_k(x) = 0, \quad k = 1, \dots, K \quad (2.3)$$

$$x_i^L \leq x_i \leq x_i^U, \quad i = 1, \dots, n \quad (2.4)$$

As J desigualdades (2.2) e as K igualdades (2.3) são chamadas de restrições. Em 2.4, x_i^L e x_i^U , representam o valor mínimo e máximo para a variável de decisão x_i . Uma solução $x \in R^n$ é um vetor de n variáveis de decisão $x = (x_1, x_2, \dots, x_n)^T$. As soluções que satisfazem as restrições e os limites das variáveis definem o Espaço Viável das variáveis de decisão $S \subset R^n$. Como o problema é multiobjetivo, as funções objetivo formam um espaço multidimensional em relação ao espaço das variáveis de decisão. Este espaço é definido como Espaço de Objetivos $Z \subset R^M$.

As soluções ótimas em POMs podem ser definidas a partir de um conceito matemático de ordenação parcial (DEB, 2011). No jargão da otimização multiobjetivo, o termo dominância é usado para esse propósito. Bardanachvili (2006) e Ticona (2008) apresentam o conceito de dominância: quando uma solução x_1 é melhor ou igual a outra solução x_2 em relação a

todas as funções objetivo ($f(x_1) \leq f(x_2)$) e estritamente melhor em relação a pelo menos uma das funções objetivo ($f(x_1) < f(x_2)$), diz-se que a solução x_1 domina a solução x_2 ($x_1 \prec x_2$). Quando uma solução é estritamente melhor em relação a pelo menos uma das funções objetivas, mas é estritamente pior com relação a pelo menos uma das demais funções objetivo, diz-se que elas são soluções não dominadas (BARDANACHVILI, 2006). Por sua vez, o conjunto das soluções não dominadas é chamada de Fronteira de Pareto (FP) (COSTA, 2015).

2.1.1 Métodos para resolução de problemas de otimização multiobjetivo

Diante da existência de objetivos conflitantes, o principal desafio na otimização multiobjetivo é encontrar soluções viáveis que otimizem todos os objetivos simultaneamente. Conforme mencionado por Horn (1997), ao lidar com problemas multiobjetivos, identificam-se dois desafios distintos: a busca por soluções e o processo de tomada de decisões.

A busca por soluções refere-se ao processo de otimização no qual o conjunto de soluções viáveis, ou factíveis, é direcionado para identificar o conjunto de soluções Pareto-Ótimas.

Já o segundo desafio, a tomada de decisões, envolve a escolha de um critério apropriado para selecionar uma solução do conjunto Pareto-Ótimo. Esse critério será utilizado pelo responsável ou decisor no processo decisório, permitindo-lhe ponderar entre as distintas soluções conflitantes e chegar a uma decisão fundamentada.

Dessa forma, dentro do contexto da Otimização Multiobjetivo, a escolha do método para resolver os problemas é feita de acordo com a função desempenhada pelo decisor no processo decisório (EHRGOTT; GANDIBLEUX, 2002). Esses métodos podem ser agrupados da seguinte forma:

- Métodos *à priori*: quando as preferências do decisor são conhecidas desde o início do processo e o decisor participa da busca de soluções antes da resolução do problema;
- Métodos *à posteriori*: quando as soluções são geradas sem prévio conhecimento das preferências do decisor, que então escolhe a opção considerada "melhor". A tomada de decisão é feita após a realização da busca de soluções ótimas de Pareto;
- Métodos iterativos: inicialmente, são geradas soluções eficientes, as quais são aprimoradas conforme as preferências do tomador de decisão. Ou seja, o decisor antes de cada iteração, define as prioridades, guiando a busca a partir de um conjunto de soluções conflitantes.

Os métodos discutidos neste capítulo visam gerar soluções sem que as preferências de um decisor sejam conhecidas. Dependendo das características e da dificuldade de resolução do problema, tanto métodos com base em abordagens clássicas (exatas) quanto em abordagens aproximativas (heurísticas) podem ser utilizados.

2.1.1.1 Métodos clássicos

As técnicas tradicionais possuem como principal vantagem, provas de convergência que garantem encontrar as soluções ótimas de Pareto (COELLO; VELDHUIZEN, 2002). Porém, um inconveniente associado a eles é a necessidade, na maioria dos casos, de um tempo computacional elevado (COSTA, 2015).

Os métodos clássicos transformam o problema multiobjetivo em um problema mono-objetivo (LAUNDY; STEUER, 1988), como por exemplo o Método da Soma Ponderada das Funções Objetivo, Método de Restrição Épsilon (ϵ) e Método de Programação por Metas.

Segundo Bardanachvili (2006) e Farias (2008), o Método da Soma Ponderada das Funções Objetivo é um dos mais simples utilizados na literatura para a resolução de problemas com múltiplos objetivos. Ele consiste na escalarização do problema multiobjetivo através da construção de uma função objetivo formada pela soma das funções objetivo do POM, ponderadas por um conjunto de pesos (JUNIOR, 2011). Esses pesos representam a importância dada a cada um dos objetivos durante o processo de tomada de decisão. Zadeh (1963) apresentou por que esse método é capaz de gerar soluções eficientes para os POMs. A vantagem deste método consiste na sua simplicidade de execução. Variando-se os valores dos pesos, diferentes soluções podem ser encontradas (EHRGOTT; GANDIBLEUX, 2002). Assim, um POM pode ser reescrito da seguinte forma:

$$\min f = \sum_{i=1}^k w_i f_i(x) \quad (2.5)$$

No qual, w_i é o peso do objetivo f_i em relação com os outros objetivos.

Um outro método clássico bastante utilizado para resolução de POMs, é através do Método da Restrição ϵ . Haimes, Lasdon e Wismer (1971) introduz o método, onde apenas uma das funções objetivo f_1 é minimizada, enquanto que as demais são transformadas em restrições com valores ϵ_i definidos pelo usuário. ϵ_i representa um limite superior da função objetivo f_i . O problema pode ser matematicamente descrito como:

$$\min f = f_1(x), \quad x \in X \quad (2.6)$$

$$f_i(x) \leq \epsilon_i, \quad i = 2, \dots, k \quad (2.7)$$

Além dos métodos acima apresentados, outro popular é o Método de Programação por Metas (ARENALES et al., 2007). Neste método, é estabelecido anteriormente, um conjunto de metas (valores) aceitáveis para cada função objetivo. Caso não exista uma solução factível com as metas estabelecidas, minimiza-se os desvios das metas estabelecidas (TICONA, 2008). Esta técnica também tem como base a conversão de múltiplos objetivos em um único objetivo. Para mais aprofundamento nesta técnica, são recomendados para o leitor o livro de Arenales et al. (2007) e os trabalhos de Ticona (2008), Farias (2008) e Junior (2011).

2.1.1.2 Técnicas aproximativas

A maioria dos métodos aproximativos é representada pelas metaheurísticas (COSTA, 2015). De acordo com Souza (2000), as metaheurísticas representam métodos que visam encontrar uma solução de alta qualidade, possivelmente ótima. Em cada iteração, essas estratégias consistem em uma heurística subordinada, a qual deve ser adaptada especificamente para cada problema (JUNIOR, 2011). A distinção fundamental entre metaheurísticas e heurísticas convencionais reside no fato de que as metaheurísticas são genéricas e têm a capacidade de escapar de ótimos locais. Existem diversas categorias de metaheurísticas, sendo a principal diferença entre elas o método de escape de ótimos locais. Essas estratégias são classificadas em duas categorias distintas de acordo com o método de exploração da região de soluções: busca local e busca populacional:

- Busca Local: exploram a região das soluções por meio de movimentos bem definidos. A cada iteração, esses movimentos são aplicados à solução atual, gerando uma nova solução;
- Busca Populacional: fundamentam-se na manutenção de um conjunto de boas soluções. Essas soluções são combinadas com o objetivo de obter uma nova solução que seja superior às duas primeiras.

Na resolução de POMs por busca local, destacam-se o 2PPLS e o PLS. O 2PPLS, proposto por Paquete e Stützle (2003), é composto de duas fases: a primeira busca determinar uma

solução de alta qualidade considerando apenas um dos objetivos, enquanto a segunda aplica procedimentos de busca local a problemas agregados formados pela soma ponderada de funções objetivo. Neste trabalho, devido a complexidade do problema, para a primeira fase foi utilizado uma heurística para encontrar o conjunto de soluções iniciais. Este método é detalhado na seção 3.3.

Já em relação ao método PLS (PAQUETE; CHIARANDINI; STÜTZLE, 2004), o mesmo consiste em gerar novas soluções a partir da exploração da vizinhança de soluções eficientes já obtidas anteriormente. Caso uma solução vizinha de alguma solução explorada não seja dominada por nenhuma outra solução do conjunto de soluções, ela é incluída no conjunto de soluções eficientes. No método original proposto, o conjunto de soluções iniciais utilizado como ponto de partida para o PLS foi gerado aleatoriamente. Porém outros métodos podem ser utilizados. Neste trabalho, foi utilizado o PLS adaptado como a segunda fase do 2PPLS que tem como primeira fase uma heurística pra determinar o conjunto inicial de soluções a ser explorado. Estes métodos são detalhados na seção 3.3.

Na resolução de POMs por busca populacional, destacam-se os algoritmos evolutivos, principalmente os algoritmos genéticos (FARIAS, 2008). Estes algoritmos foram desenvolvidos para simular um processo evolutivo através da adaptação dos indivíduos de uma população (JUNIOR, 2011). Os algoritmos genéticos são métodos flexíveis e que possuem a capacidade de gerar novas soluções de boa qualidade em problemas complexos (ARROYO, 2002).

Embora os Algoritmos Evolucionários tenham demonstrado sucesso na resolução de POMs, este trabalho utilizará exclusivamente métodos fundamentados em Busca Local. Para maior aprofundamento nos modelos de algoritmo genético, recomenda-se para o leitor, a revisão da literatura apresentada por Farias (2008), Ticona (2008) e Junior (2011).

2.1.2 Avaliação do desempenho dos algoritmos multiobjetivo

O cálculo de desempenho de um ou vários métodos em otimização multiobjetivo é uma tarefa complexa (BASTOS, 2017) e um dos principais desafios relacionados à resolução dos POMs está relacionado à comparação entre fronteiras de Pareto. Nesse contexto, torna-se necessário estabelecer uma metodologia adequada para realizar a comparação entre dois métodos multiobjetivos.

Segundo Lucena et al. (2013), podemos verificar duas metas na otimização multiobjetivo, a diversidade das soluções encontradas e a convergência do método. Para auxiliar nestes cálculos

de desempenho, (LUCENA et al., 2013) apresenta algumas métricas como a distância geracional e o espalhamento.

Embora a definição de qualidade de solução em um POM não seja totalmente clara, são três os objetivos principais de um POM (BARDANACHVILI, 2006):

1. Obter soluções as mais próximas possíveis da fronteira ótima de Pareto;
2. Obter soluções as mais diversas possíveis ao longo da fronteira,
3. Contemplar a possibilidade de não convexidade do espaço de soluções.

Esses critérios auxiliam na avaliação da qualidade das soluções obtidas em um POM, buscando alcançar um conjunto de soluções eficientes e diversificadas ao longo da FP.

Zitzler et al. (2003) destacam que vários estudos têm se dedicado ao desafio de comparar, de forma quantitativa, dois conjuntos de soluções. Os métodos de avaliação utilizados podem ser classificados principalmente em dois tipos: unários e binários:

- Métodos unários: associam a cada conjunto de soluções eficientes encontrado um número (valor) que reflete algum aspecto específico em consideração,
- Métodos Binários: atribuem valores a cada par de conjuntos de soluções, geralmente comparando-os em relação a um conjunto de referência.

Neste trabalho, serão adotados dois indicadores para avaliação dos conjuntos de soluções eficientes em um POM: o Hipervolume (H) e a Medida R (R). Esses indicadores foram selecionados por sua capacidade de avaliar os aspectos que apontam a qualidade de um conjunto de soluções. Além disso, trata-se de serem indicadores unários, o que significa que não requerem uma fronteira de referência para sua utilização. No caso do problema em questão tal aspecto se torna relevante, pois não existem na literatura outras fronteiras de Pareto conhecidas para o GPFSPSDS. Além disso, resolver o GPFSPSDS de forma exata requereria um elevado esforço computacional. Portanto, optou-se por utilizar esses indicadores que possibilitam uma avaliação adequada do desempenho dos conjuntos de soluções sem a necessidade de uma fronteira de referência.

2.1.2.1 Hipervolume

Uma métrica muito utilizada nos POMs, é o Hipervolume (ZHANG; HANSEN, 2009). O Hipervolume (H), uma métrica proposta por Zitzler (1999), é calculado o volume da região coberta entre os pontos das soluções e um ponto de referência, através da união dos politopos $(\pi_1, \pi_2, \dots, \pi_X)$ formados a partir de uma FP, como ilustrado na Figura 1.

Para garantir a adequada escolha do ponto de referência, é importante que este ponto seja, no mínimo, constituído com os piores valores das funções objetivo pertencentes a FP. Este ponto é obtido a partir de uma aproximação do *Ponto de Nadir* (ponto formado pelos piores valores para cada um dos objetivos, dentre os pontos na FP). Na prática, multiplica-se as coordenadas do *Ponto de Nadir*, por um coeficiente α , onde $\alpha > 1$. Desta forma, garante-se assim que as coordenadas do ponto de referência sejam sempre superiores às coordenadas de qualquer ponto não dominado (considerando um problema de minimização).

O valor do Hipervolume deve ser o maior possível, pois quanto maior a área (volume) sobre a figura, mais distante a fronteira estará do *Ponto de Nadir* e mais próxima do Ponto Ideal, que contém como coordenadas no Espaço de Objetivos os menores valores para cada um dos objetivos.

Segundo (DEB et al., 2002), um valor alto do Hipervolume, significa que o método teve uma boa distribuição e uma boa convergência. O trabalho de Costa (2015) apresenta exemplos que ilustram como o valor do Hipervolume se comporta diante de fronteiras com diferentes formatos, contribuindo para uma melhor compreensão e interpretação dessa métrica na avaliação de conjuntos de soluções em POMs.

O pseudocódigo 1, apresenta as etapas para o cálculo do Hipervolume para um POM biobjetivo. Na Linha 4 é calculado o Ponto de Nadir (Ponto de Referência). Na Linha 5, é calculado o primeiro volume do politopo da fronteira em relação ao Ponto de Nadir. Nas Linhas 6 e 7 são calculados os demais volumes dos politopos em relação ao ponto anterior da fronteira e o Ponto de Nadir.

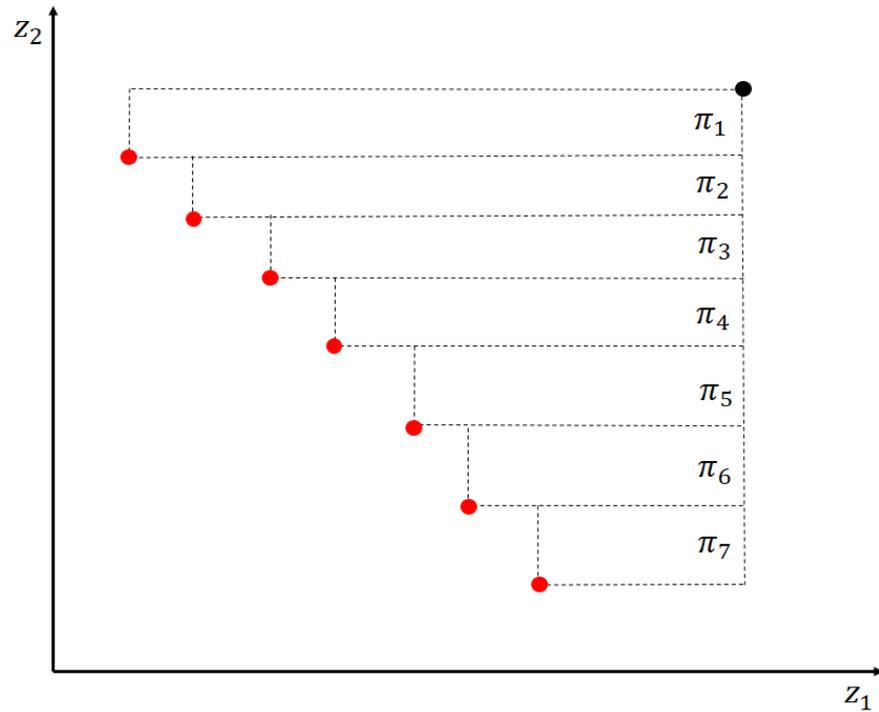
Algoritmo 1 Hipervolume

```

1: Entrada:  $P$  - Conjunto de Soluções Não Dominadas
2: Seja  $H$  o valor do Hipervolume da Fronteira
3: Seja  $N$  o número de soluções não dominadas
4:  $nadir_1, nadir_2 \leftarrow ComputeNadirPoint()$ 
5:  $H = (nadir_1 - f_1^0) * (nadir_2 - f_2^0)$ 
6: for  $i = 1 : N - 1$  do
7:    $H = H + ((nadir_1 - f_1^i) * (f_2^i - f_2^{i+1}))$ 
8: Saída:  $H$ 

```

Figura 1 – Hipervolume da Fronteira de Pareto



Fonte: O autor (2023).

2.1.2.2 Medida R

A Medida R (R), proposta por Jaszekiewicz (2002) e normalizada entre 0 e 1, é uma métrica que avalia um conjunto de soluções através da Função de Tchebycheff Ponderada (FTP) por um conjunto de vetores ponderados normalizados. Similar ao Hipervolume, o objetivo é maximizar o valor de R para indicar melhor qualidade das soluções.

O indicador de qualidade R é obtido calculando a média dos valores mínimos obtidos pelas Funções de Tchebycheff Ponderadas (FTPs) para cada vetor de pesos em Ψ . A Função de Tchebycheff, também conhecida como Distância de Tchebycheff, é definida para dois pontos como a maior distância absoluta entre suas coordenadas.

Para calcular o indicador R , para cada um dos pesos λ gerados (λ e Ψ), são obtidos os valores das FTPs por meio da expressão $\|z(x) - y^0\|_\lambda = \max_{i=1, \dots, k} \lambda_i (z_i(x) - y_i^0)$ onde $(z(x))$ representa cada um dos pontos pertencentes à FP e (y^0) é o ponto de referência adotado. Para cada peso λ , todas as distâncias ponderadas são calculadas e a menor delas é determinada (Γ_λ^*). A Medida R é, então, computada como a média dos valores (Γ_λ^*) obtidos para cada um dos pesos. O algoritmo 2 apresenta as etapas para o cálculo do indicador R .

O Algoritmo 2 recebe como parâmetro de entrada o conjunto de pontos não dominados do problema. Na Linha 2, é gerado o conjunto de pesos conforme a expressão 2.8. Neste processo, são gerados $\binom{n+k-1}{k-1}$ vetores de pesos, onde n é o número de tarefas na instância e k é o número de objetivos considerados.

$$\Psi = \left\{ \lambda = (\lambda_1, \dots, \lambda_k) \mid \sum_{j=1}^k \lambda_j = 1; \lambda_j \in \left\{ 0, \frac{1}{k}, \frac{2}{k}, \dots, \frac{k-1}{k}, 1 \right\} \right\} \quad (2.8)$$

Nas Linhas 3 e 4 são computados o *Ponto Ideal* e o *Ponto de Nadir*, e na Linha 5 é determinado o Ponto de Referência, que este pode ser informado pelo usuário, caso contrário, ele pode ser calculado como uma aproximação do *Ponto Ideal*. Em seguida, nas Linhas 6 a 8, ocorre a normalização dos objetivos para cada ponto não dominado existente na FP.

Posteriormente, nas Linhas 9 a 11, é computada a Função Normalizada de Tchebycheff, utilizando cada um dos pesos. Para cada peso λ , calcula-se o valor Γ_y para cada um dos pontos não dominados $y \in Y$. O valor de Γ_y é obtido como a maior distância entre as coordenadas do ponto e do produto do peso λ . Na Linha 12, Γ_λ^* recebe o menor valor dentre os calculados no laço anterior, buscando minimizar a FTP para cada peso λ . Ao final, na Linha 13, o valor da Medida R é calculado, sendo obtido subtraindo-se de 1 a média dos valores ótimos das FTPs.

Algoritmo 2 Medida R

```

1: Entrada:  $P$  - Conjunto de Soluções Não Dominadas
2: Conjunto  $\Psi$  com  $\binom{n+k-1}{k-1}$  pesos  $\Psi \leftarrow \{(1, 0), (0.99, 0.01), \dots, (0.01, 0.99), (0, 1)\}$ 
3:  $\min \leftarrow \text{ComputeIdealPoint}()$ 
4:  $\max \leftarrow \text{ComputeNadirPoint}()$ 
5:  $y^0 \leftarrow \text{GetReferencePoint}()$ 
6: for  $z(x) \in Y$  do
7:   for  $i \in k$  do
8:      $z_i^{norm} = \frac{z_i(x) - \min_i}{\max_i - \min_i}$ 
9: for  $\lambda \in \Psi$  do
10:  for  $y = z(x) \in Y$  do
11:     $\Gamma_y = \max_{i \in k} \{\lambda_i (y_i^{norm} - z_i^{norm})\}$ 
12:   $\Gamma_\lambda^* = \min_{y \in Y} \{\Gamma_y\}$ 
13:  $R(Y) = 1 - \frac{\sum_{\lambda \in \Psi} \Gamma_\lambda^*}{|\Psi|}$ 
14: Saída:  $R$ 

```

2.2 REVISÃO DA LITERATURA

Nos últimos anos, tem-se observado um aumento no número de trabalhos na literatura que investigam a programação do chão de fábrica. Apesar disso, ainda existem algumas questões críticas sobre este tópico, incluindo consumo de energia e pegada de carbono (DING; SONG; WU, 2016). Uma revisão da literatura sobre programação de eficiência energética na manufatura, conduzida por Gahm et al. (2016), mostrou que a programação de produção sustentável vem chamando a atenção dos fabricantes. Várias abordagens foram desenvolvidas para reduzir o consumo de energia por máquinas nas indústrias de manufatura. Essas abordagens incluem:

- Programação com restrição no pico de consumo de energia das máquinas (FANG et al., 2011; FANG et al., 2013);
- Desligar as máquinas em tempos ociosos (MOUZON; YILDIRIM; TWOMEY, 2007; MOUZON; YILDIRIM, 2008; LIU et al., 2016; LU et al., 2017; JIANG; WANG, 2019);
- Projetar máquinas com maior eficiência energética (LI et al., 2011; MORI et al., 2011).

