

---

**Indicadores de Convergência e Diversidade em  
Algoritmos Evolutivos para Otimização  
Multiobjetivo**

---

**Luiz Carlos Felix Carvalho**



UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
FACULDADE DE COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Uberlândia  
2021



**Luiz Carlos Felix Carvalho**

**Indicadores de Convergência e Diversidade em  
Algoritmos Evolutivos para Otimização  
Multiobjetivo**

Tese de doutorado apresentada ao Programa de Pós-graduação da Faculdade de Computação da Universidade Federal de Uberlândia como parte dos requisitos para a obtenção do título de Doutor em Ciência da Computação.

Área de concentração: Ciência da Computação

Orientador: Márcia Aparecida Fernandes

Uberlândia

2021

Ficha Catalográfica Online do Sistema de Bibliotecas da UFU  
com dados informados pelo(a) próprio(a) autor(a).

|              |   |
|--------------|---|
| C331<br>2021 | <p>Carvalho, Luiz Carlos Felix, 1983-<br/>Indicadores de Convergência e Diversidade em<br/>Algoritmos Evolutivos para Otimização Multiobjetivo<br/>[recurso eletrônico] / Luiz Carlos Felix Carvalho. -<br/>2021.</p> <p>Orientadora: Márcia Aparecida Fernandes.<br/>Tese (Doutorado) - Universidade Federal de Uberlândia,<br/>Pós-graduação em Ciência da Computação.<br/>Modo de acesso: Internet.<br/>Disponível em: <a href="http://doi.org/10.14393/ufu.te.2021.511">http://doi.org/10.14393/ufu.te.2021.511</a><br/>Inclui bibliografia.<br/>Inclui ilustrações.</p> <p>1. Computação. I. Fernandes, Márcia Aparecida, 1963-,<br/>(Orient.). II. Universidade Federal de Uberlândia. Pós-<br/>graduação em Ciência da Computação. III. Título.</p> <p style="text-align: right;">CDU: 681.3</p> |
|--------------|---|

Bibliotecários responsáveis pela estrutura de acordo com o AACR2:

Gizele Cristine Nunes do Couto - CRB6/2091



**UNIVERSIDADE FEDERAL DE UBERLÂNDIA**  
 Coordenação do Programa de Pós-Graduação em Ciência da Computação  
 Av. João Naves de Ávila, nº 2121, Bloco 1A, Sala 243 - Bairro Santa Mônica, Uberlândia-MG, CEP 38400-902  
 Telefone: (34) 3239-4470 - www.ppgco.facom.ufu.br - cpqfacom@ufu.br



### ATA DE DEFESA - PÓS-GRADUAÇÃO

|                                    |  |                 |     |                       |          |
|------------------------------------|--|-----------------|-----|-----------------------|----------|
| Programa de Pós-Graduação em:      | Ciência da Computação  |                 |     |                       |          |
| Defesa de:                         | Tese de doutorado, 23/2021, PPGCO  |                 |     |                       |          |
| Data:                              | 02 de setembro de 2021   | Hora de início: | 09h | Hora de encerramento: | 13h45min |
| Matrícula do Discente:             | 11523CCP003  |                 |     |                       |          |
| Nome do Discente:                  | Luiz Carlos Felix Carvalho   |                 |     |                       |          |
| Título do Trabalho:                | Indicadores de Convergência e Diversidade em Algoritmos Evolutivos para Otimização Multiobjetivo |                 |     |                       |          |
| Área de concentração:              | Ciência da Computação  |                 |     |                       |          |
| Linha de pesquisa:                 | Inteligência Artificial  |                 |     |                       |          |
| Projeto de Pesquisa de vinculação: | -  |                 |     |                       |          |

Reuniu-se, por videoconferência, a Banca Examinadora, designada pelo Colegiado do Programa de Pós-graduação em Ciência da Computação, assim composta: Professores Doutores: Paulo Henrique Ribeiro Gabriel - FACOM/UFU, Luiz Gustavo Almeida Martins - FACOM/UFU, Anderson da Silva Soares - INF/UFU, Danilo Sipoli Sanches - UTFPR e Márcia Aparecida Fernandes - FACOM/UFU orientadora do candidato.

Os examinadores participaram desde as seguintes localidades: Anderson da Silva Soares - Goiânia/GO; Danilo Sipoli Sanches - Curitiba/PR; Paulo Henrique Ribeiro Gabriel, Luiz Gustavo Almeida Martins e Márcia Aparecida Fernandes - Uberlândia/MG. O discente participou da cidade de Uberlândia/MG.

Iniciando os trabalhos a presidente da mesa, Prof.<sup>a</sup> Dr.<sup>a</sup> Márcia Aparecida Fernandes, apresentou a Comissão Examinadora e o candidato, agradeceu a presença do público, e concedeu ao Discente a palavra para a exposição do seu trabalho. A duração da apresentação do Discente e o tempo de arguição e resposta foram conforme as normas do Programa.

A seguir a senhora presidente concedeu a palavra, pela ordem sucessivamente, aos examinadores, que passaram a arguir o candidato. Ultimada a arguição, que se desenvolveu dentro dos termos regimentais, a Banca, em sessão secreta, atribuiu o resultado final, considerando o candidato:

**Aprovado.**

Esta defesa faz parte dos requisitos necessários à obtenção do título de Doutor.

O competente diploma será expedido após cumprimento dos demais requisitos, conforme as normas do Programa, a legislação pertinente e a regulamentação interna da UFU.

Nada mais havendo a tratar foram encerrados os trabalhos. Foi lavrada a presente ata que após lida e achada conforme foi assinada pela Banca Examinadora.



Documento assinado eletronicamente por **Márcia Aparecida Fernandes, Professor(a) do Magistério Superior**, em 02/09/2021, às 14:54, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Luiz Gustavo Almeida Martins, Professor(a) do Magistério Superior**, em 02/09/2021, às 15:07, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Paulo Henrique Ribeiro Gabriel, Professor(a) do Magistério Superior**, em 02/09/2021, às 15:33, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Danilo Sipoli Sanches, Usuário Externo**, em 08/09/2021, às 09:13, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Anderson da Silva Soares, Usuário Externo**, em 08/09/2021, às 12:37, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Pedro Frosi Rosa, Coordenador(a)**, em 16/09/2021, às 10:03, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site [https://www.sei.ufu.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **3013471** e o código CRC **FE20AD33**.

*Dedico este trabalho à minha mãe, minha esposa, minha irmã e, em especial, aos meus filhos. Que Deus abençoe para que este trabalho possa ser inspiração aos “meus pequenos” para um futuro brilhante baseado na ciência e que Deus possa sempre unir nossa família e nos abençoar. Dedico este trabalho, também, a todos que acreditam na ciência, mesmo com toda propaganda para o contrário.*



---

# Agradecimentos

Primeiramente, não posso agradecer outro, se não, Deus. Tenho certeza que cheguei ao fim desta pesquisa somente por que fui abençoado com muita determinação. Foram muitos os entraves e foram muitas as oportunidades para desistir. Mas, Deus me guiou para finalizar este trabalho passando por todas as dificuldades.

À minha esposa, Fernanda, e aos meus filhos, Matheus e Felipe. A base de minha vida é minha família. Eu não conseguiria elaborar uma pesquisa deste nível de dificuldade, se eu não tivesse vocês ao meu lado. Toda a compreensão, que foi necessária, não seria possível, se não houvesse amor em nosso lar. Obrigado Fernanda, pelas orações e por sempre estar ao meu lado, em todos os momentos. Obrigado Matheus e Felipe, por me ensinarem a ser feliz, por interromperem meu estudo frequentemente e por serem simplesmente vocês! Matheus, meu doutorado tem sua idade! Felipe, agora posso brincar, um pouco!

À minha irmã, Adriana, e seu esposo, Ezequiel, pessoas essenciais em minha vida, obrigado pelo apoio. Adriana, obrigado por perder a cerimônia daquele casamento e me acalmar depois do primeiro dia do vestibular. Eu não teria chegado até aqui, se você não tivesse me confortado. Aos meus sobrinhos, Vitor Hugo e Tatiana, as primeiras crianças de nossa família, que me trazem alegria gratuita. Aos meus familiares, Eduardo, Cláudia e Gledson, que são apoio em todos os momentos. Às minhas sobrinhas Stella, pelo seu belo sorriso em todos os dias, e Clara, nossa mais nova estrelinha, que já brilha em nossas vidas.

Aos meus amigos que fazem parte da minha vida e estão sempre presentes em todos os momentos, em especial, Reinis, Fify, Fransérgio, Juliano, Pipi e Wesley, que são também amigos de fé. Ao Jesus, que sempre me incentiva e me enaltece, mesmo quando não mereço. Ao ex-diretor do Centro de Tecnologia da Informação e Comunicação da UFU, professor Luís Fernando Faina, por possibilitar me afastar do trabalho para realização de parte deste trabalho e aos, além de amigos, colegas de trabalho, Marta, Álisson, Jayme, Luzencort, Santana, Muriel, João, Fernanda, Edmundo, Paulo e Jeziel, que permitiram a implementação de políticas para efetivar meu afastamento, além de trabalharem por mim durante tal período. Ao Erisvaldo e à Amanda, amigos e colegas de trabalho, que

administrativamente auxiliaram no decorrer desta pesquisa.

À professora Márcia, pessoa fantástica que Deus encaminhou em minha vida. Mais do que orientadora, é um exemplo de hombridade, honestidade e profissionalismo. A sua compreensão, das várias situações que ocorreram, foi essencial para a conclusão deste trabalho. E, claro, todo estudo desenvolvido não seria possível se não houvesse seu direcionamento. Muito obrigado, por tudo!

E, em especial, à minha mãe, Noemia, que com tantas dificuldades me colocou em condições de cursar o ensino superior. Foram muitas as abstinências em favor de nossa família e sei que muitas foram doloridas. A nossa história foi árdua, devo minha vida à senhora e este título é para coroar sua luta em favor do meu futuro e de minha irmã. Quando criança te falei que seria “doutor” e, agora, consegui. A criança queria ser médico e o adulto seguiu outro caminho, porém, indiretamente, alcançou o objetivo. Enfim, muito obrigado por lutar, por me educar e ser esse exemplo! Te amo!

*“Porque quando estou fraco, então sou forte.”*

*(2 Coríntios 12:10b)*

*“Há últimos que serão os primeiros, e há primeiros que serão os últimos.”*

*(Lucas 13:30)*

*“Autodidata é um ignorante por conta própria.”*

*(Mario Quintana)*

*“Se a educação sozinha não transforma a sociedade,  
sem ela tampouco a sociedade muda.”*

*(Paulo Freire)*



---

# Resumo

A convergência de Algoritmos Evolutivos Multiobjetivo (AEMO's) é um tema que não tem na literatura o devido retorno se comparado à importância que possui, porém é possível que esse fato esteja mudando. A relevância é destacada pois a convergência está diretamente ligada à capacidade de um AEMO obter a solução de um problema. Os estudos atuais utilizam indicadores para observar características no conjunto de soluções não dominadas do algoritmo, de modo a analisar determinado aspecto, auxiliando no estudo da convergência. Este trabalho propõe dois indicadores, Taxa de Concentração (*Concentration Rate*, CR) e Taxa de Diversidade (*Diversity Rate*, DR), para analisar características comportamentais e determinar um ponto de parada para AEMO's, além de um indicador que pode ser utilizado para determinar o ponto de convergência de um AEMO, Indicador de Estabilidade no Espaço de Soluções (*Stability in Solution Space*,  $S_3$ ). Para realizar os experimentos, dois AEMO's foram propostos, Algoritmo de Estimativa de Distribuição Simples (*Simple Estimation of Distribution Algorithm*, SEDA) e Algoritmo de Estimativa de Distribuição Simples com Inteligência de Enxame (*Simple Estimation of Distribution Algorithm with Swarm Intelligence*, SEDASI), para tratar o problema *Job Shop* Flexível. Os experimentos realizados, que contemplaram os AEMO's e indicadores propostos e da literatura, foram realizados com o problema *Job Shop* Flexível e com o *benchmark* ZDT. Os resultados mostraram a efetividade dos indicadores em apresentar as características a que se propõe e em auxiliar um processo para determinar o ponto de parada e de convergência de AEMO's, com resultados competitivos com os de métodos conhecidos da literatura.

**Palavras-chave:** Convergência. Diversidade. Otimização multiobjetivo. Algoritmos evolutivos multiobjetivo. Ponto de parada. Detecção de convergência. Otimização por Enxame de Partículas. Otimização por Enxame de Partículas com Diversidade. Algoritmos de Estimativa de Distribuição. Indicadores. Hipervolume. Problema *Job Shop* Flexível.



---

# Abstract

The convergence of Multiobjective Evolutionary Algorithms (MOEA's) is an important area in the study of this type algorithm, all the same, it is a topic that is currently not adequately represented in the literature, but this fact seems to be a changing one. This relevant importance is highlighted, as convergence is directly linked to the ability of an MOEA to obtain a solution to a problem. Current studies use indicators to observe characteristics in the set of non-dominated solutions of the algorithm, in order to analyze a certain aspect, which aids in the study of convergence. This work proposes two indicators, Concentration Rate (CR) and Diversity Rate (DR), to analyze behavioral characteristics and determine a stopping criteria for MOEA's, in addition to an indicator that can be used to determine the convergence point of a MOEA - Indicator of Stability in Solution Space ( $S_3$ ). To perform the necessary experiments, two MOEA's were proposed, the Simple Estimation of Distribution Algorithm (SEDA) and the Simple Estimation of Distribution Algorithm with Swarm Intelligence (SEDASI), to deal with the Flexible Job Shop Problem. The experiments, which included the MOEA's along with the proposed indicators, as well as those of the literature, were performed with the Flexible Job Shop Problem and the benchmark ZDT. The results showed the effectiveness of the indicators in presenting the proposed characteristics, while assisting in a process that determines the stopping and convergence point of MOEA's, with results that are competitive with those of methods already established in the literature.

**Keywords:** Convergence. Diversity. Multiobjective optimization. Multiobjective evolutionary algorithms. Stopping criteria. Convergence detection. Particle Swarm Optimization. Particle Swarm Optimization with Diversity. Estimation of Distribution Algorithms. Indicators. Hypervolume. Flexible Job Shop Problem.



---

## Lista de ilustrações

|  |     |
|--|-----|
| Figura 1 – Exemplo de cálculo de modelo probabilístico . . . . .                 | 36  |
| Figura 2 – Exemplo de criação de indivíduos . . . . .                            | 37  |
| Figura 3 – Gráficos de CR e DR - DIPSO . . . . .                                 | 59  |
| Figura 4 – Gráfico de $S_3$ no problema KC 4×5 - DIPSO . . . . .                 | 62  |
| Figura 5 – Representação do Indivíduo . . . . .                                  | 63  |
| Figura 6 – Exemplo de ES para gerar uma MP . . . . .                             | 64  |
| Figura 7 – Exemplo de uma MP . . . . .   | 64  |
| Figura 8 – Exemplo de ordenação de um indivíduo - Parâmetros . . . . .           | 66  |
| Figura 9 – Exemplo de ordenação de um indivíduo - Iteração 1, Passo 1 . . . . .  | 67  |
| Figura 10 – Exemplo de ordenação de um indivíduo - Iteração 1, Passo 2 . . . . . | 67  |
| Figura 11 – Exemplo de ordenação de um indivíduo - Iteração 1, Passo 3 . . . . . | 67  |
| Figura 12 – Exemplo de ordenação de um indivíduo - Iteração 1, Passo 4 . . . . . | 67  |
| Figura 13 – Exemplo de ordenação de um indivíduo - Iteração 2, Passo 1 . . . . . | 68  |
| Figura 14 – Exemplo de ordenação de um indivíduo - Iteração 2, Passo 2 . . . . . | 68  |
| Figura 15 – Exemplo de ordenação de um indivíduo - Iteração 2, Passo 3 . . . . . | 68  |
| Figura 16 – Exemplo de ordenação de um indivíduo - Iteração 2, Passo 4 . . . . . | 69  |
| Figura 17 – Gráficos do problema KC10X07 . . . . .                               | 97  |
| Figura 18 – Gráficos do problema KC15X10 . . . . .                               | 98  |
| Figura 19 – Gráficos do problema BR MK03 . . . . .                               | 99  |
| Figura 20 – Gráficos do problema BR MK08 . . . . .                               | 100 |
| Figura 21 – Gráficos dos problemas ZDT's do algoritmo NSGA2 . . . . .            | 103 |
| Figura 22 – Gráficos dos problemas ZDT's do algoritmo SPEA2 . . . . .            | 104 |
| Figura 23 – Gráficos do problema KC04X05 . . . . .                               | 120 |
| Figura 24 – Gráficos do problema KC08X08 . . . . .                               | 121 |
| Figura 25 – Gráficos do problema KC10X07 . . . . .                               | 122 |
| Figura 26 – Gráficos do problema KC10X10 . . . . .                               | 123 |
| Figura 27 – Gráficos do problema KC15X10 . . . . .                               | 124 |
| Figura 28 – Gráficos do problema BR MK01 . . . . .                               | 125 |

|  |     |
|--|-----|
| Figura 29 – Gráficos do problema BR MK02 . . . . . | 126 |
| Figura 30 – Gráficos do problema BR MK03 . . . . . | 127 |
| Figura 31 – Gráficos do problema BR MK04 . . . . . | 128 |
| Figura 32 – Gráficos do problema BR MK05 . . . . . | 129 |
| Figura 33 – Gráficos do problema BR MK06 . . . . . | 130 |
| Figura 34 – Gráficos do problema BR MK07 . . . . . | 131 |
| Figura 35 – Gráficos do problema BR MK08 . . . . . | 132 |
| Figura 36 – Gráficos do problema BR MK09 . . . . . | 133 |
| Figura 37 – Gráficos do problema BR MK10 . . . . . | 134 |

---

## Lista de tabelas

|   |    |
|---|----|
| Tabela 1 – Dados Estatísticos em Porcentagem da Evolução do DIPSO - CR . . . .                    | 56 |
| Tabela 2 – Dados Estatísticos em Absoluto da Evolução do DIPSO - CR . . . . .                     | 56 |
| Tabela 3 – Dados Estatísticos em Porcentagem da Evolução do DIPSO - DR . . . .                    | 57 |
| Tabela 4 – Dados Estatísticos em Absoluto da Evolução do DIPSO - DR . . . . .                     | 58 |
| Tabela 5 – SEDA - Parâmetros . . . . .  | 75 |
| Tabela 6 – Resultados do SEDA - KC . . . . .  | 75 |
| Tabela 7 – Resultados do SEDA - BR . . . . .  | 76 |
| Tabela 8 – W2-EDA (Wang; Wang; Liu, 2013) - Resultados . . . . .                                  | 77 |
| Tabela 9 – SEDASI - Parâmetros . . . . .  | 78 |
| Tabela 10 – Resultados do SEDASI - KC . . . . .   | 79 |
| Tabela 11 – SEDA - Parâmetros . . . . .   | 80 |
| Tabela 12 – DIPSO - Parâmetros . . . . .  | 81 |
| Tabela 13 – PSO - Parâmetros . . . . .  | 82 |
| Tabela 14 – Ponto de convergência (OCD-HV) . . . . .  | 82 |
| Tabela 15 – Hipervolume no Ponto de Convergência e na Geração Final . . . . .                     | 82 |
| Tabela 16 – Resultados do SEDA e do DIPSO no Ponto de Convergência (OCD-HV)                       | 84 |
| Tabela 17 – Resultados do SEDA e do DIPSO na Geração Final ( <i>MAX_GEN</i> ) . .                 | 84 |
| Tabela 18 – Análise do Ponto de Convergência de DIPSO (OCD-HV) . . . . .                          | 85 |
| Tabela 19 – DIPSO - Parâmetros . . . . .  | 88 |
| Tabela 20 – SEDA - Parâmetros . . . . .   | 88 |
| Tabela 21 – SEDASI - Parâmetros . . . . .   | 89 |
| Tabela 22 – NSGA2 - Parâmetros . . . . .  | 89 |
| Tabela 23 – SPEA2 - Parâmetros . . . . .  | 89 |
| Tabela 24 – DIPSO - FJSP - Comparação do uso de indicadores para parada do<br>algoritmo . . . . . | 90 |
| Tabela 25 – SEDA - FJSP - Comparação do uso de indicadores para parada do<br>algoritmo . . . . .  | 91 |

|   |     |
|---|-----|
| Tabela 26 – SEDASI - FJSP - Comparação do uso de indicadores para parada do algoritmo . . . . . | 92  |
| Tabela 27 – NSGA2 - FJSP - Comparação do uso de indicadores para parada do algoritmo . . . . .  | 92  |
| Tabela 28 – SPEA2 - FJSP - Comparação do uso de indicadores para parada do algoritmo . . . . .  | 93  |
| Tabela 29 – Relação de falhas na identificação do ponto de parada . . . . .                     | 95  |
| Tabela 30 – NSGA2 - ZDT - Comparação do uso de indicadores para parada do algoritmo . . . . .   | 101 |
| Tabela 31 – SPEA2 - ZDT - Comparação do uso de indicadores para parada do algoritmo . . . . .   | 101 |

---

## Lista de siglas

**AE** Algoritmo Evolutivo

**AEMO** Algoritmo Evolutivo Multiobjetivo

**AG** Algoritmo Genético

**BR** *Benchmark* de Brandimarte

**CR** *Concentration Rate* (Taxa de Concentração)

**DIPSO** *Particle Swarm Optimization with Diversity* (Otimização por Enxame de Partículas com Diversidade)

**DR** *Diversity Rate* (Taxa de Diversidade)

**EA** *Elitist Amount* (Quantidade Elitista)

**EDA** *Estimation of Distribution Algorithm* (Algoritmo de Estimativa de Distribuição)

**EGP** *Evidence Gathering Process* (Processo de Coleta de Evidências)

**ES** *Elitist Set* (Conjunto Elitista)

**FJSP** *Flexible Job Shop Problem* (Problema *Job Shop* Flexível)

**FNS** *Fast Non-dominated Sorting*

**GD** Generational Distance

**HV** Hipervolume

**IE** Inteligência de Enxame

**IC** Inteligência Computacional

**IGD** *Inverse Generational Distance*

**IGD+** *Inverse Generational Distance* modificado

**IT** Indivíduo Temporário

**JSP** (*Job Shop Problem*) (Problema *Job Shop*)

**KC** *Benchmark* de Kacem

**M** *Makespan*

**MOEA/D** *Multiobjective Evolutionary Algorithm based on Decomposition*

**MOFJSP** *Multiobjective Flexible Job Shop Problem* (Problema *Job Shop* Flexível Multiobjetivo)

**MOGA** *Multiobjective Evolutionary Genetic Algorithm*

**MP** Matriz de Probabilidade

**NSGA** *Non-dominated Sorting Genetic Algorithm*

**NSGA 2** *Non-dominated Sorting Genetic Algorithm 2*

**NSGA 3** *Non-dominated Sorting Genetic Algorithm 3*

**OCD** *Online Convergence Detection*

**OCD-HV** *Online Convergence Detection* com Hipervolume

**OCD-IGD** *Online Convergence Detection* com *Inverse Generational Distance*

**OCD-S<sub>3</sub>** *Online Convergence Detection* com S<sub>3</sub>

**OFCD** *Offline Convergence Detection*

**OPM** Operação por Máquina

**P-FJSP** FJSP Parcial

**PFA** Parada por Falta de Atividade

**PSO** *Particle Swarm Optimization* (Otimização por Enxame de Partículas)

**S<sub>3</sub>** *Stability in Solution Space* (Indicador de Estabilidade no Espaço de Solução)

**SEA** *Simple Evolutionary Algorithm*

**SEDA** *Simple Estimation of Distribution Algorithm* (Algoritmo de Estimatição de Distribuição Simples)

**SEDASI** *Simple Estimation of Distribution Algorithm with Swarm Intelligence* (Algoritmo de Estimatição de Distribuição Simples com Inteligência de Enxame)

**SND** Soluções Não Dominadas

**SPEA** *Strength Pareto Evolutionary Algorithm*

**SPEA 2** *Strength Pareto Evolutionary Algorithm 2*

**T-FJSP** FJSP Total

**TRD** *Testing-based Runlength Detection*

**TW** *Total Workload* (Carga de Trabalho Total)

**W** *Workload* (Carga de Trabalho)

**ZDT** Zitzler, Deb e Thiele



---

# Sumário

|            |  |           |
|------------|--|-----------|
| <b>1</b>   | <b>INTRODUÇÃO . . . . .</b>  | <b>25</b> |
| <b>1.1</b> | <b>Motivação . . . . .</b>   | <b>27</b> |
| <b>1.2</b> | <b>Objetivos e Desafios da Pesquisa . . . . .</b>                    | <b>28</b> |
| <b>1.3</b> | <b>Hipótese . . . . .</b>  | <b>28</b> |
| <b>1.4</b> | <b>Contribuições . . . . .</b>                                       | <b>29</b> |
| <b>1.5</b> | <b>Organização da Tese . . . . .</b>                                 | <b>29</b> |
| <b>2</b>   | <b>FUNDAMENTAÇÃO TEÓRICA . . . . .</b>                               | <b>31</b> |
| <b>2.1</b> | <b>Otimização Multiobjetivo . . . . .</b>                            | <b>31</b> |
| <b>2.2</b> | <b>Problema <i>Job Shop</i> Flexível . . . . .</b>                   | <b>33</b> |
| <b>2.3</b> | <b>Algoritmos . . . . .</b>  | <b>33</b> |
| 2.3.1      | Algoritmos de Estimação de Distribuição . . . . .                    | 35        |
| 2.3.2      | Otimização por Enxame de Partículas . . . . .                        | 37        |
| 2.3.3      | Algoritmos Evolutivos e de Inteligência de Enxame com FJSP . . . . . | 39        |
| 2.3.4      | Otimização por Enxame de Partículas com Diversidade . . . . .        | 40        |
| <b>2.4</b> | <b>Convergência . . . . .</b>  | <b>42</b> |
| 2.4.1      | Algoritmo Detecção de Convergência Online . . . . .                  | 44        |
| 2.4.2      | Indicadores . . . . .  | 45        |
| <b>2.5</b> | <b>Revisão da Literatura . . . . .</b>                               | <b>47</b> |
| <b>2.6</b> | <b>Considerações Finais . . . . .</b>                                | <b>51</b> |
| <b>3</b>   | <b>CONVERGÊNCIA E PONTO DE PARADA DE AEMO'S . . . . .</b>            | <b>53</b> |
| <b>3.1</b> | <b>Novos Indicadores . . . . .</b>                                   | <b>54</b> |
| 3.1.1      | Taxa de Concentração . . . . .                                       | 54        |
| 3.1.2      | Taxa de Diversidade . . . . .  | 56        |
| 3.1.3      | Aplicação de CR e DR . . . . .                                       | 57        |
| 3.1.4      | Indicador de Estabilidade no Espaço de Soluções . . . . .            | 61        |
| <b>3.2</b> | <b>Algoritmos para o FJSP . . . . .</b>                              | <b>62</b> |

|            |  |                |
|------------|--|----------------|
| 3.2.1      | Representação do Indivíduo . . . . .                               | 62             |
| 3.2.2      | SEDA: Um Simples Algoritmo de Estimativa de Distribuição . . . . . | 63             |
| <b>3.3</b> | <b>Considerações Finais . . . . .</b>                              | <b>71</b>      |
| <b>4</b>   | <b>EXPERIMENTOS E ANÁLISE DOS RESULTADOS . . . . .</b>             | <b>73</b>      |
| <b>4.1</b> | <b>SEDA e SEDASI . . . . .</b>                                     | <b>73</b>      |
| 4.1.1      | SEDA - Experimentos e Análise dos Resultados . . . . .             | 74             |
| 4.1.2      | SEDASI - Experimentos e Análise dos Resultados . . . . .           | 78             |
| <b>4.2</b> | <b>Convergência: PSO, DIPSO, SEDA . . . . .</b>                    | <b>78</b>      |
| 4.2.1      | Descrição dos Experimentos . . . . .                               | 80             |
| 4.2.2      | Análise . . . . .  | 81             |
| <b>4.3</b> | <b>Ponto de Parada de AEMO's com FJSP . . . . .</b>                | <b>86</b>      |
| <b>4.4</b> | <b>Experimentos de NSGA2 e SPEA2 com problemas ZDT's . . . . .</b> | <b>96</b>      |
| <b>4.5</b> | <b>Considerações Finais . . . . .</b>                              | <b>102</b>     |
| <b>5</b>   | <b>CONCLUSÃO . . . . .</b>   | <b>105</b>     |
| <b>5.1</b> | <b>Principais Contribuições . . . . .</b>                          | <b>106</b>     |
| <b>5.2</b> | <b>Trabalhos Futuros . . . . .</b>                                 | <b>107</b>     |
| <b>5.3</b> | <b>Contribuições em Produção Bibliográfica . . . . .</b>           | <b>107</b>     |
|            | <b>REFERÊNCIAS . . . . .</b>                                       | <b>109</b>     |
|            | <br><b>APÊNDICES</b>   | <br><b>117</b> |
|            | <b>APÊNDICE A – GRÁFICOS . . . . .</b>                             | <b>119</b>     |

---

# Introdução

A convergência de algoritmos evolutivos multiobjetivo (AEMO's) é um tema que ao longo dos anos não recebeu a devida importância na literatura, com poucos trabalhos elaborados (Trautmann et al., 2008). Porém, estamos em um momento em que tal fato não é mais verdade, de tal maneira que esse tema pode estar sendo um dos maiores focos de pesquisas dentro dos temas relacionados com AEMO's (Abouhawwash; Seada; Deb, 2017).