Fang et al. (2011) propuseram um modelo de programação matemática em um ambiente *flowshop* com duas máquinas e três objetivos, incluindo carga de pico de energia, pegada de carbono e *makespan*. Fang et al. (2013) consideraram um problema de *flowshop scheduling* com restrição no pico de consumo de energia. Eles apresentaram dois modelos de programação inteira mista biobjetivo para minimizar o pico de consumo de energia e *makespan*.

Mouzon, Yildirim e Twomey (2007) propuseram várias regras de despacho e um modelo de programação matemática biobjetivo para minimizar o consumo de energia e o tempo total de conclusão. Eles usaram uma estrutura *on/off* para reduzir o consumo de energia. Mouzon e Yildirim (2008) usaram a mesma estrutura *on/off* (MOUZON; YILDIRIM; TWOMEY, 2007) para minimizar o consumo de energia e o atraso total para um problema de máquina única. Nesses dois estudos, os autores não consideraram o consumo de energia durante o funcionamento das máquinas. Liu et al. (2016) apresentaram um modelo matemático biobjetivo no problema de *job shop scheduling* para minimizar o consumo total de eletricidade do tempo ocioso e o atraso total ponderado. Eles aplicaram a estrutura de *on/off* para economizar eletricidade. Lu et al. (2017) consideraram a eficiência energética na programação do *flowshop* de m máquinas com tempos de *setup* dependentes da sequência. Eles estudaram o problema *permutation flowshop*

scheduling com considerações ambientais com base na estrutura *on/off*. Para resolver o problema, eles usaram um algoritmo híbrido de busca multiobjetivo. A estrutura *on/off* também foi utilizada no trabalho de Jiang e Wang (2019). Eles apresentaram um modelo matemático biobjetivo para minimizar o *makespan* e o consumo de energia. Eles desenvolveram um algoritmo evolucionário multiobjetivo baseado em decomposição para resolver o problema. Meng et al. (2019a) consideraram o problema de *flowshop scheduling* híbrido com máquinas paralelas não relacionadas. Cinco modelos matemáticos foram propostos e um algoritmo genético melhorado foi proposto para resolver o problema. Eles aplicaram a estrutura *on/off* como método de economia de energia. Meng et al. (2019b) propuseram seis modelos de *Mixed Integer Linear Programming* (MILP) com estrutura *on/off* no problema *flexible job shop scheduling*. O objetivo de todos os modelos foi minimizar o TEC.

No âmbito de projetar máquinas com maior eficiência energética, Li et al. (2011) propõem um estudo do consumo de energia de tempo ocioso em função da máquina-ferramenta e do próprio processo de corte, propondo estratégias de melhoria. Mori et al. (2011), por sua vez, propuseram um novo método de controle de aceleração de máquinas-ferramentas para melhoria na eficiência energética. Os autores chegaram a conclusão que modificar as condições de operação e controle do equipamento reduz o consumo de energia.

Porém, as abordagens citadas anteriormente não são práticas nas indústrias de manufatura (DING; SONG; WU, 2016). Por exemplo, o consumo de energia usada para ligar as máquinas é elevado. Além disso, ligar e desligar repetidamente as máquinas podem danificá-las (ZHANG et al., 2014). Assim, é necessário fornecer soluções mais sustentáveis para reduzir o consumo de energia. O consumo de energia das máquinas depende de diferentes características, incluindo potência da máquina, velocidade de operação e tempo de processamento (BANSAL; KIMBREL; PRUHS, 2007). O fabricante pode alterar a velocidade de operação e o tempo de processamento para controlar a potência da máquina e seu consumo de energia.

Com base nesta questão, trabalhos recentes abordam o controle da velocidade e as consequências no *makespan* e consumo de energia. Mansouri, Aktas e Besikci (2016) analisaram o *trade-off* entre *makespan* e TEC no problema PFSPTSDS para um caso especial de duas máquinas. Eles desenvolveram uma heurística para obter uma análise entre os objetivos. Mansouri e Aktas (2016) estudaram o mesmo problema. Eles desenvolveram uma heurística e um algoritmo genético multiobjetivo para resolver o problema. Mokhtari e Hasani (2017) consideraram o aspecto ambiental no problema de *flexible job shop scheduling*. Eles desenvolveram um modelo matemático tri-objetivo cujo terceiro objetivo é de minimizar o custo total de energia das

operações de produção e manutenção. Eles propuseram um algoritmo evolutivo aprimorado para resolver o problema. Ramezani, Vali-Siar e Jalalian (2019) para o GPFSTSDS, propuseram uma heurística com o intuito de economizar energia a partir do controle de velocidade das máquinas sem prejuízos ao tempo de conclusão (*makespan*) da operação.

O problema PFSPTSDS, tem muitas aplicações nas indústrias de manufatura, como indústrias têxteis, de compostos químicos e indústrias de plástico (EREN, 2010; ALLAHVERDI; GUPTA; ALDOWAISAN, 1999; FRANÇA et al., 1996; DAS; GUPTA; KHUMAWALA, 1995). Além disso, os tempos de *setup* são usados nas indústrias de metal, farmacêutica e de papel (ALLAHVERDI; GUPTA; ALDOWAISAN, 1999; YANG; LIAO, 1999).

Existem poucos trabalhos de PFSPTSDS considerando o consumo de energia. Apenas os trabalhos de Lu et al. (2017) e Jiang e Wang (2019), ambos utilizando a estrutura *on/off*, e Ramezani, Vali-Siar e Jalalian (2019) consideraram o aspecto ambiental no problema. Desta forma, observa-se a necessidade de discussão sobre o problema GPFSTSDS, que é abordado nesta dissertação.

A tabela 1 apresenta um resumo dos problemas e métodos propostos das principais referências utilizadas neste trabalho.

Tabela 1 – Resumo das principais referências utilizadas da revisão da literatura

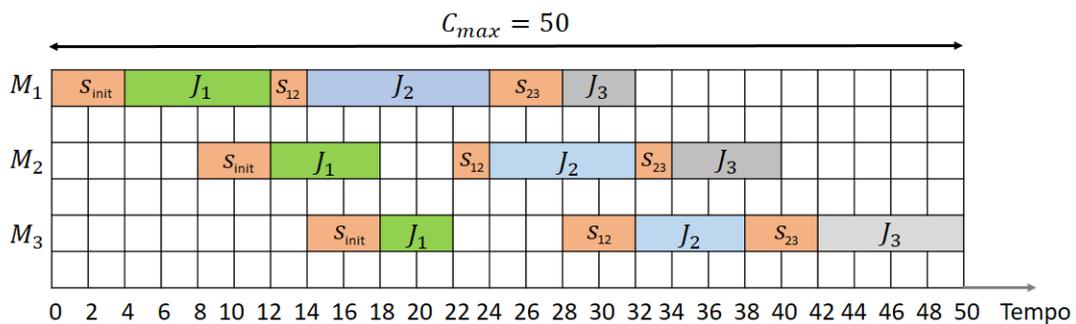
Autor	Problema	Método
Mouzon, Yildirim e Twomey (2007)	<i>Shop</i> de 4 máquinas com 2 objetivos: minimização de <i>Makespan</i> e de consumo total de energia	Exato: Soma Ponderada das Funções Objetivo
Mouzon e Yildirim (2008)	Minimização de Atraso Total e de consumo total de energia em uma máquina	Metaheurística: <i>Greedy Randomised Adaptive Search Procedure</i> (GRASP)
Fang et al. (2011)	<i>Flowshop</i> de 2 máquinas com 3 objetivos: Minimização de <i>Makespan</i> , de pegada de carbono e de pico de consumo de energia	Exato (<i>Gurobi Optimizer 4.0</i>)
Fang et al. (2013)	<i>Permutation Flowshop</i> de 2 máquinas com restrições de pico de consumo de energia para minimização de <i>Makespan</i>	Exato (<i>Gurobi Optimizer 4.5</i>)
Liu et al. (2016)	<i>Job shop</i> com 2 objetivos: minimização de atraso total ponderado e de consumo de energia nos tempos ociosos	Algoritmo Genético (NSGA-II)
Mansouri, Aktas e Besikci (2016)	<i>Permutation Flowshop</i> de 2 máquinas para minimização de <i>Makespan</i> e consumo de energia	Heurística Construtiva
Mansouri e Aktas (2016)	<i>Permutation Flowshop</i> de 2 máquinas para minimização de <i>Makespan</i> e consumo de energia	Heurística Construtiva e Algoritmo Genético
Lu et al. (2017)	<i>Permutation Flowshop</i> com 2 objetivos: Minimização de <i>Makespan</i> e de consumo de energia	Híbrido de Algoritmo Genético com busca <i>Backtracking</i>
Mokhtari e Hasani (2017)	<i>Job shop</i> com 3 objetivos: minimização do <i>Makespan</i> e do consumo total de energia e maximização da disponibilidade do sistema	Algoritmo Evolucionário
Jiang e Wang (2019)	<i>Permutation Flowshop</i> com 2 objetivos: Minimização de <i>Makespan</i> e de consumo de energia	Algoritmo Evolucionário
Meng et al. (2019a)	<i>Flowshop</i> com máquinas paralelas não relacionadas com objetivo de minimização de consumo total de energia	Algoritmo Genético
Meng et al. (2019b)	<i>Job shop</i> com objetivo de minimização de consumo total de energia	Algoritmo Genético
Ramezaniyan, Vali-Siar e Jalalian (2019)	<i>Permutation Flowshop</i> com 2 objetivos: Minimização de <i>Makespan</i> e de consumo de energia	Metaheurística: Heurística Construtiva, Busca Local e Algoritmo de Economia de Energia

3 METODOLOGIA

3.1 DESCRIÇÃO DO PROBLEMA

Neste trabalho, foi considerado o problema de sequenciamento de produção GPFSPSTSDS. A Figura 2 ilustra um *scheduling* de 3 máquinas e 3 tarefas a partir de um Diagrama de Gantt.

Figura 2 – Diagrama de Gantt de 3 tarefas em 3 máquinas



Fonte: O autor (2023).

O problema GPFSPSTSDS é denotado como $F_m/ST_s d/C_{max}, TEC$ com base na notação de três campos de Graham ($\alpha/\beta/\gamma$) (GRAHAM et al., 1979). Isto significa que n tarefas (*jobs*) devem ser processados em m máquinas sequencialmente, com consideração de tempos de *setup* dependentes da sequência, visando minimizar o *makespan* e o TEC. Este problema é NP-difícil porque pode ser reduzido para o problema $F2/ST_s d/C_{max}$, que é conhecido por ser NP-Difícil (GUPTA; DARROW, 1986).

Conforme descrito no Capítulo 2, o GPFSPSTSDS foi estudado na literatura por Ramezani, Vali-Siar e Jalalian (2019). Em seu artigo, os autores apresentam as seguintes hipóteses para o problema:

- Todas as máquinas estão sempre disponíveis e não há avarias;
- Todas as tarefas estão disponíveis no instante inicial;
- Cada uma das n tarefas pertencentes ao conjunto $J = 1, 2, \dots, n$ deve ser processado em m máquinas pertencentes ao conjunto $M = 1, 2, \dots, m$ sequencialmente na mesma ordem. Cada máquina só pode processar um trabalho por vez;

- Há um tempo inicial de *setup* para o primeiro trabalho da sequência em todas as máquinas;
- Os tempos de *setup* não estão incluídos nos tempos de processamento. Eles são dependentes da sequência, ou seja, a duração dos tempos de *setup* depende tanto do trabalho atual quanto do imediatamente anterior;
- Os tempos de *setup* são antecipados, ou seja, os tempos de *setup* podem ser iniciados antes que o trabalho correspondente fique disponível na máquina;
- Pode haver tempos ociosos para as máquinas;
- Cada máquina será desligada após a conclusão do último trabalho.

A notação do problema é a seguinte:

Índices

j, k	Índices para tarefas $j, k = 1, 2, \dots, n$
i	Índice para máquinas $i = 1, 2, \dots, m$
l	Índice para níveis de velocidade de processamento $l = 0, 1, 2$ (velocidade lenta, normal e rápida respectivamente)
n	Número de tarefas
m	Número de máquinas

Parâmetros

sp_l	Fator de velocidade de processamento l
p_{ij}	Tempo de processamento da tarefa j na máquina i
st_{ijk}	Tempo de <i>setup</i> para a mudança da tarefa j para a tarefa k na máquina i
λ_l	Fator de conversão para a velocidade de processamento l
ϑ_l	Fator de conversão para o tempo ocioso na máquina i
π_i	Potência da máquina i
M	Um número grande

Variáveis de Decisão

s_{ij}	Tempo inicial da tarefa j na máquina i
o_{ij}	Tempo de <i>setup</i> para a tarefa j na máquina i , $i \neq 1$, se a tarefa j é a primeira tarefa na sequência
c_{ij}	Tempo de conclusão da tarefa j na máquina i
θ_i	Tempo ocioso na máquina i

∂_j	1 se a tarefa j é a primeira tarefa, 0 se contrário
x_{jk}	1 se a tarefa j precede a tarefa k , 0 se contrário ($j \neq k$)
y_{ijl}	1 se a tarefa j é processada no nível de velocidade l na máquina i , 0 se contrário
C_{max}	<i>Makespan</i> , o tempo de conclusão da última tarefa denominada n na última máquina denominada m
TEC	Consumo total de energia

O modelo MILP biobjetivo para minimizar C_{max} e TEC é apresentado abaixo:

$$\min C_{max} \quad (3.1)$$

$$\min TEC \quad (3.2)$$

$$C_{max} \geq c_{mj}, \quad \forall j = 1, \dots, n \quad (3.3)$$

$$M(1 - \partial_j) + o_{ij} \geq st_{ijj} - c_{(i-1)j}, \quad i = 2, \dots, m; j = 1, \dots, n \quad (3.4)$$

$$s_{ij} \geq c_{(i-1)j} + o_{ij}, \quad i = 2, \dots, m; j = 1, \dots, n \quad (3.5)$$

$$s_{1j} \geq st_{1jj}\partial_j, \quad j = 1, \dots, n \quad (3.6)$$

$$c_{ij} \geq s_{ij} + \frac{p_{ij}}{sp_l} y_{ijl}, \quad i = 2, \dots, m; j = 1, \dots, n; l = 1, 2, 3 \quad (3.7)$$

$$s_{ik} \geq c_{ij} - M(1 - x_{jk}) + st_{ijk}x_{jk}, \quad i = 1, \dots, m; j, k = 1, \dots, n, |j \neq k \quad (3.8)$$

$$s_{ij} \geq c_{ik} - Mx_{jk} + st_{ikj}(1 - x_{jk}), \quad i = 1, \dots, m; j, k = 1, \dots, n, |j \neq k \quad (3.9)$$

$$\sum_{j=1}^n \partial_j = 1 \quad (3.10)$$

$$\sum_{l=1}^3 y_{ijl} = 1, \quad i = 1, \dots, m; j = 1, \dots, n \quad (3.11)$$

$$\theta_i = C_{max} - \sum_{j=1}^n \sum_{l=1}^3 \frac{p_{ij}}{sp_l} y_{ijl}, \quad i = 1, \dots, m \quad (3.12)$$

$$TEC = \sum_{i=1}^m \sum_{j=1}^n \sum_{l=1}^3 \frac{\pi_i p_{ij} \lambda_l}{60 sp_l} y_{ijl} + \sum_{i=1}^m \frac{\vartheta_i \pi_i}{60} \theta_i \quad (3.13)$$

$$s_{ij} \geq 0 \quad i = 1, \dots, m; j = 1, \dots, n \quad (3.14)$$

$$c_{ij} \geq 0 \quad i = 1, \dots, m; j = 1, \dots, n \quad (3.15)$$

$$o_{ij} \geq 0 \quad i = 1, \dots, m; j = 1, \dots, n \quad (3.16)$$

$$\theta_i \geq 0 \quad i = 1, \dots, m \quad (3.17)$$

$$\partial_j \in \{0, 1\} \quad j = 1, \dots, n \quad (3.18)$$

$$x_{jk} \in \{0, 1\} \quad j, k = 1, \dots, n \quad (3.19)$$

$$y_{ijl} \in \{0, 1\} \quad i = 1, \dots, m; j = 1, \dots, n; l = 1, 2, 3 \quad (3.20)$$

$$C_{max} \geq 0 \quad (3.21)$$

$$TEC \geq 0 \quad (3.22)$$

As funções objetivo 3.1 e 3.2 representam a minimização de C_{max} e TEC , respectivamente. C_{max} é a representação do nível de serviço e TEC é conhecida como uma métrica de sustentabilidade. As restrições 3.3 calculam C_{max} . Isto ocorre após a conclusão do último trabalho na última máquina. Como os tempos de *setup* antecipados são assumidos, as restrições 3.4 calculam a duração do deslocamento de *setup* apenas para o primeiro trabalho em todas as máquinas, exceto para a primeira máquina. Isso permite que o *setup* em cada máquina seja iniciado antes que o primeiro trabalho na máquina anterior seja concluído. As restrições 3.5 garantem que o instante de tempo inicial de uma tarefa em todas as máquinas, exceto na primeira máquina, seja maior ou igual ao tempo de conclusão da mesma tarefa na máquina anterior mais a duração do tempo de *setup* (se o tempo de *setup* existir). As restrições 3.6 estabelecem que uma tarefa começa na primeira máquina após terminar seu tempo de *setup* correspondente. As restrições 3.7 garantem que o tempo de conclusão de uma tarefa em uma máquina seja maior ou igual ao seu tempo de início mais seu tempo de processamento. As restrições 3.8 e 3.9 determinam a sequência de tal forma que o instante de início de uma tarefa seja maior ou igual ao tempo de conclusão da tarefa anterior na mesma máquina mais a troca de *setup*. A restrição 3.10 garante que haja apenas uma primeira tarefa (*job*). As restrições 3.11 garantem que haja apenas um nível de velocidade para cada tarefa em cada máquina. As restrições 3.12 e 3.13 calculam o tempo ocioso das máquinas e o TEC, respectivamente. As restrições 3.14 a 3.17 e as restrições 3.21 e 3.22 definem que as respectivas variáveis são não negativas. As restrições 3.18 a 3.20 definem que as respectivas variáveis são binárias.

3.2 MÉTODOS DE RAMEZANIAN

Como não se teve acesso às fronteiras geradas pelo trabalho de Ramezianian, Vali-Siar e Jalalian (2019), foi necessário reimplementar seus métodos. Porém, a implementação talvez não tenha sido a mais acurada devido à falta de informações como a potência considerada de cada máquina em cada instância. Além disso, há alguns erros no algoritmo apresentado de economia de energia (Algoritmo 6) por Ramezianian, Vali-Siar e Jalalian (2019) e foram comentados na seção 3.2.1. Mesmo assim, tentou-se fazer o melhor possível para reprodução

dos resultados.

Ramezaniyan, Vali-Siar e Jalalian (2019) desenvolveram um algoritmo heurístico construtivo para gerar soluções com redução de consumo de energia. O método é descrito no Algoritmo 3. O mesmo contém cinco etapas. Na primeira etapa (Linhas 3 a 6), todos os *jobs* são definidos para serem executados no nível de velocidade mais rápido. Na segunda etapa (Linha 8), o NEHT-RB (Algoritmo 4) é executado para encontrar uma solução inicial.

O NEHT-RB é uma heurística construtiva proposta por Rios-Mercado e Bard (1998) com o objetivo de minimizar o *makespan* para o problema PFSPTSDS. Rios-Mercado e Bard (1998) modificaram o NEH (NAWAZ; ENSCORE, 1983) e o NEHT (TAILLARD, 1990) para considerar os tempos de *setup* dependentes da sequência.

O NEHT-RB consiste em calcular os tempos totais de processamento T_j de cada *job* com seus respectivos níveis de velocidade em cada máquina e ordená-los em ordem decrescente. Com os *jobs* ordenados, o *job* com maior T_j é removido e inserido na solução na posição que minimize o *makespan* parcial $\gamma(k)$, até que todas as tarefas tenham sido incluídas. O cálculo de $\gamma(k)$ é detalhado no trabalho de Rios-Mercado e Bard (1998).

Na terceira etapa do Algoritmo 3 (Linhas 10 a 14), um método de busca local (Algoritmo 5) *best improvement* com estrutura de vizinhança *exchange* é aplicado a solução inicial. Na quarta etapa (Linhas 16 a 24), o algoritmo implementa um laço de repetição para buscar por soluções com redução do consumo de energia. Em cada iteração, o algoritmo seleciona o *job* com o menor tempo de processamento corrigido pelo nível de velocidade atual e reduz sua velocidade em um nível (ou seja, rápido para normal e normal para lento). Com novos níveis de velocidade, o NEHT-RB e a busca local são executados para encontrar uma nova solução. Posteriormente, o método de redução do consumo de energia proposto pelos autores (Algoritmo 6) é utilizado para melhorar o consumo de energia sem alterar o *makespan*. A quarta etapa (Linhas 16 a 24) continua até que o nível de velocidade de todos os *jobs* seja minimizado. Uma solução é obtida no final de cada iteração. Na quinta etapa (Linhas 26 a 27), o algoritmo de seleção de soluções não dominadas (Algoritmo 7) é utilizado nas soluções obtidas nas etapas anteriores.

O Algoritmo 6 é um método de economia de energia e é a principal contribuição do trabalho de Ramezaniyan, Vali-Siar e Jalalian (2019). Inicialmente, este método (2ª Etapa - Linhas 5 a 22) encontra os *gaps* (lacunas) entre dois *jobs* consecutivos ($GapJ_{ij}$) em uma máquina e encontra os *gaps* para um *job* entre duas máquinas consecutivas ($GapM_{ij}$). A Figura 3 ilustra os *gaps* entre *jobs* e entre máquinas em um Diagrama de Gantt.

Algoritmo 3 Heurística Construtiva - Ramezianian

- 1: **Entrada:** Conjunto de máquinas e *jobs* não programados, sp_l, p_{ij}
 - 2: **1ª Etapa (Inicialização)**
 - 3: Definir contador $z = 0$
 - 4: Definir B como o conjunto de todas as soluções viáveis
 - 5: $B = \square$
 - 6: Setar Matriz de Velocidade no nível mais alto de velocidade (Ex.: $V_z : [v_{ij} = sp_2], \forall i = 1, \dots, n; j = 1, \dots, m$)
 - 7: **2ª Etapa (Solução Inicial)**
 - 8: $seq_z \leftarrow$ NEHT-RB($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 9: **3ª Etapa**
 - 10: $seq_z \leftarrow$ Busca Local (seq_z, n) (Algoritmo 5)
 - 11: $B = B \cup seq_z$
 - 12: $z = z + 1$
 - 13: $[k]$ representa o *job* na posição k da seq_z e $op_{i[k]}$ denota a operação na posição k na máquina i
 - 14: $op_{n_{slow}}$ representa o conjunto de operações em que o nível de velocidade não é baixo (Ex.: $v_{i[k]} \neq sp_0$)
 - 15: **4ª Etapa (Busca por Soluções com consumo reduzido de energia)**
 - 16: **while** $op_{n_{slow}} \neq \square$ **do**
 - 17: Encontrar a operação $op_{i[\alpha]}$ tal que $op_{i[\alpha]} = \min[p_{i[k]}/v_{i[k]}, v_{i[k]} \in V_z, \forall i = 1, \dots, m$
 - 18: Reduzir o nível de velocidade de $op_{i[\alpha]}$ em um nível ($v_{i[\alpha]} : sp_l \rightarrow sp_{l-1}$) e atualizar V_z
 - 19: $seq_z \leftarrow$ NEHT-RB($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 20: $seq_z \leftarrow$ Busca Local Swap(seq_z, n) (Algoritmo 5)
 - 21: $seq_z \leftarrow$ Energy Saving ($p_{ij}, s_{i,j,k}, V_z, n, m$) (Algoritmo 6)
 - 22: $B = B \cup seq_z$
 - 23: Atualizar $op_{n_{slow}}$
 - 24: $z = z + 1$
 - 25: **5ª Etapa**
 - 26: Aplicar o Algoritmo (7) de seleção de soluções não dominadas no conjunto B
 - 27: **Retornar** Conjunto de soluções não dominadas
-

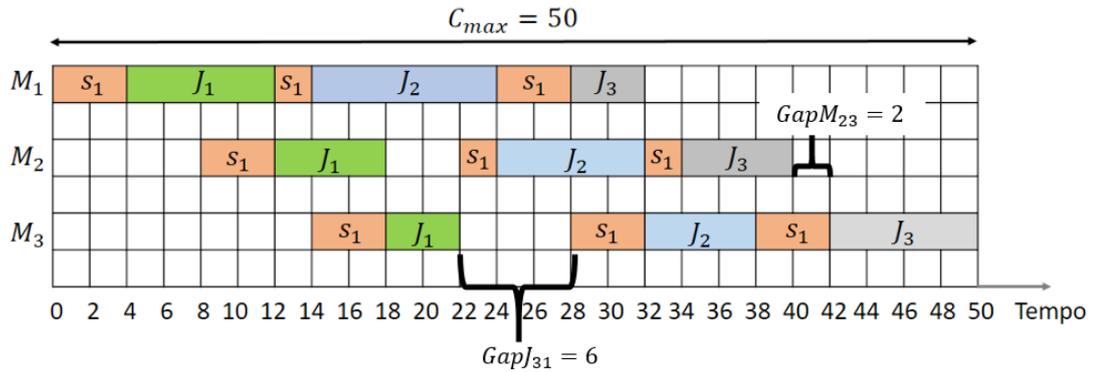
Algoritmo 4 Construtivo NEHT-RB

- 1: **Entrada:** Tarefas não sequenciadas, $p_{ij}, V_z = [v_{ij}], n, m$
 - 2: $FS = []$
 - 3: Seja U o conjunto de tarefas não sequenciadas
 - 4: Calcular a matriz pr de modo que $pr_{ij} = p_{ij}/v_{ij} (v_{ij} \in V_z); i = 1, \dots, m; j = 1, \dots, n$
 - 5: Calcular $T_j = \sum_{i=1}^m pr_{ij}; j = 1, \dots, n$
 - 6: Ordenar as tarefas no conjunto U em ordem decrescente de T_j
 - 7: **while** $U \neq \emptyset$ **do**
 - 8: Escolher a primeira tarefa h do conjunto U
 - 9: Calcular o *makespan* parcial $\gamma(k)$ para toda posição $k = 1, 2, \dots, [FS + 1]$
 - 10: Encontrar $o = \min \{ \gamma(k) \}$
 - 11: Inserir a tarefa h na posição o em FS
 - 12: Remover a tarefa h de U
 - 13: **Retornar** seq_z
-

Algoritmo 5 Busca Local - Ramezianian

- 1: **Entrada:** seq_z, n
 - 2: $seq_1 = seq_z$
 - 3: **for** $j = 1$ to $j < n$ **do**
 - 4: Trocar *job* j e $j + 1$ da seq_z
 - 5: **if** seq_z não é dominada por seq_1 **then**
 - 6: $seq_1 = seq_z$
 - 7: **else**
 - 8: $seq_z = seq_1$
 - 9: **Retornar** Solução seq_z melhorada
-

Figura 3 – Representação dos gaps



Fonte: O autor (2023).

Em seguida, o método de economia de energia verifica se os *jobs* podem ser realizados em um nível de velocidade mais baixo sem alterar o *makespan*. Entre estes *jobs*, o método de economia de energia seleciona o *job* que mais poupa energia e depois reduz a sua velocidade em um nível (3ª Etapa - Linhas 24 a 38). Se o *job* selecionado não cobrir todo o *gap*, o final deste *job* será movido para o final do novo *gap* (4ª Etapa - Linhas 40 a 44). Portanto, novos *gaps* são encontrados e novas verificações são realizadas para redução do consumo de energia. O Algoritmo 6 mostra o pseudocódigo do método de economia de energia. No Algoritmo, o tempo de conclusão do trabalho j na máquina i (e_{ij}) é calculado da seguinte forma:

$$e_{i0} = 0, \quad \forall i = 1, \dots, m \quad (3.23)$$

$$e_{0j} = r_j, \quad j = 1, \dots, o - 1 \quad (3.24)$$

$$e_{ij} = \max \{e_{i-1,j}; e_{i,j-1} + s_{i,j-1,j}\} + pr, \quad i = 1, \dots, m; j = 1, \dots, o - 1 \quad (3.25)$$

onde, r_j indica o *release time* da tarefa j assumido como zero.

O método proposto em Ramezani, Vali-Siar e Jalalian (2019) é apresentado no Algoritmo 6.

Na última etapa do Algoritmo 3, um algoritmo de seleção de solução não dominadas é implementado. Foi utilizada a abordagem de classificação rápida não dominada proposta por Deb et al. (2002). Neste procedimento, n_b é o número de soluções dominadas pela solução b , e S_b é o conjunto de soluções dominadas pela solução b . O Algoritmo 7 mostra o pseudocódigo deste procedimento.

Algoritmo 6 Energy Saving - Ramezianian

1: **Entrada:** $p_{ij}, V_Z = [v_{ij}], seq_z, n, m, spl$
2: **1ª Etapa (Inicialização)**
3: Calcular e_{ij} e formar a matriz $E = [e_{ij}], i = 1, \dots, m; j = 1, \dots, n$
4: **2ª Etapa (Encontrar os gaps)**
5: $[j]$ e $[j + 1]$ representam o *job* na posição j e o *job* na posição $j + 1$
6: $GapJ_{i,[j]}$ representa o *gap* entre o *job* $[j]$ e o *job* $[j + 1]$ na máquina i
7: $GapM_{i,[j]}$ representa o *gap* para o *job* $[j]$ entre a máquina i e a máquina $i + 1$
8: Calcular $pr_{ij} = p_{ij}/v_{ij}, (v_{ij} \in V_Z), i = 1, \dots, m; j = 1, \dots, n$
9: **for** $i = 1$ to m **do**
10: **for** $j = 1$ to $n - 1$ **do**
11: $GapJ_{i,[j]} = E_{i,[j+1]} - pr_{i,[j+1]} - E_{i,[j]}$
12: **for** $i = 1$ to $m - 1$ **do**
13: $GapJ_{i,n} = inf$
14: **for** $i = 1$ to $m - 1$ **do**
15: **for** $j = 1$ to n **do**
16: $GapM_{i,[j]} = E_{i+1,[j]} - pr_{i+1,[j]} - E_{i,[j]}$
17: **for** $j = 1$ to $n - 1$ **do**
18: $GapM_{m,j} = inf$
19: $GapJ_{m,n} = 0$ e $GapM_{m,n} = 0$
20: $Gap_{ij} = \min([GapJ_{i,j}], [GapM_{i,j}]), i = 1, \dots, m; j = 1, \dots, n$
21: G representa a matriz de máquinas e *jobs* que satisfazem $Gap_{ij} > 0$
22: β representa o número de linhas em G
23: **3ª Etapa (Encontrar a melhor economia de energia)**
24: **while** $\sum_{i \in m} \sum_{j \in n} Gap_{ij} > 0$ **do**
25: $V'_Z = V_Z, (v'_{ij} = v_{ij}), i = 1, \dots, m; j = 1, \dots, n$
26: Calcular a energia consumida EC_1 por cada *job* em cada máquina da matriz G
27: **for** $q = 1 : \beta$ **do**
28: **if** $V_{G(q,1)(q,2)} \neq sp_0$ **then**
29: $\Delta_{G(q,1)(q,2)} = Gap_{G(q,1)(q,2)}; \Delta'_{G(q,1)(q,2)} = p_{G(q,1)(q,2)}/spl - 1$
30: **while** $\Delta_{G(q,1)(q,2)} - \Delta'_{G(q,1)(q,2)} \geq 0$ **do**
31: Atualizar Matriz $V_Z, (v_{G(q,1)(q,2)} = spl - 1)$
32: **if** $v_{G(q,1)(q,2)} = sp_0$ **then**
33: Sair do *loop*
34: Atualizar $\Delta_{G(q,1)(q,2)}, \Delta'_{G(q,1)(q,2)}$
35: Calcular nova energia EC_2 consumida por cada *job* em cada máquina da matriz G
36: $EC_3 = EC_2 - EC_1$ e $M_x = \max(EC_3)$
37: a e b representa a máquina e o *job* respectivamente do elemento M_x
38: Atualizar Matriz $V'_Z, (v'_{ab} = v_{ab}); pr_{ab} = p_{ab}/v'_{ab}$
39: **4ª Etapa (Criar novo Gap)**
40: **if** $\Delta'_{G(q,1)(q,2)} < \Delta_{G(q,1)(q,2)}$ **then**
41: $e_{ab} = e_{ab} + Gap_{ab}$
42: $V_Z = V'_Z, (v_{ij} = v'_{ij}), i = 1, \dots, m; j = 1, \dots, n$
43: Atualizar $pr_{ij} = p_{ij}/v_{ij}, (v_{ij} \in V_Z)$ e $E = [e_{ij}], i = 1, \dots, m; j = 1, \dots, n$
44: Calcular $GapJ_{i,j}$ e $GapM_{i,j}, Gap_{ij}$ e Matriz G
45: **Retornar** Solução seq_z melhorada, C_{max} e TEC

Algoritmo 7 Seleção de soluções não dominadas - Ramezianian

```

1: Entrada: Conjunto de Soluções  $B$ 
2:  $NDS$  representa o conjunto de soluções não dominadas
3: for cada  $b \in B$  do
4:    $S_b = \square$ 
5:    $n_b = 0$ 
6:   for cada  $q \in B$  do
7:     if ( $b \prec q$ ) (se  $b$  domina  $q$ ) then
8:        $S_b = S_b \cup q$ 
9:     else if  $q \prec b$  then
10:       $n_b = n_b + 1$ 
11:    if  $n_b = 0$  then
12:       $NDS = NDS \cup \{b\}$ 
13: Retornar Conjunto de soluções não dominadas
  
```

3.2.1 Comentários sobre o algoritmo de economia de energia proposto por Ramezianian

Ao se tentar reproduzir o algoritmo de economia de energia (Algoritmo 6), foram observadas algumas inconsistências. Isto fez com que o mesmo não pudesse ser implementado tal como apresentado no trabalho de Ramezianian, Vali-Siar e Jalalian (2019). As inconsistências são listadas abaixo:

1. Existe uma inconsistência no cálculo dos *Gaps* entre *Jobs* da mesma máquina ($GapJ_{i[j]}$) na Linha 11. Como o problema em questão leva em consideração os tempos de *setup* entre *jobs* na mesma máquina, o tempo de *setup* deve ser considerado no cálculo do *gap* subtraindo-se seu valor. Caso estes tempos não sejam subtraídos, será encontrado um tempo de *Gap* maior que o real. Ao se fazer esta correção, o cálculo do $GapJ_{i[j]}$ na Linha 11 deveria ser calculado conforme apresentado na Equação (3.26):

$$GapJ_{i[j]} = E_{i,[j+1]} - pr_{i,[j+1]} - E_{i,[j]} - s_{i,[j][j+1]} \quad (3.26)$$

2. A condição do laço de repetição no início da 3ª Etapa (Linha 24) do Algoritmo 6 não garante a parada do algoritmo. Como apresentado no pseudocódigo, a busca pela melhor economia de energia deve prosseguir enquanto existirem *gaps*. Porém, não há como se garantir que o método de redução de energia proposto corrija totalmente todos os *gaps* (assumirem valores zero). Isto deve-se principalmente a limitação de três níveis de velocidade propostos. Com mais níveis de velocidade mais lentos, os *gaps* poderiam ser reduzidos. No entanto, não há como garantir que seriam totalmente preenchidos em todas as situações. Desta forma este laço pode entrar em um *loop* infinito.

3. Não fica claro como é realizado o cálculo da energia unitária consumida por cada *job* em cada máquina (Linhas 26 e 34). No trabalho é apenas apresentado o cálculo do consumo total de energia.
4. Existe uma inconsistência no cálculo de $\Delta'_{G(q,1)(q,2)}$ na Linha 29. O intuito desta variável é calcular a diferença do tempo de processamento do *job* selecionado no seu nível de velocidade atual e com o nível de velocidade reduzido em um nível. Isso faz com que seja possível verificar a condição da Linha 30: verificar se com a redução de velocidade, o *job* consegue preencher parcial ou totalmente o *gap* sem alterar o *makespan*. Desta forma, $\Delta'_{G(q,1)(q,2)}$ deveria ser corrigido para ser calculado conforme apresentado na Equação (3.27):

$$\Delta'_{G(q,1)(q,2)} = \frac{p_{G(q,1)(q,2)}}{sp_{l-1}} - \frac{p_{G(q,1)(q,2)}}{sp_l} \quad (3.27)$$

5. Não fica claro como $\Delta_{G(q,1)(q,2)}$ e $\Delta'_{G(q,1)(q,2)}$ devem ser atualizados na Linha 34. $\Delta_{G(q,1)(q,2)}$ deve ser atualizado reduzindo o $\Delta'_{G(q,1)(q,2)}$ de seu valor, pois o *gap* para checar novas reduções de velocidade agora é menor, visto que foi atendida a condição da linha 30 e a velocidade foi reduzida. $\Delta'_{G(q,1)(q,2)}$ por sua vez, também deve ser atualizado, visto que a velocidade foi reduzida e uma nova verificação na próxima iteração será realizada: verificar se a diferença do tempo de processamento do *job* selecionado no seu novo nível de velocidade e com o nível de velocidade reduzido em um nível, consegue preencher parcial ou totalmente o *gap* sem alterar o *makespan*. Desta forma, a atualização de $\Delta_{G(q,1)(q,2)}$ e $\Delta'_{G(q,1)(q,2)}$ na Linha 34, deveria ser executada conforme as equações (3.28) e (3.29):

$$\Delta_{G(q,1)(q,2)} = Gap_{G(q,1)(q,2)} - \Delta'_{G(q,1)(q,2)} \quad (3.28)$$

$$\Delta'_{G(q,1)(q,2)} = \frac{p_{G(q,1)(q,2)}}{sp_{l-1}} - \frac{p_{G(q,1)(q,2)}}{sp_l} \quad (3.29)$$

6. Por fim, além das inconsistências relacionadas aos cálculos unitários de energia não apresentados nas Linhas 26, 35 e 36, algumas melhorias quanto à eficiência do método poderiam ser colocadas em prática. No algoritmo proposto por Ramezani, Vali-Siar e Jalalian (2019), apenas a alteração de velocidade do *job* que resultar na maior redução de energia é de fato alterada na matriz de velocidade original da solução seq_z , a cada iteração do laço da Linha 24. Do ponto de vista computacional, estes passos poderiam ser simplificados, visto que todos os *gaps* já foram calculados e o algoritmo já possui verificações para alterar a velocidade de todos os *gaps* da matriz G no laço da Linha

27. Desta forma, pode-se dispensar o laço da Linha 27 e todos os cálculos posteriores à 34.

Na Seção 3.3, será apresentado um algoritmo de economia de energia adaptado de Ramezani, Vali-Siar e Jalalian (2019), que corrige todos os aspectos apresentados anteriormente.

3.3 DESCRIÇÃO DO MÉTODO

O problema abordado neste trabalho é NP-difícil. Devido à complexidade combinatória e limitações de tempo, métodos heurísticos são utilizados na solução de problemas de grande porte (HEJAZI; SAGHAFIAN, 2005). O método utilizado para a resolução do GPFSPSDS foi o 2PPLS proposto por Lust e Teghem (2010) e utilizado por Costa et al. (2018). As duas fases do método são apresentadas a seguir.

3.3.1 Primeira fase

A primeira fase visa encontrar um conjunto de soluções como população inicial. Neste trabalho, essas soluções são obtidas empregando algoritmos heurísticos que combinam métodos construtivos com procedimentos de busca local. Foram consideradas diferentes configurações que são apresentadas a seguir:

1. **NEHT-RB + SWAP (Algoritmo 8)**: Consiste no Algoritmo 3 sem a utilização do método de economia de energia;
2. **NEHT-RB + SWAP + economia de energia adaptado (Algoritmo 9)**: Consiste no Algoritmo 3 com a utilização do método de economia de energia adaptado (Algoritmo 10).
3. **NEHT-RB + VND + economia de energia adaptado (Algoritmo 11)**: Consiste no Algoritmo 3 com um VND (Algoritmo 12) como busca local ao invés da busca local proposta por Ramezani, Vali-Siar e Jalalian (2019). Esta configuração utiliza o método de economia de energia adaptado.