AEMO's são algoritmos que visam tratar problemas de otimização multiobjetivo, ou seja, visam otimizar uma função que é formada por várias funções objetivo. A comparação de soluções de um problema multiobjetivo pode não ser simples, pois os valores resultantes das funções objetivos podem ser conflitantes. Existem algumas abordagens para comparar essas soluções e a adotada neste trabalho é a de dominância de Pareto. Nessa abordagem, uma solução é considerada melhor do que outra, se, em todas as funções objetivo, possuir valores menores ou iguais e, em pelo menos uma função objetivo, possuir valor estritamente menor (para problemas de minimização, em problemas de maximização considerar maior ou igual e maior, ao invés de, menor ou igual e menor). Nesse sentido, a solução de um problema multiobjetivo é um conjunto de soluções não dominadas (conjunto com soluções que não possuem relação de dominância entre elas), denominada fronteira de Pareto.

Estudar a convergência de AEMO's significa observar a qualidade de soluções, estudar como o algoritmo obtém um conjunto de soluções, analisar sua capacidade em alcançar algum resultado ou verificar se o algoritmo pode obter solução ótima (Chen; Zou; Xie, 2011). Portanto, a importância de se explorar o tema está relacionada ao fato dele estar estritamente relacionado com os resultados que um algoritmo apresenta.

Aliado ao crescente número de trabalhos relacionados com a convergência, está o crescente uso dos indicadores, que são métricas que quantificam alguma característica de um AEMO e possuem finalidade própria (Zitzler et al., 2003). A necessidade de se comparar os algoritmos foi a base para que essa definição se instanciasse na literatura. É grande o número de trabalhos apresentados na literatura sobre convergência que utilizam

indicadores para dar valor ao que se quer analisar, como pode ser observado em (He; Lin, 2016) e (Wagner; Trautmann; Naujoks, 2009). O primeiro utiliza os indicadores para efetuar comparações de características dos algoritmos e o segundo para determinar o ponto de convergência.

Outra importante característica associada à convergência de AEMO's é a diversidade. Deb e Jain (2002) ressaltam tal importância quando apresentam dois indicadores para avaliar a convergência e a diversidade de AEMO's. Os autores também indicam que, devido à importância de tais características, esses dois indicadores são suficientes para realizar a avaliação dos algoritmos. Por outro lado, Zitzler et al. (2003) destacam que não se pode limitar a quantidade de indicadores para avaliar de forma geral um AEMO.

Encontrar o ponto de convergência de um AEMO associa a convergência à identificação de um ponto de parada para o algoritmo, pois se tal ponto é identificado, o algoritmo chegou em seu resultado e pode ser finalizado. A identificação do ponto de convergência é tema do trabalho desenvolvido por Trautmann et al. (2008), em que, utilizando teste estatístico, verifica a estabilidade do conjunto de aproximação. Determinar um ponto de parada para um AEMO é um tema que pode ser analisado como proposto por Martí et al. (2016), que apresenta um indicador para verificar a evolução do conjunto de aproximação a cada geração, observando a dominância do conjunto da geração atual em relação à anterior.

Nesse sentido, este trabalho propõe três novos indicadores para se observar a convergência e o ponto de parada de AEMO's, baseando-se empiricamente no estudo da convergência e da diversidade populacional: 1) Taxa de Concentração (*Concentration Rate*, CR); 2) Taxa de Diversidade (*Diversity Rate*, DR); e, 3) Indicador de Estabilidade no Espaço de Soluções (*Stability in Solution Space*,  $S_3$ ). Além desses, um algoritmo para identificar um ponto de parada para um AEMO, através de CR e DR, denominado de algoritmo de Parada por Falta de Atividade (PFA), e dois AEMO's, Algoritmo de Estimativa de Distribuição Simples (*Simple Estimation of Distribution Algorithm*, SEDA) e Algoritmo de Estimativa de Distribuição Simples com Inteligência de Enxame (*Simple Estimation of Distribution Algorithm with Swarm Intelligence*, SEDASI), são propostos.

Os indicadores CR e DR permitem analisar o nível de atividade de um AEMO, através da observância da concentração de indivíduos no conjunto de soluções não dominadas e da efetividade do algoritmo em buscar novas posições no espaço de busca. Ao serem associados ao PFA, estimam um ponto de parada para um AEMO de forma que o algoritmo esteja próximo de sua estagnação. O indicador  $S_3$  visa identificar a convergência analisando o espaço de soluções, identificando a estabilidade do conjunto de soluções não dominadas.

Os experimentos utilizaram algoritmos já conhecidos pela literatura: Otimização por Enxame de Partículas com Diversidade (*Particle Swarm Optimization with Diversity*, DIPSO) (Carvalho; Fernandes, 2014), *Non-dominated Sorting Genetic Algorithm 2* (NSGA2)

(Deb et al., 2002) e *Strength Pareto Evolutionary Algorithm 2* (SPEA2) (Zitzler; Laumanns; Thiele, 2001), além de SEDA e SEDASI, que foram propostos com esse objetivo. Esses algoritmos são a base do experimento que utilizou o problema *Job Shop* Flexível (*Flexible Job Shop Problem*, FJSP), um problema de escalonamento, para verificar a eficiência dos indicadores. Além desse, um outro experimento foi realizado com o *benchmark* ZDT (sigla que referencia os nomes dos autores: Zitzler, Deb e Thiele) (Zitzler; Deb; Thiele, 2000) com os algoritmos NSGA2 e SPEA2, de forma a verificar a eficiência dos indicadores em problemas elaborados para experimentar AEMO's clássicos da literatura. Os experimentos mostraram que PFA, com CR e DR, e  $S_3$  possuem um comportamento competitivo quando comparados com outros algoritmos e indicadores conhecidos na literatura. No primeiro experimento contemplando o FJSP,  $S_3$  apresentou melhores resultados, enquanto no segundo experimento, com o *benchmark* ZDT, PFA, com CR e DR, obteve os melhores resultados.

## 1.1 Motivação

Devido aos resultados promissores com o DIPSO (Carvalho, 2015), um algoritmo de Otimização por Enxame de Partículas (*Particle Swarm Optimization*, PSO) híbrido para o FJSP multiobjetivo, que obteve soluções não dominadas ainda não mencionadas na literatura para algumas instâncias do problema, além de que observações sobre a evolução do algoritmo foram feitas, visando estudo da convergência e diversidade da população desse algoritmo. Essas observações consistiram, principalmente, em acompanhar as alterações das posições de cada partícula, bem como o surgimento de novas, já que esse algoritmo apresenta a capacidade de incluir diversidade no enxame.

Um outro fator que motiva essa proposta é o fato da convergência estar diretamente associada ao resultado do algoritmo, por isso, é importante estudar ou detectar a convergência dos AEMO's. De acordo com Derrac et al. (2014), tal importância se apresenta, principalmente, para a proposição de novos algoritmos, pois é esperado que um novo algoritmo garanta um bom resultado. Entretanto, a maioria dos trabalhos da literatura avaliam o conjunto de aproximação ao final do algoritmo, não observando o caminho que foi traçado para se atingir tal ponto (Deb; Jain, 2002). Os trabalhos de Martí et al. (2016) e Trautmann et al. (2008) seguem essa narrativa e buscam observar o conjunto de aproximação, no espaço objetivo, a cada geração, para determinar um ponto de parada e um ponto de convergência, respectivamente.

Conforme apresentado, o conceito de convergência está relacionado com o resultado do algoritmo, porém observar somente o conjunto de aproximação no espaço objetivo pode não ser suficiente para determinar o final da execução de um AEMO, pois, por exemplo, o algoritmo pode estar em um platô de sua curva de busca e otimização. Observando somente o conjunto de aproximação, essa característica pode não ser detectada.

## 1.2 Objetivos e Desafios da Pesquisa

Assim, baseado nas observações de Carvalho (2015) e na importância da convergência, citadas na motivação, o objetivo deste trabalho foi propor indicadores para estudo da convergência de algoritmos evolutivos e que também pudessem ser critério de parada do algoritmo. Sendo assim, os parâmetros para esses indicadores seriam características como evolução de cada partícula/indivíduo no enxame/população, diferentemente de indicadores que consideram o conjunto de aproximação. Como a ideia inicial surgiu a partir de problema e algoritmo particulares, para atingir esse objetivo que considera outros paradigmas para algoritmos evolutivos, os seguintes objetivos específicos foram propostos:

- ❑ Determinar indicadores que pudessem quantificar a capacidade de introduzir novas posições e quantificar o direcionamento do enxame para posições cujos valores de objetivo se encontravam no conjunto de aproximação.
- ❑ Avaliar os indicadores em DIPSO.
- ❑ Propor AEMO's baseados em outros paradigmas, como Algoritmos de Estimação de Distribuição (*Estimation Distribution Algorithms*, EDA), para também avaliar as métricas.
- ❑ Avaliar e comparar as métricas propostas com outros indicadores, como hipervolume (Zitzler; Thiele, 1998), a partir dos algoritmos propostos e utilizando o *Online Convergence Detection - Hipervolume* (OCD-HV) (Wagner; Trautmann, 2010).

Além desses objetivos específicos, após avaliações dos indicadores com os algoritmos propostos, foram realizadas novas avaliações utilizando outros problemas (*benchmark ZDT*), outros algoritmos evolutivos clássicos (NSGA2 e SPEA2) e outro indicador (*Inverse Generational Distance* (IGD) (Coello; Sierra, 2004)), a fim de diversificar a aplicação dos indicadores propostos e mostrar como utilizá-los. Assim, de acordo com esses objetivos, os dois principais desafios deste trabalho foram propor indicadores de uso geral e que fossem competitivos quando comparados aos mais utilizados da literatura, como hipervolume e IGD.

## 1.3 Hipótese

Conforme descrito, a maioria dos trabalhos propostos para análise de convergência e definição de um ponto de parada para um AEMO, o faz observando o conjunto de aproximação, no espaço objetivo. Entretanto, acredita-se que existem diversas características que podem estabilizar o conjunto de aproximação em pontos que representam um platô no processo de busca. Assim, a hipótese é que utilizar características como a evolução do enxame/população pode melhorar o ponto de convergência e, conseqüentemente, a

solução de um AEMO, pois observa-se o espaço de soluções e não somente o conjunto de aproximação do espaço objetivo. Assim, as principais questões de pesquisa são:

- QP1: A convergência e a diversidade de algoritmos evolutivos podem ser estudadas, a partir de observações da evolução de cada partícula/indivíduo no espaço de soluções?
- QP2: Indicadores baseados nas observações mencionadas em QP1 podem apontar pontos de parada para AEMO's, melhorando o conjunto de aproximação? Essa melhoria pode ser entendida como a descoberta de novas soluções não dominadas ou a presença de um grande número de soluções não dominadas.
- QP3: Esses indicadores são competitivos em relação ao hipervolume?

## 1.4 Contribuições

Neste trabalho foram propostos dois novos indicadores para se observar de forma comportamental os AEMO's, Taxa de Concentração (*Concentration Rate*, CR) e Taxa de Diversidade (*Diversity Rate*, DR), além de um novo indicador que visa ser utilizado para observar a estabilidade no espaço de busca através do quão espaçadas estão as soluções a cada geração, denominado indicador de Estabilidade no Espaço de Soluções (*Stability in Solution Space*,  $S_3$ ).

Outras contribuições são os AEMO's que foram propostos para tratar o FJSP, denominados SEDA e SEDASI. O primeiro é um EDA e o segundo é um EDA híbrido, com características de inteligência de enxame aplicadas, sendo que ambos apresentam resultados equivalentes àqueles conhecidos pela literatura.

## 1.5 Organização da Tese

A organização deste trabalho é detalhada a seguir.

O Capítulo 2 apresenta a fundamentação deste trabalho. Assim, os conceitos utilizados são detalhados nesse capítulo. Além disso, o capítulo apresenta os trabalhos correlatos à nossa proposta, apresentando um estado da arte relacionado com o estudo da convergência e de indicadores.

O Capítulo 3 descreve o estudo realizado. Os indicadores propostos, CR, DR e  $S_3$ , são descritos, sua utilização é especificada e a possibilidade de se identificar um ponto de parada para AEMO's, aplicando tais indicadores, é apresentada. Além desses, os algoritmos, SEDA e SEDASI, que foram propostos para tratar o FJSP, são apresentados.

O Capítulo 4 apresenta os experimentos realizados. Os experimentos são organizados em grupos que contemplam a experimentação para: 1) identificar a eficiência de SEDA e

SEDASI; 2) verificar a necessidade de novos indicadores; 3) comparar e validar o ponto de parada que os novos indicadores apresentam em AEMO's tratando o FJSP; e 4) comparar e validar o ponto de parada que os novos indicadores apresentam, quando aplicados a algoritmos e problemas clássicos da literatura.

O Capítulo 5 traz a conclusão do estudo resumindo o trabalho realizado, apontando os principais resultados e descrevendo os trabalhos futuros.

---

## Fundamentação Teórica

O trabalho desenvolvido foca no estudo da convergência de algoritmos evolutivos multiobjetivo. O estudo foi desenvolvido com os algoritmos: Otimização por Enxame de Partículas com Diversidade (*Particle Swarm Optimization with Diversity*, DIPSO), Algoritmo de Estimativa de Distribuição Simples (*Simple Estimation of Distribution Algorithm*, SEDA), Algoritmo de Estimativa de Distribuição Simples com Inteligência de Enxame (*Simple Estimation of Distribution Algorithm with Swarm Intelligence*, SEDASI), *Non-dominated Sorting Genetic Algorithm 2* (NSGA2) e *Strength Pareto Evolutionary Algorithm 2* (SPEA2). DIPSO e SEDASI são algoritmos híbridos, com características de algoritmos evolutivos e inteligência de enxame, porém não possuem características que inviabilizem o estudo como AEMO. Portanto, os temas trabalhados na pesquisa foram computação evolutiva, inteligência de enxame, convergência (definição e análise), indicadores de qualidade e de progresso, além do FJSP. Neste capítulo, os conceitos que envolvem a pesquisa e a revisão da literatura a respeito do tema são apresentados.

### 2.1 Otimização Multiobjetivo

Problemas de otimização são aqueles em que as soluções possuem valores de função mínimos ou máximos, dependendo do sentido da otimização. Boyd e Vandenberghe (2004) definem um problema de minimização com a seguinte equação:

$$\begin{aligned} & \min f(x) \\ & \text{sujeito a } g_i(x) \leq b_i, \quad i = 1, \dots, m \end{aligned} \quad (1)$$

em que  $f$  é a função objetivo a ser otimizada,  $x = (x_1, x_2, \dots, x_n)$  é o vetor de variáveis do problema,  $g_i$ ,  $i = 1, \dots, m$  são funções de restrição limitadas, respectivamente, pelas constantes  $b_1, \dots, b_m$ . A definição de um problema de maximização é obtida negando essa equação.

A Equação 1 define um problema de otimização (minimização) com um único objetivo. Um problema de otimização multiobjetivo possui mais de uma função objetivo, ou seja,  $f(x) = \{f_1(x), f_2(x), \dots, f_n(x)\}$ , sendo que, cada  $f_i(x)$ , para  $i = 1, \dots, n$ , é uma função objetivo e  $n$  é o número de objetivos. A análise de soluções desses problemas gera um conflito entre os objetivos, de modo que é necessário considerar uma abordagem. Por exemplo, sejam  $a = (3, 4)$  e  $b = (2, 5)$  duas soluções para um problema de minimização multiobjetivo com dois objetivos,  $f(x) = \{f_1(x), f_2(x)\}$ . Então,  $f_1(b) < f_1(a)$  e  $f_2(a) < f_2(b)$ . Portanto, há um conflito entre as soluções  $a$  e  $b$ , de modo que, ao comparar os dois objetivos das soluções, a melhor solução não pode ser determinada com uma simples comparação entre os valores das funções  $f_1$  e  $f_2$ .

Existem algumas abordagens para realizar a comparação entre as soluções de problemas de otimização multiobjetivo. Por exemplo, o método da função utilitária e o da dominância de Pareto. No primeiro método, o problema é transformado de modo que se tenha um objetivo a ser tratado, através de uma função utilitária que, tradicionalmente, é uma soma ponderada entre os objetivos. No segundo método, os conceitos de dominância de Pareto são utilizados para gerar as fronteiras de Pareto e, então, apresentar um conjunto de soluções não dominadas como resposta para o problema. A seguir, esses conceitos são apresentados.

**Definição 1 (Relação de Dominância)** *Sejam  $S$  um conjunto de soluções,  $x$  e  $y$  duas soluções que são diferentes e pertencem ao conjunto  $S$ ,  $f(x) = \{f_1(x), f_2(x), f_3(x), \dots, f_{numobj}(x)\}$  funções objetivo, em que  $numobj$  é o número de objetivos. Então,  $x$  domina  $y$ , se e somente se:*

- **Minimização:**  $f_1(x) \leq f_1(y), f_2(x) \leq f_2(y), \dots, f_{numobj}(x) \leq f_{numobj}(y)$  e para pelo menos um objetivo  $c, 1 \leq c \leq numobj, f_c(x) < f_c(y)$ .
- **Maximização:**  $f_1(x) \geq f_1(y), f_2(x) \geq f_2(y), \dots, f_{numobj}(x) \geq f_{numobj}(y)$  e para pelo menos um objetivo  $c, 1 \leq c \leq numobj, f_c(x) > f_c(y)$ .

**Definição 2 (Conjuntos de não dominadas - Fronteiras)** *Conjuntos de não dominadas são subconjuntos obtidos a partir do conjunto  $S$ , que contém soluções que não são dominadas por outras soluções. Assim, fronteira 1 ( $F_1$ ) é o conjunto de todas soluções que não são dominadas por qualquer outra solução de  $S$ . Fronteira 2 ( $F_2$ ) é o conjunto de todas soluções que não são dominadas por qualquer outra solução de  $S \setminus F_1$  ( $S - F_1$ ), e assim por diante.*

**Definição 3 (Ótimo de Pareto)** *O conjunto ótimo de Pareto é um subconjunto de  $S$ , que contém soluções que não são dominadas por qualquer outra. Em outras palavras, o conjunto  $F_1$ .*

## 2.2 Problema *Job Shop* Flexível

O problema *Job Shop* Flexível (*Flexible Job Shop Problem*, FJSP) é um problema de escalonamento de  $j$  jobs, que são sequências de  $n$  operações cada, e  $m$  máquinas, sendo que apresentar uma solução para o problema significa apresentar: 1) a associação de cada operação a uma máquina, observando um ou mais objetivos, como o *makespan*; 2) uma sequência de execução das operações. O FJSP é derivado do problema *Job Shop* (*Job Shop Problem*, JSP), em que uma operação pode ser executada somente por uma máquina. Assim, o FJSP é mais complexo, pois uma solução do JSP apresenta apenas uma sequência de execução das operações.

O FJSP pode ser classificado em parcial (P-FJSP) e total (T-FJSP). O primeiro é quando pelo menos uma operação não tem disponível todas as máquinas para ser executada. O segundo é quando todas as operações possuem todas as máquinas disponíveis para execução. Neste trabalho, é considerado a minimização de três objetivos ao tratar o FJSP, isto é, os algoritmos abordam o FJSP multiobjetivo (MOFJSP), sendo que os objetivos tratados são: o *makespan* (M), a carga de trabalho total (*total workload*, TW) e a carga de trabalho máxima (*workload*, W), definidos a seguir.

- **Makespan:** definido pelo maior instante de finalização de processamento entre as máquinas. A Equação 2 define o *makespan*, em que  $t_k$  é o instante de finalização de processamento da máquina  $k$ .

$$M = \max_{1 \leq k \leq m} t_k \quad (2)$$

- **Carga de trabalho total:** é a soma da carga de trabalho de todas as máquinas, sendo que entende-se por carga de trabalho, o tempo em que a máquina está efetivamente executando um processo. A Equação 3 apresenta a fórmula da carga de trabalho total, sendo que  $W_k$  é a carga de trabalho da máquina  $k$ .

$$TW = \sum_{k=1}^m W_k \quad (3)$$

- **Carga de trabalho máxima:** é a maior carga de trabalho entre todas as máquinas. A Equação 4 define a carga de trabalho máxima, em que  $W_k$ , assim como na Equação 3, é a carga de trabalho da máquina  $k$ .

$$W = \max_{1 \leq k \leq m} W_k \quad (4)$$

## 2.3 Algoritmos

Os algoritmos utilizados nesta pesquisa fazem parte de um ramo da Inteligência Artificial chamado de Inteligência Computacional (IC), que visa desenvolver sistemas inteligentes baseados em elementos da natureza, imitando características tais como, aprendizado

ou adaptação a novas situações, generalização, abstração, descobrimento e associação (Engelbrecht, 2007). São paradigmas que fazem parte da IC, Computação Evolutiva, Inteligência de Enxame, entre outros. Não há consenso quanto a distinção entre Inteligência de Enxame e Computação Evolutiva. Existem trabalhos que consideram que a Inteligência de Enxame faz parte da Computação Evolutiva, por exemplo na pesquisa de Shao et al. (2013). Porém, neste trabalho utilizaremos a abordagem proposta por Engelbrecht (2007), que classifica de forma distinta.

A Computação Evolutiva compreende os métodos que se baseiam na evolução biológica (Engelbrecht, 2007). O termo evolução na biologia foi primeiramente utilizado por Lamarck (1809), porém foi o naturalista Charles Robert Darwin que ficou conhecido pela teoria da evolução (Darwin, 1860). Essa teoria busca explicar a evolução dos seres vivos baseando na seleção natural, na qual espécies com características mais adequadas ao meio onde vivem são selecionadas naturalmente por melhor se adaptarem às necessidades que o local apresenta. Ou seja, há uma competição entre os indivíduos pela sobrevivência e aqueles que se adaptam melhor ao meio sobrevivem e se reproduzem, sendo que, na reprodução, as características dos indivíduos são levadas para os seus descendentes. Outra característica é que, no decorrer das gerações, uma espécie pode gerar outras, através da mutação.

Assim, os Algoritmos Evolutivos (AE's), que são algoritmos de busca e otimização baseados na teoria da evolução, é um método que codifica os principais elementos dessa teoria: a seleção natural, o cruzamento e a mutação. Os AE's utilizam uma população de soluções candidatas que, por meio de um processo iterativo de evolução, busca encontrar a solução de um problema. O Algoritmo 1 apresenta os passos básicos de um algoritmo evolutivo. A inicialização da população é o processo para criar um conjunto de soluções candidatas inicial. Esse conjunto é chamado de população e cada solução candidata, de indivíduo. Depois de criada, a população inicial será submetida aos próximos passos (criação de novos indivíduos e a seleção dos indivíduos da nova população) até que se atinja um critério de parada. A seleção dos indivíduos é realizada com base na avaliação realizada através da função de aptidão, a qual mede o desempenho de cada indivíduo segundo um critério associado ao problema que está sendo tratado. Cada iteração realizada é chamada de geração e um exemplo de critério de parada é uma quantidade limite de gerações.

---

**Algoritmo 1** Algoritmo Evolutivo

---

```
1: pop = inicializarPopulacao();
2: while não atingir critério de parada do
3:   inds = criarNovosIndividuos();
4:   pop = selecaoNovaPopulacao(pop, inds);
5: end while
```

---

A repetição dos passos 3 e 4 geram uma contínua atualização da população e dois importantes aspectos são consequências dessa característica: a diversidade da população

e a convergência. A diversidade está associada à capacidade do algoritmo em manter uma grande quantidade de soluções candidatas diferentes na população. Essa característica é importante, pois influencia diretamente na busca de novas soluções candidatas. A criação de novos indivíduos utiliza, principalmente, as informações da população no espaço de soluções. Se a diversidade é baixa, pode não haver informações suficientes para que o processo evolutivo tenha sucesso na execução. Nesse sentido, tal diversidade influencia a capacidade do algoritmo em chegar em uma solução, ou seja, a capacidade de convergir. A relação entre esses dois aspectos é nítida quando analisamos a convergência de AE's e fenômenos como a convergência prematura, que pode ser interpretada como a convergência do algoritmo antes do esperado e/ou para uma solução inapropriada. Geralmente, uma baixa diversidade na população pode acarretar a convergência prematura.

Outro paradigma da IC é Inteligência de Enxame (IE), ou Inteligência Coletiva, que, compreende a ideia de resolução de problemas em que o compartilhamento do conhecimento, entre os indivíduos envolvidos, é realizado para que o problema seja solucionado. Um enxame é formado por agentes que, para alcançar o objetivo, fazem interações entre eles, trocando informações, para que um problema possa ser solucionado de forma mais eficiente do que se cada agente atuasse individualmente (Engelbrecht, 2007). O comportamento de resolução de problemas baseado nessas interações é a base da IE e os algoritmos de tal técnica implementam esse comportamento, sendo que são exemplos de algoritmos baseados em população de soluções candidatas.