Nas configurações **NEHT-RB + SWAP** e **NEHT-RB + SWAP + economia de energia adaptado**, será analisada a influência de um método de economia de energia. Nas configurações **NEHT-RB + SWAP + economia de energia adaptado** e **NEHT-RB +**

Algoritmo 8 NEHT-RB + SWAP

- 1: **Entrada:** Conjunto de máquinas e *jobs* não programados, sp_l, p_{ij}
 - 2: **1ª Etapa (Inicialização)**
 - 3: Definir contador $z = 0$
 - 4: Definir B como o conjunto de todas as soluções viáveis
 - 5: $B = \emptyset$
 - 6: Setar Matriz de Velocidade no nível mais alto de velocidade (Ex.: $V_z : [v_{ij} = sp_2], \forall i = 1, \dots, n; j = 1, \dots, m$)
 - 7: **2ª Etapa (Solução Inicial)**
 - 8: $seq_z \leftarrow$ NEHT-RB($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 9: **3ª Etapa**
 - 10: $seq_z \leftarrow$ Busca Local *Swap* (seq_z, n)
 - 11: $B = B \cup seq_z$
 - 12: $z = z + 1$
 - 13: $[k]$ representa o *job* na posição k da seq_z e $op_{i[k]}$ denota a operação na posição k na máquina i
 - 14: $op_{n_{slow}}$ representa o conjunto de operações em que o nível de velocidade não é baixo (Ex.: $v_{i[k]} \neq sp_0$)
 - 15: **4ª Etapa**
 - 16: **while** $op_{n_{slow}} \neq \emptyset$ **do**
 - 17: Encontrar a operação $op_{i[\alpha]}$ tal que $op_{i[\alpha]} = \min[p_{i[k]}/v_{i[k]}], v_{i[k]} \in V_z, \forall i = 1, \dots, m$
 - 18: Reduzir o nível de velocidade de $op_{i[\alpha]}$ em um nível ($v_{i[\alpha]} : sp_l \rightarrow sp_{l-1}$) e atualizar V_z
 - 19: $seq_z \leftarrow$ NEHT-RB($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 20: $seq_z \leftarrow$ Busca Local *Swap* (seq_z, n)
 - 21: $B = B \cup seq_z$
 - 22: Atualizar $op_{n_{slow}}$
 - 23: $z = z + 1$
 - 24: **5ª Etapa**
 - 25: Aplicar o Algoritmo (7) de seleção de soluções não dominadas no conjunto B
 - 26: **Retornar** Conjunto de soluções não dominadas
-

Algoritmo 9 NEHT-RB + SWAP + Economia de Energia Adaptado

- 1: **Entrada:** Conjunto de máquinas e *jobs* não programados, sp_l, p_{ij}
 - 2: **1ª Etapa (Inicialização)**
 - 3: Definir contador $z = 0$
 - 4: Definir B como o conjunto de todas as soluções viáveis
 - 5: $B = \emptyset$
 - 6: Setar Matriz de Velocidade no nível mais alto de velocidade (Ex.: $V_z : [v_{ij} = sp_2], \forall i = 1, \dots, n; j = 1, \dots, m$)
 - 7: **2ª Etapa (Solução Inicial)**
 - 8: $seq_z \leftarrow$ NEHT-RB($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 9: **3ª Etapa**
 - 10: $seq_z \leftarrow$ Busca Local *Swap* (seq_z, n)
 - 11: $B = B \cup seq_z$
 - 12: $z = z + 1$
 - 13: $[k]$ representa o *job* na posição k da seq_z e $op_{i[k]}$ denota a operação na posição k na máquina i
 - 14: $op_{n_{slow}}$ representa o conjunto de operações em que o nível de velocidade não é baixo (Ex.: $v_{i[k]} \neq sp_0$)
 - 15: **4ª Etapa (Busca por Soluções Energeticamente Eficientes)**
 - 16: **while** $op_{n_{slow}} \neq \emptyset$ **do**
 - 17: Encontrar a operação $op_{i[\alpha]}$ tal que $op_{i[\alpha]} = \min[p_{i[k]}/v_{i[k]}], v_{i[k]} \in V_z, \forall i = 1, \dots, m$
 - 18: Reduzir o nível de velocidade de $op_{i[\alpha]}$ em um nível ($v_{i[\alpha]} : sp_l \rightarrow sp_{l-1}$) e atualizar V_z
 - 19: $seq_z \leftarrow$ NEHT-RB($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 20: $seq_z \leftarrow$ Busca Local *Swap* (seq_z, n)
 - 21: $seq_z \leftarrow$ Econ. de Energia Adaptado ($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 22: $B = B \cup seq_z$
 - 23: Atualizar $op_{n_{slow}}$
 - 24: $z = z + 1$
 - 25: **5ª Etapa**
 - 26: Aplicar o Algoritmo (7) de seleção de soluções não dominadas no conjunto B
 - 27: **Retornar** Conjunto de soluções não dominadas
-

Algoritmo 10 Economia de Energia Adaptado

```

1: Entrada:  $p_{ij}, V'_Z = [v'_{ij}], e_{ij}, seq_z, n, m, spl$ 
2:  $V_Z = V'_Z, (v'_{ij} = v_{ij}), i = 1, \dots, m; j = 1, \dots, n$ 
3: 1ª Etapa (Encontrar os gaps)
4:  $[j]$  e  $[j + 1]$  representam o job na posição  $j$  e o job na posição  $j + 1$ 
5:  $GapJ_{i,[j]}$  representa o gap entre o job  $[j]$  e o job  $[j + 1]$  na máquina  $i$ 
6:  $GapM_{i,[j]}$  representa o gap para o job  $[j]$  entre a máquina  $i$  e a máquina  $i + 1$ 
7: Calcular  $pr_{ij} = p_{ij}/v_{ij}, (v_{ij} \in V_Z), i = 1, \dots, m; j = 1, \dots, n$ 
8: for  $i = 1$  to  $m$  do
9:   for  $j = 1$  to  $n - 1$  do
10:      $GapJ_{i[j]} = E_{i,[j+1]} - pr_{i,[j+1]} - E_{i,[j]} - s_{i,[j][j+1]}$ 
11:   for  $i = 1$  to  $m - 1$  do
12:      $GapJ_{i,n} = inf$ 
13:   for  $i = 1$  to  $m - 1$  do
14:     for  $j = 1$  to  $n$  do
15:        $GapM_{i[j]} = E_{i+1,[j]} - pr_{i+1,[j]} - E_{i,[j]}$ 
16:   for  $j = 1$  to  $n - 1$  do
17:      $GapM_{m,j} = inf$ 
18:  $GapJ_{m,n} = 0$  e  $GapM_{m,n} = 0$ 
19:  $Gap_{ij} = \min([GapJ_{i,j}], [GapM_{i,j}]), i = 1, \dots, m; j = 1, \dots, n$ 
20:  $G$  representa a matriz de máquinas e jobs que satisfazem  $Gap_{ij} > 0$ 
21:  $\beta$  representa o número de linhas em  $G$ 
22: 2ª Etapa (Correção de Gaps)
23: for  $q = 1 : \beta$  do
24:   if  $V_{G(q,1)(q,2)} \neq sp_0$  then
25:      $\Delta_{G(q,1)(q,2)} = Gap_{G(q,1)(q,2)}$ 
26:      $\Delta'_{G(q,1)(q,2)} = p_{G(q,1)(q,2)}/spl_{-1} - p_{G(q,1)(q,2)}/spl$ 
27:     while  $\Delta_{G(q,1)(q,2)} - \Delta'_{G(q,1)(q,2)} \geq 0$  do
28:       Atualizar Matriz  $V_Z, (v_{G(q,1)(q,2)} = spl_{-1})$ 
29:       if  $v_{G(q,1)(q,2)} = sp_0$  then
30:         Sair do loop
31:        $\Delta_{G(q,1)(q,2)} = Gap_{G(q,1)(q,2)} - \Delta'_{G(q,1)(q,2)}$ 
32:        $\Delta'_{G(q,1)(q,2)} = p_{G(q,1)(q,2)}/spl_{-1} - p_{G(q,1)(q,2)}/spl$ 
33: Retornar Solução  $seq_z$  melhorada,  $V_Z, C_{max}$  e  $TEC$ 

```

VND + economia de energia adaptado será analisada o impacto de incluir mais estruturas de vizinhança na busca local.

Desta forma, o procedimento de busca local utilizado na configuração **NEHT-RB + VND + economia de energia adaptado** (Algoritmo 11), foi a Descida em Vizinhança Variável (VND) proposta por Mladenović e Hansen (1997) (Algoritmo 12) com abordagem *Best Improvement* utilizando 6 movimentos de vizinhança:

- Exchange (Figura 5(a)): Troca de posição entre duas tarefas;
- Relocate (Figura 5(b)): Mudança de posição de uma tarefa na sequência de produção;
- Or-opt (Figura 5(c) e 5(d)): Realoca um bloco contíguo de tarefas em outra posição da sequência (se 2 tarefas, Or-opt2, se 3 tarefas Or-opt3);

Algoritmo 11 NEHT-RB + VND + Economia de Energia Adaptado

- 1: **Entrada:** Conjunto de máquinas e *jobs* não programados, sp_l, p_{ij}
 - 2: **1ª Etapa (Inicialização)**
 - 3: Definir contador $z = 0$
 - 4: Definir B como o conjunto de todas as soluções viáveis
 - 5: $B = \emptyset$
 - 6: Setar Matriz de Velocidade no nível mais alto de velocidade (Ex.: $V_z : [v_{ij} = sp_2], \forall i = 1, \dots, n; j = 1, \dots, m$)
 - 7: **2ª Etapa (Solução Inicial)**
 - 8: $seq_z \leftarrow$ NEHT-RB($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 9: **3ª Etapa**
 - 10: $seq_z \leftarrow$ Busca Local VND (seq_z, n)
 - 11: $B = B \cup seq_z$
 - 12: $z = z + 1$
 - 13: $[k]$ representa o *job* na posição k de seq_z e $op_{i[k]}$ denota a operação na posição k na máquina i
 - 14: $op_{n_{slow}}$ representa o conjunto de operações em que o nível de velocidade não é baixo (Ex.: $v_{i[k]} \neq sp_0$)
 - 15: **4ª Etapa (Busca por Soluções Energeticamente Eficientes)**
 - 16: **while** $op_{n_{slow}} \neq \emptyset$ **do**
 - 17: Encontrar a operação $op_{i[\alpha]}$ tal que $op_{i[\alpha]} = \min[p_{i[k]}/v_{i[k]}], v_{i[k]} \in V_z, \forall i = 1, \dots, m$
 - 18: Reduzir o nível de velocidade de $op_{i[\alpha]}$ em um nível ($v_{i[\alpha]} : sp_l \rightarrow sp_{l-1}$) e atualizar V_z
 - 19: $seq_z \leftarrow$ NEHT-RB($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 20: $seq_z \leftarrow$ Busca Local VND (seq_z, n)
 - 21: $seq_z \leftarrow$ Econ. de Energia Adaptado ($p_{ij}, s_{i,j,k}, V_z, n, m$)
 - 22: $B = B \cup seq_z$
 - 23: Atualizar $op_{n_{slow}}$
 - 24: $z = z + 1$
 - 25: **5ª Etapa**
 - 26: Aplicar o Algoritmo (7) de seleção de soluções não dominadas no conjunto B
 - 27: **Retornar** Conjunto de soluções não dominadas
-

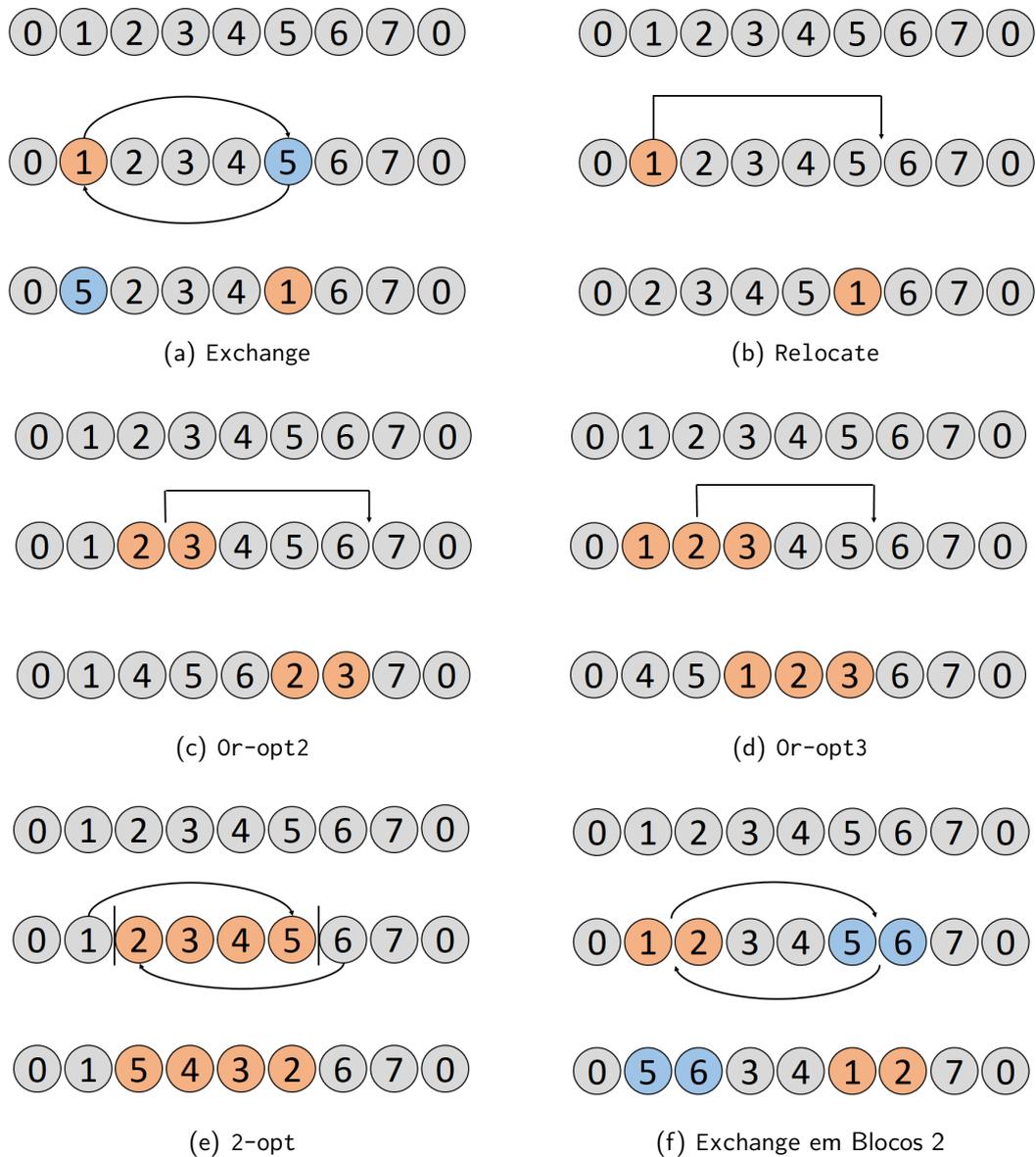
- 2-Opt (Figura 5(e)): Remove duas arestas (arcos) não adjacentes e reconecta a sequência usando duas arestas alternativas;
- Exchange em blocos (Figura 5(f)): Troca de Posição entre blocos contíguos de n tarefas.

Algoritmo 12 Busca Local - VND

- 1: **Entrada:** $s, P, st, V, n, m, nivel_v$
 - 2: Seja r o número de estruturas diferentes de vizinhança
 - 3: $k \leftarrow 0$
 - 4: **while** $k < r$ **do**
 - 5: Encontre o melhor vizinho s'
 - 6: **if** $makespan(s') < makespan(s)$ **then**
 - 7: $s \leftarrow s'$
 - 8: $k \leftarrow 0$
 - 9: **else**
 - 10: $k \leftarrow k + 1$
 - 11: **Retornar** s
-

Po fim, o algoritmo de economia de energia adaptado, considera os apontamentos realizados na Seção 3.2 e é apresentado no pseudocódigo 10.

Figura 4 – Movimentos para gerar vizinhança



Fonte: O autor (2023).

3.3.2 Segunda fase

Na segunda fase, procede-se com uma busca por soluções aproximadas, localizadas entre as soluções já localizadas na fronteira atual. Nesta fase, as soluções são geradas a partir da aplicação de um procedimento de busca local (Algoritmo 13) *first improvement* com as vizinhanças: exchange e relocate combinados com o algoritmo de economia de energia adaptado.

A segunda fase do método tem por objetivo encontrar soluções eficientes entre as soluções

já contidas na fronteira, e assim aumentar o número e/ou qualidade de soluções no conjunto de soluções obtidos durante a primeira fase. Esse processo se dá através da utilização de um procedimento de PLS adaptado. Ele é apresentado no Algoritmo 13.

O método inicia-se com uma população P , composta por potenciais soluções contidas na população inicial P_0 . Esse conjunto de soluções foi gerado durante a primeira fase do método.

Durante a execução do procedimento, todos os vizinhos p' para cada uma das soluções $p \in P$ são gerados, e tem seus *gaps* corrigidos com o algoritmo de economia de energia adaptado (Linha 8). Se uma solução p' gerada não for dominada por p , tenta-se adicioná-la ao conjunto de soluções (Linha 10). O método *AddSolution* não só adiciona novas soluções ao conjunto de soluções, como também remove soluções presentes em X_E , que passam a ser dominadas (Linha 12). Sempre que uma solução $p' \in N$ é adicionada ao conjunto de soluções, ela é também colocada em uma lista de soluções que será analisada uma vez que todas as soluções da atual população tenham sido exploradas.

Neste trabalho, as soluções da segunda fase foram analisadas em diferentes configurações apresentadas a seguir:

1. **NEHT-RB + SWAP + 2ª Fase**: Consiste no Algoritmo 8 com a inclusão da segunda fase (Algoritmo 13);
2. **NEHT-RB + SWAP + economia de energia adaptado + 2ª Fase**: Consiste no Algoritmo 9 com a inclusão da segunda fase (Algoritmo 13);
3. **NEHT-RB + VND + 2ª Fase**: Consiste no Algoritmo 14 com a inclusão da segunda fase (Algoritmo 13);
4. **NEHT-RB + VND + economia de energia adaptado + 2ª Fase**: Consiste no Algoritmo 11 com a inclusão da segunda fase (Algoritmo 13)

Nas configurações **NEHT-RB + SWAP** e **NEHT-RB + SWAP + 2ª Fase** será analisada a influência da inclusão de um método de segunda fase. Nas configurações **NEHT-RB + SWAP + 2ª Fase** e **NEHT-RB + SWAP + economia de energia adaptado + 2ª Fase** será avaliado o impacto de corrigir os *gaps* das soluções geradas na primeira fase anterior ou posteriormente a segunda fase. O mesmo será realizado com as configurações **NEHT-RB + VND + 2ª Fase** e **NEHT-RB + VND + economia de energia adaptado + 2ª Fase**, porém com a busca local VND.

Algoritmo 13 Pareto Local Search

```

1: Entrada: População Inicial  $P_0$ 
2:  $X_E \leftarrow P_0$ 
3:  $P \leftarrow P_0$ 
4:  $P_a \leftarrow \emptyset$ 
5: while  $P \neq \emptyset$  do
6:   for  $p \in P$  do
7:     for  $p' \in N(p)$  do
8:       Aplicar o Algoritmo de economia de energia adaptado (Alg. 10) em  $p'$ 
9:       if  $z(p) \leq z(p')$  then
10:        AddSolution( $X_E \uparrow, p' \downarrow, z(p') \downarrow, Added \uparrow$ )
11:        if  $Added = true$  then
12:          AddSolution( $P_a \uparrow, p' \downarrow, z(p') \downarrow$ )
13:    $P \leftarrow P_a$ 
14:    $P_a \leftarrow \emptyset$ 
15: Retornar Uma aproximação  $X_E$  do conjunto de soluções

```

Algoritmo 14 NEHT-RB + VND

```

1: Entrada: Conjunto de máquinas e jobs não programados,  $sp_l, p_{ij}$ 
2: 1ª Etapa (Inicialização)
3: Definir contador  $z = 0$ 
4: Definir  $B$  como o conjunto de todas as soluções viáveis
5:  $B = \square$ 
6: Setar Matriz de Velocidade no nível mais alto de velocidade (Ex.:  $V_z : [v_{ij} = sp_2], \forall i = 1, \dots, n; j = 1, \dots, m$ )
7: 2ª Etapa (Solução Inicial)
8:  $seq_z \leftarrow \text{NEHT-RB}(p_{ij}, s_{i,j,k}, V_z, n, m)$ 
9: 3ª Etapa
10:  $seq_z \leftarrow \text{Busca Local VND}(seq_z, n)$ 
11:  $B = B \cup seq_z$ 
12:  $z = z + 1$ 
13:  $[k]$  representa o job na posição  $k$  da  $seq_z$  e  $op_{i[k]}$  denota a operação na posição  $k$  na máquina  $i$ 
14:  $op_{n_{slow}}$  representa o conjunto de operações em que o nível de velocidade não é baixo (Ex.:  $v_{i[k]} \neq sp_0$ )
15: 4ª Etapa
16: while  $op_{n_{slow}} \neq \square$  do
17:   Encontrar a operação  $op_{i[\alpha]}$  tal que  $op_{i[\alpha]} = \min[p_{i[k]}/v_{i[k]}], v_{i[k]} \in V_z, \forall i = 1, \dots, m$ 
18:   Reduzir o nível de velocidade de  $op_{i[\alpha]}$  em um nível ( $v_{i[\alpha]} : sp_l \rightarrow sp_{l-1}$ ) e atualizar  $V_z$ 
19:    $seq_z \leftarrow \text{NEHT-RB}(p_{ij}, s_{i,j,k}, V_z, n, m)$ 
20:    $seq_z \leftarrow \text{Busca Local VND}(seq_z, n)$ 
21:    $B = B \cup seq_z$ 
22:   Atualizar  $op_{n_{slow}}$ 
23:    $z = z + 1$ 
24: 5ª Etapa
25: Aplicar o Algoritmo (7) de seleção de soluções não dominadas no conjunto  $B$ 
26: Retornar Conjunto de soluções não dominadas

```

4 RESULTADOS E DISCUSSÕES

Neste capítulo, são reportados os resultados dos experimentos realizados com o 2PPLS na resolução do GPFSPSTSDS. Os algoritmos propostos foram implementados em linguagem de programação C++ e executado em um computador com processador Intel Core i5-3230 2.60GHz, 8 GB de memória RAM e Sistema Operacional Windows 10. Foram utilizadas as instâncias geradas por Ruiz, Maroto e Alcaraz (2005) para o PFSPTSDS. Essas instâncias foram utilizadas em dois conjuntos denominados SDST100 e SDST125 com os tempos de *setup* uniformemente distribuídos em $[1, 99]$ e $[1, 124]$ respectivamente. Estas instâncias estão disponíveis em <http://www.upv.es>. As instâncias utilizadas nos experimentos foram as mesmas utilizadas no trabalho de Ramezani, Vali-Siar e Jalalian (2019). Neste trabalho, um problema com m máquinas e n tarefas é denotado como $n \times m$. Os valores e intervalos dos fatores usados nos experimentos são os seguintes:

- Número de máquinas (m): $\{5, 10, 20\}$ para *large-sized instances*(RUIZ; MAROTO; ALCARAZ, 2005);
- Número de tarefas (n): $\{20, 50\}$ para *large-sized instances*(RUIZ; MAROTO; ALCARAZ, 2005);
- Distribuição do tempo de *setup*: $U(1, 99)$, $U(1, 124)$ (TAILLARD, 1990; RUIZ; STÜTZLE, 2008; MANSOURI; AKTAS; BESI KCI, 2016);
- Parâmetros do problema (MANSOURI; AKTAS; BESI KCI, 2016):
 - ϑ_i : 0,05 (Fator de conversão para o tempo ocioso na máquina i);
 - π_i : 50 (Potência da máquina i);
 - sp_l : $\{0, 8; 1; 1, 2\}$ (Fator de velocidade de processamento l);
 - λ_l : $\{0, 6; 1; 1, 5\}$ (Fator de conversão para a velocidade de processamento l).