### 2.3.1 Algoritmos de Estimação de Distribuição

Os Algoritmos de Estimação de Distribuição (*Estimation Distribution Algorithms*, EDA) (Mühlenbein; Paaß, 1996) são algoritmos evolutivos que utilizam uma estimativa de distribuição de probabilidades para criar novos indivíduos para a população, diferenciando-se dos Algoritmos Genéticos (AGs), o mais tradicional AE, nesse ponto, pois não utilizam o cruzamento para criar novos indivíduos (Hauschild; Pelikan, 2011). A modelagem de como estimar as probabilidades de distribuição é chamada de modelo probabilístico e é o ponto central do EDA. Tal modelo é baseado no paradigma do modelo gráfico probabilístico (Larrañaga, 2002).

No Algoritmo 2, o EDA é apresentado e podemos observar que, como citado, o modelo probabilístico é o responsável por gerar novos indivíduos, ou seja, direcionar o algoritmo no espaço de busca. Nos passos iniciais, 1 e 2, a população é inicializada com tamanho  $tamPop$  e depois avaliada segundo uma função de aptidão. No passo 3, o laço de iteração que evolui a população é definido e os passos de 4 a 8 realizam a busca. No passo 4, são selecionados os indivíduos para serem utilizados na construção do modelo probabilístico e no passo seguinte o modelo é criado. A quantidade de indivíduos selecionados é parametrizado por  $tamPm$ . Em seguida, no passo 6, novos indivíduos são gerados através do modelo probabilístico e, no passo 7, irão substituir aqueles que não foram selecionados no

passo 4. Assim, a quantidade de indivíduos gerados deve ser igual a diferença do tamanho da população ( $tamPop$ ) e a quantidade de indivíduos selecionados para a criação do modelo probabilístico ( $tamPm$ ). Finalizando a iteração, os novos indivíduos são avaliados, pela função de aptidão, no passo 8. O laço de iteração é executado até que um critério de parada seja alcançado, por exemplo, até atingir uma quantidade de gerações.

---

**Algoritmo 2** EDA
 

---

**Require:**  $tamPop$ 
 $tamPm$ 

```

1: pop = inicializarPopulacao( tamPop );
2: avaliar( pop );
3: while não atingir critério de parada do
4:   es = selecionar( pop, tamPm );
5:   mp = construirModeloProbabilistico( es );
6:   inds = criarNovosIndividuos( mp, tamPop - tamPm );
7:   pop = substituirIndividuos( pop, es, inds );
8:   avaliar( inds );
9: end while

```

---

O modelo probabilístico objetiva dar suporte para formar novos indivíduos, de modo que as escolhas de cada elemento do indivíduo sejam feitas de forma probabilística, utilizando as informações do modelo. Assim, sua estrutura e construção estão associadas à definição do problema e à representação da solução. Se um indivíduo é formado por um vetor de  $m$  posições e cada posição possui  $n$  possibilidades de valor, uma possível proposta para modelo probabilístico é uma matriz  $n \times m$ , em que a posição  $ixj$  representaria a probabilidade da posição  $j$  do vetor assumir o valor representado pela linha  $i$ , na respectiva coluna  $j$ . Nesse sentido, a Figura 1 apresenta a seleção de indivíduos para geração do modelo probabilístico e o modelo gerado. Nesse exemplo, o indivíduo é representado por um vetor de quatro posições, sendo que os valores permitidos para as posições do vetor são 0 e 1. O modelo probabilístico apresentado (Figura 1b) foi obtido determinando a porcentagem da quantidade de 1's em cada posição do vetor, considerando a seleção de indivíduos (Figura 1a). Portanto, o método de geração de um novo indivíduo fará escolhas probabilísticas entre 1 e 0, observando os valores probabilísticos do modelo.

|   |   |   |   |
|---|---|---|---|
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 |

(a) Seleção de indivíduos

|     |     |     |      |
|-----|-----|-----|------|
| 60% | 60% | 40% | 100% |
|-----|-----|-----|------|

(b) Modelo probabilístico

Figura 1 – Exemplo de cálculo de modelo probabilístico

A Figura 2 apresenta um exemplo de criação de indivíduos utilizando o modelo probabilístico da Figura 1. Três indivíduos foram criados utilizando escolhas probabilísticas em que as duas primeiras posições possuíam 60% de probabilidade de ser 1, a terceira posição possuía 40% e a quarta posição não tinha possibilidade de ser 0.

|   |   |   |   |
|---|---|---|---|
| 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |

Figura 2 – Exemplo de criação de indivíduos

### 2.3.2 Otimização por Enxame de Partículas

O algoritmo Otimização por Enxame de Partículas (*Particle Swarm Optimization*, PSO) (Kennedy; Eberhart, 1995) é um algoritmo de inteligência de enxame baseado no comportamento social de pássaros em busca de comidas, que utilizam conhecimento próprio e do bando. Assim, sua caracterização como IE, se dá pela troca de conhecimento citada. No PSO, o comportamento do enxame é de buscar conhecimento para uma nova e melhor posição no espaço de busca. Assim, no algoritmo, cada partícula (agente do enxame) conhece sua posição atual e velocidade, além de ter conhecimento de outras posições no espaço de busca, por sua própria experiência e pela troca de conhecimento com a vizinhança (Engelbrecht, 2007). Com esses dados, cada partícula tem sua posição modificada a cada geração.

O Algoritmo 3 apresenta o PSO, utilizando o modelo  $g_{best}$  (melhor posição global, *global best position*) para interações entre as partículas, o que significa que elas conhecerão a melhor posição visitada pelo enxame, ou que o tamanho da vizinhança para troca de informações é do tamanho do enxame. Nesse sentido, a modificação da velocidade e da posição de uma partícula no PSO é realizada através das equações:

$$v_i^{c+1} = \omega v_i^c + \phi_1 (g_{best} - x_i^c) + \phi_2 (p_{best} - x_i^c) \quad (5)$$

$$x_i^{c+1} = x_i^c + v_i^{c+1} \quad (6)$$

em que  $\phi_1 = r_1 \cdot a_g$ ,  $\phi_2 = r_2 \cdot a_l$ ,  $\omega \in \mathfrak{R}$  significa a inércia,  $a_g$  e  $a_l \in \mathfrak{R}$  são, respectivamente, as constantes de aceleração global (fator social) e local (fator cognitivo);  $r_1$  e  $r_2$  são números aleatórios uniformemente distribuídos em  $(0, 1)$ ,  $\phi_1$  e  $\phi_2$  são acelerações aleatórias global e local.

O primeiro passo apresentado no Algoritmo 3, PSO, é gerar um enxame inicial de tamanho  $b$  (parâmetro que identifica a quantidade de partículas no enxame), inicializando cada partícula (posição inicial  $(x_i^0)$  e velocidade inicial  $(v_i^0)$ ). No passo 2, a avaliação de cada partícula gerada é realizada, através do cálculo da aptidão. Os passos seguintes,

de 3 a 6, são para inicializar o  $p_{best}$  (melhor posição pessoal, *personal best position*) de cada partícula e o  $g_{best}$ , sendo que o  $p_{best}$  é a memória local de cada partícula e  $g_{best}$  está relacionado com a memória global. Como, no início, cada partícula conhece somente a posição inicial, o  $p_{best}$  é definido como sua própria posição. Já o  $g_{best}$  é definido como a melhor posição criada dentre todas as partículas do enxame. No passo 7, o laço iterativo para evoluir o enxame é definido e os passos de 8 a 21 evoluem o enxame a cada iteração, de modo que a busca seja realizada. Os passos de 8 a 11 realizam a atualização da velocidade e da posição de cada partícula do enxame, através, respectivamente, das Equações 5 e 6. No passo 12, as partículas são avaliadas através da função de aptidão. Nos passos de 13 a 17, o  $p_{best}$  é atualizado, caso seja necessário. O passo 14 realiza uma verificação para identificar se a nova posição de cada partícula é melhor do que seu respectivo  $p_{best}$ . Se for, o  $p_{best}$  da partícula é atualizado com sua posição atual (passo 15). O passo 18 identifica qual a melhor posição que o enxame está ocupando, para, no passo 19, verificar se o  $g_{best}$  precisa ser atualizado. A atualização do  $g_{best}$  é necessária, caso uma partícula melhor seja encontrada. Portanto, caso a partícula identificada no passo 18 for melhor, o  $g_{best}$  é atualizado com a posição de partícula em questão no passo 20, finalizando a iteração. O laço é repetido até que um critério de parada seja atingido, como um limite de número de gerações.

---

### Algoritmo 3 PSO

---

**Require:**  $b$

```

1:  $enxame = inicializarEnxame( b )$ 
2:  $calcularAptidao( enxame )$ 
3: for cada partícula  $p$  do  $enxame$  do
4:    $p.p_{best} = p$ 
5: end for
6:  $g_{best} = encontrarMelhor( enxame )$ 
7: while não atingir critério de parada do
8:   for cada partícula  $p$  do  $enxame$  do
9:      $atualizarVelocidade( p )$ 
10:     $atualizarPosicao( p )$ 
11:   end for
12:    $calcularAptidao( enxame )$ 
13:   for cada partícula  $p$  do  $enxame$  do
14:     if a posição de  $p$  é melhor do que  $p.p_{best}$  then
15:        $p.p_{best} = p$ 
16:     end if
17:   end for
18:    $p = encontrarMelhor( enxame )$ 
19:   if a  $p$  é melhor do que  $g_{best}$  then
20:      $g_{best} = p$ 
21:   end if
22: end while

```

---

Por ser um algoritmo com fórmulas matemáticas bem definidas para a troca de posições da partícula, compreende-se que o PSO é um técnica para tratar problemas do domínio contínuo. Porém, para aplicar o PSO a algum problema de domínio discreto, é necessário implementar alguma estratégia para que o problema seja compatível com o algoritmo. Na literatura, existem dois tipos de estratégias: a transformação do problema de domínio discreto em um problema contínuo ou a transformação do PSO em um algoritmo discreto (Shao et al., 2013).

### 2.3.3 Algoritmos Evolutivos e de Inteligência de Enxame com FJSP

A lista de trabalhos encontrados na literatura relacionados com algoritmos evolutivos e de inteligência de enxame tratando o FJSP é extensa. Dentre as técnicas mais utilizadas, podemos destacar, AGs (Guimarães, 2007) (Ling-li; Feng-Xing; Xiao-hong, 2009) (Wang et al., 2010) (Gen; Tsujimura; Kubota, 1994) (Cuiyu; Yang; Xinyu, 2021) (Liu; Zhang; Sun, 2021) e PSO (Carvalho; Fernandes, 2014) (Jia; Chen; Tang, 2007) (Zhang et al., 2009) (Ling-li; Feng-Xing; Xiao-hong, 2009) (Xiao-hong; Ling-li; Yue-wen, 2010) (Zhang; Hu, 2019), sendo que (Carvalho; Fernandes, 2014), (Jia; Chen; Tang, 2007), (Wang et al., 2010), (Gen; Tsujimura; Kubota, 1994), (Xiao-hong; Ling-li; Yue-wen, 2010) e (Liu; Zhang; Sun, 2021) consideram o aspecto multiobjetivo. Já a utilização de EDA com o FJSP pode ser observada em (Wang; Wang; Liu, 2013) (Wang et al., 2012) (Wang et al., 2013) (Hao et al., 2014).

Um dos trabalhos de maior destaque é o apresentado por Chiang e Lin (2013), pois a quantidade de resultados apresentados como não dominados, quando comparados a literatura, foi grande, o que outros trabalhos até aquele momento não tinham apresentado. O algoritmo desenvolvido, denominado Algoritmo Evolutivo Simples (*Simple Evolutionary Algorithm*, SEA), é um AEMO para tratar o FJSP multiobjetivo, no qual a maioria das soluções conhecidas para os problemas do benchmark de Brandimarte (BR) (Brandimarte, 1993) foram apresentadas. SEA usa um processo para gerar a população inicial elaborado com conhecimento específico do problema, com a justificativa de que o espaço de busca é mais promissor e a diversidade é mantida, evitando a convergência prematura. Na evolução da população, SEA usa duas operações de cruzamento (sendo a escolha entre elas feita de forma aleatória) e um processo que utiliza conhecimento específico do problema com cinco passos. A aplicação de mutação é realizada sempre que um indivíduo repetido é gerado. Os resultados apresentados por SEA aumentaram a quantidade de soluções conhecidas para os problemas BR, mostrando maior diversidade quando comparado a outras pesquisas (Wang et al., 2010) (Li; Pan; Liang, 2010) e (Xing; Chen; Yang, 2009).

O EDA é uma técnica que tem sido usada para resolver problemas complicados com bons resultados, por exemplo, problemas da mochila multi-objetivo (Li; Zhou; Zhang,

2012) e o problema do caixeiro viajante multiobjetivo (Gao; Zhou; Zhang, 2012). Dos trabalhos conhecidos (quando esta revisão foi elaborada) de EDA com FJSP, apenas (Wang; Wang; Liu, 2013) e (Wang et al., 2013) observam o aspecto multiobjetivo no problema. Wang et al. (2013) apresentaram um EDA para o FJSP multiobjetivo, dividindo o problema em dois sub-problemas, o primeiro para associar cada operação à uma máquina e o segundo para sequenciar as operações. Assim, o modelo probabilístico do EDA foi elaborado com duas matrizes, uma para cada sub-problema. A atualização das duas matrizes é realizada através de um método elaborado pelos autores e não é uma simples contagem, aumentando a complexidade do algoritmo. Os autores apresentam os resultados do algoritmo como sendo equivalentes a outros trabalhos, porém a quantidade de soluções apresentadas é pequena quando comparado com os resultados apresentados em (Chiang; Lin, 2013).

Wang, Wang e Liu (2013) trataram o FJSP multiobjetivo com um EDA híbrido. Apesar dos autores não o definir assim, é fácil ter essa conclusão quando o comparamos com outros Algoritmos Evolutivos (Carvalho; Fernandes, 2014) (Ling-li; Feng-Xing; Xiaohong, 2009). Eles associaram ao EDA dois operadores genéticos (também adotados em outros trabalhos (Carvalho; Fernandes, 2014)) e uma busca local que utiliza conhecimento específico do problema. Para construir a população inicial, foi construído um método que utiliza conhecimento específico do problema, a fim de melhor explorar o espaço de busca. Os resultados apresentados são equivalentes aos resultados de Chiang e Lin (2013), porém os autores compararam os resultados com outros trabalhos da literatura.

### 2.3.4 Otimização por Enxame de Partículas com Diversidade

O algoritmo Otimização por Enxame de Partículas com Diversidade (*Particle Swarm Optimization with Diversity*, DIPS0) (Carvalho; Fernandes, 2014) é um algoritmo PSO multiobjetivo híbrido, com operações definidas nos Algoritmos Genéticos (AG's) e no *Non-dominated Sorting Genetic Algorithm 2* (NSGA2) (Deb et al., 2002).

DIPS0 utiliza o modelo de vizinhança global e, sendo um algoritmo multiobjetivo, uma pequena modificação foi realizada no  $g_{best}$ . No PSO,  $g_{best}$  representa a melhor posição visitada pelo algoritmo naquele momento. No ambiente multiobjetivo, definir apenas uma melhor posição não é tão simples e como DIPS0 utiliza o conceito de fronteiras de Pareto para classificar as soluções, o  $g_{best}$  foi alterado para armazenar um conjunto de soluções, as melhores posições visitadas pelo algoritmo, ou seja,  $g_{best}$  é o conjunto de fronteira 1 de todas posições visitadas por todas partículas.

DIPS0 foi desenvolvido para tratar problemas de domínio discreto, ao contrário do PSO, que com sua equação de mudança de posição da partícula (Equação 6) direciona o algoritmo para tratar problemas de domínio contínuo. Assim, a equação de mudança de posição da partícula no DIPS0 não é igual à do PSO e, ainda, possui elementos dos AGs incorporados (operações de cruzamento e mutação). Tal adaptação é baseada no trabalho

de Niu, Jiao e Gu (2008) que propõem uma versão modificada da Equação 6, para aplicar o PSO em problemas de escalonamento, na qual a nova posição de uma partícula é a melhor opção entre: o cruzamento da partícula em questão com um elemento de  $g_{best}$ , o cruzamento da partícula em questão com um elemento de  $p_{best}$  e a mutação da mesma partícula.

A Equação 8 apresenta a fórmula para mudança de posição de uma partícula no DIPSO.

$$\begin{aligned} P_{k+1} &= \text{selecionarMelhor}(P'_1, P'_2, P'_3), \text{ em que} & (7) \\ P'_1 &= \text{cruzamento}(P_k, \text{selecionarUm}(g_{best})), \\ P'_2 &= \text{cruzamento}(P_k, p_{best}), \\ P'_3 &= \text{mutacao}(P_k) \end{aligned}$$

em que  $P_k$  é uma partícula do enxame na geração  $k$ ,  $p_{best}$  possui o mesmo significado do PSO e cada função utilizada na fórmula é descrita a seguir:

- **selecionarMelhor:** função que compara as partículas recebidas com parâmetro  $e$ , através da classificação em fronteiras de Pareto, retorna a melhor delas. Caso o conjunto da fronteira 1 tenha mais de uma partícula, um sorteio é realizado para retornar apenas uma partícula.
- **cruzamento:** função que realiza a operação de cruzamento utilizando as duas partículas de parâmetro.
- **selecionarUm:** função que retorna uma partícula aleatória a partir do conjunto de partículas de parâmetro.
- **mutacao:** função que realiza a operação de mutação utilizando a partícula de parâmetro.

Os operadores genéticos (cruzamento e mutação) utilizados são os mesmos comumente empregados nos AG's. Outra importante característica de DIPSO está relacionada com o operador de cruzamento. O algoritmo possui dois operadores de cruzamento, um para convergir e outro para diversificar. A justificativa está relacionada com a convergência prematura do PSO, principalmente quando aplicado a problemas complexos. A determinação de qual utilizar se faz por meio de uma escolha probabilística que utiliza os parâmetros do algoritmo  $p_{cc}$  e  $p_{dx}$ , em que  $p_{cc} + p_{dx} = 1$ .  $p_{cc}$  representa a probabilidade de escolher o operador de convergência e  $p_{dx}$  a probabilidade de escolha do operador de diversidade.

O algoritmo NSGA2 (Deb et al., 2002), possui um procedimento definido para realizar a classificação em conjuntos de fronteiras de Pareto de um determinado conjunto, chamado de *Fast Non-dominated Sorting* (FNS). DIPSO faz a utilização desse procedimento

em toda operação de classificação em conjunto de fronteiras, porém somente o conjunto fronteira 1 é calculado, pois os outros (fronteira 2, fronteira 3 ...) não tem utilidade no algoritmo.

O Algoritmo 4 apresenta o DIPSO. Os dois primeiros passos de DIPSO são para inicializar o enxame de tamanho  $tamEnx$  e avaliar as partículas recém criadas. No passo 3, o procedimento FNS é executado para obter o conjunto de fronteira 1 ( $F_1$ ) do enxame. No passo 4, o  $p_{best}$  de cada partícula é inicializado com o valor da posição da partícula e no passo 5, o  $g_{best}$  é inicializado com o conjunto  $F_1$ . No passo 6, a iteração para evoluir o enxame no algoritmo é definida e no passo 7 a iteração para percorrer todas as partículas do enxame é apresentada. Assim, os passos de 8 a 17 são para evoluir cada partícula a cada geração do algoritmo. Nos passos de 8 a 12, a mudança de posição da partícula em questão é realizada. No passo 8, o operador de cruzamento é escolhido segundo probabilidades  $p_{cc}$  e  $p_{dx}$  e nos passos de 9 a 12, a Equação 8 é executada, sendo que a nova posição da partícula é a melhor opção entre  $P'_1$ ,  $P'_2$  e  $P'_3$ . No passo 13, a partícula pode ser submetida a operação de mutação, de acordo com a respectiva probabilidade. Nos passos de 14 a 17, a partícula é avaliada em sua nova posição, se necessário seu  $p_{best}$  é atualizado (se a nova posição for melhor que a posição memorizada em  $p_{best}$ ) e o procedimento FNS é executado para obter o conjunto de fronteira 1 do enxame unido com o conjunto  $g_{best}$ . Se o resultado for diferente do conjunto  $g_{best}$  atual, ele é atualizado. Assim, os passos de 7 a 17 são executados até que se atinja um critério de parada, como, por exemplo, o limite de gerações, critério que foi utilizado neste trabalho.

## 2.4 Convergência

O termo convergência tem origem na matemática e é muito relevante quando se trata de sequência, séries e métodos de otimização. Sequências são listas de números ordenados e indexados e séries são somatórios de elementos de uma sequência. Assim, a série  $S_m$  (Equação 8) é a soma dos  $m$  primeiros termos da sequência  $(s_1, s_2, s_3, \dots, s_m, \dots, s_n)$ .

$$S_m = \sum_{i=1}^m s_i \quad (8)$$

Uma sequência é convergente quando o limite do seu  $n$ -ésimo termo tende a um valor. Por exemplo, a sequência  $(1, -1/2, 1/3, -1/4, \dots)$  definida na Equação 9 pelo termo geral  $a_n$  converge para 0, conforme limite na Equação 10.

$$a_n = \frac{(-1)^{n+1}}{n} \quad (9)$$

$$\lim_{n \rightarrow \infty} \frac{(-1)^{n+1}}{n} = 0 \quad (10)$$

**Algoritmo 4** Algoritmo DIPSO**Require:**  $tamEnx$  $probMutacao$  $p_{cc}$  $p_{dx}$ 


---

```

1:  $enxame = \text{inicializarEnxame}( tamEnx )$ 
2:  $\text{avaliar}( enxame )$ 
3:  $F_1 = \text{FNS}( enxame )$ ;
4:  $\text{inicializarPBest}( enxame )$ ;
5:  $g_{best} = F_1$ ;
6: while não atingir critério de parada do
7:   for cada partícula  $P_k$  de  $enxame$  do
8:      $\text{cruzamento} = \text{selecionarCruzamento}( p_{cc}, p_{dx} )$ ;
9:      $P'_1 = \text{cruzamento}( P_k, \text{selecionarUm}( g_{best} ) )$ 
10:     $P'_2 = \text{cruzamento}( P_k, p_{best} )$ 
11:     $P'_3 = \text{mutacao}( P_k )$ 
12:     $P_{k+1} = \text{selecionarMelhor}( P'_1, P'_2, P'_3 )$ 
13:     $\text{mutacao}( P_{k+1}, probMutacao )$ ;
14:     $\text{avaliar}( P_{k+1} )$ ;
15:     $\text{atualizarPBest}( P_{k+1} )$ ;
16:     $pop' = pop \cup g_{best}$ 
17:     $g_{best} = \text{FNS}( pop' )$ ;
18:   end for
19: end while

```

---

Nesse sentido, uma série é convergente quando a sequência formada por suas somas,  $(S_1, S_2, S_3, \dots)$ , tende a um valor.

Métodos de otimização são processos definidos para se obter o mínimo ou o máximo de uma função objetivo, realizando minimização ou a maximização. Dentre tais métodos, os métodos iterativos, que utilizam um processo que realiza diversas iterações, possuem analogia na definição de convergência com os algoritmos da inteligência computacional baseados em população. Os métodos iterativos utilizam um ponto inicial  $x_0$  para começar o processo e, a cada iteração, um ponto  $x_k$  é determinado, traçando uma rota e criando uma sequência e, com um  $k$  grande o suficiente, uma solução é apresentada. Se o método traçar uma rota a partir de um ponto  $x_0$ , de modo que um  $x_k$  seja apresentado como uma solução para o problema, entende-se que houve convergência finita. Caso não tenha como provar que um  $x_k$  é solução para o problema, então entende-se que há convergência assintótica e prova-se que  $x_k$  é uma boa aproximação da solução, considerando um  $k$  suficientemente grande (Izmailov; Solodov, 2010).

Nesse sentido, Trelea (2003) apresentou uma análise de convergência do PSO. No estudo, os parâmetros foram divididos por intervalos e, matematicamente, as fórmulas foram analisadas de forma iterativa. Apesar de que, o estudo foi realizado com um enxame de uma partícula, generalizando, entende-se que em uma faixa de valores dos

parâmetros, o PSO converge, porém não se garante a qualidade da solução.

A definição de convergência para os AEMO's mais utilizada na literatura é a capacidade do algoritmo em chegar na solução do problema (Chen; Zou; Xie, 2011). São poucos os trabalhos que apresentem um estudo formal, como (He; Lin, 2016), que propõe a taxa de convergência média assintótica.

Com o crescente número de trabalhos em torno do tema de convergência de AEMO's, podemos citar algoritmos como: o Detecção de Convergência *Online* (*Online Convergence Detection*, OCD) (Wagner; Trautmann; Naujoks, 2009) e Detecção de Convergência *Offline* (*Offline Convergence Detection*, OFCD) (Trautmann et al., 2008). Esses algoritmos têm o objetivo de indicar o ponto de convergência de um AEMO, utilizam indicadores para quantificar características das soluções e realizam cálculos estatísticos em torno desses valores para indicar se o algoritmo convergiu.

O ponto de convergência é o instante em que o algoritmo convergiu e, conseqüentemente, é um ponto de parada para o algoritmo. Entende-se por ponto de parada, um instante da execução do algoritmo em que se identifica, pela observância de alguma característica, que não é mais interessante manter ele executando. Por exemplo, identificar um ponto de parada através da identificação da estagnação. Portanto, um ponto de convergência pode ser utilizado como um ponto de parada para um algoritmo. Porém, um ponto de parada não pode ser utilizado como um ponto de convergência, pois não necessariamente a convergência está sendo observada.

### 2.4.1 Algoritmo Detecção de Convergência Online

Para ajudar na análise de convergência dos algoritmos, o algoritmo Detecção de Convergência *Online* (*Online Convergence Detection*, OCD) foi elaborado objetivando identificar o ponto de convergência de um AEMO enquanto o mesmo está sendo executado. O algoritmo propõe que ao final de cada geração, um teste estatístico seja realizado para determinar se o algoritmo convergiu. O teste estatístico usa um ou mais indicadores. No Algoritmo 5 é apresentado a versão do OCD em que é utilizado o hipervolume (Zitzler; Thiele, 1998) para avaliação do estado do algoritmo. Tal versão é apresentada em (Wagner; Trautmann, 2010), denominada OCD-HV.

Os parâmetros do OCD-HV são *varLimit*, *nPreGen*, *maxGen* e *ref. varLimit* é o limite de variação, que é usado com o valor do indicador hipervolume para determinar a convergência do algoritmo em análise. A quantidade de gerações a ser considerada na análise de convergência é chamada *nPreGen*. O número máximo de gerações do algoritmo analisado é o *maxGen* e *ref* é o ponto de referência usado para calcular o valor do hipervolume. A lógica por trás do OCD-HV é encontrada nas linhas 3 a 12. Na linha 5, o conjunto fronteira 1 (*Front\_1*) da população na respectiva geração é calculado. As linhas 7 e 8 mostram o cálculo do valor do indicador hipervolume (procedimento PHV).  $Hv_i$  é o valor do indicador hipervolume da *i*-ésima geração.  $\vec{Hv}$  são os valores

**Algoritmo 5** OCD-HV**Require:**  $varLimit$  $nPreGen$  $maxGen$  $ref$ 1:  $\alpha = 0,05$ 2:  $i = 0$ 3: **repeat**4:    $i = i + 1$ 5:   Obter  $Front\_1$  of  $i$ -th geração do AEMO6:   **if**  $i > nPreGen$  **then**7:      $Hv_i = PHv(Front\_1_i, ref)$ 8:      $\overrightarrow{DHv} = (PHv(Front\_1_{i-nPreGen}, ref), \dots, PHv(Front\_1_{i-1}, ref))$ 9:      $\overrightarrow{DHv} = (|Hv_i - Hv_{i-nPreGen}|, \dots, |Hv_i - Hv_{i-1}|)$ 10:      $pChi2(i) = Chi2(\overrightarrow{DHv}, varLimit)$ 11:   **end if**12: **until**  $(pChi2(i) \leq \alpha) \wedge (pChi2(i-1) \leq \alpha) \vee i = maxGen$ 

13: Parar o AEMO

14: **return** o critério de finalização ( $maxGen$  ou teste estatístico) e  $i$  (o número da geração em que o AEMO foi finalizado)

do indicador hipervolume das últimas  $nPreGen$  gerações. Na linha 9, a diferença dos valores do indicador hipervolume das últimas  $nPreGen$  gerações e o valor do indicador hipervolume da  $i$ -ésima geração é calculada e o resultado é armazenado em  $\overrightarrow{DHv}$ . Na linha 10, o teste de variância estatística  $\chi^2$  (Sheskin, 2007) é executado (procedimento Chi2) para testar se a variância de  $\overrightarrow{DHv}$  é menor que  $varLimit$ . Se o valor  $p$  do teste de variação  $\chi^2$  for menor que o  $\alpha$  definido em duas gerações consecutivas, o OCD-HV conclui a convergência do AEMO.

**2.4.2 Indicadores**

A comparação da qualidade de uma solução de problemas de um objetivo é simples, basta verificar a solução melhor avaliada. Porém em problemas multiobjetivo, a comparação não se mostra tão simples. Os AEMO's mais difundidos na literatura, por exemplo NSGA2 e SPEA2, se baseiam nas definições de dominância e nas fronteiras de Pareto para formar uma solução.

Aplicando as definições de dominância de Pareto, apresentadas na Seção 2.1, o algoritmo gera um conjunto de soluções não dominadas, que chamamos de conjunto de aproximação, e é a dificuldade de comparação desses conjuntos que motivou o surgimento dos indicadores de qualidade (Audet et al., 2021).

**Definição 4 (Indicadores de qualidade)** *Seja  $A$  um conjunto de aproximação, um indicador de qualidade  $IQ_x$  é um método que associa  $A$  a um valor numérico  $y$ , de forma a*

representar alguma característica de  $A$  em número.

$$IQ_x(A) = y \quad (11)$$

É comum a utilização de um ponto de referência na definição de um indicador. Trata-se de um ponto a ser escolhido que é dependente do problema e da definição do indicador, sendo determinante no valor resultante. A definição do ponto de referência é importante, pois, já que interfere no cálculo, uma escolha ruim pode interferir na análise, determinando conclusões incorretas (Naujoks; Hochstrate; Emmerich, 2005).

Existem os mais diversos indicadores de qualidade na literatura e, na sequência, listamos alguns.

- **Hipervolume (HV)** (Zitzler; Thiele, 1998): o volume entre um ponto de referência e o conjunto de aproximação formado no espaço objetivo, sendo que, o ponto de referência é, comumente, definido como o pior ponto conhecido do respectivo problema, incrementado de um valor de segurança (Naujoks; Hochstrate; Emmerich, 2005). Hipervolume é definido pela Equação 12.

$$HV(A, Z) = \lambda_r \left( \bigcup_{a \in A} [a, Z] \right) \quad (12)$$

em que  $A$  é o conjunto de aproximação,  $Z$  é o ponto de referência ( $Z \in \mathbb{R}^r$ ), sendo que para todo  $a \in A$ ,  $a \leq Z$  e  $\lambda_r$  é a medida Lebesgue  $m$ -dimensional (Audet et al., 2021).

- **Generational Distance (GD)** (Veldhuizen, 1999): é a média do somatório da distância de cada elemento do conjunto de aproximação  $A$  ao elemento mais próximo do ponto de referência  $Z$ , que deve ser definido como a solução ótima do problema analisado, para o caso de problemas com solução conhecida, ou como a melhor solução conhecida, para o caso de problemas em aberto, acrescido de um valor de segurança. GD é definido pela Equação 13.

$$GD(A, Z) = \frac{(\sum_{i=1}^{|A|} d_i^p)^{1/p}}{|A|} \quad (13)$$

em que  $d_i$  é a distância euclideana de um elemento de  $A$  ao elemento mais próximo de  $Z$  e, geralmente,  $p = 2$  (Veldhuizen, 1999).

- **Inverse Generational Distance (IGD)** (Coello; Sierra, 2004): é a média do somatório da distância de cada elemento do ponto de referência  $Z$  ao elemento mais próximo do conjunto de aproximação  $A$ . A definição do ponto de referência para

esse indicador é análoga à definição do ponto de referência do GD. IGD é definido pela Equação 14.

$$IGD(A, Z) = \frac{(\sum_{i=1}^{|Z|} d_i^p)^{1/p}}{|Z|} \quad (14)$$

em que  $d_i$  é a distância euclideana de um elemento de  $Z$  ao elemento mais próximo de  $A$  e, geralmente,  $p = 2$  (Audet et al., 2021).

Indicadores, inicialmente criados para medir e comparar a qualidade de conjuntos de aproximação, também são utilizados para acompanhar a evolução do algoritmo e Deb e Jain (2002) citam a importância de realizar essa análise a cada geração. Essa proposta foi evoluída e o conceito de indicadores de progresso aborda essa narrativa.

**Definição 5 (Indicadores de progresso)** *Um indicador de progresso  $IP_x$  é um método que qualifica determinado algoritmo a cada geração para avaliar a evolução do mesmo, ou seja, o progresso. (Martí et al., 2010)*

Ao contrário dos indicadores de qualidade, que visam observar somente características do conjunto de aproximação, os indicadores de progresso não delimitam o que analisar. Nesse sentido, os indicadores propostos nesta pesquisa se enquadram nessa definição, pois são indicadores que analisam a evolução do algoritmo a cada geração, não utilizando necessariamente o conjunto de aproximação.

## 2.5 Revisão da Literatura

O aumento no número de pesquisas sobre convergência de algoritmos evolutivos pode ser devido ao fato delas permitirem caracterizar melhor algoritmos a fim de especificar a qual classe de problema um determinado algoritmo é bem aplicado (Deb; Jain, 2002). Isso pode ajudar no ajuste fino da parametrização (Bansal; Gopal; Nagar, 2018) ou identificar os critérios de parada. No estudo de Abouhawwash, Seada e Deb (2017), é apresentado uma linha do tempo que descreve o foco das pesquisas relacionadas com AEMO's, mostrando justamente que as pesquisas sobre convergência podem ser consideradas o foco atual. Segundo os autores, o primeiro foco foi implementar algoritmos para resolver problemas multiobjetivos, *Multiobjective Genetic Algorithm* (MOGA) (Fonseca; Fleming, 1993) e *Non-dominated Sorting Genetic Algorithm* (NSGA) (Srinivas; Deb, 1994) são exemplos desse tema. Algoritmos que incluem estratégia de elitismo para manter soluções não dominadas na população constituem o segundo foco, como o NSGA2 (Deb et al., 2002), *Strength Pareto Evolutionary Algorithm* (SPEA) (Zitzler; Thiele, 1999) e SPEA2 (Zitzler; Laumanns; Thiele, 2001). Depois disso, o foco foi implementar algoritmos para problemas de muitos objetivos, como o *Multiobjective Evolutionary Algorithm based on*

*Decomposition* (MOEA/D) (Zhang; Li, 2007) e o *Non-dominated Sorting Genetic Algorithm 3* (NSGA3) (Deb; Jain, 2014; Jain; Deb, 2014). O último foco relatado descrevem dois temas que foram mais explorados. O primeiro está relacionado à convergência e o segundo diz respeito à integração de AEMO's com os tomadores de decisão.

A importância da convergência não é discutida na literatura, pois é assumida como um aspecto inerente aos algoritmos de otimização. Como citado por Derrac et al. (2014), uma técnica, ao ser apresentada, deve considerar a convergência. Outras características importantes a serem analisadas em um AE são eficiência, aplicabilidade em diferentes domínios e diversidade (Derrac et al., 2014). A importância dos estudos que analisam a convergência é citada por Bansal, Gopal e Nagar (2018) e um dos motivos é para ajustar os parâmetros do algoritmo. Nesse sentido, He e Lin (2016) apresentaram uma métrica para medir a velocidade de convergência de um AE, através de um cálculo baseado na média da aptidão dos indivíduos da população. Tal medida pode auxiliar na comparação entre as convergências dos AE's. Um outro estudo que observa a velocidade de convergência, foi elaborado por González-Álvarez, Vega-Rodríguez e Rubio-Largo (2014), que mede a quantidade de avaliações do algoritmo até que uma boa solução seja alcançada, na qual uma boa solução é definida como sendo uma solução com hipervolume maior do que uma porcentagem do hipervolume da solução ótima.

A definição de convergência que é vista na literatura de AEMO é a capacidade do algoritmo em alcançar a solução ótima (Chen; Zou; Xie, 2011), o que associa o estudo de tal característica diretamente com a diversidade, seja a preservação da diversidade na população, seja a diversidade do conjunto de aproximação, como pode ser visto em (Deb; Jain, 2002; González-Álvarez; Vega-Rodríguez; Rubio-Largo, 2014; Liu et al., 2017). Essa relação é destacada por alguns autores. Deb (2001) cita que o objetivo geral de um AEMO é encontrar uma solução próxima da ótima e diversa o suficiente para cobrir toda a fronteira de Pareto, preservando a diversidade da população. Liu et al. (2017) citam a importância de se manter a diversidade da população para alcançar a convergência, ressaltando que a maioria dos algoritmos são projetados com maior ênfase na convergência do que na diversidade. González-Álvarez, Vega-Rodríguez e Rubio-Largo (2014) ressaltam a importância da diversidade da população ao comparar algoritmos considerando a qualidade e a diversidade dos resultados e a quantidade de soluções encontradas pelo algoritmo.

O estudo apresentado por Deb e Jain (2002) destaca a importância da convergência e da diversidade de tal forma, que eleger essas duas características para comparar quaisquer conjuntos de aproximação funcionalmente, mesmo com alguma perda de informação na comparação. Como apresentado pelos autores, é intuitivo que o uso de métricas em uma quantidade menor do que o de objetivos torna a avaliação imprecisa (Deb; Jain, 2002). Assim, os autores classificam a convergência e a diversidade como as principais métricas em AEMO's, as definindo como objetivos funcionais.

Nesse cenário, em que a análise de convergência e de diversidade ganha importância, a utilização de métricas para quantificar as características dos AEMO's, como convergência e diversidade, é indispensável e a extensa utilização em estudos relacionados ao tema comprova sua necessidade (Falcón-Cardona; Coello, 2019). Os já citados trabalhos de He e Lin (2016) e González-Álvarez, Vega-Rodríguez e Rubio-Largo (2014) que apresentaram métricas para medir a velocidade de convergência são exemplos de uso e necessidade. Porém, o indicador de qualidade mais difundido entre as pesquisas é o hipervolume (Zitzler; Thiele, 1998). Como definido na Seção 2.4.2, um indicador de qualidade quantifica em um número real um conjunto de aproximação, através da análise de um ou mais aspectos de qualidade (Li; Yao, 2019). Zitzler e Thiele apresentaram o hipervolume para medir a performance de algoritmos. Com os avanços das proposições de novos indicadores e das utilizações dessas medidas, o hipervolume se destacou por ser um indicador compatível com o conceito de dominância de Pareto (Zitzler et al., 2003). Além disso, seu cálculo tem características de convergência e de diversidade. Outros exemplos de indicadores da literatura são: *additive  $\epsilon$*  (Zitzler et al., 2003), *Inverse Generational Distance* modificado (IGD+) (Ishibuchi et al., 2015) e, os definidos na Seção 2.4.2, *generational distance* (Veldhuizen, 1999) e *inverse generational distance* (Coello; Sierra, 2004).

A utilização dos indicadores pode ser aplicada para fazer comparações entre algoritmos a fim de apresentar resultados em torno de uma característica medida e a fim de se comparar conjuntos de aproximação (conjuntos de soluções não dominadas) resultantes de AEMO's, sendo que a maioria dos indicadores são para avaliar tais conjuntos (Deb; Jain, 2002). Nesse sentido, a avaliação de AEMO's com indicadores é estabelecida e, por ser algoritmos multiobjetivos, a utilização de mais de um indicador é indicada (Deb; Jain, 2002). Não é possível determinar um número finito de indicadores para avaliar um AEMO de  $m$  objetivos, ou seja, são necessários, no mínimo,  $m$  indicadores, para que não haja perda de informação (Zitzler et al., 2003).

Com a crescente utilização de indicadores nas pesquisas sobre AEMO's, processos de coleta de evidências (*Evidence Gathering Process*, EGP) foram propostos para auxiliar na decisão do ponto de parada de um algoritmo (Martí et al., 2010). Segundo os autores, um AEMO necessita de um processo local (indicadores) e um processo global (EGP) para determinar o ponto de parada de um algoritmo. O processo local quantifica características do algoritmo a cada geração, enquanto o processo global combina essas informações locais para fazer uma análise mais ampla, a fim de decidir sobre a parada do algoritmo (Martí et al., 2010). São exemplos de EGP, *Offline Convergence Detection* (Trautmann et al., 2008) e *Online Convergence Detection* (Wagner; Trautmann; Naujoks, 2009). EGP combinam a utilização de um ou mais indicadores, considerando uma ou mais gerações e fazem uma análise específica para determinar a parada de um AEMO (Wagner; Trautmann; Martí, 2011).

Nesse contexto, aplicando a utilização de indicadores para a detecção do ponto de

convergência, os trabalhos de Trautmann et al. (2008), Wagner, Trautmann e Naujoks (2009) e Trautmann et al. (2009) apresentaram dois EGP's para indicar a convergência de um AEMO. O primeiro, inicialmente chamado de *Testing-based Runlength Detection* (TRD) (Trautmann et al., 2008) e posteriormente renomeado para *Offline Convergence Detection* (OFCD) (Trautmann et al., 2009), retorna a geração de convergência de um algoritmo. O segundo, *Online Convergence Detection* (OCD) (Wagner; Trautmann; Naujoks, 2009), foi desenvolvido para determinar quando um algoritmo converge utilizando informações durante o processamento. Indicadores de qualidade são usados em ambos os algoritmos, OFCD e OCD, como o hipervolume. A diferença entre eles é que, o OFCD executa o algoritmo várias vezes para determinar a geração de convergência, enquanto o OCD indica a convergência durante a execução do algoritmo. Portanto, o OFCD, através de um experimento prévio, apresenta a geração de convergência para qualquer execução de forma fixa e o OCD analisa cada execução durante o processamento. Uma outra versão do OCD é apresentado no trabalho de Wagner e Trautmann (Wagner; Trautmann, 2010), o OCD-HV, simplificando os testes estatísticos e utilizando somente o hipervolume como indicador. A simplificação deve-se ao fato da utilização de somente um indicador, ao contrário da versão clássica, que faz o uso de três indicadores.

A identificação do ponto de convergência pode ser aplicado como ponto de parada para um AEMO. Porém, em alguns casos, é melhor obter uma boa solução com um custo computacional muito menor do que gastar um tempo consideravelmente maior para obter uma solução ligeiramente superior (González-Álvarez; Vega-Rodríguez; Rubio-Largo, 2014). Outra possibilidade para critério de parada é identificar se o algoritmo não tem mais possibilidades de melhorar a solução atual (Trautmann et al., 2009), ou seja, identificar se entrou em um estado de estagnação (Martí et al., 2016). Além desse, pode-se verificar se uma solução obtida é satisfatória, identificar se o método é incapaz de convergir, ou se o custo computacional aplicado é suficiente (Martí et al., 2016).

Na análise de convergência, a utilização de indicadores de qualidade é intuitiva, visto que suas definições se confundem, ou seja, ambos observam aspectos do conjunto de aproximação. Em relação à definição de um ponto de parada para o algoritmo, por ser uma questão mais ampla, os indicadores de progresso também são utilizados. Os indicadores de progresso, diferentemente dos indicadores de qualidade, objetivam avaliar a evolução do algoritmo, seja com o conjunto de aproximação, seja com outro aspecto do algoritmo medido a cada geração (Martí et al., 2010). Não existe consenso quanto a denominação da medida efetuada nesse cenário em que se avalia a evolução do algoritmo. Por exemplo, os autores apresentam dois indicadores em (Deb; Jain, 2002), um de convergência e outro de diversidade, ressaltando a importância de se analisar toda a evolução do algoritmo, não somente seu resultado final. Assim, os indicadores apresentados se encaixam na definição de indicadores de progresso, porém Deb e Jain denominaram de métricas de performance.

## 2.6 Considerações Finais

Observamos que a utilização de AE's em problemas complexos, como o MOFJSP, é alta e que os trabalhos que apresentam algoritmos para tratar o problema não fazem análises sobre a convergência do algoritmo. Por outro lado, vários trabalhos que estudam a convergência e a diversidade apresentam a importância dessas características. No próximo capítulo, a proposta de três indicadores para auxiliar no estudo da convergência são apresentados, além de dois algoritmos evolutivos multiobjetivo.



---

## Convergência e Ponto de Parada de AEMO's

No capítulo anterior foi discutida a importância do estudo da convergência de AEMO's, que está relacionado ao ponto de parada de um algoritmo. Assim, é essencial entender a evolução e o comportamento do algoritmo para atingir a solução e não somente observar a qualidade da solução final.

A proposta apresentada neste trabalho, que consiste de dois indicadores para estudo da evolução ou comportamento de AEMO's e um indicador para se observar a convergência a partir da estabilidade dos indivíduos no espaço de soluções, teve origem empírica, isto é, a partir de observações do comportamento do algoritmo DIPSO (descrito na seção 2.3.4). Para efetivar as conclusões dos experimentos iniciais, dois novos algoritmos foram projetados, SEDA e SEDASI, que se juntaram ao DIPSO. Assim como o anterior, esses novos algoritmos também foram concebidos para tratar o MOFJSP. Os experimentos com esses três algoritmos mostraram a necessidade de se propor novos indicadores para entender os seus comportamentos, principalmente o do DIPSO, no qual o OCD-HV não foi capaz de determinar o ponto de melhor resultado do algoritmo.

Assim, foram propostos dois novos indicadores que permitem deduzir, probabilisticamente, se o algoritmo possui condições de melhorar o conjunto de aproximação:

- Taxa de Concentração (*Concentration Rate*, CR): porcentagem de indivíduos da população que estão referenciadas no conjunto de aproximação;
- Taxa de Diversidade (*Diversity Rate*, DR): porcentagem de indivíduos gerados em determinada geração, em relação ao tamanho da população, que são posições do espaço de busca ainda desconhecidas pelo algoritmo.

A proposta desses indicadores permitiu identificar probabilisticamente que DIPSO ainda possuía condições de melhorar seu conjunto de aproximação no ponto de convergência apresentado por OCD-HV, ou seja, mostrou que o nível de atividade de DIPSO

após o ponto de convergência permitia identificar novas soluções não dominadas no espaço de busca.

Os experimentos realizados para analisar os AEMO's utilizando CR e DR identificaram que, se os indicadores permitem deduzir a possibilidade de um AEMO melhorar seu conjunto de aproximação, podem identificar um bom ponto de parada. Assim, baseado nesses resultados, foi proposto um algoritmo para identificar o ponto de parada de um AEMO utilizando CR e DR, o algoritmo de Parada por Falta de Atividade (PFA).

Paralelamente, com o intuito de melhorar o ponto de convergência identificado pelo OCD-HV, um novo indicador foi proposto. Trata-se do Indicador de Estabilidade no Espaço de Soluções (*Stability in Solution Space, S<sub>3</sub>*).

Portanto, na seção 3.1 deste capítulo apresentaremos os indicadores CR, DR,  $S_3$  e o algoritmo PFA, que tratam sobre convergência e ponto de parada de um AEMO. Na seção 3.2, os algoritmos SEDA e SEDASI são apresentados, que são algoritmos multiobjetivo propostos para tratar o MOFJSP.

## 3.1 Novos Indicadores

A necessidade de avaliar o comportamento de AEMO's foi identificada pelo experimento descrito na seção 4.2. Os resultados mostraram que a estabilidade identificada pelo OCD-HV no espaço objetivo pode não ser o melhor ponto para apresentar uma solução. Nesse sentido, indicadores que permitam identificar se um AEMO possui possibilidade de melhorar seu conjunto de aproximação são úteis. Para esse fim, foram propostos os indicadores Taxa de Concentração (*Concentration Rate, CR*) e Taxa de Diversidade (*Diversity Rate, DR*). Além disso, outra necessidade apresentada no estudo é a de possibilitar a identificação de um ponto de convergência mais próximo da melhor solução que o algoritmo consegue buscar. Nesse sentido, o indicador de Estabilidade no Espaço de Soluções (*Stability in Solution Space, S<sub>3</sub>*) foi proposto para que, associado ao OCD, possa atender tal finalidade. Nas seções seguintes, esses três indicadores, CR, DR e  $S_3$ , serão apresentados.

### 3.1.1 Taxa de Concentração

A movimentação realizada na população dentro do espaço de busca por técnicas como AE's e IE pode levar os indivíduos da população a se concentrarem em uma região restrita do espaço de busca. Esse comportamento dos algoritmos iterativos baseados em população aumenta a dificuldade em identificar novas posições no espaço de busca, pois quanto maior a concentração, menor a quantidade de informações disponíveis para se chegar em um ponto melhor do espaço de busca. Nesse sentido, identificar o nível de concentração da população, a cada geração, pode nos informar se tal algoritmo está em condições de

continuar a melhorar a solução. Assim, com base nessa ideia, a Taxa de Concentração (*Concentration Rate*, CR), para cada geração  $i$ , é definida pela Equação 15.

$$CR_i(S_i, P) = \frac{|S_i|}{|P|} \quad (15)$$

em que  $P$  é a população,  $S_i = \{x \in P | f(x) = (f_1(x), f_2(x), \dots, f_r(x)) \in A_i\}$ , sendo  $x$  uma solução,  $f(x)$  é a tupla que contém os valores de cada função objetivo  $f_i(x)$ ,  $1 \leq i \leq r$ ,  $A_i$  é o conjunto de aproximação e  $r$  é a dimensão do espaço objetivo. Daí,  $|P|$  é o tamanho da população e  $|S_i|$  é o número de soluções cujas tuplas de valores de funções objetivo estão no conjunto de aproximação  $A_i$ .

Analisando CR, tem-se que: 1) quanto maior o valor de CR, maior a quantidade de indivíduos no conjunto de aproximação; 2) Quanto menor o valor de CR, menor a concentração da população dentro do conjunto de soluções não dominadas; 3)  $CR_i = 1$  indica que toda a população está no conjunto de aproximação; 4) quanto maior o número de gerações, maior tende a ser o valor de CR. Entretanto, não se pode afirmar, de forma geral, que para um número grande de gerações, CR seja 1. Essa situação depende de outras condições no algoritmo como, por exemplo, elitismo e aleatoriedade baixa.

Para embasar essa análise de CR, a qual afirma que, quanto maior o CR, menor a probabilidade de um AEMO conseguir melhorar o conjunto de aproximação, as Tabelas 1 e 2 apresentam dados estatísticos de um experimento com DIPSO processando quinze vezes cada problema do *benchmark* de Kacem, Hammadi e Borne (2002). Nessas tabelas, cada coluna representa um intervalo de valores de CR, ou seja, a coluna  $[0, 0.1]$  apresenta informações sobre o experimento quando CR está no intervalo representado. Todos os números representados nos intervalos, que possuem casas decimais, serão apresentados utilizando o separador ‘ . ’ (ponto), pois a notação de intervalos utilizam a ‘ , ’ (vírgula) para separação dos valores. Os dados estatísticos apresentados são a relação da quantidade de incremento do conjunto de aproximação do DIPSO pela quantidade de gerações em que o algoritmo permaneceu com tais faixas de valores de CR. Nesse sentido, a informação do problema KC 4×5 para o intervalo  $[0, 0.1]$  de CR representa que, de todas as gerações em que DIPSO permaneceu com valores de CR nesse intervalo, houve melhora do conjunto de aproximação em 88,2% delas. Os números apresentados não são médias, ou seja, é um somatório de todas as execuções. O algoritmo foi configurado com oitocentas gerações em cada execução, portanto são um total de doze mil gerações. A contabilização desconsidera a primeira geração, pois é a geração em que o primeiro conjunto de aproximação é determinado. A Tabela 1 apresenta os dados em porcentagem e a Tabela 2 apresenta os dados em valores absolutos.

Os dados da Tabela 1 mostram a clara tendência de baixo incremento conforme o aumento de CR. No problema mais simples, KC 4×5, o momento de maior atividade do algoritmo se concentra com  $CR \leq 0,3$ . No problema mais complexo, KC 15×10, esse momento concentra-se com  $CR \leq 0,4$ . A relação (quantidade de incremento no

Tabela 1 – Dados Estatísticos em Porcentagem da Evolução do DIPSO - CR

| Problema | [0, 0.1] | (0.1, 0.2] | (0.2, 0.3] | (0.3, 0.4] | (0.4, 0.5] | (0.5, 0.6] | (0.6, 0.7] | (0.7, 0.8] | (0.8, 1] |
|----------|----------|------------|------------|------------|------------|------------|------------|------------|----------|
| KC 4x5   | 88,2%    | 42,5%      | 22,2%      | 1%         | 1,6%       | 1,2%       | 0,1%       | 0%         | -        |
| KC 8x8   | 89,9%    | 29,1%      | 3,3%       | 1,1%       | 0,8%       | 0,8%       | 0,9%       | 0,9%       | -        |
| KC 10x7  | 85,8%    | 36,8%      | 2,1%       | 1,4%       | 0,8%       | 0,4%       | 0%         | -          | -        |
| KC 10x10 | 84,1%    | 45,7%      | 21,7%      | 1,3%       | 0,6%       | 0,6%       | 0,5%       | 6,3%       | -        |
| KC 15x10 | 83,8%    | 41,9%      | 11,2%      | 7%         | 2,7%       | 0,7%       | 1,6%       | 0%         | -        |

Tabela 2 – Dados Estatísticos em Absoluto da Evolução do DIPSO - CR

| Problema | [0, 0.1]  | (0.1, 0.2] | (0.2, 0.3] | (0.3, 0.4] | (0.4, 0.5] | (0.5, 0.6] | (0.6, 0.7] | (0.7, 0.8] | (0.8, 1] |
|----------|-----------|------------|------------|------------|------------|------------|------------|------------|----------|
| KC 4x5   | 142/161   | 17/40      | 8/36       | 5/510      | 88/5505    | 56/4820    | 1/911      | 0/2        | 0/0      |
| KC 8x8   | 533/593   | 50/172     | 36/1079    | 36/3294    | 37/4430    | 12/1446    | 8/856      | 1/115      | 0/0      |
| KC 10x7  | 436/508   | 63/171     | 38/1778    | 48/3471    | 26/3384    | 7/1988     | 0/685      | 0/0        | 0/0      |
| KC 10x10 | 498/592   | 74/162     | 31/143     | 16/1216    | 35/6277    | 16/2817    | 4/762      | 1/16       | 0/0      |
| KC 15x10 | 1128/1346 | 235/561    | 147/1313   | 121/1738   | 90/3275    | 20/2788    | 15/953     | 0/11       | 0/0      |

conjunto de aproximação pela quantidade de gerações) apresentada na Tabela 2 mostra que há um volume grande de gerações no intervalo  $(0.2, 0.7]$ , observado pelas colunas  $(0.2, 0.3], \dots, (0.6, 0.7]$ , e o incremento no conjunto de aproximação tende a diminuir conforme CR aumenta. Portanto, nesse experimento, há evidências estatísticas de que conforme CR aumenta, o nível de atividade de DIPSO diminui.