Na literatura do GPFSPSTSDS, o único trabalho que propõe métodos de resolução e apresenta resultados é o de Ramezani, Vali-Siar e Jalalian (2019). No entanto, uma comparação direta com os resultados propostos neste trabalho não é possível, uma vez que: 1) As fronteiras de Pareto geradas em Ramezani, Vali-Siar e Jalalian (2019) não estão disponíveis; 2) Uma implementação alternativa do método de Ramezani, Vali-Siar e Jalalian (2019) não seria

possível devido às inconsistências listadas na Seção 3.2 e também devido à ausência de informações de parâmetros de entrada relacionadas com o problema. A ausência destes parâmetros fez como que fossem adotados nesta dissertação os parâmetros usados em Mansouri, Aktas e Besikci (2016).

A fim de avaliar o desempenho do método proposto, foram utilizadas quatro métricas, tal como indicado no Quadro 1.

Quadro 1 – Indicadores de Desempenho

Indicador	Descrição	Referência
NDS	Quanto maior, melhor	-
Tempo de Execução (s)	Quanto menor, melhor	-
Hipervolume (H)	Quantifica a qualidade da FP; Quanto maior, melhor	2.1.2.1
Medida R (R)	Quantifica a qualidade da FP; Normalizado $[0, 1]$; Quanto maior, melhor	2.1.2.2

Fonte: O autor (2023).

Foram utilizadas como medida de desempenho o Hipervolume (ZITZLER, 1999), a Medida R Jaskiewicz (2002), o NDS, para as quais quanto maior o valor melhor, e o tempo de execução, que por sua vez, quanto menor melhor.

Nas Seções 4.1 a 4.4, devido ao grande volume de instâncias, os resultados serão apresentados e discutidos com tabelas agregadas e figuras de algumas fronteiras obtidas. Os resultados completos para cada instância serão incluídos no Apêndice A. Cada combinação de $n \times m$ possui 10 instâncias para cada conjunto de distribuição de tempos de *setup*. Os valores reportados neste capítulo são as médias das medidas de desempenho.

4.1 ANÁLISE DO ALGORITMO DE ECONOMIA DE ENERGIA ADAPTADO

A Tabela 2 apresenta os resultados obtidos para as configurações **NEHT-RB + SWAP** e **NEHT-RB + SWAP + ECON. DE ENERGIA ADAPTADO**. Estas configurações são as que mais se aproximam das configurações de Ramezani, Vali-Siar e Jalalian (2019).

Na Tabela 2 observa-se que com um pequeno acréscimo de tempo computacional, o método que inclui o Algoritmo 10 (Economia de Energia Adaptado), mesmo obtendo menos NDS, as medidas R e H são melhores. A partir da Figura 5 é possível observar que o algoritmo adaptado

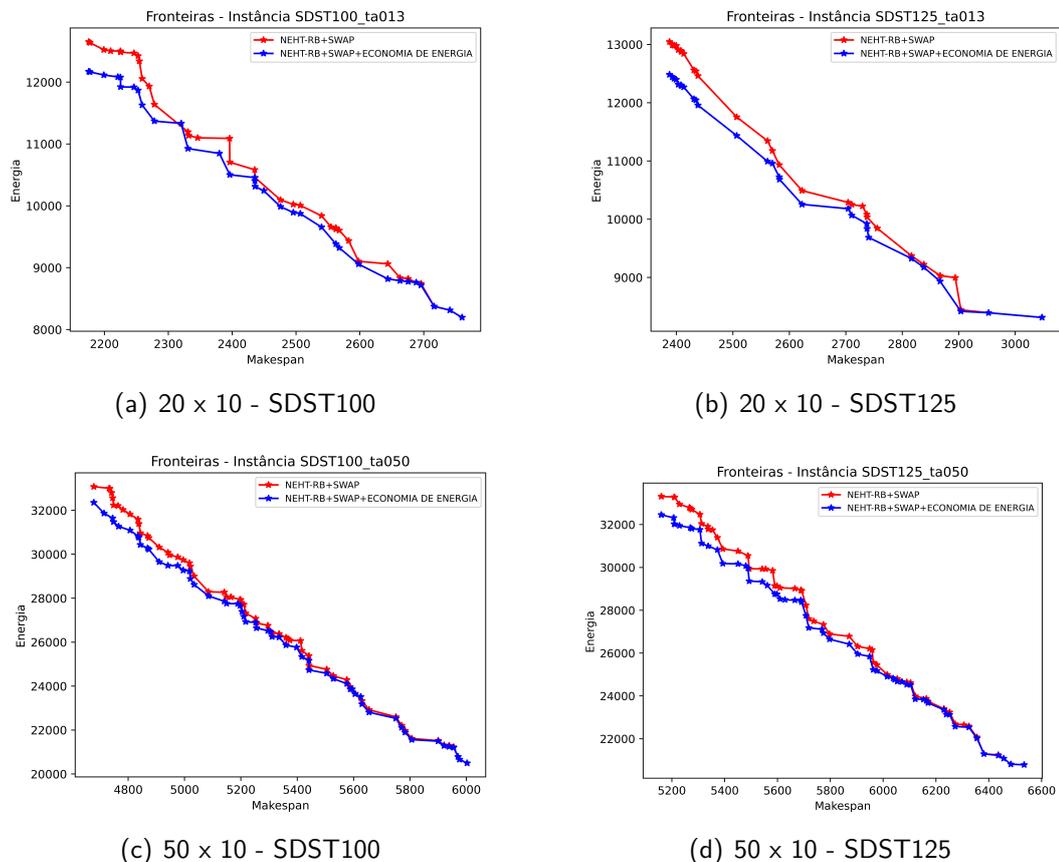
de economia de energia (Algoritmo 10) é capaz de encontrar soluções com menor consumo de energia.

Tabela 2 – Resultados comparativos entre NEHT-RB + SWAP e NEHT-RB + SWAP + ECON. DE ENERGIA

$n \times m$	NEHT-RB + SWAP				NEHT-RB + SWAP + ECON. DE ENERGIA				
	SDST	Tempo (s)	H (10^6)	NDS	R	Tempo (s)	H (10^6)	NDS	R
20×5	SDST100	0,512	1,7802	28	0,804777	0,5987	1,8352	27,4	0,806689
	SDST125	0,5124	1,9473	28,2	0,804037	0,5641	1,9667	26,3	0,806425
20×10	SDST100	1,6093	4,3391	39,1	0,81567	1,7059	4,4135	35,5	0,818655
	SDST125	1,5974	4,5634	35,2	0,811946	1,7232	4,7668	34,1	0,815375
20×20	SDST100	5,0669	12,1989	58,5	0,81981	5,2283	12,4814	57,3	0,822844
	SDST125	5,0703	12,1752	48,9	0,8133	5,3793	12,3636	45,4	0,816862
50×5	SDST100	18,8946	9,7863	41,2	0,805685	19,1133	9,8563	40,9	0,806463
	SDST125	19,0674	10,299	36,7	0,80162	19,4111	10,2407	34,6	0,80256
50×10	SDST100	57,4988	22,0244	51	0,808901	56,6986	22,4858	49,5	0,810594
	SDST125	57,2977	23,2165	47,3	0,806987	59,7646	23,7552	46	0,809103
50×20	SDST100	197,746	56,165	66	0,81312	192,479	59,04	71	0,81561
	SDST125	219,321	56,556	56	0,80989	204,919	57,985	57	0,81266

Fonte: O autor (2023).

Figura 5 – Fronteiras obtidas para NEHT-RB + SWAP e NEHT-RB + SWAP + ECON. DE ENERGIA



Fonte: O autor (2023).

4.2 ANÁLISE DO IMPACTO DA BUSCA LOCAL PROPOSTA

A Tabela 3 apresenta os resultados obtidos para as configurações **NEHT-RB + SWAP + ECON. DE ENERGIA ADAPTADO** e **NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO**. Na Tabela 3 observa-se que com um custo mais elevado de tempo computacional, o método que utiliza a busca local VND com 6 estruturas de vizinhança, melhora significativamente as NDS e as medidas R e H .

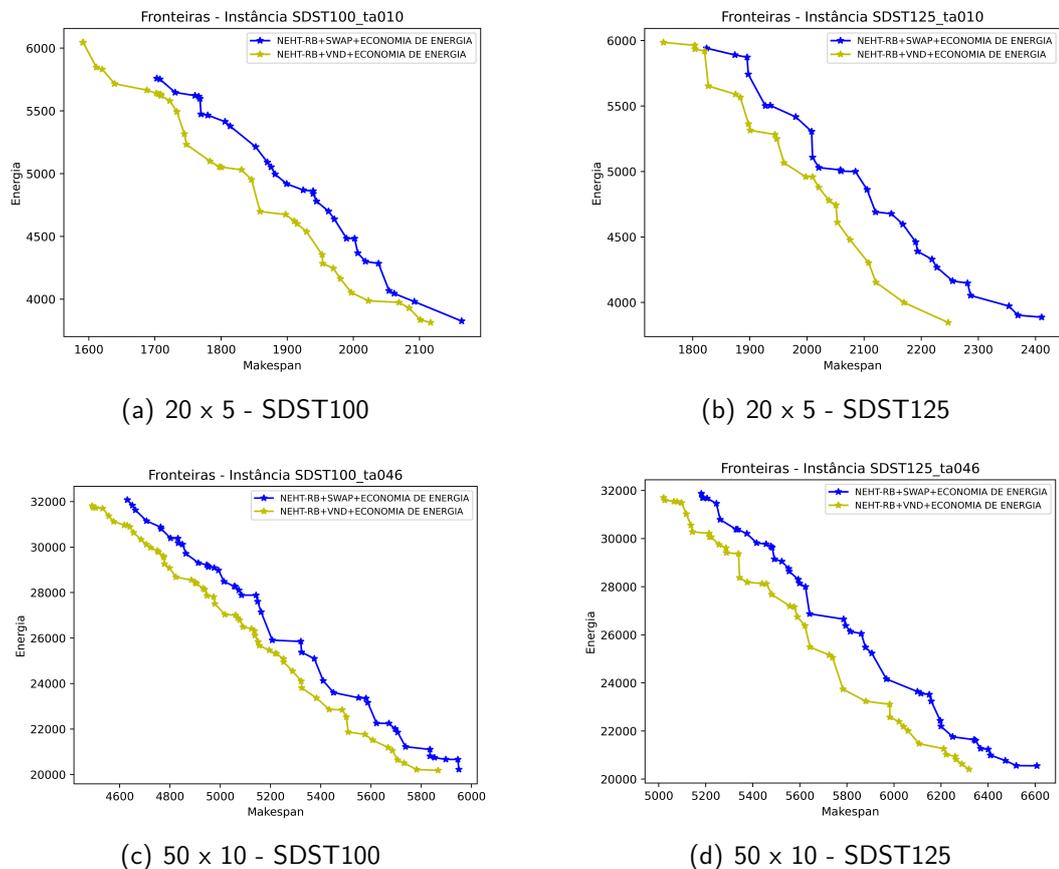
A partir da Figura 6 é possível observar que a busca local VND foi superior a busca local proposta por Ramezani, Vali-Siar e Jalalian (2019). O elevado tempo computacional deve-se ao fato de que a quantidade de movimentos realizados aumenta com o acréscimo de estruturas de vizinhança e conseqüentemente a quantidade de vezes que o *makespan* é calculado. Dessa forma, métodos mais ágeis para cálculo do *makespan* e para avaliação de movimentos seriam cruciais para instâncias com maior número de *jobs* e máquinas.

Tabela 3 – Resultados comparativos entre NEHT-RB + SWAP + ECON. DE ENERGIA e NEHT-RB + VND + ECON. DE ENERGIA

$n \times m$	NEHT-RB + SWAP + ECON. DE ENERGIA					NEHT-RB + VND + ECON. DE ENERGIA			
	SDST	Tempo (s)	H (10^6)	NDS	R	Tempo (s)	H (10^6)	NDS	R
20 × 5	SDST100	0,5987	1,8352	27,4	0,806689	3,222	2,0426	32,5	0,833021
	SDST125	0,5641	1,9667	26,3	0,806425	3,3469	2,2436	27,6	0,836031
20 × 10	SDST100	1,7059	4,4135	35,5	0,818655	8,1155	4,7656	39,8	0,833513
	SDST125	1,7232	4,7668	34,1	0,815375	8,3096	5,0382	34,9	0,833202
20 × 20	SDST100	5,2283	12,4814	57,3	0,822844	20,9314	13,2452	57,2	0,833851
	SDST125	5,3793	12,3636	45,4	0,816862	23,0025	13,2144	51,6	0,830133
50 × 5	SDST100	19,1133	9,8563	40,9	0,806463	211,7679	11,1311	42,9	0,83455
	SDST125	19,4111	10,2407	34,6	0,80256	213,8244	12,118	35,3	0,835742
50 × 10	SDST100	56,6986	22,4858	49,5	0,810594	468,4269	24,8164	51,5	0,831379
	SDST125	59,7646	23,7552	46	0,809103	472,8483	26,7058	42	0,831875
50 × 20	SDST100	192,479	59,04	71	0,81561	1497,36	62,654	65	0,83053
	SDST125	204,919	57,985	57	0,81266	1457,92	64,357	58	0,82891

Fonte: O autor (2023).

Figura 6 – Fronteiras obtidas para NEHT-RB + SWAP + ECON. DE ENERGIA e NEHT-RB + VND + ECON. DE ENERGIA



Fonte: O autor (2023).

4.3 ANÁLISE DO IMPACTO DA 2ª FASE PARA UMA 1ª FASE OBTIDA POR NEHT-RB + SWAP

A Tabela 4 apresenta os resultados obtidos para as configurações **NEHT-RB + SWAP** e **NEHT-RB + SWAP + 2ª FASE**. Na Tabela 4 observa-se que com um pequeno acréscimo de tempo computacional, o método que inclui a segunda fase (Algoritmo 13), mesmo obtendo menos NDS, as medidas R e H são melhores. A partir da Figura 7 é possível observar que a segunda fase realizada a partir de um PLS adaptado (Algoritmo 13) é capaz de encontrar novas soluções com menor consumo de energia. Neste caso, a correção dos *gaps* só é realizada na segunda fase, poupando tempo computacional e podendo possibilitar que as estruturas de vizinhança da segunda fase encontrem soluções melhores com mais facilidade.

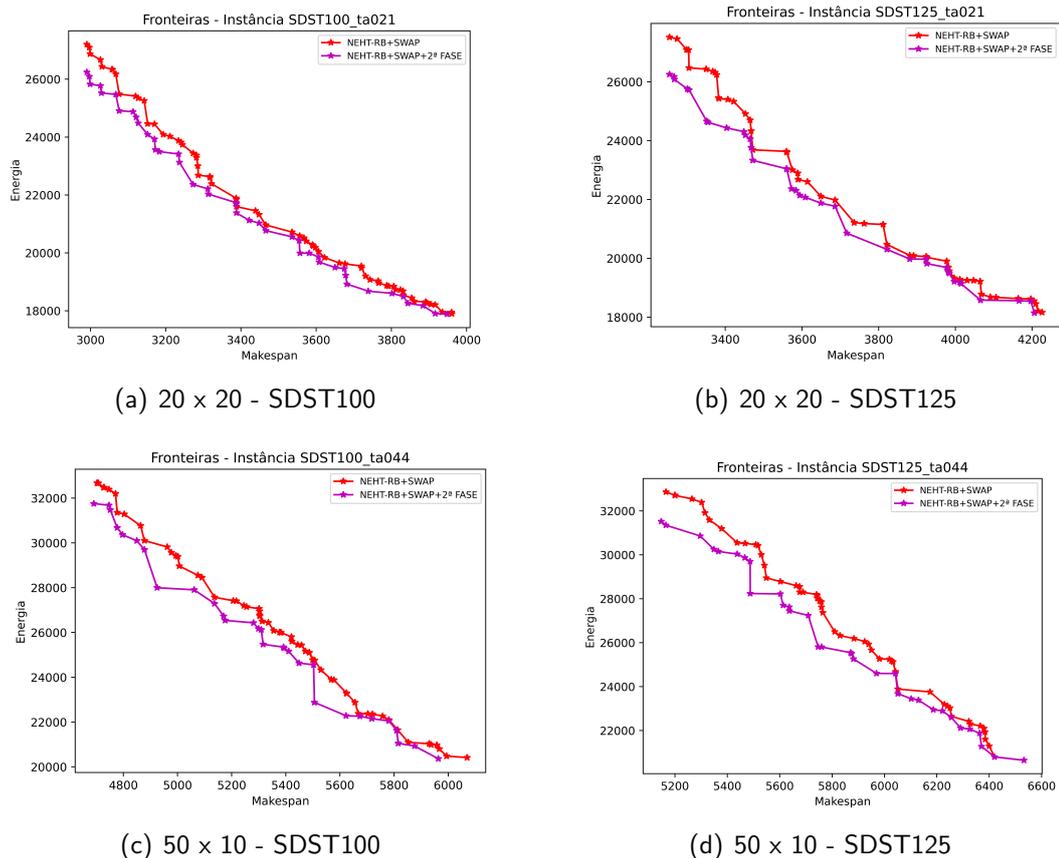
A Tabela 5 apresenta os resultados obtidos para as configurações **NEHT-RB + SWAP + 2ª FASE** e **NEHT-RB + SWAP + ECON. DE ENERGIA ADAPTADO + 2ª FASE**.

Tabela 4 – Resultados comparativos entre NEHT-RB + SWAP e NEHT-RB + SWAP + 2ª FASE

$n \times m$	NEHT-RB + SWAP					NEHT-RB + SWAP + 2ª FASE				
	SDST	Tempo (s)	H (10^6)	NDS	R	Tempo (s)	H (10^6)	NDS	R	
20 × 5	SDST100	0,512	1,7802	28	0,804777	0,7651	1,8702	21,8	0,81015	
	SDST125	0,5124	1,9473	28,2	0,804037	0,806	1,9968	20,2	0,810678	
20 × 10	SDST100	1,6093	4,3391	39,1	0,81567	2,0018	4,5284	29,5	0,820865	
	SDST125	1,5974	4,5634	35,2	0,811946	2,0222	4,6591	25,5	0,818496	
20 × 20	SDST100	5,0669	12,1989	58,5	0,81981	5,8606	12,5238	41,7	0,825013	
	SDST125	5,0703	12,1752	48,9	0,8133	5,8592	12,5815	35,5	0,818453	
50 × 5	SDST100	18,8946	9,7863	41,2	0,805685	24,5939	10,2213	28,6	0,812389	
	SDST125	19,0674	10,299	36,7	0,80162	23,4515	10,7297	22,7	0,807917	
50 × 10	SDST100	57,4988	22,0244	51	0,808901	64,5327	22,9713	37,7	0,814312	
	SDST125	57,2977	23,2165	47,3	0,806987	63,6871	24,0689	31,5	0,812777	
50 × 20	SDST100	197,746	56,165	66	0,81312	214,905	57,802	51	0,81729	
	SDST125	219,321	56,556	56	0,80989	232,255	60,754	47	0,81601	

Fonte: O autor (2023).

Figura 7 – Fronteiras obtidas para NEHT-RB + SWAP e NEHT-RB + SWAP + 2ª FASE



Fonte: O autor (2023).

A diferença entre as duas configurações consiste que na NEHT-RB + SWAP + 2ª FASE, a correção dos *gaps* só é realizada na segunda fase. Já na configuração NEHT-RB + SWAP + ECON. DE ENERGIA ADAPTADO + 2ª FASE, a correção dos *gaps* é realizada em ambas as

fases.

Na Tabela 5 observa-se que realizar a correção dos *gaps* somente na segunda fase, em média, poupa tempo computacional e possibilita que as estruturas de vizinhança da segunda fase encontrem soluções melhores com mais facilidade. A configuração NEHT-RB + SWAP + 2ª FASE, possui em média, todas as medidas de desempenho um pouco melhores.

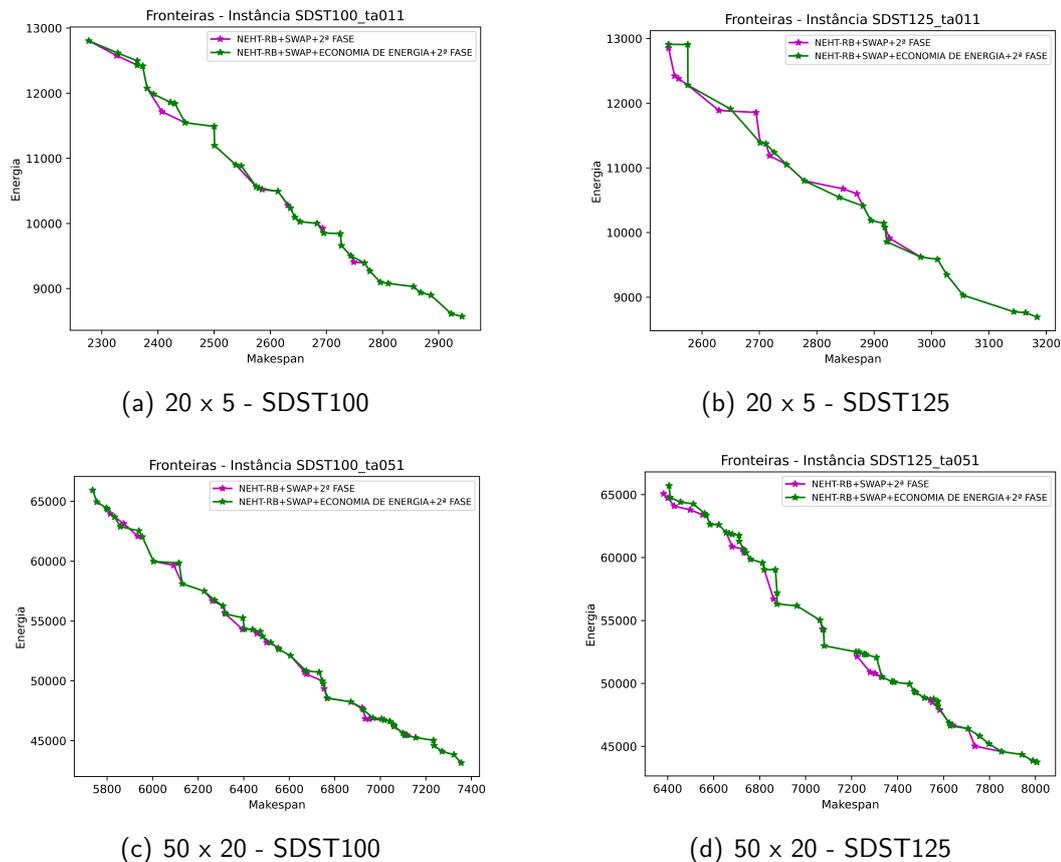
A partir da Figura 8 é possível observar que a segunda fase realizada a partir de um PLS adaptado (Algoritmo 13), pode ter pior ou melhor desempenho dependendo do método da 1ª Fase.