### 3.1.2 Taxa de Diversidade

Quando iniciado, um algoritmo baseado em população distribui seus indivíduos pelo espaço de busca. Na geração seguinte, uma nova quantidade de posições no espaço de busca são determinadas e assim ocorre até que o algoritmo seja finalizado. Obviamente, na primeira geração todas as posições são novas (ainda não visitadas pelo algoritmo). Na segunda geração, não necessariamente e, quanto mais gerações forem efetivadas pelo algoritmo, maior a probabilidade de posições do espaço de busca se repetirem. E quanto mais posições repetidas forem acessadas, ou quanto menos posições novas forem buscadas, menor é a condição do algoritmo de melhorar a solução. Nesse contexto, a Taxa de Diversidade (*Diversity Rate*, DR) é definida como sendo a porcentagem, em relação à população, de posições novas acessadas em determinada geração. A Equação 16 define formalmente DR, para cada geração  $i$ .

$$DR_i(P_i) = \frac{|new(P_i, i)|}{|P_i|} \quad (16)$$

em que  $P_i$  é a população da geração  $i$  e  $new(P_i, i)$  é o conjunto das novas soluções encontradas em  $P_i$ , assim  $new(P_i, i) = P_i - P_k$ , com  $0 < k < i$  e, considerando  $P_i$  e  $P_k$  como

conjuntos,  $P_i - P_k$  é a diferença entre a população da geração atual e as populações das gerações anteriores, isto é, até a geração  $i - 1$ .

Conforme definição, temos: 1) quanto maior o valor de DR, maior a possibilidade do algoritmo encontrar soluções não dominadas; 2) quanto menor o valor de DR, menor a possibilidade do algoritmo encontrar uma solução não dominada; 3)  $DR_i(P_i) = 0$  indica que o algoritmo não acessou nenhuma posição nova do espaço de busca na geração  $i$ ; 4)  $DR_i(P_i) = 1$  indica que todas as posições do espaço de busca acessadas na geração  $i$  são novas. Em relação ao comportamento de DR na evolução do algoritmo, tem-se que: seja  $k$  o número de gerações executadas por um algoritmo e considerando um espaço de busca finito, podemos concluir que, para um  $k$  suficientemente grande,  $DR_k(P_k) = 0$  (Equação 17).

$$\lim_{k \rightarrow \infty} DR_k(P_k) = 0 \quad (17)$$

Para exemplificar as afirmações sobre DR, as Tabelas 3 e 4 são apresentadas, as quais são análogas às Tabelas 1 e 2 e foram elaboradas através do mesmo experimento. Os dados apresentados na Tabela 3 mostram a tendência de uma quantidade menor de incremento do conjunto de aproximação, com DR menores. No problema KC  $10 \times 7$  não há atividade com  $DR \leq 0,3$ , sendo que a maior concentração dos incrementos acontece com  $DR > 0,6$ . Generalizando, com  $DR \leq 0,2$  a atividade apresentada por DIPSO é pequena, sendo que houve incremento do conjunto de aproximação somente nos problemas KC  $4 \times 5$  e KC  $8 \times 8$ . A Tabela 4, mostra que a atividade de incremento do conjunto de aproximação, com  $DR \leq 0,2$ , foi considerável somente no problema KC  $4 \times 5$ , pois no problema KC  $8 \times 8$  foi somente um incremento em 157 gerações que foram executadas com DR nesse intervalo. Assim, a respectiva tabela mostra a queda de efetividade do algoritmo com valores menores de DR. Assim, podemos afirmar que, nesse experimento, DIPSO apresenta o comportamento de quanto menor o valor de DR, menor a probabilidade de melhorar o conjunto de aproximação.

Tabela 3 – Dados Estatísticos em Porcentagem da Evolução do DIPSO - DR

| Problema | [0, 0.1] | (0.1, 0.2] | (0.2, 0.3] | (0.3, 0.4] | (0.4, 0.5] | (0.5, 0.6] | (0.6, 0.7] | (0.7, 0.8] | (0.8, 0.9] | (0.9, 1] |
|----------|----------|------------|------------|------------|------------|------------|------------|------------|------------|----------|
| KC 4x5   | 0%       | 0,8%       | 1,7%       | 2,2%       | 7,4%       | 28,6%      | 61,8%      | 94,7%      | 100%       | 100%     |
| KC 8x8   | -        | 0,6%       | 0,9%       | 0,9%       | 0,4%       | 0,9%       | 2,7%       | 53,6%      | 90,2%      | 88,5%    |
| KC 10x7  | -        | 0%         | 0%         | 0,1%       | 0,4%       | 1%         | 1,7%       | 26,2%      | 87,9%      | 93,3%    |
| KC 10x10 | -        | 0%         | 0,4%       | 0,4%       | 0,3%       | 0,7%       | 2,9%       | 60,2%      | 82,8%      | 26,7%    |
| KC 15x10 | -        | -          | 0,9%       | 1,3%       | 1%         | 2%         | 6,5%       | 37%        | 74,1%      | 100%     |

### 3.1.3 Aplicação de CR e DR

Os indicadores CR e DR foram propostos para identificar o nível de atividade durante a evolução de um AEMO. Além disso, ambos podem determinar uma menor probabilidade

Tabela 4 – Dados Estatísticos em Absoluto da Evolução do DIPSO - DR

| Problema | [0, 0.1] | (0.1, 0.2] | (0.2, 0.3] | (0.3, 0.4] | (0.4, 0.5] | (0.5, 0.6] | (0.6, 0.7] | (0.7, 0.8] | (0.8, 0.9] | (0.9, 1] |
|----------|----------|------------|------------|------------|------------|------------|------------|------------|------------|----------|
| KC 4x5   | 0/274    | 46/5785    | 78/4673    | 21/946     | 6/81       | 14/49      | 34/55      | 72/76      | 31/31      | 15/15    |
| KC 8x8   | 0/0      | 1/157      | 15/1728    | 22/2522    | 6/1623     | 27/3093    | 51/1909    | 393/733    | 175/194    | 23/26    |
| KC 10x7  | 0/0      | 0/20       | 0/1169     | 1/1690     | 6/1363     | 30/3004    | 54/3255    | 331/1262   | 182/207    | 14/15    |
| KC 10x10 | 0/0      | 0/3        | 4/1116     | 11/2647    | 8/3008     | 21/3226    | 32/1091    | 354/588    | 241/291    | 4/15     |
| KC 15x10 | 0/0      | 0/0        | 6/633      | 33/2543    | 19/1852    | 42/2119    | 96/1484    | 927/2505   | 617/833    | 16/16    |

de um algoritmo melhorar sua solução e, nesse contexto, podem ser utilizados com o objetivo de identificar um ponto de parada para um AEMO.

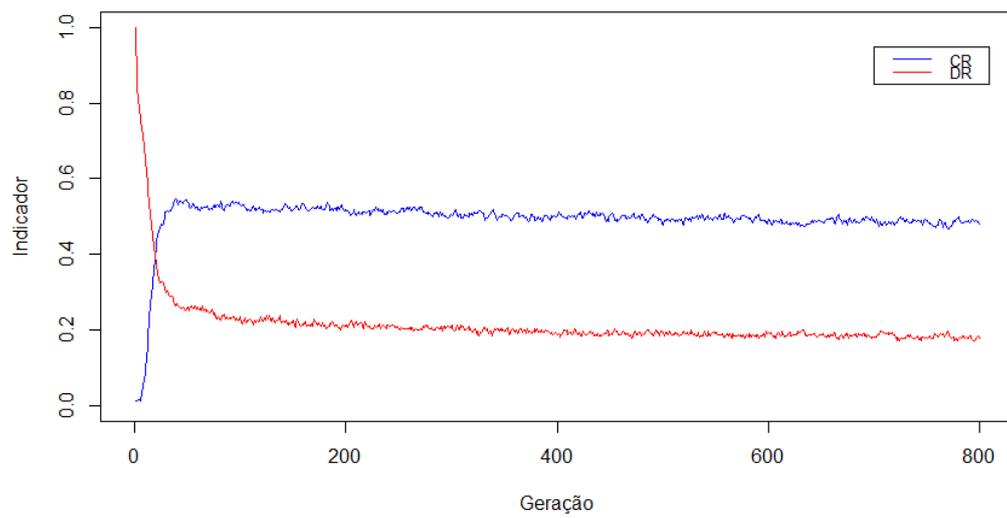
### 3.1.3.1 Observando o Nível de Atividade de um AEMO

Como exposto nas seções anteriores, CR se aproximando de 1 e DR de 0 indicaria que o algoritmo possui uma probabilidade baixa de melhorar seu conjunto de aproximação, mas não necessariamente pode acontecer. A explicação é devido ao fato de que há várias características que podem influenciar nesse comportamento. No caso de CR, como já citado, a falta de elitismo e, no caso de DR, um problema com espaço de busca muito grande e com um número insuficiente de gerações. Para esboçar o comportamento desses indicadores, os gráficos de CR e DR são apresentados na Figura 3 e foram construídos a partir das médias dos respectivos valores do experimento citado nas Seções 3.1.1 e 3.1.2. Em todos esses exemplos, CR não se aproxima de 1 e DR não se aproxima de 0.

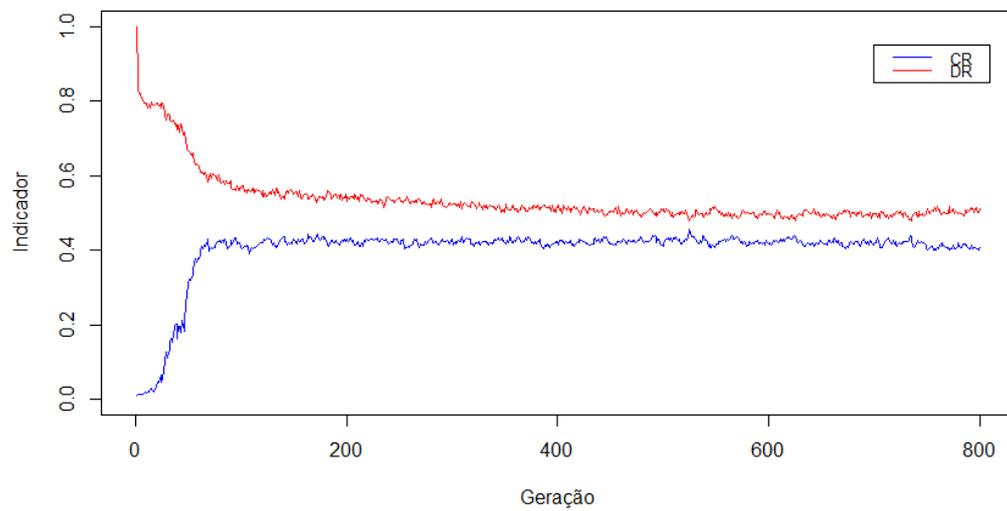
Apesar desse fato, CR e DR representam dois aspectos que caracterizam o nível de atividade de um AEMO. Então, ao entender que o algoritmo não chegará nos valores máximo, para o CR, e mínimo, para o DR, é indicado realizar um estudo prévio que estime o máximo e o mínimo atingido por tais indicadores. Sugere-se um experimento com execuções em que se observará CR e DR. Para cada execução realizada, coleta-se o maior CR e o menor DR. Ao fim de todas as execuções, determina-se  $CR_M$  e  $DR_m$  como sendo as médias de todos os respectivos valores coletados. A utilização da denominação  $CR_M$  faz referência à média do valor máximo de CR e  $DR_m$  à média do valor mínimo. Assim, entende-se que, para um dado algoritmo e um dado problema, tendo o número de gerações especificado,  $CR_i = x, 0 < x \leq CR_M$  e  $DR_i = y, DR_m \leq y \leq 1$ . Nesse sentido, ao realizar uma análise do algoritmo com base nos indicadores propostos, utiliza-se  $CR_M$  e  $DR_m$  para observar o quão próximo ele está do local com a menor probabilidade de evoluir a população.

### 3.1.3.2 Identificando um Ponto de Parada com CR e DR

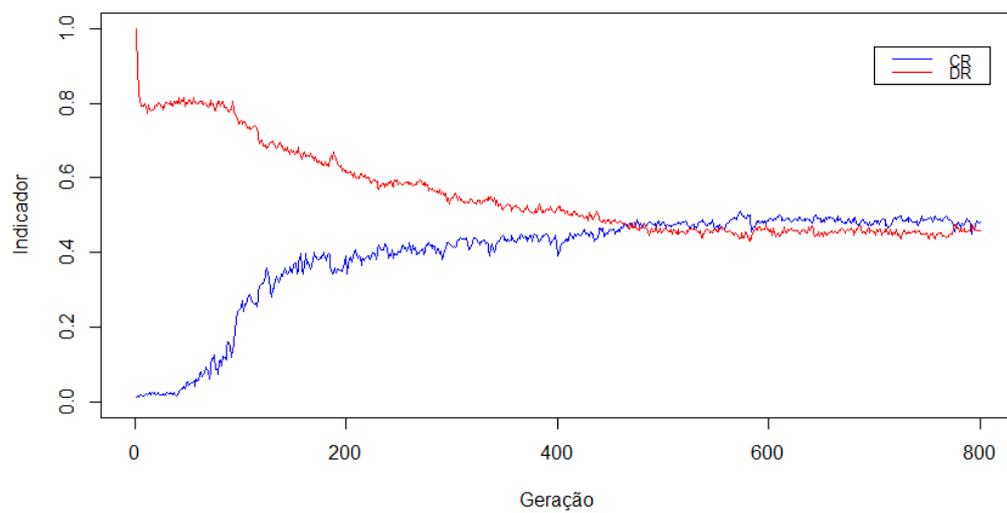
Conforme apresentado, CR e DR permitem estimar o nível de atividade de um AEMO. Nesse sentido, é razoável concluir que podem estimar um ponto de parada para um algoritmo, sendo que esse ponto tende a ser aquele de menor atividade. Para tal, o Algoritmo



(a) KC 4x5



(b) KC 10x7



(c) KC 15x10

Figura 3 – Gráficos de CR e DR - DIPSO

6, denominado de algoritmo de Parada por Falta de Atividade (PFA), é apresentado e seu objetivo é efetuar a parada de um AEMO quando CR e DR ultrapassar limites pré-determinados. PFA possui quatro parâmetros: 1) AEMO, que é o algoritmo a ser analisado; 2) erro, que possui valor padrão de 0,05 e tem a função de suavizar a regra de parada do algoritmo permitindo que valores próximos aos estabelecidos sejam considerados; 3)  $CR_M$ ; 4)  $DR_m$ . Os parâmetros 3 e 4 são identificados como apresentado na seção anterior e são os limites considerados para a análise.

---

### Algoritmo 6 PFA

---

**Require:** AEMO

```

     $CR_M$ 
     $DR_m$ 
    erro = 0,05
1:  $MAX\_LIMITE = 5$ 
2:  $CR'_M = CR_M - erro$ 
3:  $DR'_m = DR_m + erro$ 
4:  $cr_0 = 0$ 
5:  $dr_0 = DR'_m + 0,1$ 
6:  $gen = 0$ 
7:  $contadorParada = 0$ 
8: while  $gen++ < AEMO.MAX\_GEN \ \&\& \ contadorParada < MAX\_LIMITE$ 
   do
9:    $populacao = AEMO.executarGeracao()$ 
10:   $cr_{gen} = CR(populacao.S_{gen}, populacao)$ 
11:   $dr_{gen} = DR(populacao)$ 
12:  if  $cr_{gen} > CR'_M \ \&\& \ dr_{gen} < DR'_m$  then
13:     $contadorParada++$ 
14:  else
15:     $contadorParada = 0$ 
16:  end if
17: end while
18:  $AEMO.parar()$ 
19: return  $gen, cr_{gen}, dr_{gen}$ 

```

---

O algoritmo PFA, nas linhas de 1 a 7 inicializa as variáveis utilizadas no processo iterativo de análise a cada geração. Na primeira linha, a constante  $MAX\_LIMITE$  é definida com o valor 5 e indica a quantidade necessária de gerações consecutivas que os limites devem ser atingidos. Nas linhas 2 e 3,  $CR'_M$  e  $DR'_m$  são inicializados, sendo que são os valores dos parâmetros,  $CR_M$  e  $DR_m$ , considerando o parâmetro *erro* (subtraído no caso de  $CR_M$  e acrescido no caso de  $DR_m$ ). O objetivo dessa modificação é possibilitar a aproximação do algoritmo a esses valores, pois  $CR_M$  e  $DR_m$  são médias. Nas linhas 4 e 5,  $cr_0$  e  $dr_0$  são inicializados, de forma que não interfiram na análise da geração 0. Na linha 6, o contador de gerações  $gen$  é inicializado e, na linha 7, a variável  $contadorParada$  é inicializada com 0 e é utilizada para armazenar a quantidade de vezes que CR e DR ultrapassaram seus respectivos limites consecutivamente durante a análise. Nas linhas

de 8 a 17, uma geração do AEMO é executada (linha 8), os valores de  $cr$  e  $dr$  são calculados (linhas 9 e 10) e, na linha 12, é verificado se os valores calculados de  $cr$  e  $dr$  ultrapassaram seus limites simultaneamente,  $CR'_M$  (limite superior) e  $DR'_m$  (limite inferior), respectivamente. Caso ultrapassaram, *contadorParada* será incrementado e, caso não ultrapassaram, o *contadorParada* recebe o valor 0, para garantir a contagem consecutiva. Na linha 7, a condição de parada é testada, a qual monitora o número de gerações limite do AEMO e o número de vezes em que  $cr$  e  $dr$  ultrapassaram seus limites, de forma simultânea e consecutiva, através da variável *contadorParada*. Caso o limite de gerações ou o limite de *contadorParada* sejam atingidos, PFA determina a parada do algoritmo na linha 18 e retorna os valores dos indicadores CR e DR da última geração executada, junto com o número da respectiva geração.

Por fim, ressalta-se que PFA é baseado nos dados estatísticos que foram apresentados nas Tabelas 1, 2, 3 e 4, que mostram estatisticamente o baixo incremento do conjunto de aproximação, conforme CR aumenta e DR diminui.

### 3.1.4 Indicador de Estabilidade no Espaço de Soluções

O indicador de Estabilidade no Espaço de Soluções (*Stability in Solution Space*,  $S_3$ ) é calculado pela média da soma euclidiana das distâncias entre as soluções, ordenadas por algum critério, do conjunto de aproximação no espaço de soluções. A observação única de um valor de  $S_3$  nos mostra uma ideia de cobertura do espaço. Quanto maior seu valor mais soluções presentes no conjunto e mais espalhadas elas estão. Entretanto, ao medirmos em gerações consecutivas nos permite observar se há estabilidade no conjunto de aproximação, ou seja, se não estão ocorrendo alterações em tal conjunto. A Equação 18 apresenta o cálculo de  $S_3$ :

$$S_3(S'_i) = \frac{\sum_{k=1}^{|S'_i|} \sqrt{\sum_{y=0}^s (S'_i[k].y - S'_i[k-1].y)^2}}{|S_i|} \quad (18)$$

em que  $S_i = \{x | f(x) = (f_1(x), f_2(x), \dots, f_r(x)) \in A_i\}$ ,  $S'_i$  é o conjunto  $S_i$  ordenado por algum critério,  $r$  é a dimensão do espaço objetivo,  $i$  é a geração,  $s$  é a dimensão do espaço de soluções,  $f(x)$  é o vetor de objetivos formado por  $f_1(x), f_2(x), \dots, f_r(x)$ , as funções objetivo, e  $y$  é cada dimensão do espaço de soluções.

$S_3$  é utilizado no algoritmo OCD, para indicar a convergência através da estabilidade apresentada pela sequência dos valores desse indicador. Para exemplificar tal uso, a Figura 4 apresenta o gráfico do  $S_3$  no problema K  $4 \times 5$  executado por DIPSO no experimento citado nas seções 3.1.1 e 3.1.2. No gráfico, observa-se a estabilização com pequenas variações após algumas gerações. Isto ocorre, por causa que a diversidade no espaço de soluções é alta, de modo que a curva não apresente estabilidade em um único valor.

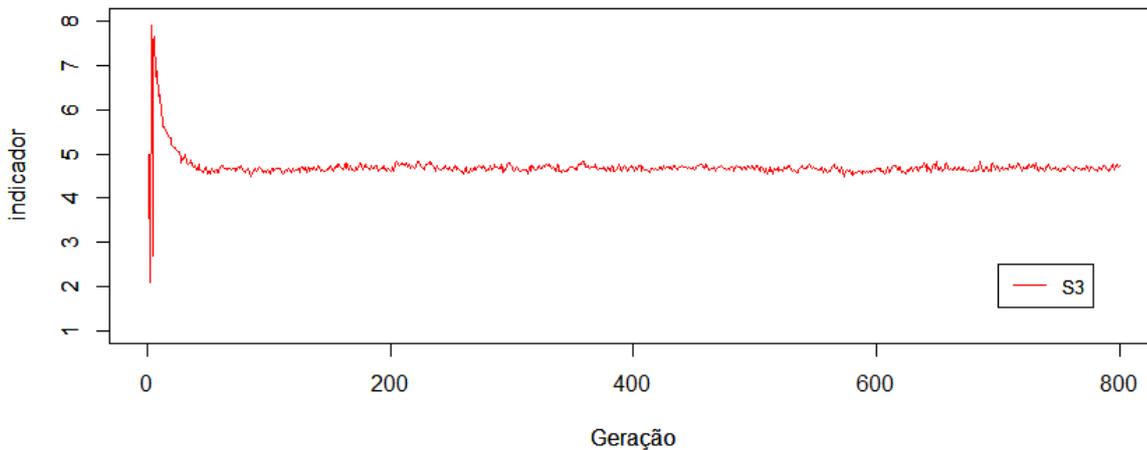


Figura 4 – Gráfico de  $S_3$  no problema KC  $4 \times 5$  - DIPSO

Porém, o OCD utiliza um teste estatístico (seção 2.4.1) para identificar a estabilidade, o qual aceita uma determinada variância.

## 3.2 Algoritmos para o FJSP

Nesta seção, os algoritmos implementados para agregar ao estudo da convergência são apresentados. O primeiro, denominado SEDA, é um EDA e o segundo, SEDASI, é um EDA híbrido com inteligência de enxame. Ambos os algoritmos são AEMO's e foram desenvolvidos para tratar o MOFJSP. A escolha do problema se justifica pela complexidade, pela possibilidade de aplicação no mundo real e pelo interesse em efetuar um quadro comparativo com DIPSO, algoritmo implementado para tal problema.

### 3.2.1 Representação do Indivíduo

Existem várias representações de um indivíduo para o FJSP com AE's. Cheng, Gen e Tsujimura (1996) mostraram algumas possibilidades para problemas de escalonamento usando um AG, sendo que essas representações podem ser usadas com o EDA ou com o PSO (aqui chamado de partícula) para tratar o FJSP (ou o MOFJSP). Gen, Tsujimura e Kubota (1994), Ling-li, Feng-Xing e Xiao-hong (2009), Xiao-hong, Ling-li e Yue-wen (2010), Wang et al. (2010) e Carvalho e Fernandes (2014) utilizaram a representação baseada em operações e devido à possibilidade de recuperação imediata do escalonamento e simplicidade de cálculo de aptidão, neste estudo, essa representação foi adotada.

Na representação baseada em operações, um indivíduo é composto de dois vetores com tamanho  $N$  (total de operações),  $X_{op}$  e  $X_{mach}$ .  $X_{op}$  contém a sequência de execução de cada  $job$ , para  $j = 1, \dots, n$ , e cada operação do  $job$  é representada pelo inteiro  $j$  que corresponde ao  $job$ . Assim, as operações  $n_j$  do  $j^{th}$   $job$  são representadas por  $n_j$  posições

em  $X_{op}$  com valor  $j$ . O primeiro  $j$  em  $X_{op}$  indica a primeira operação, e o último  $j$  indica a última operação do *job*  $j$ . Cada posição do vetor  $X_{mach}$  contém um valor  $k$ ,  $1 \leq k \leq m$ , que indica a máquina na qual a operação da posição correspondente a  $X_{op}$  é executada.

Na Figura 5, um exemplo dessa representação é apresentado para um problema com  $n = 6$  e  $m = 7$ , em que cada *job* possui duas operações.

|              |   |   |   |   |   |   |   |   |   |   |   |   |
|--------------|---|---|---|---|---|---|---|---|---|---|---|---|
| $X_{op}$ :   | 5 | 1 | 1 | 3 | 4 | 2 | 5 | 3 | 6 | 2 | 4 | 6 |
| $X_{mach}$ : | 7 | 3 | 1 | 3 | 4 | 2 | 5 | 2 | 4 | 2 | 5 | 6 |

Figura 5 – Representação do Indivíduo

### 3.2.2 SEDA: Um Simples Algoritmo de Estimativa de Distribuição

O Algoritmo de Estimativa de Distribuição Simples (*Simple Estimation of Distribution Algorithm*, SEDA) foi proposto para tratar o FJSP multiobjetivo (MOFJSP) considerando a minimização dos três objetivos, *makespan* (M), carga de trabalho total das máquinas (*Total Workload*, TW) e carga de trabalho máxima (*Workload*, W). O SEDA trata o MOFJSP em duas etapas. Na primeira, determina qual máquina é responsável por realizar cada operação e, na segunda, constrói uma ordem de execução das operações. Ambas as etapas são baseadas em uma matriz de probabilidade (MP). A construção da MP é baseada na contagem de ocorrências, utilizando a ideia básica do EDA, que determina a construção do modelo probabilístico através do número de aparições dos elementos em um conjunto de elite.

#### 3.2.2.1 Modelo Probabilístico

O modelo probabilístico do SEDA é formado pela MP, que contém as informações probabilísticas das associações entre operações e máquinas. O conjunto de indivíduos considerados na construção da MP é formado pelos indivíduos melhores avaliados (*Elitist Set*, ES) e o tamanho do conjunto é determinado pelo parâmetro Tamanho do Elitismo (*Elitism Amount*, EA).

A MP é formada contando a quantidade de aparições de cada associação formada pela operação do *job* e pela máquina, tal associação foi denominada de Operação por Máquina (OPM). Cada linha  $i$  da MP representa uma operação do *job* em específico e cada coluna  $j$  representa uma máquina. Assim, a posição  $i \times j$  de MP indica a quantidade de aparições que a  $op[i] \times mach[j]$  possui em ES, em que  $op[i]$  é a operação representada pela linha  $i$  e  $mach[j]$  é a máquina representada pela coluna  $j$ . Essa operação simples dá origem ao nome do algoritmo, já que a ideia da MP é baseada apenas no ato de contar.

Na Figura 6, um exemplo de ES é apresentado, sendo um vetor de 5 posições e o problema associado possui  $n = 3$  jobs e  $m = 5$  máquinas, em que cada job contém duas operações. Na Figura 7, um exemplo de uma MP é apresentado, se baseando no ES da Figura 6. Nesse exemplo, podemos observar que a linha 1 representa a operação 1 do job 1 (Op 1.1) e que houveram três associações de Op 1.1 com a máquina 1 (ES[0], ES[1], ES[2]), uma com a máquina 4 (ES[3]) e 5 (ES[4]) e nenhuma com as demais.

### 3.2.2.2 Mecanismo de Atualização

Em cada processo de atualização da população do algoritmo SEDA, todos os indivíduos que não estão presentes em ES são substituídos. Portanto, a cada geração do SEDA, são gerados a quantidade de  $POP - EA$  indivíduos para atualizar a população, em que  $POP$  é o parâmetro que indica o tamanho da população. A geração de novos indivíduos ocorre em duas etapas: associar cada operação do job a uma máquina e sequenciar as operações.

Na primeira etapa, o SEDA utiliza uma roleta para realizar a atribuição das máquinas às operações dos jobs e construir as associações OPM. O processo usa a MP para montar as probabilidades de escolha para a roleta. Por exemplo, usando as probabilidades da MP da Figura 7, a possibilidade da Op. 1.1 ser associada à máquina 1 seria 0,6, para ser associada à máquina 4 ou 5 seria 0,2 e para ser associada à máquina 2 ou 3 seria 0. Essas probabilidades são usadas na roleta e determinam as OPM's de maneira probabilística. Esse processo termina quando todas as operações dos jobs tiverem uma máquina associada.

|           |              |   |   |   |   |   |   |           |              |   |   |   |   |   |   |
|-----------|--------------|---|---|---|---|---|---|-----------|--------------|---|---|---|---|---|---|
| ES[ 0 ] = | $X_{op}$ :   | 1 | 1 | 3 | 2 | 3 | 2 | ES[ 3 ] = | $X_{op}$ :   | 3 | 1 | 1 | 2 | 2 | 3 |
|           | $X_{mach}$ : | 1 | 2 | 2 | 1 | 1 | 3 |           | $X_{mach}$ : | 3 | 4 | 2 | 2 | 5 | 1 |
| ES[ 1 ] = | $X_{op}$ :   | 2 | 3 | 1 | 2 | 1 | 3 | ES[ 4 ] = | $X_{op}$ :   | 1 | 3 | 1 | 3 | 2 | 2 |
|           | $X_{mach}$ : | 4 | 3 | 1 | 5 | 2 | 1 |           | $X_{mach}$ : | 5 | 3 | 2 | 1 | 5 | 3 |
| ES[ 2 ] = | $X_{op}$ :   | 3 | 1 | 1 | 2 | 3 | 2 |           |              |   |   |   |   |   |   |
|           | $X_{mach}$ : | 1 | 1 | 2 | 3 | 1 | 3 |           |              |   |   |   |   |   |   |

Figura 6 – Exemplo de ES para gerar uma MP

| Máquina | 1 | 2 | 3 | 4 | 5 |
|---------|---|---|---|---|---|
| Op 1.1  | 3 | 0 | 0 | 1 | 1 |
| Op 1.2  | 0 | 5 | 0 | 0 | 0 |
| Op 2.1  | 1 | 1 | 1 | 1 | 1 |
| Op 2.2  | 0 | 0 | 3 | 0 | 2 |
| Op 3.1  | 1 | 1 | 3 | 0 | 0 |
| Op 3.2  | 5 | 0 | 0 | 0 | 0 |

Figura 7 – Exemplo de uma MP

ada, criando um indivíduo temporário (IT) sem nenhuma ordem definida nas operações dos *jobs*. Para executar a segunda etapa, o Algoritmo 7 é utilizado.

---

**Algoritmo 7** Ordenar Operações
 

---

**Require:** *IT*

*MP*

*ES*

1: **repeat**

2:  $E\_OPM = \text{roleta}(IT.OPMs, \text{probabilidades}(MP, IT.OPMs))$

3:  $SES = \text{subConjunto}(ES, E\_OPM)$

4:  $E\_SES = \text{sortea}(SES)$

5:  $p = \text{posicao}(E\_SES.X_{op}, E\_OPM)$

6:  $\text{ordenar}(E\_SES.X_{op}, p, IT.X_{op});$

7: **until**  $\text{fimOrdenacao}(IT)$

8: **return** *IT* /\* ordenado \*/

---

As linhas de 1 a 7 são executadas até a ordenação finalizar. Na linha 2, uma roleta é construída e o sorteio probabilístico é realizado para selecionar uma *OPM* de *IT*. As probabilidades utilizadas para construir a roleta são aquelas definidas em *MP*. A *OPM* sorteada é armazenada em *E\_OPM*. Na linha 3, um subconjunto de *ES* é formado com todos indivíduos que possuem a associação *E\_OPM* e atribuído a *SES* e então, na linha 4, uma associação de *SES* é sorteada (*E\_SES*). As linhas 5 e 6 são para efetivar parcialmente a ordenação de *IT*. Primeiro, na linha 5, a posição da operação da associação sorteada no passo anterior *E\_OPM* em *E\_SES.X<sub>op</sub>* é identificada e, na linha seguinte, a ordenação de todas as operações com posição igual ou menor do que *p* é copiada de *E\_SES.X<sub>op</sub>* para *IT.X<sub>op</sub>*. Esse processo de ordenação das operações não altera as associações definidas anteriormente. A linha 7 verifica se a ordenação de *IT* está finalizada, caso não estiver, as linhas de 1 a 7 são executadas novamente. As novas iterações não consideram as operações já ordenadas, ou seja, a repetição desses passos não altera a ordenação de iterações anteriores.

As Figuras 8, 9, 10, 11, 12, 13, 14, 15 e 16 apresentam um exemplo de ordenação de um indivíduo, sendo que a primeira iteração é representada pelas Figuras 8, 9, 10, 11 e 12. A segunda iteração é representada pelas Figuras 13, 14, 15 e 16. Nesse exemplo, para indicar uma operação será utilizado o modelo  $\langle \text{job} \rangle . \langle \text{operação} \rangle$ , assim, a operação 2.3 é a operação 3 do *job* 2. Na Figura 8, os parâmetros do algoritmo são apresentados: 1) o indivíduo *IT* com as associações entre operações e máquinas, *OPMs*, já realizadas; 2) a matriz de probabilidade, *MP*, definida pelo exemplo da Figura 7; e, 3) o conjunto elitista, *ES*, que deu origem à *MP*.

No primeiro passo, Figura 9, a linha 2 do algoritmo é executada. Assim, uma roleta é criada para selecionar uma associação *OPM*, com as probabilidades, definidas pela *MP*, de cada associação do indivíduo *IT*. Todas as associações estão disponíveis para seleção, pois, nenhuma delas foi ordenada. Portanto, a operação 1.1, que está associada

à máquina 1 (posição 0 dos vetores  $X_{op}$  e  $X_{mach}$  de  $IT.OPMs$ ), possui o valor 3, linha 1 e coluna 1 da  $MP$ ; a operação 1.2, que está associada à máquina 2 (posição 1 dos vetores  $X_{op}$  e  $X_{mach}$  de  $IT.OPMs$ ), possui o valor 5, linha 2 e coluna 2 da  $MP$ . Os outros valores para formação da roleta seguem a mesma lógica. Identificado cada valor para formação da roleta, as probabilidades de seleção de cada associação é apresentada. Dessa forma, a operação 1.1 possui a probabilidade 0,15 de ser selecionada e a operação 1.2 possui a probabilidade 0,25. Efetuando a execução da roleta, a associação escolhida foi a da posição 1 de  $IT.OPMs$ , que está em negrito na figura. Assim,  $E\_OPM$  receberá a  $IT.OPMs[1]$ , que é a operação 1.2.

O segundo passo, Figura 10, apresenta a execução da linha 3 do algoritmo. Nessa linha, o conjunto  $SES$  é definido com todos os indivíduos do conjunto elitista  $ES$  que possuem a associação  $E\_OPM$ . Dessa forma, na figura é apresentado o conjunto  $SES$  com os indivíduos que possuem a associação  $E\_OPM$ , definida no passo anterior, que está destacada em negrito em todos os indivíduos.

O terceiro passo, Figura 11, apresenta a execução das linhas 4 e 5 do algoritmo. Na linha 4, um sorteio aleatório é realizado para escolher um indivíduo do conjunto  $SES$ . Portanto, na figura é apresentado o resultado de um sorteio hipotético para definir a escolha do indivíduo de  $SES$ , o atribuindo à variável  $E\_SES$ , sendo que tal sorteio resultou no indivíduo de índice 4. Ainda nesse passo, a identificação do índice da associação  $E\_OPM$  em  $E\_SES$  é realizada e atribuída a  $p$  (linha 5 do algoritmo). No exemplo, a associação  $E\_OPM$  está em negrito e está na posição 2, portanto  $p = 2$ .

O último passo dessa iteração, linha 6 do algoritmo, é apresentado na Figura 12, em que é realizada a cópia da ordenação das operações de  $E\_OPM$  para  $IT$ , do índice 0 até o índice  $p = 2$ . As operações ordenadas em tais índices de  $E\_OPM$  são: 1.1 (índice 0), 3.1 (índice 1) e 1.2 (índice 2). Como nenhuma operação foi ordenada em  $IT$ , toda

$$IT.OPMs = \begin{array}{l} X_{op}: \\ X_{mach}: \end{array} \begin{array}{|c|c|c|c|c|c|} \hline 1 & 1 & 2 & 2 & 3 & 3 \\ \hline 1 & 2 & 4 & 3 & 3 & 1 \\ \hline \end{array}$$

MP = exemplo da Figura 7

$$\begin{array}{l} ES[0] = \\ ES[1] = \\ ES[2] = \end{array} \begin{array}{l} X_{op}: \\ X_{mach}: \\ X_{op}: \\ X_{mach}: \\ X_{op}: \\ X_{mach}: \end{array} \begin{array}{|c|c|c|c|c|c|} \hline 1 & 1 & 3 & 2 & 3 & 2 \\ \hline 1 & 2 & 2 & 1 & 1 & 3 \\ \hline 2 & 3 & 1 & 2 & 1 & 3 \\ \hline 4 & 3 & 1 & 5 & 2 & 1 \\ \hline 3 & 1 & 1 & 2 & 3 & 2 \\ \hline 1 & 1 & 2 & 3 & 1 & 3 \\ \hline \end{array} \begin{array}{l} ES[3] = \\ ES[4] = \end{array} \begin{array}{l} X_{op}: \\ X_{mach}: \\ X_{op}: \\ X_{mach}: \end{array} \begin{array}{|c|c|c|c|c|c|} \hline 3 & 1 & 1 & 2 & 2 & 3 \\ \hline 3 & 4 & 2 & 2 & 5 & 1 \\ \hline 1 & 3 & 1 & 3 & 2 & 2 \\ \hline 5 & 3 & 2 & 1 & 5 & 3 \\ \hline \end{array}$$

Figura 8 – Exemplo de ordenação de um indivíduo - Parâmetros

|           |              |   |          |   |   |   |   |
|-----------|--------------|---|----------|---|---|---|---|
| IT.OPMs = | $X_{op}$ :   | 1 | <b>1</b> | 2 | 2 | 3 | 3 |
|           | $X_{mach}$ : | 1 | <b>2</b> | 4 | 3 | 3 | 1 |
|           | Ordenado?    | N | N        | N | N | N | N |

|          |                 |      |      |      |      |      |      |
|----------|-----------------|------|------|------|------|------|------|
| Roleta = | Contagem:       | 3    | 5    | 1    | 3    | 3    | 5    |
|          | Probabilidades: | 0,15 | 0,25 | 0,05 | 0,15 | 0,15 | 0,25 |

$$E\_OPM = IT.OPMs[ 1 ] = \text{Operação 1.2, Máquina 2}$$

Figura 9 – Exemplo de ordenação de um indivíduo - Iteração 1, Passo 1

|            |              |   |          |   |   |   |   |
|------------|--------------|---|----------|---|---|---|---|
| SES[ 0 ] = | $X_{op}$ :   | 1 | <b>1</b> | 3 | 2 | 3 | 2 |
|            | $X_{mach}$ : | 1 | <b>2</b> | 2 | 1 | 1 | 3 |

|              |   |   |          |   |   |   |
|--------------|---|---|----------|---|---|---|
| $X_{op}$ :   | 3 | 1 | <b>1</b> | 2 | 2 | 3 |
| $X_{mach}$ : | 3 | 4 | <b>2</b> | 2 | 5 | 1 |

|              |   |   |   |   |          |   |
|--------------|---|---|---|---|----------|---|
| $X_{op}$ :   | 2 | 3 | 1 | 2 | <b>1</b> | 3 |
| $X_{mach}$ : | 4 | 3 | 1 | 5 | <b>2</b> | 1 |

|              |   |   |          |   |   |   |
|--------------|---|---|----------|---|---|---|
| $X_{op}$ :   | 1 | 3 | <b>1</b> | 3 | 2 | 2 |
| $X_{mach}$ : | 5 | 3 | <b>2</b> | 1 | 5 | 3 |

|              |   |   |          |   |   |   |
|--------------|---|---|----------|---|---|---|
| $X_{op}$ :   | 3 | 1 | <b>1</b> | 2 | 3 | 2 |
| $X_{mach}$ : | 1 | 1 | <b>2</b> | 3 | 1 | 3 |

Figura 10 – Exemplo de ordenação de um indivíduo - Iteração 1, Passo 2

$$E\_SES = SES[ 4 ] = \begin{array}{l} X_{op}: \\ X_{mach}: \\ p = 2 \end{array} \begin{array}{|c|c|c|c|c|c|} \hline 1 & 3 & \mathbf{1} & 3 & 2 & 2 \\ \hline 5 & 3 & \mathbf{2} & 1 & 5 & 3 \\ \hline \end{array}$$

Figura 11 – Exemplo de ordenação de um indivíduo - Iteração 1, Passo 3

a ordenação é copiada, como pode ser visto no exemplo apresentado pela figura, com destaque em negrito.

|           |              |          |          |          |   |   |   |
|-----------|--------------|----------|----------|----------|---|---|---|
| IT.OPMs = | $X_{op}$ :   | <b>1</b> | <b>3</b> | <b>1</b> | 2 | 2 | 3 |
|           | $X_{mach}$ : | <b>1</b> | <b>3</b> | <b>2</b> | 4 | 3 | 1 |
|           | Ordenado?    | <b>S</b> | <b>S</b> | <b>S</b> | N | N | N |

Figura 12 – Exemplo de ordenação de um indivíduo - Iteração 1, Passo 4

O exemplo continua na Figura 13, que apresenta o primeiro passo da segunda iteração, linha 2 do Algoritmo 7. Dessa forma, uma roleta é criada, como na iteração anterior, porém considerando apenas as associações de IT.OPMs que ainda não foram ordenadas, índices 3, 4 e 5. A execução hipotética da roleta resulta no valor 4 e, portanto,  $E\_OPM =$

|           |                 |   |   |   |      |          |      |
|-----------|-----------------|---|---|---|------|----------|------|
| IT.OPMs = | $X_{op}$ :      | 1 | 3 | 1 | 2    | <b>2</b> | 3    |
|           | $X_{mach}$ :    | 1 | 3 | 2 | 4    | <b>3</b> | 1    |
|           | Ordenado?       | S | S | S | N    | N        | N    |
|           |                 |   |   |   |      |          |      |
| Roleta =  | Contagem:       | - | - | - | 1    | 3        | 5    |
|           | Probabilidades: | - | - | - | 0,11 | 0,33     | 0,56 |

$$E\_OPM = IT.OPMs[4] = \text{Operação 2.2, Máquina 3}$$

Figura 13 – Exemplo de ordenação de um indivíduo - Iteração 2, Passo 1

$IT.OPMs[4]$ , operação 2.2 associada à máquina 3, em negrito na figura.

No passo 2, Figura 14, o conjunto  $SES$  é definido com os elementos de  $ES$  que possuem a  $E\_OPM$  definida no passo anterior (linha 3 do algoritmo).

|            |              |   |   |   |   |   |          |            |              |   |   |   |   |   |          |
|------------|--------------|---|---|---|---|---|----------|------------|--------------|---|---|---|---|---|----------|
| SES[ 0 ] = | $X_{op}$ :   | 1 | 1 | 3 | 2 | 3 | <b>2</b> | SES[ 2 ] = | $X_{op}$ :   | 1 | 3 | 1 | 3 | 2 | <b>2</b> |
|            | $X_{mach}$ : | 1 | 2 | 2 | 1 | 1 | <b>3</b> |            | $X_{mach}$ : | 5 | 3 | 2 | 1 | 5 | <b>3</b> |
|            |              |   |   |   |   |   |          |            |              |   |   |   |   |   |          |
| SES[ 1 ] = | $X_{op}$ :   | 3 | 1 | 1 | 2 | 3 | <b>2</b> |            |              |   |   |   |   |   |          |
|            | $X_{mach}$ : | 1 | 1 | 2 | 3 | 1 | <b>3</b> |            |              |   |   |   |   |   |          |

Figura 14 – Exemplo de ordenação de um indivíduo - Iteração 2, Passo 2

No passo 3, Figura 15, o sorteio de um indivíduo de  $SES$  é realizado, resultando no indivíduo  $SES[0]$ , sendo esse atribuído a  $E\_SES$  (linha 4 do algoritmo). Como a associação  $E\_OPM$  está na posição 5 de  $E\_SES$ ,  $p = 5$  (linha 5 do algoritmo).

|                    |              |   |   |   |   |   |          |
|--------------------|--------------|---|---|---|---|---|----------|
| E_SES = SES[ 0 ] = | $X_{op}$ :   | 1 | 1 | 3 | 2 | 3 | <b>2</b> |
|                    | $X_{mach}$ : | 1 | 2 | 2 | 1 | 1 | <b>3</b> |
| p = 5              |              |   |   |   |   |   |          |

Figura 15 – Exemplo de ordenação de um indivíduo - Iteração 2, Passo 3

O passo 4, Figura 16, apresenta a cópia da ordenação de  $E\_SES$  para  $IT$  das operações que estão entre os índices 0 e  $p = 5$ , desconsiderando àquelas que já foram ordenadas. Desse modo, as operações 2.1, 3.2 e 2.2 são ordenadas em  $IT$ , como pode ser visto em destaque de negrito na figura. Com a execução dessa iteração, todas as operações foram ordenadas e o processo se encerra.

### 3.2.2.3 SEDA - Pseudocódigo

Aplicando os procedimentos apresentados anteriormente ao EDA formamos o SEDA, apresentado no Algoritmo 8. São parâmetros para o algoritmo, o tamanho da população

|           |              |   |   |   |          |          |          |
|-----------|--------------|---|---|---|----------|----------|----------|
| IT.OPMs = | $X_{op}$ :   | 1 | 3 | 1 | <b>2</b> | <b>3</b> | <b>2</b> |
|           | $X_{mach}$ : | 1 | 3 | 2 | <b>4</b> | <b>1</b> | <b>3</b> |
|           | Ordenado?    | S | S | S | <b>S</b> | <b>S</b> | <b>S</b> |

Figura 16 – Exemplo de ordenação de um indivíduo - Iteração 2, Passo 4

(*POP*), o número máximo de gerações (*MAX\_GEN*) e o tamanho do conjunto de elite *ES* (*EA*).

---

### Algoritmo 8 SEDA

---

**Require:** *POP*

*MAX\_GEN*

*EA*

- 1: *populacao* = inicializarPopulacao( *POP* )
  - 2: *gen* = 0
  - 3: **while** *gen*++ < *MAX\_GEN* **do**
  - 4:   calcularAptidao( *populacao* )
  - 5:   *fronts*[] = classificarFronteiras( *populacao* )
  - 6:   *ES* = selecionarMelhores( *fronts*, *EA* )
  - 7:   *MP* = construirMP( *ES* )
  - 8:   *novosInds* = gerarIndividuos( *MP*, *POP* – *EA* )
  - 9:   *populacao* = *ES* + *novosInds*
  - 10: **end while**
  - 11: calcularAptidao( *populacao* )
  - 12: **return** classificarFronteiras( *populacao* )[ 1 ]
- 

As primeiras linhas de SEDA são para inicializar a população de tamanho *POP* aleatoriamente (linha 1) e o contador de gerações *gen* (linha 2). Entre as linhas 3 e 10, o processo iterativo de evolução do algoritmo é definido. O primeiro método executado nesse processo, linha 4, é para fazer o cálculo da aptidão. Como SEDA foi construído para tratar o MOFJSP, esse cálculo é realizado por meio das Equações 2, 3 e 4, referentes aos objetivos considerados no problema. No passo 5, a população é classificada em fronteiras usando o conceito de não dominância de Pareto e, no passo 6, o conjunto de elite *ES* é selecionado na população, para que, no passo 7, o modelo probabilístico seja construído. A seleção dos melhores elementos é realizada através das fronteiras (*fronts*) definidas no passo 5. Quando não for possível selecionar todos os elementos de uma fronteira para formar o *ES*, a seleção será feita aleatoriamente. Isto pode ocorrer devido ao tamanho do *ES*, que é limitado pelo parâmetro *EA*. No passo 8, novos indivíduos são criados através do mecanismo de atualização, definido anteriormente, que faz uso do modelo probabilístico (MP). Por fim, no passo 9, a população é atualizada com a nova formação que contempla o conjunto *ES* e os novos indivíduos, ou seja, os indivíduos que não foram selecionados para *ES* são substituídos. Esse processo iterativo é realizado até

que o número de gerações, definido pelo parâmetro  $MAX\_GEN$ , seja atingido. Na linha 11, a aptidão é calculada novamente para que, linha 12, SEDA retorne a fronteira 1 da população recém atualizada.

#### 3.2.2.4 SEDA com Inteligência de Enxame

A definição de inteligência de enxame (IE) é baseada no compartilhamento do conhecimento entre os indivíduos de uma população para a resolução de um problema (Seção 2.3). Uma possibilidade de implementação da IE é o modelo global ( $g_{Best}$ ), quando uma partícula (ou indivíduo) acessa informações de todos os vizinhos. Portanto, a partir do conhecimento de cada partícula do enxame, o melhor global é identificado e todas as partículas tem conhecimento dessa informação. Esse é o modelo adotado pelo DIPSO, o qual armazena um conjunto de melhores posições conhecidas pelo enxame, ao invés de uma, e utiliza essas informações para atualizar a posição de cada partícula. Fazendo uma analogia com o SEDA, temos que esse utiliza informações de um conjunto elitista para realizar a descoberta de uma nova posição. Assim, pode-se contemplar mais informações do que as melhores posições conhecidas pelo algoritmo, pois o tamanho de tal conjunto é determinado por um parâmetro. Além disso, os novos indivíduos que são criados substituem apenas os indivíduos que não fazem parte do conjunto elitista, enquanto no DIPSO todas as partículas atualizam a posição.

Nesse sentido, o processo evolutivo de SEDA foi alterado baseando-se no do DIPSO. A proposta foi implementar a IE no SEDA, criando então o Algoritmo de Estimativa de Distribuição Simples com Inteligência de Enxame (*Simple Estimation of Distribution Algorithm with Swarm Intelligence*, SEDASI). As alterações aplicadas ao SEDA foram trocar o conjunto elitista ES pelo conhecimento global da população mesclado com o conhecimento específico de cada indivíduo e determinar que todos os indivíduos devem ser substituídos no processo de atualização de posição da população. Assim, os conceitos de DIPSO representados por  $g_{Best}$ , conhecimento global, e  $p_{Best}$ , conhecimento local, foram adaptados para serem aplicados na formação do modelo probabilístico, transformando o SEDA em uma técnica híbrida, o SEDASI.

#### 3.2.2.5 SEDASI - Pseudocódigo

Aplicando as alterações descritas, SEDASI é apresentado no Algoritmo 9. Como não há mais o conjunto elitista  $ES$ , o parâmetro  $EA$  deixa de ser necessário. Assim, os parâmetros para o algoritmo se resumem ao tamanho da população ( $POP$ ) e ao número máximo de gerações ( $MAX\_GEN$ ).

Assim como no SEDA, nas primeiras linhas do SEDASI, a população e o contador de gerações são inicializados. Além disso,  $g_{Best}$  é inicializado como um conjunto vazio. As linhas de 4 a 12 definem a evolução do algoritmo. Os dois primeiros passos são o cálculo da aptidão e a classificação da população em fronteiras. Nas linhas 7 e 8,

**Algoritmo 9** SEDASI**Require:**  $POP$  $MAX\_GEN$ 


---

```

1:  $populacao = inicializarPopulacao( POP )$ 
2:  $g_{Best} = \{ \}$ 
3:  $gen = 0$ 
4: while  $gen++ < MAX\_GEN$  do
5:    $calcularAptidao( populacao )$ 
6:    $fronts[] = classificarFronteiras( populacao )$ 
7:    $g_{Best} = classificarFronteiras( fronts[1], g_{Best} )[ 1 ]$ 
8:    $atualizar(populacao.p_{Best}[0 \leq i < POP])$ 
9:    $MP = construirMP( g_{Best}, populacao.p_{Best}[0 \leq i < POP] )$ 
10:   $novosInds = gerarIndividuos( MP, POP )$ 
11:   $populacao = novosInds$ 
12: end while
13: return  $classificarFronteiras( populacao )[ 1 ]$ 

```

---

$g_{Best}$  e  $p_{Best}$  são atualizados. O conjunto  $g_{Best}$  recebe o valor do conjunto de soluções não dominadas entre o conjunto fronteira 1, calculado no passo anterior, e o conjunto  $g_{Best}$  atual. Os elementos  $p_{Best}$  de cada indivíduo é determinado como sendo o indivíduo melhor avaliado, comparando o valor atual do  $p_{Best}$  e o indivíduo. O próximo passo é construir o modelo probabilístico MP utilizando  $g_{Best}$  e o  $p_{Best}$  de cada indivíduo e criar os novos indivíduos através da MP. Nesse passo são criados indivíduos em quantidade para substituir a população, o que acontece na linha 11. Assim como em SEDA, esses passos acontecem até que  $MAX\_GEN$  seja alcançado e o conjunto fronteira 1 da população é retornado.

### 3.3 Considerações Finais

Nesta seção foram apresentados os indicadores CR e DR, além de um método para identificar um ponto de parada para AEMO's, denominado PFA, que utiliza esses indicadores. Um outro indicador,  $S_3$ , foi apresentado para ser utilizado com o OCD, de forma que o ponto de convergência de um AEMO seja determinado observando o espaço de soluções. Além desses, dois AEMO's foram propostos, SEDA e SEDASI, para tratar o problema *Job Shop* Flexível. No próximo capítulo, os experimentos realizados para verificar a efetividade dos indicadores e dos AEMO's são apresentados e seus resultados discutidos.