Tabela 5 – Resultados comparativos entre NEHT-RB + SWAP + 2ª FASE e NEHT-RB + SWAP + ECON. DE ENERGIA ADAPTADO + 2ª FASE

$n \times m$	SDST	NEHT-RB + SWAP + 2ª FASE				NEHT-RB + SWAP + ENERGIA + 2ª FASE			
		Tempo (s)	H (10^6)	NDS	R	Tempo (s)	H (10^6)	NDS	R
20 × 5	SDST100	0,7651	1,8702	21,8	0,81015	0,6671	1,8572	21,7	0,809435
	SDST125	0,806	1,9968	20,2	0,810678	0,6706	1,998	20	0,809326
20 × 10	SDST100	2,0018	4,5284	29,5	0,820865	1,9007	4,4828	29,6	0,820182
	SDST125	2,0222	4,6591	25,5	0,818496	1,9045	4,6197	24,9	0,817397
20 × 20	SDST100	5,8606	12,5238	41,7	0,825013	5,4575	12,5273	43,8	0,824093
	SDST125	5,8592	12,5815	35,5	0,818453	5,8107	12,5073	35,3	0,818094
50 × 5	SDST100	24,5939	10,2213	28,6	0,812389	22,8191	10,0607	26,2	0,810882
	SDST125	23,4515	10,7297	22,7	0,807917	22,0429	10,6167	21,7	0,807464
50 × 10	SDST100	64,5327	22,9713	37,7	0,814312	62,2951	22,6346	36,3	0,813227
	SDST125	63,6871	24,0689	31,5	0,812777	60,5891	24,0769	31,5	0,811788
50 × 20	SDST100	214,905	57,802	51	0,81729	215,514	57,603	45	0,81706
	SDST125	232,255	60,754	47	0,81601	201,346	58,629	49	0,81369

Fonte: O autor (2023).

Figura 8 – Fronteiras obtidas para NEHT-RB + SWAP + 2ª FASE e NEHT-RB + SWAP + ECON. DE ENERGIA ADAPTADO + 2ª FASE



Fonte: O autor (2023).

4.4 ANÁLISE DO IMPACTO DA 2ª FASE PARA A 1ª FASE NEHT-RB + VND E PARA A 1ª FASE NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO

A Tabela 6 apresenta os resultados obtidos para as configurações **NEHT-RB + VND + 2ª FASE** e **NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO + 2ª FASE**. A diferença entre as duas configurações consiste que na NEHT-RB + VND + 2ª FASE, a correção dos *gaps* só é realizada na segunda fase. Já na configuração NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO + 2ª FASE, a correção dos *gaps* é realizada em ambas as fases.

Na Tabela 6 observa-se que realizar ou não a correção dos *gaps* em ambas as fases, em média, possui mesmo tempo computacional. Porém, a configuração NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO + 2ª FASE, possui em média, todas as medidas de desempenho um pouco melhores, principalmente NDS.

A partir da Figura 9 é possível observar que a segunda fase realizada após a correção de

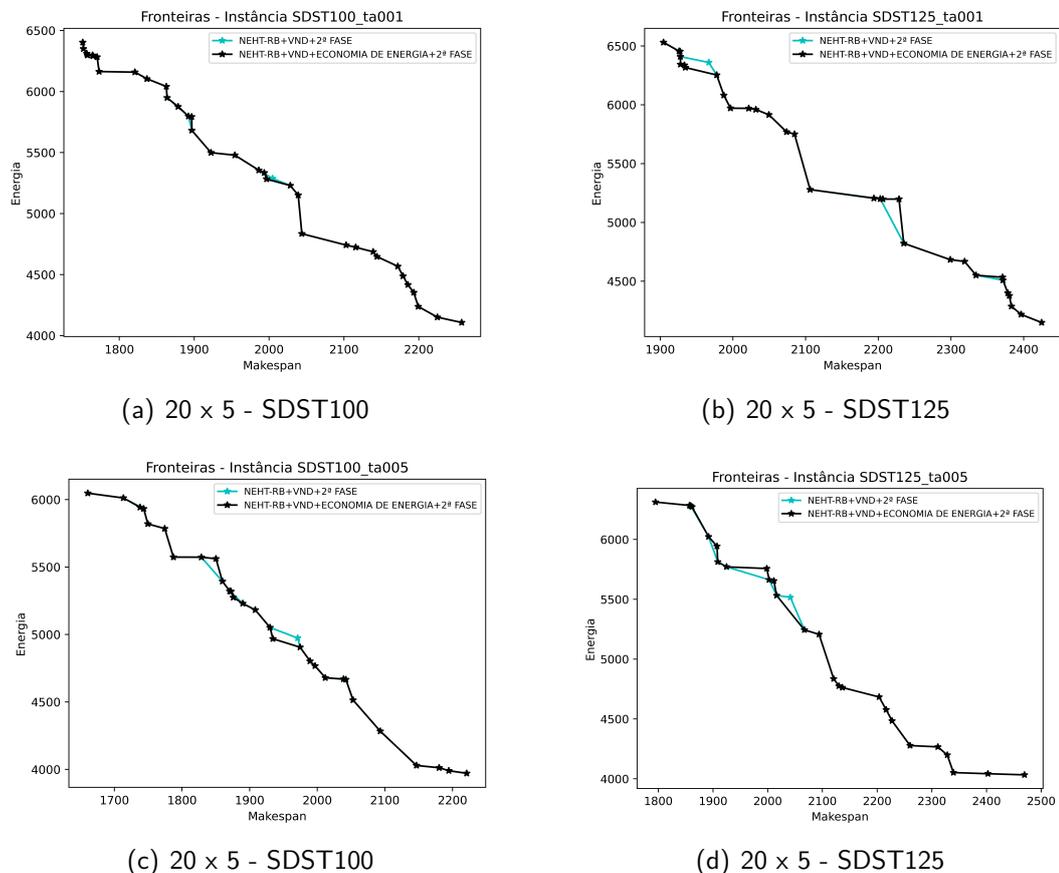
gaps na 1ª Fase, leva a resultados melhores quando utilizada a busca local VND.

Tabela 6 – Resultados comparativos entre NEHT-RB + VND + 2ª FASE e NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO + 2ª FASE

$n \times m$	NEHT-RB + VND + 2ª FASE					NEHT-RB + VND + ENERGIA + 2ª FASE				
	SDST	Tempo (s)	H (10^6)	NDS	R	Tempo (s)	H (10^6)	NDS	R	
20 × 5	SDST100	3,4059	2,0171	28,4	0,832907	3,303	2,0295	32,1	0,833021	
	SDST125	3,3424	2,2427	24,8	0,835957	3,4241	2,2437	27,4	0,836035	
20 × 10	SDST100	8,3012	4,7686	35	0,833439	8,1689	4,7609	38,9	0,833513	
	SDST125	8,26	4,9434	30,1	0,833131	8,2393	5,0205	34,1	0,833207	
20 × 20	SDST100	20,894	13,0555	49,1	0,833807	21,7713	13,1877	56,2	0,833854	
	SDST125	22,3236	13,0933	42,6	0,830026	22,073	13,1024	50,2	0,830133	
50 × 5	SDST100	216,2843	11,0938	38,6	0,834535	209,9128	11,1328	42,6	0,834554	
	SDST125	213,255	12,0932	33,7	0,835717	214,217	12,089	35,2	0,835742	
50 × 10	SDST100	481,0486	24,3602	46,2	0,831353	476,4346	24,7976	51,1	0,831382	
	SDST125	479,7501	26,471	36,1	0,83115	479,5486	26,7021	41,7	0,831876	
50 × 20	SDST100	1417,76	62,758	57	0,83053	1425,06	62,654	65	0,83053	
	SDST125	1434,91	64,441	51	0,8289	1436,04	64,358	58	0,82892	

Fonte: O autor (2023).

Figura 9 – Fronteiras obtidas para NEHT-RB + VND + 2ª FASE e NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO + 2ª FASE

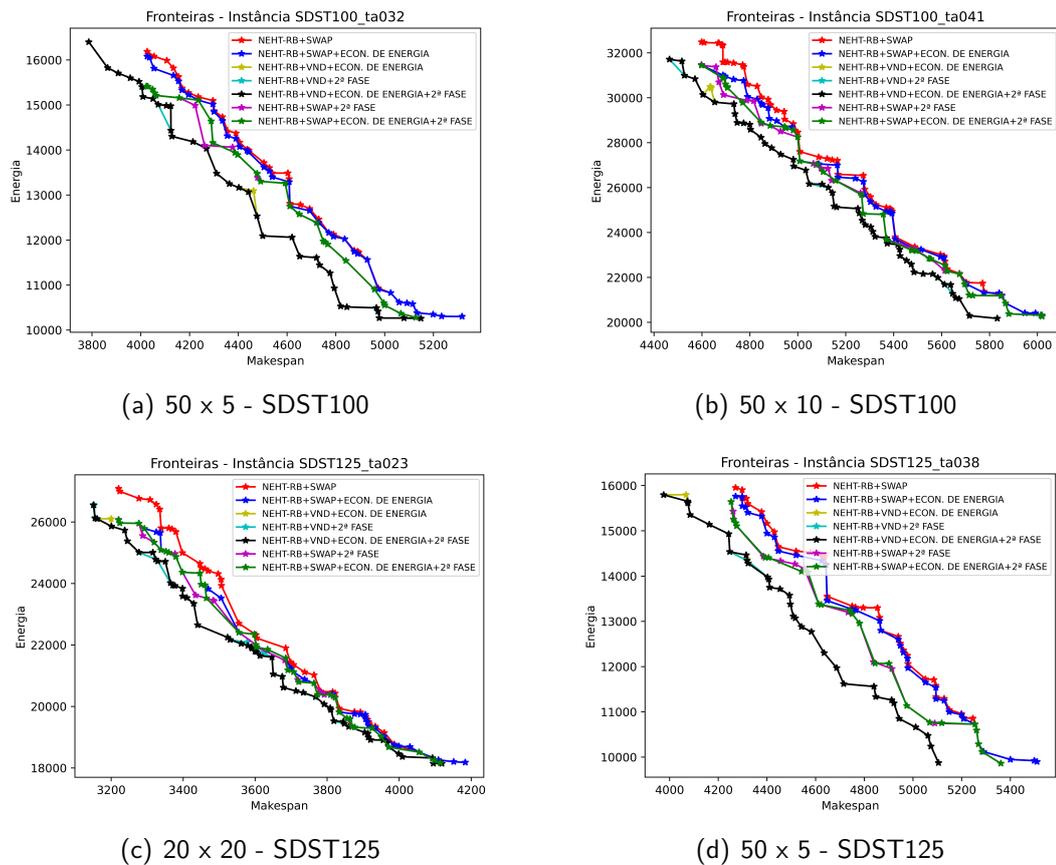


Fonte: O autor (2023).

4.5 RESUMO DAS CONFIGURAÇÕES

A Figura 10 apresenta algumas instâncias com as fronteiras obtidas em cada configuração. Na média, a configuração NEHT-RB + VND + ECON. DE ENERGIA ADAPTADO + 2ª FASE possui desempenho superior nas medidas H e R , boa quantidade de NDS quando comparada com as outras configurações.

Figura 10 – Resumo das Fronteiras obtidas com as diferentes configurações



Fonte: O autor (2023).

5 CONCLUSÕES

A pesquisa em questão considerou o GPFSTSDS biobjetivo, que visa minimizar duas funções objetivos conflitantes simultaneamente: o *makespan* e o TEC. O método 2PPLS foi utilizado para a resolução do problema.

Para a primeira fase foi proposto um algoritmo de economia de energia adaptado de Ramezani, Vali-Siar e Jalalian (2019). Foi visto que na configuração com o algoritmo de economia de energia, com baixo tempo computacional, é possível encontrar soluções com menor consumo de energia. Na primeira fase também foi considerada uma configuração com busca local VND com 6 estruturas de vizinhança e outra configuração com busca local apenas com a estrutura relocate proposta na literatura por Ramezani, Vali-Siar e Jalalian (2019). Na comparação de resultados, adicionar mais estruturas de vizinhança, melhora significativamente as fronteiras obtidas tanto em NDS, como nas medidas de desempenho H e R , porém, com um tempo computacional mais elevado. Para mitigar este problema, poderia ser utilizado métodos de avaliação de movimentos e de cálculo de *makespan* mais eficientes.

Para a segunda fase foi proposto um PLS adaptado. Esta busca local ao realizar os movimentos corrige os *gaps* a partir do algoritmo de economia de energia também proposto neste trabalho. Foi visto que com baixo tempo computacional é possível encontrar novas soluções com menor *makespan* e menor TEC, porém com um menor NDS. Para melhorar esta métrica, poderia ser considerada novas estruturas de vizinhança.

Ainda nas análises da segunda fase, foi discutido sobre a correção dos *gaps* somente na segunda fase ou em ambas as fases. Foi visto que a configuração que utiliza a busca local proposta por Ramezani, Vali-Siar e Jalalian (2019) performa melhor com a correção dos *gaps* sendo realizada somente na segunda fase. Isto pode ser relacionado ao fato de que por não ter um NDS alto, o PLS fique mais limitado a encontrar vizinhos melhores, caso a correção dos *gaps* seja realizada já na primeira fase.

Também foi verificado que a configuração que utiliza a busca local VND performa melhor com a correção dos *gaps* em ambas as fases. Esta mesma configuração mostrou-se como a configuração que gerou as melhores fronteiras para o GPFSTSDS. Isto pode ser relacionado ao fato de que por ter um NDS mais alto quando comparada a busca local proposta por Ramezani, Vali-Siar e Jalalian (2019), o PLS verificará uma maior quantidade de vizinhos e tenderá a encontrar vizinhos melhores.

Como trabalhos futuros, sugere-se a avaliação de outras estruturas de vizinhança para a primeira e segunda fase, bem como um método mais ágil para cálculo do *makespan*. Além disso, pode-se ainda desenvolver heurísticas construtivas e de buscas locais híbridas para analisar a matriz de velocidade, visando avaliar a minimização de TEC em conjunto com a minimização do *makespan*.

REFERÊNCIAS

- ALLAHVERDI, A.; GUPTA, J. N.; ALDOWAISAN, T. A review of scheduling research involving setup considerations. *Omega*, v. 27, 1999. ISSN 03050483.
- ARENALES, M.; ARMENTANO, V.; MORABITO, R.; YANASSE, H. *Pesquisa Operacional para Cursos de Engenharia*. [S.l.: s.n.], 2007. ISSN 16130073.
- ARROYO, J. E. C. *Heurísticas e metaheurísticas para otimização combinatória multiobjetivo*. 232 p. Tese (Doutorado) — Universidade Estadual de Campinas, Campinas, 2002.
- BANSAL, N.; KIMBREL, T.; PRUHS, K. Speed scaling to manage energy and temperature. *Journal of the ACM*, v. 54, 2007. ISSN 00045411.
- BARDANACHVILI, C. A. *Otimização Multiobjetivo com Estratégias Evolutivas Aplicada a Projetos Estruturais*. 104 p. Tese (Doutorado) — Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2006.
- BASTOS, H. K. de O. *Otimização multiobjetivo para seleção simultânea de variáveis e objetos em cromossomo duplo de representação inteira para calibração multivariada*. 76 p. Dissertação (Mestrado) — Universidade Federal de Goiás, Goiânia, 2017.
- COELLO, G. B. L. C. A. C.; VELDHIJZEN, D. A. V. *Evolutionary Algorithms for Solving Multi-Objective Problems*. [S.l.: s.n.], 2002.
- COSTA, L.; LUST, T.; KRAMER, R.; SUBRAMANIAN, A. A two-phase Pareto local search heuristic for the bi-objective pollution-routing problem. *Networks*, v. 72, n. 3, p. 311–336, oct 2018.
- COSTA, L. C. A. da. *Uma heurística baseada em busca local de pareto para o pollution-routing problem bi-objetivo*. 157 p. Dissertação (Mestrado) — Universidade Federal da Paraíba, João Pessoa, 2015.
- DAS, S. R.; GUPTA, J. N.; KHUMAWALA, B. M. A savings index heuristic algorithm for flowshop scheduling with sequence dependent set-up times. *Journal of the Operational Research Society*, v. 46, 1995. ISSN 14769360.
- DEB, K. Multi-objective optimisation using evolutionary algorithms: An introduction. In: _____. [S.l.: s.n.], 2011.
- DEB, K.; PRATAP, A.; AGARWAL, S.; MEYARIVAN, T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, IEEE, PISCATAWAY, v. 6, n. 2, p. 182–197, 2002. ISSN 1089-778X.
- DING, J. Y.; SONG, S.; WU, C. Carbon-efficient scheduling of flow shops by multi-objective optimization. *European Journal of Operational Research*, v. 248, 2016. ISSN 03772217.
- EHRGOTT, M. *Multicriteria optimization: Second edition*. [S.l.: s.n.], 2005.
- EHRGOTT, M.; GANDIBLEUX, X. Multiple criteria optimization: State of the art annotated bibliographic surveys. *Multiple Criteria Optimization*, 2002. ISSN 1-4020-7128-0.
- EREN, T. A bicriteria m-machine flowshop scheduling with sequence-dependent setup times. *Applied Mathematical Modelling*, v. 34, 2010. ISSN 0307904X.

- FANG, K.; UHAN, N.; ZHAO, F.; SUTHERLAND, J. W. A new approach to scheduling in manufacturing for power consumption and carbon footprint reduction. In: . [S.l.: s.n.], 2011. v. 30. ISSN 02786125.
- FANG, K.; UHAN, N. A.; ZHAO, F.; SUTHERLAND, J. W. Flow shop scheduling with peak power consumption constraints. *Annals of Operations Research*, v. 206, 2013. ISSN 02545330.
- FARIAS, M. S. R. *Algoritmos evolucionários aplicados ao problema do caixeiro viajante multiobjetivo*. 107 p. Dissertação (Mestrado) — Universidade Federal de Alagoas, Maceió, 2008.
- FENG, L.; MEARS, L.; BEAUFORT, C.; SCHULTE, J. Energy, economy, and environment analysis and optimization on manufacturing plant energy supply system. *Energy Conversion and Management*, v. 117, 2016. ISSN 01968904.
- FERREIRA, J. H. I. *Um modelo de otimização multiobjetivo aplicado no dimensionamento da capacidade instalada de uma Pequena Central Hidrelétrica*. 214 p. Tese (Doutorado) — Universidade Federal de Uberlândia, Uberlândia, 2020.
- FRANÇA, P. M.; GENDREAU, M.; LAPORTE, G.; MÜLLER, F. M. A tabu search heuristic for the multiprocessor scheduling problem with sequence dependent setup times. *International Journal of Production Economics*, v. 43, 1996. ISSN 09255273.
- GAHM, C.; DENZ, F.; DIRR, M.; TUMA, A. Energy-efficient scheduling in manufacturing companies: A review and research framework. *European Journal of Operational Research*, v. 248, 2016. ISSN 03772217.
- GRAHAM, R.; LAWLER, E.; LENSTRA, J.; KAN, A. Optimization and Approximation in Deterministic Sequencing and Scheduling: a Survey. *Annals of Discrete Mathematics*, n. 5, p. 287–326, 1979.
- GUPTA, J. N.; DARROW, W. P. The two-Machine Sequence Dependent Flowshop Scheduling Problem. *European Journal of Operational Research*, v. 3, n. 24, p. 439–446, 1986.
- HAIMES, Y. Y.; LASDON, L. S.; WISMER, D. A. On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Trans. on Systems, Man and Cybernetics*, v. 1, p. 296–297, 1971.
- HEJAZI, S. R.; SAGHAFIAN, S. Flowshop-scheduling problems with makespan criterion: A review. *International Journal of Production Research*, v. 43, 2005. ISSN 00207543.
- HORN, J. Multicriterion decision making. In: _____. *Handbook of Evolutionary Computation*. [S.l.]: Oxford University, 1997.
- JASZKIEWICZ, A. Genetic local search for multi-objective combinatorial optimization. *European Journal of Operational Research*, v. 137, n. 1, p. 50–71, 2002.
- JIANG, E. da; WANG, L. An improved multi-objective evolutionary algorithm based on decomposition for energy-efficient permutation flow shop scheduling problem with sequence-dependent setup time. *International Journal of Production Research*, v. 57, 2019. ISSN 1366588X.