---

## Experimentos e Análise dos Resultados

Os experimentos que serão apresentados neste capítulo visam embasar a proposta dos indicadores, CR, DR e  $S_3$ , apresentados no capítulo anterior, além de verificar a eficiência desses e dos algoritmos propostos, SEDA e SEDASI. Assim, de acordo com a finalidade, foram realizados experimentos para:

- Verificação da eficiência dos algoritmos: experimentos realizados com SEDA e SEDASI tratando o FJSP a fim de comparar seus resultados com aqueles da literatura;
- Embasar a proposta dos indicadores: experimento em que se identificou a necessidade de novos indicadores;
- Verificação da eficiência dos indicadores no FJSP: experimentos em que se aplicou os indicadores propostos para se identificar um ponto de parada para os AEMO's DIPSO, SEDA, SEDASI, NSGA2 e SPEA2 tratando o FJSP;
- Verificação da eficiência dos indicadores em problemas e algoritmos clássicos: experimento análogo ao anterior, porém utilizando algoritmos multiobjetivo clássicos, NSGA2 e SPEA2, com um *benchmark* clássico da literatura, o ZDT (sigla que referencia os nomes dos autores: Zitzler, Deb e Thiele) (Zitzler; Deb; Thiele, 2000).

Assim, a seção 4.1 apresenta o experimento a respeito dos algoritmos desenvolvidos, SEDA e SEDASI. O experimento que motivou a criação dos indicadores é descrito na seção 4.2. Na seção 4.3, os experimentos para se identificar um ponto de parada para AEMO's utilizando o FJSP com DIPSO, SEDA, SEDASI, NSGA2 e SPEA2 são apresentados e a seção 4.4 descreve o experimento para identificar um ponto de parada para o NSGA2 e o SPEA2, algoritmos clássicos da literatura, utilizando o *benchmark* ZDT.

### 4.1 SEDA e SEDASI

Os experimentos desta seção são dedicados a validar os resultados de SEDA e SEDASI e, a fim de analisar a eficiência desses algoritmos, seus resultados são comparados

com algoritmos já conhecidos pela literatura. Os experimentos foram realizados em um computador pessoal com processador Intel Core I7 (2GHz) e 8 GB de memória RAM, sendo que os resultados apresentados são o conjunto de soluções não dominadas de todas as soluções de trinta execuções independentes de cada problema.

#### 4.1.1 SEDA - Experimentos e Análise dos Resultados

Para a realização dos experimentos com o algoritmo SEDA, os *benchmarks* de Kacem, Hammadi e Borne (2002) (KC) e Brandimarte (1993) (BR) foram utilizados. Juntos, constituem um conjunto de quinze problemas. As instâncias dos problemas KC são referidas como  $n \times m$ , em que  $n$  e  $m$  são, respectivamente, a quantidade de *jobs* e de máquinas. Assim, os problemas KC utilizados são KC  $4 \times 5$ , KC  $8 \times 8$ , KC  $10 \times 7$ , KC  $10 \times 10$  e KC  $15 \times 10$ . As instâncias dos problemas BR são designadas por BR Mk1, BR Mk2, ..., BR Mk10 e variam de  $10 \times 6$  (BR Mk1) a  $20 \times 15$  (BR Mk10).

Na Tabela 5, os parâmetros utilizados para as execuções dos problemas são apresentados. A coluna ‘Problema’ especifica o problema em que os parâmetros foram utilizados. As demais colunas identificam os parâmetros do algoritmo. A coluna ‘POP’ é o tamanho da população utilizado, ‘MAX\_GEN’ é o número máximo de gerações e a coluna ‘EA’ é o tamanho do conjunto elitista. A distinção aplicada aos parâmetros se deve ao fato de que as instâncias de cada problema possuem características específicas. Assim, os valores dos parâmetros foram empiricamente determinados, sendo que dez execuções do algoritmo, variando os valores dos parâmetros, foram realizadas com cada configuração para verificar seu comportamento e os resultados foram analisados usando as definições do Ótimo de Pareto. Portanto, com base nos resultados obtidos, os valores dos parâmetros foram determinados.

As Tabelas 6 e 7 apresentam os resultados dos experimentos do SEDA com as instâncias dos problemas KC e BR comparando com os algoritmos propostos em (Li; Pan; Liang, 2010) (HTSA) e (Wang et al., 2013) (W1-EDA). O primeiro é um algoritmo que muitos estudos usam para comparar resultados e o segundo é um EDA como o SEDA. Nas tabelas, os resultados são apresentados agrupados por problema e algoritmo. Assim, o problema é identificado em uma linha, subdividido em algoritmos na próxima linha e, para cada algoritmo, as colunas ‘M’, ‘TW’ e ‘W’ são apresentadas indicando os valores dos objetivos considerados para o MOFJSP, respectivamente, *makespan*, carga de trabalho total e carga de trabalho máxima.

Os resultados apresentados pela Tabela 6 mostram que nos problemas KC, os algoritmos, SEDA, HTSA e W1-EDA, são equivalentes, pois, ao contabilizar todas as soluções dos algoritmos, SEDA encontrou doze, HTSA onze e W1-EDA treze soluções e todas as soluções são não dominadas. No entanto, nos resultados dos problemas BR apresentados na Tabela 7, HTSA e W1-EDA encontraram dez soluções para todos os problemas BR, enquanto SEDA encontrou 39 soluções, ou seja, SEDA apresentou 290% soluções a mais

Tabela 5 – SEDA - Parâmetros

| Problema          | POP  | MAX_GEN | EA(%) |
|-------------------|------|---------|-------|
| KC $4 \times 5$   | 200  | 30      | 70    |
| KC $8 \times 8$   | 500  | 200     | 70    |
| KC $10 \times 7$  | 400  | 200     | 70    |
| KC $10 \times 10$ | 400  | 200     | 70    |
| KC $15 \times 10$ | 800  | 200     | 70    |
| BR Mk01           | 1000 | 100     | 70    |
| BR Mk02           | 1000 | 100     | 70    |
| BR Mk03           | 1000 | 150     | 70    |
| BR Mk04           | 1000 | 100     | 70    |
| BR Mk05           | 1000 | 100     | 70    |
| BR Mk06           | 1000 | 200     | 70    |
| BR Mk07           | 1000 | 100     | 70    |
| BR Mk08           | 1000 | 100     | 70    |
| BR Mk09           | 1000 | 200     | 70    |
| BR Mk10           | 1000 | 200     | 70    |

do que os demais, HTSA e W1-EDA. Outra característica observada nos resultados é que o *makespan* (M) e a carga de trabalho (W) são muito maiores em SEDA do que em HTSA e W1-EDA. Uma razão para isso é que no SEDA não existem métodos de busca local e no HTSA e W1-EDA existem, sendo que esses métodos observam o *makespan*.

Tabela 6 – Resultados do SEDA - KC

| KC - $4 \times 5$   |    |    |        |    |    |      |    |    | KC - $8 \times 8$   |    |    |        |    |    |      |    |    |
|---------------------|----|----|--------|----|----|------|----|----|---------------------|----|----|--------|----|----|------|----|----|
| HTSA                |    |    | W1-EDA |    |    | SEDA |    |    | HTSA                |    |    | W1-EDA |    |    | SEDA |    |    |
| M                   | TW | W  | M      | TW | W  | M    | TW | W  | M                   | TW | W  | M      | TW | W  | M    | TW | W  |
| 11                  | 32 | 10 | 11     | 34 | 9  | 11   | 32 | 10 | 14                  | 77 | 12 | 14     | 77 | 12 | 15   | 75 | 12 |
| 12                  | 32 | 8  | 11     | 32 | 10 | 11   | 34 | 9  | 15                  | 75 | 12 | 15     | 75 | 12 | 16   | 73 | 13 |
|                     |    |    | 12     | 32 | 8  | 12   | 32 | 8  |                     |    |    |        |    |    |      |    |    |
|                     |    |    |        |    |    | 13   | 33 | 7  |                     |    |    |        |    |    |      |    |    |
| KC - $10 \times 7$  |    |    |        |    |    |      |    |    | KC - $10 \times 10$ |    |    |        |    |    |      |    |    |
| HTSA                |    |    | W1-EDA |    |    | SEDA |    |    | HTSA                |    |    | W1-EDA |    |    | SEDA |    |    |
| M                   | TW | W  | M      | TW | W  | M    | TW | W  | M                   | TW | W  | M      | TW | W  | M    | TW | W  |
| 11                  | 61 | 11 | 11     | 61 | 11 | 11   | 61 | 11 | 7                   | 43 | 5  | 7      | 43 | 5  | 7    | 42 | 6  |
| 11                  | 62 | 10 | 11     | 62 | 10 | 12   | 60 | 12 | 7                   | 42 | 6  | 7      | 42 | 6  | 8    | 41 | 7  |
|                     |    |    |        |    |    |      |    |    | 8                   | 42 | 5  | 8      | 41 | 7  | 8    | 42 | 5  |
|                     |    |    |        |    |    |      |    |    |                     |    |    | 8      | 42 | 5  |      |    |    |
| KC - $15 \times 10$ |    |    |        |    |    |      |    |    |                     |    |    |        |    |    |      |    |    |
| HTSA                |    |    | W1-EDA |    |    | SEDA |    |    |                     |    |    |        |    |    |      |    |    |
| M                   | TW | W  | M      | TW | W  | M    | TW | W  |                     |    |    |        |    |    |      |    |    |
| 11                  | 91 | 11 | 11     | 91 | 11 | 11   | 91 | 11 |                     |    |    |        |    |    |      |    |    |
| 11                  | 93 | 10 | 11     | 93 | 10 |      |    |    |                     |    |    |        |    |    |      |    |    |

Tabela 7 – Resultados do SEDA - BR

| BR Mk01 |      |     |        |      |     |      |      |     | BR Mk02 |      |     |        |      |     |      |      |     |
|---------|------|-----|--------|------|-----|------|------|-----|---------|------|-----|--------|------|-----|------|------|-----|
| HTSA    |      |     | W1-EDA |      |     | SEDA |      |     | HTSA    |      |     | W1-EDA |      |     | SEDA |      |     |
| M       | TW   | W   | M      | TW   | W   | M    | TW   | W   | M       | TW   | W   | M      | TW   | W   | M    | TW   | W   |
| 40      | 167  | 36  | 40     | 167  | 36  | 42   | 162  | 42  | 26      | 151  | 26  | 26     | 151  | 26  | 29   | 143  | 29  |
|         |      |     |        |      |     | 43   | 159  | 43  |         |      |     |        |      |     | 31   | 141  | 31  |
|         |      |     |        |      |     | 44   | 165  | 37  |         |      |     |        |      |     | 33   | 140  | 33  |
| BR Mk03 |      |     |        |      |     |      |      |     | BR Mk04 |      |     |        |      |     |      |      |     |
| HTSA    |      |     | W1-EDA |      |     | SEDA |      |     | HTSA    |      |     | W1-EDA |      |     | SEDA |      |     |
| M       | TW   | W   | M      | TW   | W   | M    | TW   | W   | M       | TW   | W   | M      | TW   | W   | M    | TW   | W   |
| 204     | 852  | 204 | 204    | 850  | 204 | 210  | 848  | 210 | 61      | 366  | 61  | 60     | 382  | 60  | 84   | 334  | 84  |
|         |      |     |        |      |     | 213  | 844  | 213 |         |      |     |        |      |     | 78   | 337  | 78  |
|         |      |     |        |      |     | 221  | 842  | 221 |         |      |     |        |      |     |      |      |     |
|         |      |     |        |      |     | 222  | 838  | 222 |         |      |     |        |      |     |      |      |     |
|         |      |     |        |      |     | 231  | 834  | 231 |         |      |     |        |      |     |      |      |     |
|         |      |     |        |      |     | 240  | 832  | 240 |         |      |     |        |      |     |      |      |     |
|         |      |     |        |      |     | 249  | 830  | 249 |         |      |     |        |      |     |      |      |     |
|         |      |     |        |      |     | 258  | 828  | 258 |         |      |     |        |      |     |      |      |     |
| BR Mk05 |      |     |        |      |     |      |      |     | BR Mk06 |      |     |        |      |     |      |      |     |
| HTSA    |      |     | W1-EDA |      |     | SEDA |      |     | HTSA    |      |     | W1-EDA |      |     | SEDA |      |     |
| M       | TW   | W   | M      | TW   | W   | M    | TW   | W   | M       | TW   | W   | M      | TW   | W   | M    | TW   | W   |
| 172     | 687  | 172 | 172    | 687  | 172 | 178  | 680  | 178 | 65      | 398  | 62  | 63     | 423  | 59  | 74   | 367  | 67  |
|         |      |     |        |      |     | 183  | 677  | 183 |         |      |     |        |      |     | 74   | 365  | 69  |
|         |      |     |        |      |     | 185  | 676  | 185 |         |      |     |        |      |     | 74   | 366  | 68  |
|         |      |     |        |      |     | 191  | 675  | 191 |         |      |     |        |      |     | 74   | 364  | 70  |
|         |      |     |        |      |     | 197  | 674  | 197 |         |      |     |        |      |     |      |      |     |
|         |      |     |        |      |     | 203  | 673  | 203 |         |      |     |        |      |     |      |      |     |
|         |      |     |        |      |     | 209  | 672  | 209 |         |      |     |        |      |     |      |      |     |
| BR Mk07 |      |     |        |      |     |      |      |     | BR Mk08 |      |     |        |      |     |      |      |     |
| HTSA    |      |     | W1-EDA |      |     | SEDA |      |     | HTSA    |      |     | W1-EDA |      |     | SEDA |      |     |
| M       | TW   | W   | M      | TW   | W   | M    | TW   | W   | M       | TW   | W   | M      | TW   | W   | M    | TW   | W   |
| 140     | 695  | 140 | 140    | 685  | 140 | 151  | 667  | 151 | 523     | 2524 | 523 | 523    | 2524 | 523 | 542  | 2509 | 542 |
|         |      |     |        |      |     | 156  | 664  | 156 |         |      |     |        |      |     | 551  | 2504 | 551 |
|         |      |     |        |      |     | 162  | 659  | 162 |         |      |     |        |      |     | 560  | 2499 | 560 |
|         |      |     |        |      |     | 166  | 657  | 166 |         |      |     |        |      |     | 569  | 2494 | 569 |
|         |      |     |        |      |     |      |      |     |         |      |     |        |      |     | 578  | 2489 | 578 |
| BR Mk09 |      |     |        |      |     |      |      |     | BR Mk10 |      |     |        |      |     |      |      |     |
| HTSA    |      |     | W1-EDA |      |     | SEDA |      |     | HTSA    |      |     | W1-EDA |      |     | SEDA |      |     |
| M       | TW   | W   | M      | TW   | W   | M    | TW   | W   | M       | TW   | W   | M      | TW   | W   | M    | TW   | W   |
| 310     | 2294 | 301 | 309    | 2301 | 299 | 375  | 2273 | 299 | 214     | 2053 | 210 | 219    | 1992 | 201 | 265  | 1889 | 235 |
|         |      |     |        |      |     |      |      |     |         |      |     |        |      |     | 265  | 1901 | 222 |

Por outro lado, algoritmos como o de Wang, Wang e Liu (2013) (W2-EDA), Chiang e Lin (2013) (SEA) e Carvalho (2015) (DIPSO) apresentam muito mais soluções do que o SEDA. O primeiro é um EDA híbrido com operadores genéticos e busca local. O segundo é um AE projetado para o MOFJSP, sendo que ambos usam conhecimento específico do problema. O terceiro, como já descrito na seção 2.3.4, é um PSO híbrido. Os resultados do W2-EDA estão na Tabela 8, que é análoga às Tabelas 6 e 7. Para aqueles problemas que o algoritmo apresenta mais de onze soluções, o símbolo ‘...’ substituiu o restante das soluções, mostrando que existem outras que não estão listadas. Ao comparar as soluções do W2-EDA e do SEDA, observa-se que, além do número de soluções do W2-EDA serem

Tabela 8 – W2-EDA (Wang; Wang; Liu, 2013) - Resultados

| KC 4x5 |     |    | KC 8x8 |     |     | KC 10x7 |      |     | KC 10x10 |      |     | KC 15x10 |      |     |
|--------|-----|----|--------|-----|-----|---------|------|-----|----------|------|-----|----------|------|-----|
| M      | TW  | W  | M      | TW  | W   | M       | TW   | W   | M        | TW   | W   | M        | TW   | W   |
| 11     | 32  | 10 | 14     | 77  | 12  | 11      | 61   | 11  | 7        | 42   | 6   | 11       | 91   | 11  |
| 11     | 34  | 9  | 15     | 75  | 12  | 11      | 62   | 10  | 8        | 41   | 7   | 11       | 93   | 10  |
| 12     | 32  | 8  | 16     | 73  | 13  | 12      | 60   | 12  | 8        | 42   | 5   |          |      |     |
| 13     | 33  | 7  | 16     | 77  | 11  |         |      |     | 7        | 43   | 5   |          |      |     |
| ...    |     |    |        |     |     |         |      |     |          |      |     |          |      |     |
| BR Mk1 |     |    | BR Mk2 |     |     | BR Mk3  |      |     | BR Mk4   |      |     | BR Mk5   |      |     |
| M      | TW  | W  | M      | TW  | W   | M       | TW   | W   | M        | TW   | W   | M        | TW   | W   |
| 40     | 167 | 36 | 26     | 151 | 26  | 204     | 850  | 204 | 60       | 382  | 60  | 173      | 683  | 173 |
| 40     | 165 | 37 | 27     | 145 | 27  | 210     | 848  | 210 | 61       | 366  | 61  | 175      | 682  | 175 |
| 41     | 160 | 38 | 28     | 144 | 28  | 213     | 844  | 213 | 62       | 379  | 60  | 178      | 680  | 178 |
| 41     | 163 | 37 | 29     | 143 | 29  | 221     | 842  | 221 | 63       | 362  | 62  | 179      | 679  | 179 |
| 42     | 157 | 40 | 30     | 142 | 30  | 222     | 838  | 222 | 64       | 355  | 62  | 183      | 677  | 183 |
| 42     | 158 | 39 | 31     | 141 | 31  | 231     | 834  | 231 | 64       | 365  | 61  | 185      | 676  | 185 |
| 42     | 165 | 36 | 31     | 150 | 26  | 240     | 832  | 240 | 65       | 348  | 63  | 190      | 687  | 172 |
| 43     | 155 | 40 | 33     | 140 | 33  | 249     | 830  | 249 | 67       | 344  | 66  | 191      | 675  | 191 |
| 44     | 154 | 40 |        |     |     | 266     | 828  | 258 | 69       | 343  | 67  | 197      | 674  | 197 |
| 46     | 153 | 46 |        |     |     |         |      |     | 72       | 340  | 72  | 203      | 673  | 203 |
| 47     | 153 | 42 |        |     |     |         |      |     | 72       | 355  | 62  | 209      | 672  | 209 |
| ...    |     |    |        |     |     |         |      |     |          |      |     |          |      |     |
| BR Mk6 |     |    | BR Mk7 |     |     | BR Mk8  |      |     | BR Mk9   |      |     | BR Mk10  |      |     |
| M      | TW  | W  | M      | TW  | W   | M       | TW   | W   | M        | TW   | W   | M        | TW   | W   |
| 63     | 423 | 59 | 139    | 693 | 139 | 523     | 2524 | 523 | 309      | 2301 | 299 | 224      | 1980 | 219 |
| 64     | 411 | 62 | 140    | 689 | 140 | 524     | 2519 | 524 | 311      | 2282 | 299 | 225      | 1976 | 211 |
| 65     | 405 | 62 | 143    | 683 | 143 | 533     | 2514 | 533 | 312      | 2278 | 306 | 233      | 1919 | 214 |
| 66     | 396 | 62 | 144    | 673 | 144 | 542     | 2509 | 542 | 313      | 2269 | 304 | 235      | 1895 | 255 |
| 69     | 376 | 63 | 150    | 669 | 150 | 551     | 2504 | 551 | 313      | 2273 | 302 | 235      | 1897 | 218 |
| 71     | 371 | 66 | 151    | 667 | 151 | 560     | 2499 | 560 | 314      | 2265 | 312 | 240      | 1905 | 215 |
| 71     | 390 | 62 | 153    | 664 | 156 | 569     | 2494 | 569 | 316      | 2279 | 299 | 240      | 1888 | 216 |
| 72     | 368 | 70 | 157    | 662 | 157 | 578     | 2489 | 578 | 318      | 2263 | 310 | 242      | 1913 | 214 |
| 72     | 370 | 68 | 161    | 660 | 161 | 587     | 2484 | 587 | 319      | 2277 | 299 | 246      | 1896 | 215 |
| 72     | 374 | 65 | 162    | 659 | 162 |         |      |     | 320      | 2255 | 315 | 252      | 1884 | 224 |
| 72     | 384 | 61 | 166    | 657 | 166 |         |      |     | 321      | 2256 | 312 | 256      | 1919 | 211 |
| ...    |     |    | ...    |     |     | ...     |      |     | ...      |      |     | ...      |      |     |

superiores ao SEDA, também existem soluções do SEDA que são dominadas por soluções do W2-EDA para os problemas BR Mk01, BR Mk06, BR Mk07, BR Mk09 e BR Mk10. O conceito do SEDA é baseado no EDA, sem utilizar conhecimento do problema ou utilizar métodos de outras técnicas, ou seja, SEDA é um EDA puro. O uso de conhecimento específico, pesquisa local e operadores genéticos transformariam o algoritmo em uma técnica híbrida.

### 4.1.2 SEDASI - Experimentos e Análise dos Resultados

O experimento realizado para validar os resultados apresentados pelo SEDASI foi baseado no experimento do SEDA, porém foi utilizado somente o *benchmark* KC. Na Tabela 9 são apresentados os parâmetros utilizados, que são os mesmos do SEDA, exceto pela exclusão do parâmetro EA que não foi utilizado nesse algoritmo.

Tabela 9 – SEDASI - Parâmetros

| Problema          | <i>POP</i> | <i>MAX_GEN</i> |
|-------------------|------------|----------------|
| KC $4 \times 5$   | 200        | 30             |
| KC $8 \times 8$   | 500        | 200            |
| KC $10 \times 7$  | 400        | 200            |
| KC $10 \times 10$ | 400        | 200            |
| KC $15 \times 10$ | 800        | 200            |

Os resultados determinados pelo SEDASI são apresentados na Tabela 10. Um comparativo é feito com os resultados dos algoritmos HTSA, W-EDA, SEDA e DIPSO. Essa tabela é análoga às Tabelas 6 e 7, que apresentam os resultados de SEDA. Os resultados mostram que os algoritmos comparados são equivalentes, entretanto, SEDASI possui duas soluções dominadas para os problemas KC  $8 \times 8$  e KC  $15 \times 10$ , enquanto o DIPSO possui apenas uma para o KC  $15 \times 10$ . Nos demais algoritmos não há soluções dominadas. Apesar disso, SEDASI apresentou uma quantidade de soluções maior do que de SEDA. Conforme apresentado, SEDA e SEDASI são similares, diferenciando-se pelo conjunto que auxilia a evolução da população e pela quantidade de indivíduos que são substituídos, o que nos leva a concluir que são os fatores determinantes para uma maior diversidade no conjunto não dominado apresentado como solução.

## 4.2 Convergência: PSO, DIPSO, SEDA

Nesta seção, é apresentada uma análise de convergência considerando os algoritmos PSO, DIPSO e SEDA. A escolha desses algoritmos foi baseada no fato de que o PSO e o EDA possuem uma convergência rápida (Zhang et al., 2009) (Mühlenbein; Paaß, 1996), além de que DIPSO é um PSO híbrido que lida com a convergência prematura do PSO. Assim como SEDA e SEDASI, DIPSO foi projetado para tratar o MOFJSP, portanto, neste experimento os algoritmos são aplicados para tratar esse problema. Para auxiliar, o OCD foi utilizado para identificar o ponto de convergência dos algoritmos. Assim, um comparativo será apresentado com o conjunto de aproximação, hipervolume e geração de tais informações, a fim de traçar um perfil evolutivo de cada técnica. Os pontos de referência para o cálculo do hipervolume foram determinados por um experimento em que se executou cada problema dez vezes com cada algoritmo. Para cada problema, o pior

Tabela 10 – Resultados do SEDASI - KC

| KC 4 × 5   |    |    |       |    |    |      |    |    |       |    |    |        |    |    |
|------------|----|----|-------|----|----|------|----|----|-------|----|----|--------|----|----|
| HTSA       |    |    | W-EDA |    |    | SEDA |    |    | DIPSO |    |    | SEDASI |    |    |
| M          | TW | W  | M     | TW | W  | M    | TW | W  | M     | TW | W  | M      | TW | W  |
| 11         | 32 | 10 | 11    | 34 | 9  | 11   | 32 | 10 | 11    | 32 | 10 | 11     | 32 | 10 |
| 12         | 32 | 8  | 11    | 32 | 10 | 11   | 34 | 9  | 11    | 34 | 9  | 11     | 34 | 9  |
|            |    |    | 12    | 32 | 8  | 12   | 32 | 8  | 12    | 32 | 8  | 12     | 32 | 8  |
|            |    |    |       |    |    | 13   | 33 | 7  | 13    | 33 | 7  | 13     | 33 | 7  |
| KC 8 × 8   |    |    |       |    |    |      |    |    |       |    |    |        |    |    |
| HTSA       |    |    | W-EDA |    |    | SEDA |    |    | DIPSO |    |    | SEDASI |    |    |
| M          | TW | W  | M     | TW | W  | M    | TW | W  | M     | TW | W  | M      | TW | W  |
| 14         | 77 | 12 | 14    | 77 | 12 | 15   | 75 | 12 | 14    | 77 | 12 | 14     | 77 | 12 |
| 15         | 75 | 12 | 15    | 75 | 12 | 16   | 73 | 13 | 15    | 75 | 12 | 15     | 75 | 12 |
|            |    |    |       |    |    |      |    |    | 16    | 73 | 13 | 16     | 73 | 13 |
|            |    |    |       |    |    |      |    |    | 16    | 77 | 11 | 17     | 77 | 11 |
| KC 10 × 7  |    |    |       |    |    |      |    |    |       |    |    |        |    |    |
| HTSA       |    |    | W-EDA |    |    | SEDA |    |    | DIPSO |    |    | SEDASI |    |    |
| M          | TW | W  | M     | TW | W  | M    | TW | W  | M     | TW | W  | M      | TW | W  |
| 11         | 61 | 11 | 11    | 61 | 11 | 11   | 61 | 11 | 11    | 61 | 11 | 11     | 61 | 11 |
| 11         | 62 | 10 | 11    | 62 | 10 | 12   | 60 | 12 | 11    | 62 | 10 | 11     | 62 | 10 |
|            |    |    |       |    |    |      |    |    | 12    | 60 | 12 | 12     | 60 | 12 |
| KC 10 × 10 |    |    |       |    |    |      |    |    |       |    |    |        |    |    |
| HTSA       |    |    | W-EDA |    |    | SEDA |    |    | DIPSO |    |    | SEDASI |    |    |
| M          | TW | W  | M     | TW | W  | M    | TW | W  | M     | TW | W  | M      | TW | W  |
| 7          | 43 | 5  | 7     | 43 | 5  | 7    | 42 | 6  | 7     | 42 | 6  | 7      | 42 | 6  |
| 7          | 42 | 6  | 7     | 42 | 6  | 8    | 41 | 7  | 7     | 43 | 5  | 8      | 41 | 7  |
| 8          | 42 | 5  | 8     | 41 | 7  | 8    | 42 | 5  | 8     | 41 | 7  | 8      | 42 | 5  |
|            |    |    | 8     | 42 | 5  |      |    |    | 8     | 42 | 5  |        |    |    |
| KC 15 × 10 |    |    |       |    |    |      |    |    |       |    |    |        |    |    |
| HTSA       |    |    | W-EDA |    |    | SEDA |    |    | DIPSO |    |    | SEDASI |    |    |
| M          | TW | W  | M     | TW | W  | M    | TW | W  | M     | TW | W  | M      | TW | W  |
| 11         | 91 | 11 | 11    | 91 | 11 | 11   | 91 | 11 | 12    | 91 | 11 | 12     | 91 | 11 |
| 11         | 93 | 10 | 11    | 93 | 10 |      |    |    |       |    |    |        |    |    |

valor encontrado de cada objetivo foi coletado separadamente, sendo acrescido de uma margem de dez por cento, formando o ponto de referência do respectivo problema.