- JUNIOR, G. P. *Métodos de otimização multiobjetivo e de simulação aplicados ao problema de planejamento operacional de lavra em minas a céu aberto*. 89 p. Dissertação (Mestrado) — Universidade Federal de Ouro Preto, Ouro Preto, 2011.
- KONSTANTINAVICIUTE, I.; BOBINAITE, V. Comparative Analysis of Carbon Dioxide Emission Factors for Energy Industries in European Union Countries. *Renewable and Sustainable Energy Reviews*, n. 51, p. 603–612, 2015.
- LAUNDY, R. S.; STEUER, R. E. Multiple criteria optimisation: Theory, computation and application. *The Journal of the Operational Research Society*, v. 39, 1988. ISSN 01605682.
- LI, W.; ZEIN, A.; KARA, S.; HERRMANN, C. An investigation into fixed energy consumption of machine tools. In: . [S.l.: s.n.], 2011.
- LIU, Y.; DONG, H.; LOHSE, N.; PETROVIC, S. A multi-objective genetic algorithm for optimisation of energy consumption and shop floor production performance. *International Journal of Production Economics*, v. 179, 2016. ISSN 09255273.
- LU, C.; GAO, L.; LI, X.; PAN, Q.; WANG, Q. Energy-efficient permutation flow shop scheduling problem using a hybrid multi-objective backtracking search algorithm. *Journal of Cleaner Production*, v. 144, 2017. ISSN 09596526.
- LUCENA, D. V. D.; LIMA, T. W. D.; SOARES, A. D. S.; DELBEM, A. C.; FILHO, A. R. G.; COELHO, C. J.; LAUREANO, G. T. Multi-objective evolutionary algorithm for variable selection in calibration problems: A case study for protein concentration prediction. In: . [S.l.: s.n.], 2013.
- LUST, T.; TEGHEM, J. Two-phase pareto local search for the biobjective traveling salesman problem. *Journal of Heuristics*, v. 16, n. 3, p. 475–510, 2010.
- MANSOURI, S. A.; AKTAS, E. Minimizing energy consumption and makespan in a two-machine flowshop scheduling problem. *Journal of the Operational Research Society*, v. 67, n. 9, p. 1382–1394, 2016.
- MANSOURI, S. A.; AKTAS, E.; BESIKCI, U. Green scheduling of a two-machine flowshop: Trade-off between makespan and energy consumption. *European Journal of Operational Research*, v. 248, n. 3, p. 772–788, 2016.
- MENG, L.; ZHANG, C.; SHAO, X.; REN, Y.; REN, C. Mathematical modelling and optimisation of energy-conscious hybrid flow shop scheduling problem with unrelated parallel machines. *International Journal of Production Research*, v. 57, 2019. ISSN 1366588X.
- MENG, L.; ZHANG, C.; SHAO, X.; REN, Y. Milp models for energy-aware flexible job shop scheduling problem. *Journal of Cleaner Production*, v. 210, 2019. ISSN 09596526.
- MLADENović, N.; HANSEN, P. Variable neighborhood search. *Computers & Operations Research*, Pergamon, v. 24, p. 1097–1100, 11 1997. ISSN 0305-0548.
- MOKHTARI, H.; HASANI, A. An energy-efficient multi-objective optimization for flexible job-shop scheduling problem. *Computers and Chemical Engineering*, v. 104, 2017. ISSN 00981354.

- MORI, M.; FUJISHIMA, M.; INAMASU, Y.; ODA, Y. A study on energy efficiency improvement for machine tools. *CIRP Annals - Manufacturing Technology*, v. 60, 2011. ISSN 00078506.
- MOUZON, G.; YILDIRIM, M. B. A framework to minimise total energy consumption and total tardiness on a single machine. *International Journal of Sustainable Engineering*, v. 1, 2008. ISSN 19397038.
- MOUZON, G.; YILDIRIM, M. B.; TWOMEY, J. Operational methods for minimization of energy consumption of manufacturing equipment. *International Journal of Production Research*, v. 45, 2007. ISSN 00207543.
- NAWAZ, M.; ENSCORE, E. E. A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Jl of Mgmt Sci*, v. 11, p. 95, 1983.
- PAQUETE, L.; CHIARANDINI, M.; STÜTZLE, T. Pareto local optimum sets in the biobjective traveling salesman problem: An experimental study. In: _____. [S.l.: s.n.], 2004.
- PAQUETE, L.; STÜTZLE, T. A two-phase local search for the biobjective traveling salesman problem. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 2632, 2003. ISSN 16113349.
- PINEDO, M. L. *Planning and scheduling in manufacturing services*. [S.l.: s.n.], 2005. ISSN 0-387-22198-0.
- RAMEZANIAN, R.; VALI-SIAR, M. M.; JALALIAN, M. Green permutation flowshop scheduling problem with sequence-dependent setup times: a case study. *International Journal of Production Research*, v. 57, n. 10, p. 3311–3333, 2019.
- RIOS-MERCADO, R. Z.; BARD, J. F. Heuristics for the flow line problem with setup costs. *European Journal of Operational Research*, v. 110, 1998. ISSN 03772217.
- RUIZ, R.; MAROTO, C.; ALCARAZ, J. Solving the flowshop scheduling problem with sequence dependent setup times using advanced metaheuristics. *European Journal of Operational Research*, v. 165, 2005. ISSN 03772217.
- RUIZ, R.; STÜTZLE, T. An iterated greedy heuristic for the sequence dependent setup times flowshop problem with makespan and weighted tardiness objectives. *European Journal of Operational Research*, v. 187, 2008. ISSN 03772217.
- SOUZA, M. J. F. *Programação de horários em escolas: uma aproximação por metaheurísticas*. 149 p. Tese (Doutorado) — Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2000.
- TAILLARD, E. Some efficient heuristic methods for the flow shop sequencing problem. *European Journal of Operational Research*, v. 47, 1990. ISSN 03772217.
- TICONA, W. G. C. *Algoritmos evolutivos multi-objetivo para a reconstrução de árvores filogenéticas*. 134 p. Tese (Doutorado) — Universidade de São Paulo, São Carlos, 2008.
- YANG, W. H.; LIAO, C. J. Survey of scheduling research involving setup times. *International Journal of Systems Science*, v. 30, 1999. ISSN 14645319.
- ZADEH, L. A. Optimality and non-scalar-valued performance criteria. *IEEE Transactions on Automatic Control*, v. 8, 1963. ISSN 15582523.

ZHANG, H.; ZHAO, F.; FANG, K.; SUTHERLAND, J. W. Energy-conscious flow shop scheduling under time-of-use electricity tariffs. *CIRP Annals - Manufacturing Technology*, v. 63, 2014. ISSN 17260604.

ZHANG, W.; HANSEN, K. M. An evaluation of the nsga-ii and mocell genetic algorithms for self-management planning in a pervasive service middleware. In: . [S.l.: s.n.], 2009. ISSN 27708535.

ZITZLER, E. Evolutionary algorithms for multiobjective optimization: Methods and applications. *Ph.D. Thesis*, 1999.

ZITZLER, E.; THIELE, L.; LAUMANN, M.; FONSECA, C. M.; GRUNERT, V. Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Transactions on Evolutionary Computation*, v. 7, n. 2, p. 117–132, 2003.

APÊNDICE A - RESULTADOS POR INSTÂNCIA

Instância	NEHT-RB + SWAP				NEHT-RB + SWAP + ENERGIA ADAPTADO			
	Tempo (s)	H (10^6)	NDS	R	Tempo (s)	H (10^6)	NDS	R
SDST100_ta001	1,688	1,598	27	0,80355	0,985	1,651	31	0,80622
SDST125_ta001	0,51	2,022	33	0,81507	0,531	1,913	32	0,81816
SDST100_ta002	0,494	1,758	27	0,80658	0,501	1,896	28	0,80853
SDST125_ta002	0,544	1,966	31	0,8012	0,656	2,004	29	0,8036
SDST100_ta003	0,586	1,507	21	0,80116	0,609	1,523	18	0,80251
SDST125_ta003	0,508	1,49	22	0,79864	0,561	1,504	21	0,80109
SDST100_ta004	0,531	2,22	29	0,81375	0,613	2,248	27	0,81588
SDST125_ta004	0,515	2,131	34	0,79619	0,585	2,174	32	0,79885
SDST100_ta005	0,443	1,713	28	0,79998	0,518	1,762	25	0,80208
SDST125_ta005	0,512	1,898	24	0,79371	0,578	1,938	24	0,79638
SDST100_ta006	0,429	1,852	29	0,79903	0,449	1,926	28	0,80083
SDST125_ta006	0,478	2,023	26	0,80952	0,504	2,048	20	0,81234
SDST100_ta007	0,449	1,828	32	0,81441	0,476	1,863	32	0,81603
SDST125_ta007	0,454	2,238	26	0,81206	0,482	2,265	26	0,81358
SDST100_ta008	0,486	2,011	31	0,80649	0,548	2,05	28	0,80815
SDST125_ta008	0,44	2,061	29	0,79945	0,454	2,098	30	0,80097
SDST100_ta009	0,557	1,856	23	0,81083	0,608	1,941	27	0,81245
SDST125_ta009	0,553	1,843	26	0,80635	0,638	1,9	22	0,80841
SDST100_ta010	0,625	1,459	33	0,79199	0,68	1,492	30	0,79421
SDST125_ta010	0,61	1,801	31	0,80818	0,652	1,823	27	0,81087
SDST100_ta011	1,84	4,511	48	0,81527	1,952	4,512	42	0,81802
SDST125_ta011	1,968	4,552	33	0,80475	2,086	5,004	39	0,80778
SDST100_ta012	1,749	5,279	33	0,82191	1,835	5,222	34	0,82492
SDST125_ta012	1,696	5,478	39	0,81159	1,807	5,593	38	0,81527
SDST100_ta013	1,448	3,706	37	0,80351	1,531	3,821	33	0,80705
SDST125_ta013	1,456	4,619	29	0,81551	1,523	4,764	28	0,81945
SDST100_ta014	1,494	3,738	38	0,81742	1,606	3,85	36	0,82039
SDST125_ta014	1,323	4,139	35	0,81459	1,422	4,26	31	0,81832
SDST100_ta015	1,484	4,085	30	0,82037	1,578	4,177	31	0,82298
SDST125_ta015	1,712	4,648	33	0,82051	1,964	4,773	33	0,82374
SDST100_ta016	1,734	3,907	34	0,81289	1,843	4,025	30	0,81629
SDST125_ta016	1,711	3,991	28	0,81417	1,843	4,509	32	0,81759
SDST100_ta017	1,628	4,131	43	0,81555	1,674	4,299	38	0,8182
SDST125_ta017	1,68	4,161	33	0,81108	1,784	4,265	27	0,81424
SDST100_ta018	1,798	4,757	55	0,82677	1,896	4,885	48	0,82996
SDST125_ta018	1,626	4,037	37	0,79731	1,789	4,173	33	0,80064
SDST100_ta019	1,622	4,58	36	0,81188	1,752	4,641	32	0,81458
SDST125_ta019	1,48	5,038	41	0,81911	1,605	5,15	39	0,82198
SDST100_ta020	1,296	4,697	37	0,81113	1,392	4,703	31	0,81416
SDST125_ta020	1,322	4,971	44	0,81084	1,409	5,177	41	0,81474
SDST100_ta021	4,588	12,783	68	0,81903	4,909	12,995	63	0,82185
SDST125_ta021	5,909	13,221	50	0,82021	6,152	13,474	47	0,82342
SDST100_ta022	4,774	10,869	50	0,81573	5,031	10,852	50	0,81847
SDST125_ta022	5,301	11,494	57	0,81213	5,626	11,786	51	0,81545
SDST100_ta023	5,882	12,934	53	0,82206	6,441	12,995	56	0,82449
SDST125_ta023	4,468	12,46	45	0,81427	4,792	12,856	48	0,81742
SDST100_ta024	4,601	11,11	45	0,81273	4,815	11,608	48	0,816
SDST125_ta024	4,741	10,148	51	0,80741	5,006	10,604	44	0,81148
SDST100_ta025	4,993	12,839	60	0,81886	5,304	13,52	58	0,82214
SDST125_ta025	6,233	13,044	42	0,81642	6,52	12,615	38	0,82034
SDST100_ta026	7,302	12,401	77	0,82377	6,157	12,78	67	0,82755
SDST125_ta026	4,636	12,409	49	0,81111	5,007	12,75	49	0,81456

Instância	NEHT-RB + SWAP				NEHT-RB + SWAP + ENERGIA ADAPTADO			
	Tempo (s)	H (10^6)	NDS	R	Tempo (s)	H (10^6)	NDS	R
SDST100_ta027	5,093	13,053	52	0,8285	5,366	13,372	56	0,83198
SDST125_ta027	4,629	14,091	50	0,81955	4,881	14,396	45	0,82314
SDST100_ta028	4,401	11,968	62	0,81794	4,547	12,376	52	0,82091
SDST125_ta028	4,728	11,892	56	0,81178	5,155	12,248	44	0,81578
SDST100_ta029	4,599	12,617	66	0,82183	4,887	12,633	68	0,82473
SDST125_ta029	5,379	13,523	48	0,8175	5,7	13,442	49	0,82105
SDST100_ta030	4,436	11,415	52	0,81765	4,826	11,683	55	0,82032
SDST125_ta030	4,679	9,47	41	0,80262	4,954	9,465	39	0,80598
SDST100_ta031	21,893	8,966	29	0,80478	21,54	9,046	31	0,80564
SDST125_ta031	17,48	8,974	39	0,79965	17,836	9,018	36	0,80042
SDST100_ta032	18,264	10,67	41	0,80467	18,605	10,907	38	0,80543
SDST125_ta032	17,234	11,786	40	0,80716	17,473	12,096	36	0,80823
SDST100_ta033	17,339	9,661	46	0,80515	17,55	9,73	41	0,80608
SDST125_ta033	19,159	9,755	30	0,80369	19,434	8,508	31	0,80449
SDST100_ta034	20,043	9,173	43	0,80113	20,368	9,349	42	0,80183
SDST125_ta034	21,664	10,08	44	0,79527	21,973	10,436	44	0,79614
SDST100_ta035	19,27	10,043	44	0,80414	19,142	10,165	41	0,80502
SDST125_ta035	18,682	11,208	37	0,80786	18,989	10,755	35	0,80873
SDST100_ta036	17,167	11,033	49	0,81181	17,271	10,829	49	0,81249
SDST125_ta036	19,429	11,92	33	0,81196	21,826	11,843	31	0,81286
SDST100_ta037	19,547	9,679	39	0,80869	19,913	9,68	44	0,80933
SDST125_ta037	18,787	10,212	31	0,80382	19,237	10,29	35	0,80446
SDST100_ta038	17,246	9,625	46	0,80233	18,167	9,727	45	0,80313
SDST125_ta038	20,81	10,276	42	0,79505	18,429	10,461	36	0,79592
SDST100_ta039	19,645	8,712	36	0,80526	19,814	8,778	41	0,80591
SDST125_ta039	18,922	8,486	33	0,7903	19,621	8,571	26	0,79144
SDST100_ta040	18,532	10,301	39	0,80889	18,763	10,352	37	0,80977
SDST125_ta040	18,507	10,293	38	0,80144	19,293	10,429	36	0,80291
SDST100_ta041	54,905	23,599	53	0,81296	55,044	23,181	46	0,81478
SDST125_ta041	62,224	23,745	47	0,80655	53,307	24,161	46	0,80856
SDST100_ta042	56,034	20,293	51	0,80501	55,44	20,684	42	0,80662
SDST125_ta042	55,833	22,392	34	0,80631	53,886	23,349	36	0,80826
SDST100_ta043	57,931	18,52	42	0,80003	54,152	19,212	43	0,8016
SDST125_ta043	55,692	23,361	49	0,80642	57,028	23,922	47	0,80864
SDST100_ta044	50,123	23,456	59	0,80853	51,759	23,854	57	0,81023
SDST125_ta044	50,239	22,96	54	0,80648	59,492	23,39	47	0,80895
SDST100_ta045	57,775	23,604	41	0,81096	58,195	24,091	50	0,81275
SDST125_ta045	59,609	21,355	44	0,80482	69,783	22,515	40	0,80705
SDST100_ta046	55,377	22,408	49	0,8081	65,61	22,777	46	0,80997
SDST125_ta046	60,191	24,158	43	0,80736	55,238	24,239	43	0,80932
SDST100_ta047	55,5	22,209	45	0,8124	52,455	23,485	48	0,81397
SDST125_ta047	54,107	24,201	52	0,8033	60,616	24,643	52	0,80526
SDST100_ta048	61,157	22,797	51	0,81582	54,585	23,964	51	0,81747
SDST125_ta048	62,215	23,876	51	0,81073	66,547	24,165	50	0,81274
SDST100_ta049	62,39	20,588	58	0,80971	56,463	21,229	55	0,81106
SDST125_ta049	49,157	22,229	41	0,80971	56,516	22,769	40	0,8116
SDST100_ta050	63,796	22,77	61	0,80549	63,283	22,381	57	0,80749
SDST125_ta050	63,71	23,888	58	0,80819	65,233	24,399	59	0,81065
SDST100_ta051	197,746	56,165	66	0,81312	192,479	59,04	71	0,81561
SDST125_ta051	219,321	56,556	56	0,80989	204,919	57,985	57	0,81266

Instância	NEHT-RB + SWAP + 2ª Fase				NEHT-RB + SWAP + ENERGIA + 2ª Fase			
	Tempo (s)	H (10 ⁶)	NDS	R	Tempo (s)	H (10 ⁶)	NDS	R
SDST100_ta001	0,766	1,69	23	0,80877	0,711	1,685	23	0,80812
SDST125_ta001	0,805	1,983	22	0,82198	0,743	2	23	0,82216
SDST100_ta002	0,729	1,915	19	0,81049	0,774	1,959	22	0,81062
SDST125_ta002	1,103	2,05	23	0,80634	0,692	2,049	24	0,8061
SDST100_ta003	0,733	1,534	18	0,80421	0,625	1,532	18	0,80347
SDST125_ta003	0,699	1,583	16	0,8047	0,62	1,596	14	0,80468
SDST100_ta004	0,899	2,28	26	0,82212	0,643	2,336	22	0,82088
SDST125_ta004	0,826	2,147	23	0,80472	0,688	2,13	22	0,80137
SDST100_ta005	0,696	1,753	26	0,80279	0,575	1,772	24	0,80294
SDST125_ta005	0,682	1,949	19	0,80269	0,634	1,983	18	0,79804
SDST100_ta006	0,677	2,017	25	0,80471	0,557	1,957	24	0,80285
SDST125_ta006	0,721	2,083	22	0,81436	0,582	2,081	18	0,81382
SDST100_ta007	0,744	1,923	24	0,82031	0,626	1,926	21	0,8192
SDST125_ta007	0,745	2,322	23	0,81972	0,625	2,192	23	0,81846
SDST100_ta008	0,753	2,107	17	0,81277	0,665	1,904	17	0,81148
SDST125_ta008	0,721	2,063	19	0,80444	0,587	2,151	19	0,80487
SDST100_ta009	0,745	1,951	19	0,81302	0,684	1,953	22	0,81302
SDST125_ta009	0,814	1,901	19	0,81376	0,724	1,933	19	0,8102
SDST100_ta010	0,909	1,532	21	0,80231	0,811	1,548	24	0,80177
SDST125_ta010	0,944	1,887	16	0,81407	0,811	1,865	20	0,81356
SDST100_ta011	2,388	4,545	34	0,81966	2,179	4,565	34	0,81933
SDST125_ta011	2,402	4,415	25	0,81023	2,353	4,843	22	0,80992
SDST100_ta012	2,095	5,716	27	0,8294	2,046	5,508	30	0,82685
SDST125_ta012	2,059	5,513	32	0,81773	1,965	5,663	30	0,8164
SDST100_ta013	1,726	3,944	25	0,80924	1,661	3,866	28	0,80875
SDST125_ta013	1,957	4,804	26	0,8206	1,671	4,785	27	0,82018
SDST100_ta014	1,844	3,881	31	0,82224	1,82	3,847	28	0,82215
SDST125_ta014	1,635	4,357	24	0,82125	1,572	4,336	25	0,82058
SDST100_ta015	1,818	4,258	26	0,82397	1,748	4,218	25	0,82378
SDST125_ta015	2,279	4,866	24	0,82573	2,185	4,855	25	0,82536
SDST100_ta016	2,176	4,131	28	0,81948	2,095	4,125	27	0,81936
SDST125_ta016	2,062	4,581	19	0,82334	2,058	3,927	19	0,81997
SDST100_ta017	2,006	4,34	32	0,82088	1,867	4,33	33	0,81962
SDST125_ta017	1,991	3,879	22	0,81593	1,906	3,867	23	0,81594
SDST100_ta018	2,265	4,97	37	0,83118	2,11	4,939	37	0,83116
SDST125_ta018	2,216	4,303	30	0,8063	1,952	4,09	26	0,80321
SDST100_ta019	1,994	4,692	26	0,81598	1,938	4,688	27	0,81561
SDST125_ta019	1,883	4,818	28	0,82492	1,785	4,795	27	0,82448
SDST100_ta020	1,706	4,807	29	0,81662	1,543	4,742	27	0,81521
SDST125_ta020	1,738	5,055	25	0,81893	1,598	5,036	25	0,81793
SDST100_ta021	5,749	13,151	42	0,82324	5,386	13,116	45	0,82296
SDST125_ta021	6,684	13,749	36	0,82464	6,606	13,685	33	0,82475
SDST100_ta022	5,433	11,117	36	0,82038	5,317	10,942	40	0,81971
SDST125_ta022	6,345	11,96	41	0,81705	6,075	11,955	37	0,8168
SDST100_ta023	6,446	13,192	40	0,82534	5,498	13,12	45	0,82533
SDST125_ta023	5,179	12,878	38	0,81863	5,152	12,833	38	0,81837
SDST100_ta024	5,243	11,849	36	0,82159	5,06	11,932	36	0,81693
SDST125_ta024	5,522	10,591	37	0,81296	5,313	10,724	35	0,81241
SDST100_ta025	5,908	12,811	35	0,82384	5,65	13,384	42	0,82353
SDST125_ta025	7,029	13,616	28	0,82215	6,868	12,801	29	0,82179
SDST100_ta026	7,671	12,971	57	0,8291	6,44	12,916	52	0,82868
SDST125_ta026	5,36	12,986	34	0,81678	5,276	12,927	41	0,81615

Instância	NEHT-RB + SWAP + 2ª Fase				NEHT-RB + SWAP + ENERGIA + 2ª Fase			
	Tempo (s)	H (10 ⁶)	NDS	R	Tempo (s)	H (10 ⁶)	NDS	R
SDST100_ta027	5,776	13,08	35	0,83382	5,739	13,522	43	0,83401
SDST125_ta027	5,43	14,563	36	0,82496	5,104	14,541	36	0,8239
SDST100_ta028	5,38	12,562	46	0,82325	5,063	12,497	42	0,8228
SDST125_ta028	5,578	12,278	42	0,8183	5,238	12,482	36	0,81784
SDST100_ta029	5,577	13,059	49	0,82585	5,301	12,036	51	0,82592
SDST125_ta029	6,2	13,601	33	0,82196	7,301	13,56	36	0,82193
SDST100_ta030	5,423	11,446	41	0,82372	5,121	11,808	42	0,82106
SDST125_ta030	5,265	9,593	30	0,8071	5,174	9,565	32	0,807
SDST100_ta031	21,495	9,193	26	0,80736	20,644	9,201	24	0,80736
SDST125_ta031	22,905	9,413	27	0,80535	20,812	9,426	19	0,80418
SDST100_ta032	24,212	11,385	27	0,80969	22,007	11,344	26	0,80919
SDST125_ta032	20,89	12,386	24	0,8124	19,691	11,873	25	0,81183
SDST100_ta033	25,231	10,391	30	0,81276	23,135	10,263	23	0,8113
SDST125_ta033	23,363	9,129	16	0,81054	22,339	9,033	16	0,81016
SDST100_ta034	24,357	9,576	29	0,80425	23,192	9,306	28	0,80392
SDST125_ta034	23,643	11,134	22	0,80274	22,272	11,083	24	0,80075
SDST100_ta035	25,079	10,548	26	0,81467	23,305	10,587	22	0,81472
SDST125_ta035	22,552	11,518	21	0,81429	21,614	11,095	23	0,81164
SDST100_ta036	22,356	11,089	32	0,81701	21,581	11,134	28	0,81639
SDST125_ta036	27,974	12,315	26	0,81716	24,848	12,162	24	0,81625
SDST100_ta037	25,377	10,25	29	0,81624	23,747	9,919	31	0,81218
SDST125_ta037	22,218	10,64	23	0,80783	21,656	10,6	23	0,80742
SDST100_ta038	27,019	10,334	33	0,80964	24,977	9,337	29	0,8063
SDST125_ta038	23,522	10,761	24	0,80308	21,946	10,456	23	0,80288
SDST100_ta039	26,808	8,871	27	0,816	23,447	8,778	25	0,81246
SDST125_ta039	23,394	8,988	26	0,7984	21,839	9,12	19	0,80154
SDST100_ta040	24,005	10,576	27	0,81627	22,156	10,738	26	0,815
SDST125_ta040	24,054	11,013	18	0,80738	23,412	11,319	21	0,80799
SDST100_ta041	61,775	24,463	38	0,81686	59,641	23,449	37	0,81663
SDST125_ta041	59,906	24,906	30	0,8118	57,75	24,83	30	0,81104
SDST100_ta042	62,918	21,242	30	0,81234	60,351	21,262	25	0,81045
SDST125_ta042	57,837	23,033	27	0,81141	58,218	23,489	30	0,81004
SDST100_ta043	60,994	20,287	33	0,80727	64,741	20,11	27	0,80666
SDST125_ta043	62,72	23,689	32	0,81273	61,004	24,27	32	0,81264
SDST100_ta044	66,52	24,432	30	0,81485	57,513	24,276	38	0,81431
SDST125_ta044	58,595	24,169	33	0,81333	54,63	23,47	29	0,81121
SDST100_ta045	63,72	23,95	38	0,81418	62,291	24,425	39	0,81433
SDST125_ta045	68,088	24,362	32	0,81193	65,783	23,122	27	0,80938
SDST100_ta046	65,735	22,85	37	0,81408	60,83	23,252	39	0,81152
SDST125_ta046	61,173	23,322	24	0,81241	59,104	25,229	29	0,8123
SDST100_ta047	59,43	22,756	42	0,81617	57,423	23,732	38	0,8161
SDST125_ta047	69,087	24,96	34	0,80894	63,589	24,28	34	0,80759
SDST100_ta048	61,431	24,488	45	0,8212	66,362	23,955	43	0,81942
SDST125_ta048	63,333	24,869	32	0,81627	68,624	23,945	32	0,81548
SDST100_ta049	77,527	21,246	43	0,8146	72,659	21,011	44	0,8127
SDST125_ta049	67,936	23,405	31	0,81526	57,49	23,037	29	0,81503
SDST100_ta050	65,277	23,999	41	0,81157	61,14	20,874	33	0,81015
SDST125_ta050	68,196	23,974	40	0,81369	59,699	25,097	43	0,81317
SDST100_ta051	214,905	57,802	51	0,81729	215,514	57,603	45	0,81706
SDST125_ta051	232,255	60,754	47	0,81601	201,346	58,629	49	0,81369

Instância	NEHT-RB + VND + ENERGIA ADAPTADO				NEHT-RB + VND + 2ª Fase			
	Tempo (s)	H (10 ⁶)	NDS	R	Tempo (s)	H (10 ⁶)	NDS	R
SDST100_ta001	2,923	1,749	35	0,82339	3,149	1,751	30	0,82323
SDST125_ta001	3,529	2,222	31	0,83516	3,397	2,23	25	0,83496
SDST100_ta002	3,583	2,134	33	0,83148	4,074	2,052	27	0,83138
SDST125_ta002	3,914	2,179	34	0,83165	3,346	2,192	30	0,83146
SDST100_ta003	3,057	1,595	22	0,83293	3,168	1,595	22	0,83293
SDST125_ta003	3,011	1,818	25	0,83144	3,141	1,804	22	0,83142
SDST100_ta004	3,066	2,521	32	0,83664	3,242	2,511	27	0,83661
SDST125_ta004	3,15	2,513	29	0,83119	3,155	2,511	27	0,83109
SDST100_ta005	2,896	1,951	29	0,83191	2,952	1,953	24	0,83155
SDST125_ta005	3,104	2,176	27	0,83341	3,063	2,18	23	0,8334
SDST100_ta006	3,112	2,307	32	0,83989	3,246	2,307	30	0,83977
SDST125_ta006	3,112	2,316	26	0,84184	3,124	2,316	25	0,84183
SDST100_ta007	2,924	2,051	27	0,83552	3,037	1,965	22	0,83539
SDST125_ta007	3,23	2,561	26	0,84573	3,37	2,562	24	0,84564
SDST100_ta008	2,873	2,278	40	0,83559	3,029	2,194	35	0,83548
SDST125_ta008	3,101	2,415	29	0,83699	3,313	2,415	27	0,83699
SDST100_ta009	3,668	2,121	40	0,83336	3,887	2,121	37	0,8333
SDST125_ta009	3,505	2,122	26	0,83429	3,605	2,123	24	0,83427
SDST100_ta010	4,118	1,719	35	0,8295	4,275	1,722	30	0,82943
SDST125_ta010	3,813	2,114	23	0,83861	3,91	2,094	21	0,83851
SDST100_ta011	8,387	4,981	46	0,83087	8,558	4,98	41	0,83079
SDST125_ta011	10,126	4,939	38	0,82817	9,459	4,962	30	0,82811
SDST100_ta012	8,788	5,877	44	0,8339	9,039	5,889	39	0,83381
SDST125_ta012	8,967	6,036	36	0,83249	7,81	6,077	33	0,83245
SDST100_ta013	7,708	3,971	33	0,82696	7,74	3,982	28	0,82682
SDST125_ta013	7,157	4,991	37	0,83406	7,262	4,947	28	0,83393
SDST100_ta014	7,654	4,025	33	0,8348	7,683	4,027	32	0,83479
SDST125_ta014	6,609	4,481	28	0,8346	6,645	3,926	22	0,83456
SDST100_ta015	7,27	4,434	37	0,83332	7,414	4,444	33	0,83323
SDST125_ta015	9,171	5,111	34	0,83726	9,301	4,945	28	0,83711
SDST100_ta016	9,288	4,386	40	0,83689	9,617	4,389	36	0,83689
SDST125_ta016	9,249	4,907	33	0,83946	9,335	4,9	33	0,83945
SDST100_ta017	8,24	4,615	39	0,8364	8,531	4,629	34	0,83634
SDST125_ta017	8,797	4,447	38	0,83181	9,311	4,315	34	0,83172
SDST100_ta018	8,501	5,179	41	0,83895	8,645	5,19	30	0,83881
SDST125_ta018	8,949	4,398	31	0,82757	9,117	4,432	26	0,82737
SDST100_ta019	8,596	5,053	41	0,83327	8,738	5,061	36	0,83323
SDST125_ta019	7,134	5,521	33	0,8366	7,223	5,379	28	0,83663
SDST100_ta020	6,723	5,135	44	0,82977	7,047	5,095	41	0,82968
SDST125_ta020	6,937	5,551	41	0,83	7,137	5,551	39	0,82998
SDST100_ta021	19,805	13,986	54	0,8354	20,091	13,986	50	0,83534
SDST125_ta021	24,023	14,116	48	0,83441	21,765	13,876	37	0,83431
SDST100_ta022	20,806	11,64	53	0,831	20,617	11,509	43	0,83096
SDST125_ta022	25,466	12,602	50	0,83175	23,229	12,63	40	0,8317
SDST100_ta023	20,7	13,905	62	0,83522	21,154	13,484	52	0,83519
SDST125_ta023	25,574	13,605	57	0,8312	26,029	13,634	51	0,8311
SDST100_ta024	20,351	12,766	55	0,82968	20,594	12,388	47	0,82961
SDST125_ta024	20,203	11,334	47	0,82152	20,057	10,646	38	0,82137
SDST100_ta025	21,954	14,14	71	0,83373	23,011	14,143	63	0,83371
SDST125_ta025	24,935	14,265	39	0,83136	21,953	14,273	34	0,83132
SDST100_ta026	23,783	13,291	62	0,83429	20,482	13,312	53	0,83424
SDST125_ta026	22,694	13,3	53	0,8316	22,554	13,333	41	0,83142

Instância	NEHT-RB + VND + ENERGIA ADAPTADO				NEHT-RB + VND + 2ª Fase			
	Tempo (s)	H (10 ⁶)	NDS	R	Tempo (s)	H (10 ⁶)	NDS	R
SDST100_ta027	21,543	13,824	52	0,83727	21,871	12,761	45	0,83727
SDST125_ta027	19,528	15,307	54	0,83797	19,773	14,864	42	0,83779
SDST100_ta028	20,618	12,942	52	0,83359	20,941	12,98	45	0,8336
SDST125_ta028	20,845	12,623	49	0,82726	21,016	12,654	40	0,82713
SDST100_ta029	20,155	13,652	51	0,83608	20,223	13,661	46	0,83602
SDST125_ta029	25,915	14,598	66	0,83501	25,89	14,617	56	0,83495
SDST100_ta030	19,599	12,306	60	0,83225	19,956	12,331	47	0,83213
SDST125_ta030	20,842	10,394	53	0,81925	20,97	10,406	47	0,81917
SDST100_ta031	186,141	9,854	42	0,82914	189,308	9,869	37	0,82909
SDST125_ta031	210,626	10,361	30	0,83186	199,8	10,361	30	0,83186
SDST100_ta032	227,052	11,432	34	0,83772	225,118	11,473	32	0,83776
SDST125_ta032	216,015	13,939	38	0,83799	218,255	13,95	35	0,83797
SDST100_ta033	217,243	11,028	38	0,83766	216,925	11,044	35	0,83761
SDST125_ta033	215,683	11,378	33	0,83754	214,098	11,38	32	0,83751
SDST100_ta034	204,029	10,831	42	0,82971	223,134	10,839	39	0,82971
SDST125_ta034	220,916	12,768	32	0,83551	239,231	12,768	32	0,83551
SDST100_ta035	215,051	11,51	45	0,83439	212,805	11,524	37	0,83437
SDST125_ta035	205,907	12,815	37	0,8391	217,614	12,826	35	0,83907
SDST100_ta036	199,367	12,419	53	0,83712	201,647	12,428	47	0,83711
SDST125_ta036	201,225	13,556	41	0,8393	211,293	13,404	39	0,83928
SDST100_ta037	219,214	10,96	50	0,83601	218,624	10,97	46	0,83599
SDST125_ta037	211,141	11,229	29	0,83328	205,824	11,229	29	0,83328
SDST100_ta038	214,957	11,441	40	0,83427	242,62	11,454	35	0,83426
SDST125_ta038	222,631	12,524	34	0,83591	213,684	12,244	31	0,83586
SDST100_ta039	213,664	9,923	41	0,83396	218,347	9,927	39	0,83396
SDST125_ta039	220,668	10,072	41	0,83179	214,074	10,207	37	0,83169
SDST100_ta040	220,961	11,913	44	0,83552	214,315	11,41	39	0,83549
SDST125_ta040	213,432	12,538	38	0,83514	198,677	12,563	37	0,83514
SDST100_ta041	449,666	26,05	54	0,83338	448,367	24,752	47	0,83337
SDST125_ta041	456,399	26,873	44	0,83202	463,606	26,888	39	0,83194
SDST100_ta042	443,23	22,833	53	0,82877	447,662	21,726	47	0,82875
SDST125_ta042	479,004	25,422	36	0,83384	473,797	25,427	35	0,83383
SDST100_ta043	456,559	21,758	50	0,8269	497,525	21,105	43	0,82685
SDST125_ta043	441,199	26,365	45	0,83425	443,704	25,701	35	0,83422
SDST100_ta044	464,311	26,45	44	0,83373	460,313	25,513	41	0,83373
SDST125_ta044	446,096	26,033	44	0,82865	449,517	24,732	38	0,82859
SDST100_ta045	465,178	26,164	47	0,83501	467,952	26,261	42	0,83495
SDST125_ta045	491,911	27,56	42	0,83208	501,133	27,6	39	0,83207
SDST100_ta046	510,583	24,75	59	0,82933	520,697	24,772	55	0,82932
SDST125_ta046	481,702	27,181	41	0,8327	480,092	27,111	34	0,83265
SDST100_ta047	485,975	25,822	54	0,8323	501,175	25,617	49	0,83227
SDST125_ta047	462,924	27,426	39	0,83072	502,546	27,453	38	0,83071
SDST100_ta048	454,723	25,883	54	0,8341	462,438	25,309	47	0,83405
SDST125_ta048	470,182	26,529	56	0,83088	480,302	25,914	39	0,82392
SDST100_ta049	464,327	23,205	50	0,83033	488,927	23,266	46	0,8303
SDST125_ta049	481,496	26,43	36	0,83354	487,667	26,582	32	0,83353
SDST100_ta050	489,717	25,249	50	0,82994	515,43	25,281	45	0,82994
SDST125_ta050	517,57	27,239	37	0,83007	515,137	27,302	32	0,83004
SDST100_ta051	1497,36	62,654	65	0,83053	1417,76	62,758	57	0,83053
SDST125_ta051	1457,92	64,357	58	0,82891	1434,91	64,441	51	0,8289

NEHT-RB + VND + ENERGIA + 2ª FASE				
Instância	Tempo (s)	H (10^6)	NDS	R
SDST100_ta001	3,003	1,749	35	0,82339
SDST125_ta001	3,614	2,223	31	0,8352
SDST100_ta002	3,849	2,135	32	0,83148
SDST125_ta002	3,67	2,179	34	0,83165
SDST100_ta003	3,134	1,595	22	0,83293
SDST125_ta003	3,12	1,818	24	0,83144
SDST100_ta004	3,142	2,521	32	0,83664
SDST125_ta004	3,091	2,513	29	0,83119
SDST100_ta005	2,931	1,951	29	0,83191
SDST125_ta005	3,064	2,176	26	0,83341
SDST100_ta006	3,238	2,241	31	0,83989
SDST125_ta006	3,231	2,316	26	0,84184
SDST100_ta007	2,939	2,051	27	0,83552
SDST125_ta007	3,29	2,561	26	0,84573
SDST100_ta008	2,928	2,212	38	0,83559
SDST125_ta008	3,77	2,415	29	0,83699
SDST100_ta009	3,675	2,121	40	0,83336
SDST125_ta009	3,59	2,122	26	0,83429
SDST100_ta010	4,191	1,719	35	0,8295
SDST125_ta010	3,801	2,114	23	0,83861
SDST100_ta011	8,449	4,976	45	0,83087
SDST125_ta011	9,53	4,943	36	0,82817
SDST100_ta012	9,044	5,886	44	0,8339
SDST125_ta012	7,695	6,036	36	0,83249
SDST100_ta013	7,733	3,971	33	0,82696
SDST125_ta013	7,336	5	34	0,83406
SDST100_ta014	7,683	4,025	33	0,8348
SDST125_ta014	6,776	4,481	28	0,8346
SDST100_ta015	7,395	4,434	35	0,83332
SDST125_ta015	9,315	4,919	32	0,83726
SDST100_ta016	9,461	4,385	39	0,83689
SDST125_ta016	9,25	4,908	33	0,83948
SDST100_ta017	8,38	4,616	38	0,8364
SDST125_ta017	8,943	4,448	38	0,83184
SDST100_ta018	8,726	5,179	40	0,83895
SDST125_ta018	9,172	4,398	31	0,82757
SDST100_ta019	7,938	5,053	41	0,83327
SDST125_ta019	7,2	5,521	33	0,8366
SDST100_ta020	6,88	5,084	41	0,82977
SDST125_ta020	7,176	5,551	40	0,83
SDST100_ta021	22,443	13,986	54	0,83541
SDST125_ta021	20,21	14,116	48	0,83441
SDST100_ta022	20,763	11,626	52	0,831
SDST125_ta022	22,397	12,602	50	0,83175
SDST100_ta023	23,646	13,701	61	0,83522
SDST125_ta023	24,039	13	54	0,8312
SDST100_ta024	20,463	12,393	51	0,82968
SDST125_ta024	20,194	11,275	45	0,82152
SDST100_ta025	28,07	14,14	70	0,83373
SDST125_ta025	22,444	14,265	39	0,83136
SDST100_ta026	19,702	13,291	61	0,83429
SDST125_ta026	22,634	13,301	50	0,8316

NEHT-RB + VND + ENERGIA + 2ª FASE				
Instância	Tempo (s)	H (10 ⁶)	NDS	R
SDST100_ta027	21,552	13,831	51	0,83727
SDST125_ta027	19,961	14,835	52	0,83797
SDST100_ta028	20,68	12,951	51	0,83361
SDST125_ta028	21,172	12,638	46	0,82726
SDST100_ta029	20,507	13,652	51	0,83608
SDST125_ta029	24,938	14,598	65	0,83501
SDST100_ta030	19,887	12,306	60	0,83225
SDST125_ta030	22,741	10,394	53	0,81925
SDST100_ta031	188,418	9,854	42	0,82914
SDST125_ta031	201,444	10,361	30	0,83186
SDST100_ta032	222,113	11,444	34	0,83776
SDST125_ta032	236,788	13,939	38	0,83799
SDST100_ta033	202,572	11,028	38	0,83766
SDST125_ta033	208,706	11,378	33	0,83754
SDST100_ta034	213,139	10,831	42	0,82971
SDST125_ta034	229,261	12,768	32	0,83551
SDST100_ta035	202,63	11,515	42	0,83439
SDST125_ta035	217,991	12,815	37	0,8391
SDST100_ta036	199,953	12,419	53	0,83712
SDST125_ta036	199,652	13,556	41	0,8393
SDST100_ta037	215,258	10,96	50	0,83601
SDST125_ta037	203,983	11,229	29	0,83328
SDST100_ta038	219,108	11,441	40	0,83427
SDST125_ta038	222,015	12,234	33	0,83591
SDST100_ta039	226,371	9,923	41	0,83396
SDST125_ta039	213,112	10,072	41	0,83179
SDST100_ta040	209,566	11,913	44	0,83552
SDST125_ta040	209,218	12,538	38	0,83514
SDST100_ta041	451,35	26,073	51	0,83338
SDST125_ta041	491,925	26,873	44	0,83202
SDST100_ta042	457,185	22,833	53	0,82878
SDST125_ta042	483,347	25,422	36	0,83384
SDST100_ta043	483,144	21,759	50	0,82691
SDST125_ta043	445,564	26,369	44	0,83426
SDST100_ta044	487,071	26,45	44	0,83373
SDST125_ta044	445,956	26,034	44	0,82865
SDST100_ta045	467,886	26,164	47	0,83501
SDST125_ta045	484,634	27,56	41	0,83208
SDST100_ta046	501,216	24,75	59	0,82933
SDST125_ta046	493,402	27,181	41	0,8327
SDST100_ta047	476,893	25,61	53	0,8323
SDST125_ta047	478,135	27,426	39	0,83072
SDST100_ta048	466,664	25,883	54	0,83411
SDST125_ta048	511,719	26,529	56	0,83088
SDST100_ta049	466,173	23,205	50	0,83033
SDST125_ta049	492,645	26,43	36	0,83354
SDST100_ta050	506,764	25,249	50	0,82994
SDST125_ta050	468,159	27,197	36	0,83007
SDST100_ta051	1425,06	62,654	65	0,83053
SDST125_ta051	1436,04	64,358	58	0,82892