O PSO é um algoritmo evolutivo baseado na inteligência de enxame e é um algoritmo para a otimização de problemas de domínio contínuo. Por outro lado, o FJSP é um problema de domínio discreto e, portanto, para tratar o FJSP com o PSO, é necessário adaptar ou o PSO ou o problema, seção 2.3.2. Portanto, para realizar o experimento com

o PSO, o FJSP foi modificado para ser interpretado como um problema de domínio contínuo e uma função (baseada na operação de módulo) foi implementada para transformar um valor de domínio contínuo em um valor de domínio discreto, a fim de representar a partícula como uma solução do problema. Essa função valida a posição resultante (*jobs*, operações dos *jobs* e número de máquinas) e, se necessário, modifica a posição para que ela se torne uma solução válida para o FJSP. Em decorrência desse processo, tanto os valores contínuos, quanto os valores discretos são armazenados na partícula, pois o valor discreto é interpretado pelo FJSP e o valor contínuo é utilizado pelo algoritmo a cada geração.

Para auxiliar na análise de convergência, o algoritmo OCD-HV foi implementado associado aos algoritmos. O OCD realiza um teste estatístico no final de cada geração, que utiliza um ou mais indicadores para determinar se o algoritmo convergiu. Conforme já descrito na seção 2.4.1, OCD-HV é a versão do OCD utilizando somente o hipervolume como indicador.

### 4.2.1 Descrição dos Experimentos

Os experimentos usaram o *benchmark* KC e foram realizados no mesmo computador descrito na seção 4.1. Os resultados apresentados neste experimento são o conjunto de soluções não dominadas de todas as execuções, enquanto que os dados utilizados na análise da evolução do algoritmo e da convergência (geração do ponto de convergência e valor de hipervolume do conjunto de aproximação) estão relacionados com a melhor execução, ou seja, aquela com o maior valor de hipervolume dentre as trinta execuções de cada problema. Os algoritmos foram executados até o número máximo de gerações (*MAX\_GEN*) configurado, isto é, mesmo quando o algoritmo OCD-HV identificou a convergência, os algoritmos continuaram sua execução. Esse comportamento foi determinado para avaliar o quão exato é o ponto de convergência apontado pelo OCD-HV.

Os parâmetros para o SEDA são apresentados na Tabela 11, que é análoga à Tabela 5. A Tabela 12 apresenta os parâmetros do DIPSO. A coluna ' $p_{cc}$ ' mostra a probabilidade do operador genético de evolução ser selecionado e aplicado à fórmula do DIPSO, a coluna ' $p_{dx}$ ' é análoga à coluna ' $p_{cc}$ ', mas considerando a probabilidade do operador de diversi-

Tabela 11 – SEDA - Parâmetros

| Problema | <i>POP</i> | <i>MAX_GEN</i> | <i>EA</i> (%) |
|----------|------------|----------------|---------------|
| KC 4×5   | 200        | 100            | 70            |
| KC 8×8   | 500        | 200            | 70            |
| KC 10×7  | 400        | 200            | 70            |
| KC 10×10 | 400        | 200            | 70            |
| KC 15×10 | 800        | 200            | 70            |

dade. Na coluna ‘Mutaç o’, a porcentagem para aplicar a mutaç o   apresentada. Os significados das outras colunas, ‘POP’ e ‘MAX\_GEN’, representam, respectivamente, o tamanho da populaç o e o n mero m ximo de geraç es. Os valores dos par metros do PSO s o apresentados na Tabela 13. As colunas ‘a’, ‘b<sub>1</sub>’ e ‘b<sub>2</sub>’ mostram os coeficientes que determinam a influ ncia de  $v_i$ ,  $g_{best}$  e  $p_{best}$ , respectivamente. Os significados das colunas ‘POP’ e ‘MAX\_GEN’ s o iguais  s respectivas colunas da Tabela 12.

Os valores dos par metros foram empiricamente determinados, sendo que v rias execuç es do algoritmo, variando os valores, foram realizadas para verificar seu comportamento. No caso do PSO, o estudo apresentado por Trelea (Seç o 2.4) foi base para escolher os intervalos de valores dos par metros a serem analisados. Entretanto, os intervalos utilizados foram ligeiramente superiores e inclu ram valores sem converg ncia para o PSO. O algoritmo foi executado com cada problema dez vezes e os resultados foram analisados usando as definiç es do  timo de Pareto. Assim, com base nos resultados obtidos, os valores dos par metros foram determinados. Os par metros do PSO,  $r_1$  e  $r_2$ , s o n meros aleat rios e definidos considerando  $r_1 + r_2 = 1$ ,  $1$  e  $0$ ,  $1 \leq r_1 \leq 1$ ,  $1$ .

## 4.2.2 An lise

As principais informaç es observadas nos experimentos foram o ponto de converg ncia (geraç o de converg ncia determinada pelo OCD-HV), o valor do hipervolume (HV) no ponto de converg ncia, a variaç o do indicador hipervolume (Evoluç o do HV) e os resultados dos algoritmos.

A Tabela 14 apresenta o ponto de converg ncia determinado pelo OCD-HV para cada algoritmo, sendo que na coluna ‘Problema’   informado a inst ncia do problema e as demais colunas, nomeadas com o respectivo algoritmo, apresentam a geraç o de converg ncia. Observa-se que, o valor apresentado para o problema KC  $10 \times 7$  com o algoritmo PSO est  entre par nteses. Isso ocorre, porque a parametrizaç o selecionada para esse problema no PSO n o converge (Seç o 2.4) e o n mero apresentado   a geraç o em que o algoritmo encontrou a melhor part cula durante a execuç o. O fato da n o converg ncia   explicada pela escolha da parametrizaç o, a qual foi baseada apenas no resultado experimental que, no caso do PSO, foi determinado pelo  $g_{best}$  da  ltima geraç o e n o pela possibilidade de converg ncia.

O HV no ponto de converg ncia determinado pelo OCD-HV e na geraç o final de execuç o dos algoritmos s o apresentados na Tabela 15, sendo que a coluna ‘Problema’ informa a inst ncia do problema e as colunas nomeadas pelo algoritmos est o agrupadas

Tabela 12 – DIPSO - Par metros

| $p_{cc}$ | $p_{dx}$ | Mutaç o | POP | MAX_GEN |
|----------|----------|---------|-----|---------|
| 80%      | 20%      | 5%      | 400 | 800     |

Tabela 13 – PSO - Parâmetros

| Problema | $a$ | $b_1$ | $b_2$ | $POP$ | $MAX\_GEN$ |
|----------|-----|-------|-------|-------|------------|
| KC 4×5   | 0,9 | 0,5   | 0,7   | 500   | 300        |
| KC 8×8   | 0,8 | 0,5   | 1     | 500   | 300        |
| KC 10×7  | 1   | 0,9   | 1     | 1.200 | 300        |
| KC 10×10 | 0,6 | 0,1   | 0,7   | 1.250 | 300        |
| KC 15×10 | 0,5 | 0,1   | 0,4   | 2.300 | 300        |

pelo ‘HV’. Dessa forma, apresentam o hipervolume no ponto de convergência e na geração final considerando o algoritmo e o problema enumerados. Em negrito estão destacados os maiores valores do hipervolume no ponto de convergência e na geração final, para cada problema.

Os valores apresentados na Tabela 14 indicam uma convergência rápida nos problemas KC 8 × 8, KC 10 × 10 e KC 15 × 10 com PSO, mas com o problema KC 4 × 5 essa característica não é apresentada. Na Tabela 15, observa-se que o HV do PSO de KC 4 × 5, tanto no ponto de convergência quanto na geração final, é semelhante ao dos de SEDA e DIPSO, porém o mesmo não ocorre com os outros problemas, que são bem diferentes, sendo que o HV do SEDA e do DIPSO são muito maiores que do PSO. Isso indica uma convergência prematura no PSO ao tratar os problemas KC 8 × 8, KC 10 × 7, KC 10 × 10 e KC 15 × 10.

Tabela 14 – Ponto de convergência (OCD-HV)

| Problema | PSO  | SEDA | DIPSO |
|----------|------|------|-------|
| KC 4×5   | 61   | 33   | 18    |
| KC 8×8   | 24   | 64   | 48    |
| KC 10×7  | (71) | 57   | 57    |
| KC 10×10 | 21   | 64   | 47    |
| KC 15×10 | 14   | 99   | 112   |

Tabela 15 – Hipervolume no Ponto de Convergência e na Geração Final

| Problema | HV (Ponto de Convergência) |               |             | HV (Geração Final) |        |               |
|----------|----------------------------|---------------|-------------|--------------------|--------|---------------|
|          | PSO                        | SEDA          | DIPSO       | PSO                | SEDA   | DIPSO         |
| KC 4×5   | 504K                       | 556K          | <b>562K</b> | 504K               | 544K   | <b>562K</b>   |
| KC 8×8   | 752K                       | <b>2.381K</b> | 2.294K      | 752K               | 2.331K | <b>2.381K</b> |
| KC 10×7  | 869K                       | <b>1.726K</b> | 1.725K      | 869K               | 1.713K | <b>1.750K</b> |
| KC 10×10 | 163K                       | <b>510K</b>   | 509K        | 163K               | 504K   | <b>515K</b>   |
| KC 15×10 | 1.158K                     | <b>4.008K</b> | 3.883K      | 1.158K             | 4.009K | <b>4.037K</b> |

A Tabela 14 mostra que o DIPSO converge na décima oitava geração com KC  $4 \times 5$ , o que poderia indicar uma convergência prematura. Porém, os valores de HV apresentados na Tabela 15 para esse problema mostram que DIPSO possui o maior valor, o que indica uma tendência de que DIPSO tenha a melhor solução entre os algoritmos. Outra observação efetuada na Tabela 14 é que, para todos os problemas, exceto KC  $15 \times 10$ , o DIPSO converge mais rápido que o SEDA, ou no mesmo momento, caso do KC  $10 \times 7$ . Por outro lado, as informações apresentadas pela Tabela 15 mostram que, no ponto de convergência do OCD-HV, os HV's do SEDA são maiores que os do DIPSO, indicando que os resultados do SEDA são mais desejáveis que os do DIPSO, naquele momento. Como os experimentos não pararam na geração de convergência do OCD-HV, os algoritmos foram executados até a geração final, determinada por *MAX\_GEN*. Assim, os HV's da geração final (*MAX\_GEN*) apresentados na Tabela 15 mostram que os resultados do DIPSO são mais desejáveis que os resultados do SEDA (na geração *MAX\_GEN*), existindo diferenças dos HVs entre o ponto de convergência e a geração final. Os HVs do SEDA diminuíram e os do DIPSO aumentaram, o que sugere que a qualidade dos resultados do SEDA se deteriorou e a qualidade dos resultados do DIPSO melhorou.

A Tabela 16 apresenta os resultados do SEDA e do DIPSO, considerando a geração do ponto de convergência do OCD-HV. A coluna 'Problema' especifica o problema em questão. As colunas representadas pelos nomes dos algoritmos apresentam os resultados do respectivo algoritmo para o problema relacionado, sendo que essas colunas estão subdivididas em 'M', 'TW' e 'W', objetivos tratados pelo MOFJSP, respectivamente *makespan*, carga de trabalho total e carga de trabalho máxima. Os resultados destacados em negrito representam a solução com melhor qualidade. A Tabela 17 é análoga à Tabela 16 e apresenta os resultados do SEDA e do DIPSO considerando a geração final.

Comparando os resultados do SEDA e do DIPSO apresentados nas Tabelas 16 e 17 com os valores de hipervolume apresentados na Tabela 15, observa-se que as preferências de resultados dos algoritmos mostradas pelo hipervolume estão presentes nas soluções dos algoritmos, ou seja, os valores maiores de hipervolume apresentaram melhores resultados no ponto de convergência e na geração final, como pode ser observado pelos destaques em negrito. Comparando os resultados no ponto de convergência (Tabela 16) com os resultados da última geração (Tabela 17), observa-se a perda de qualidade nos resultados do SEDA, que foi uma perda de diversidade e é justificada pela lógica do algoritmo que permite a substituição de uma boa solução. Outra observação é o ganho de qualidade dos resultados do DIPSO, que deve-se ao fato de que o DIPSO possui um operador que gera diversidade em toda a população enquanto o algoritmo está em execução. Portanto, após o OCD-HV indicar a convergência, o operador de diversidade gera novas possibilidades no espaço de busca e a população deixa o estado de estabilidade, ocorrendo a possibilidade do algoritmo buscar soluções em áreas do espaço de busca ainda não exploradas.

Em relação aos resultados que o OCD-HV retornou, foi satisfatório para o PSO e para

Tabela 16 – Resultados do SEDA e do DIPSO no Ponto de Convergência (OCD-HV)

| Problema | SEDA      |           |           | DIPSO     |           |          |
|----------|-----------|-----------|-----------|-----------|-----------|----------|
|          | M         | TW        | W         | M         | TW        | W        |
| KC 4×5   | 11        | 32        | 10        | <b>12</b> | <b>32</b> | <b>8</b> |
|          | 12        | 32        | 8         | <b>11</b> | <b>34</b> | <b>9</b> |
|          |           |           |           | <b>13</b> | <b>33</b> | <b>7</b> |
| KC 8×8   | <b>15</b> | <b>75</b> | <b>12</b> | 16        | 76        | 13       |
|          | <b>19</b> | <b>81</b> | <b>11</b> | 16        | 75        | 14       |
|          | <b>16</b> | <b>73</b> | <b>13</b> |           |           |          |
| KC 10×7  | <b>12</b> | <b>61</b> | <b>11</b> | 13        | 60        | 13       |
|          | <b>12</b> | <b>60</b> | <b>12</b> | 12        | 62        | 11       |
|          |           |           |           | 12        | 61        | 12       |
| KC 10×10 | <b>8</b>  | <b>42</b> | <b>6</b>  | 9         | 41        | 8        |
|          | <b>8</b>  | <b>41</b> | <b>7</b>  | 8         | 42        | 6        |
| KC 15×10 | <b>13</b> | <b>91</b> | <b>11</b> | 15        | 96        | 12       |
|          | <b>12</b> | <b>91</b> | <b>12</b> | 15        | 93        | 15       |
|          |           |           |           | 14        | 94        | 13       |

Tabela 17 – Resultados do SEDA e do DIPSO na Geração Final (*MAX\_GEN*)

| Problema | SEDA |    |    | DIPSO     |           |           |
|----------|------|----|----|-----------|-----------|-----------|
|          | M    | TW | W  | M         | TW        | W         |
| KC 4×5   | 11   | 32 | 10 | <b>12</b> | <b>32</b> | <b>8</b>  |
|          |      |    |    | <b>11</b> | <b>34</b> | <b>9</b>  |
|          |      |    |    | <b>13</b> | <b>33</b> | <b>7</b>  |
|          |      |    |    | <b>11</b> | <b>32</b> | <b>10</b> |
| KC 8×8   | 16   | 73 | 13 | <b>16</b> | <b>76</b> | <b>13</b> |
|          |      |    |    | <b>15</b> | <b>75</b> | <b>12</b> |
|          |      |    |    | <b>18</b> | <b>77</b> | <b>11</b> |
| KC 10×7  | 12   | 60 | 12 | <b>11</b> | <b>62</b> | <b>10</b> |
|          |      |    |    | <b>11</b> | <b>61</b> | <b>11</b> |
|          |      |    |    | <b>12</b> | <b>60</b> | <b>12</b> |
| KC 10×10 | 8    | 41 | 7  | <b>7</b>  | <b>42</b> | <b>6</b>  |
|          |      |    |    | <b>8</b>  | <b>41</b> | <b>7</b>  |
| KC 15×10 | 12   | 91 | 11 | <b>13</b> | <b>93</b> | <b>10</b> |
|          |      |    |    | <b>12</b> | <b>91</b> | <b>11</b> |

o SEDA. Para o DIPSO, a possibilidade de evolução da população exercida pelo algoritmo não permitiu que o OCD-HV retornasse os mesmos resultados, pois o algoritmo possui um operador que gera diversidade dentro da população e o OCD-HV é incapaz de retornar um resultado definitivo de convergência. Ou seja, o algoritmo estava em uma estabilidade prévia, mas a população continuava em movimento no espaço de busca. Os valores para hipervolume mostraram que SEDA e DIPSO são algoritmos com boas possibilidades na evolução populacional, enquanto o PSO não converge para uma boa posição no espaço de busca, considerando a adaptação realizada e o problema. Os parâmetros do PSO para quatro problemas garantem a convergência da população, mas não garantem um bom resultado. Por outro lado, a estabilidade prévia detectada pelo OCD-HV, indica que DIPSO, um PSO híbrido, tem a capacidade de evitar mínimos locais.

Esse experimento indica que a verificação da estabilidade do resultado no espaço objetivo, realizada pelo OCD-HV, não foi suficiente para determinar o melhor ponto de parada para DIPSO, entendendo que o melhor ponto de parada seria o de melhor solução. O OCD-HV identificou um ponto de estabilidade no conjunto de aproximação, porém o experimento mostrou que o algoritmo estava em um estado em que poderia encontrar melhores soluções. Nesse sentido, há a necessidade de se entender qual o estado que o algoritmo estava quando o OCD-HV identificou a estabilidade e sugeriu a convergência, ou seja, há a necessidade de se observar de forma comportamental o algoritmo, para melhor compreender a evolução do mesmo.

Para analisar o ponto de convergência apresentado pela experimento (Tabela 14), de modo a entender o comportamento de DIPSO, utilizaremos CR e DR. A Tabela 18 apresenta, além das colunas com a geração e hipervolume: 1) os valores de CR e DR relacionados com tal ponto de convergência; 2)  $CR_M$  e  $DR_m$  determinados previamente por um experimento de quinze execuções independentes de DIPSO para cada problema.

Tabela 18 – Análise do Ponto de Convergência de DIPSO (OCD-HV)

| Problema | Geração | Hipervolume | CR     | DR     | $CR_M$ | $DR_m$ |
|----------|---------|-------------|--------|--------|--------|--------|
| KC 4×5   | 18      | 562K        | 0,4350 | 0,3775 | 0,61   | 0,13   |
| KC 8×8   | 48      | 2.294K      | 0,3050 | 0,6975 | 0,56   | 0,35   |
| KC 10×7  | 57      | 1.725K      | 0,4875 | 0,5225 | 0,54   | 0,42   |
| KC 10×10 | 47      | 509K        | 0,3325 | 0,6275 | 0,59   | 0,35   |
| KC 15×10 | 112     | 3.883K      | 0,2850 | 0,7650 | 0,60   | 0,36   |

Observando CR e DR, tem-se que CR não era superior a 0,50 em nenhum problema e que DR apresentava valor menor do 0,50 apenas no problema KC 4×5, que é o único problema que DR já mostra uma exaustão na pesquisa, pois perdeu mais de 60% da capacidade de busca. Comparando CR e DR com  $CR_M$  e  $DR_m$ , observa-se que para o KC 4×5, DIPSO já estava próximo do ponto de menor produtividade. O mesmo pode

ser considerado para o KC  $10 \times 7$ . Em relação aos outros problemas, CR e DR possuem uma distância considerável de seus valores de referência, estão entre 40% e 60% de seus intervalos de valores. Assim, analisando esses valores, tem-se que DIPSO possui uma probabilidade razoável de melhorar seus resultados de momento, pois, DR mostra atividade ( $DR > 0,60$ ) em quatro problemas e em três problemas CR não chegou a 0,40, sendo que, a melhora dos resultados realmente ocorreu, conforme apresentado no comparativo entre as Tabelas 16 e 17.

### 4.3 Ponto de Parada de AEMO's com FJSP

Os experimentos da seção anterior apresentaram a necessidade de indicadores para avaliar características do comportamento de um algoritmo. Assim, a proposta de CR e DR auxiliou na análise, indicando propriedades importantes sobre a possibilidade do algoritmo conseguir melhorar o conjunto de aproximação em um determinado instante. Nesta seção, o foco do experimento é identificar diversos pontos de parada de AEMO's e comparar os resultados encontrados. Os algoritmos utilizados foram DIPSO, SEDA e SEDASI, além de NSGA2 e SPEA2 (dois algoritmos consolidados na literatura). Os pontos de parada foram determinados por PFA e três versões do OCD, que se diferenciam pelo indicador utilizado, hipervolume (OCD-HV), IGD (OCD-IGD) e  $S_3$  (OCD- $S_3$ ), sendo eles independentes para encontrar o respectivo ponto de parada. Os pontos de referência utilizados para o hipervolume foram os mesmos adotados na seção anterior e os do IGD foram as melhores soluções encontradas na literatura para cada problema, que podem ser obtidas em (Carvalho, 2015) e (Chiang; Lin, 2013).

A fim de generalizar, somente o termo “ponto de parada” será utilizado para todos os algoritmos. Entende-se que o OCD-HV, conforme apresentado na seção 2.4.1, foi proposto para identificar o ponto de convergência de um AEMO. Como IGD realiza uma medida de distância entre o conjunto de aproximação e um conjunto de referência, em que se pode observar a evolução do conjunto de aproximação, consideramos que OCD-IGD, assim como OCD-HV, identifica um ponto de convergência. A utilização de OCD associado ao indicador  $S_3$  permite observar a estabilidade do espaço de soluções. Se há estabilidade no espaço de soluções, é provável que há estabilidade no espaço de objetivo e, assim como OCD-HV e OCD-IGD, entende-se que OCD- $S_3$  determina um ponto de convergência. Em relação ao PFA, observa-se que ele identifica um ponto de parada para o algoritmo de modo que seja o mais próximo de um momento de pouca possibilidade de evoluir o resultado, pois utiliza uma comparação baseada no CR e DR. Assim, conforme discutido na Seção 2.5, o ponto de convergência tende a ser um ponto de parada para um algoritmo e, como entende-se que PFA não determina convergência, utilizaremos o termo “ponto de parada” para melhor efetuarmos as comparações neste experimento.

O experimento foi realizado com os *benchmarks* KC e BR, descritos na seção 4.1.1.

Cada algoritmo foi executado quarenta e cinco vezes para cada problema, sendo que, quinze execuções foram para realizar a estimativa de  $CR_M$  e  $DR_m$  e trinta para observar o resultado. O parâmetro ‘erro’ do PFA foi mantido com o valor padrão. A variância (parâmetro *varLimit*) utilizada pelo OCD-HV e OCD-IGD foi de 0,0005 e para o OCD- $S_3$  foi de 0,09. A diferença entre as variâncias é devido à diferença dos indicadores, hipervolume e IGD observam o espaço objetivo e o  $S_3$  observa o espaço de soluções. Todos as implementações do OCD utilizam  $nPreGen = 5$ , ou seja, a quantidade de gerações analisadas para detectar a convergência é cinco. Os algoritmos foram processados em quatro computadores com as seguintes configurações de processador e memória: 1) Intel Xeon E7 de 16 GB de RAM; 2 e 3) dois Intel Core I5 de 8 GB de RAM, um de quarta e outro de oitava geração; e 4) Intel Core 2 DUO com 2 GB de RAM. A divisão de processamento dos algoritmos e problemas entre as máquinas foi realizada de forma aleatória segundo a demanda. Todos os resultados apresentados são médias calculadas.

As Tabelas 19, 20, 21, 22 e 23 apresentam os parâmetros utilizados para a execução, respectivamente, de DIPSO, SEDA, SEDASI, NSGA2 e SPEA2. A Tabela 19, que apresenta os parâmetros de DIPSO, é análoga à Tabela 12. A Tabela 20, que apresenta os parâmetros de SEDA, é análoga à Tabela 11. A Tabela 21 apresenta os parâmetros de SEDASI e é análoga à Tabela 9. As Tabelas 22 e 23 apresentam os parâmetros de NSGA2 e SPEA2, respectivamente, sendo que possuem as mesmas colunas, ‘Problema’, ‘Mutaç o’, ‘POP’ e ‘MAX\_GEN’, com os mesmos significados das tabelas anteriores.

Os parâmetros de DIPSO foram replicados de Carvalho (2015), exceto em rela o aos problemas BR Mk08, BR Mk09 e BR Mk10, que foram executados com um n mero de gera o menor. Os parâmetros de SEDA foram os mesmos utilizados no experimento da se o 4.1.1, por m com uma maior quantidade de gera es, a fim de melhor observar o comportamento do algoritmo. Os parâmetros de SEDASI foram os mesmos do algoritmo SEDA, exceto em rela o aos problemas BR Mk08, BR Mk09 e BR Mk10, que foram executados com uma popula o menor. Os parâmetros de NSGA2 e SPEA2 foram definidos baseados no DIPSO, algoritmo que mais se assemelha a eles neste experimento, por m experimentos mostraram que o tamanho da popula o usado nos problemas BR poderia ser menor para ambos os algoritmos. Al m disso, o tamanho da popula o e a quantidade de gera es de NSGA2 para os problemas BR Mk08, BR Mk09 e BR Mk10 foram menores, em rela o aos outros problemas BR. Como descrito, os problemas BR Mk08, BR Mk09 e BR Mk10 foram executados com parâmetros menores do que os outros problemas BR nos algoritmos DIPSO, SEDASI e NSGA2. Isso foi estabelecido devido aos recursos computacionais dispon veis para realiza o do experimento.

As Tabelas 24, 25, 26, 27 e 28 apresentam um comparativo entre as solu es de cada ponto de parada. Esse comparativo foi constru do a partir da obten o das solu es n o dominadas entre todos os conjuntos de aproxima o de cada ponto de parada determinado pelos algoritmos. Portanto, ap s a coluna ‘Problema’, que lista os problemas do estudo, a

Tabela 19 – DIPSO - Parâmetros

| Problema          | $p_{cc}$ | $p_{dx}$ | Mutação | $POP$ | $MAX\_GEN$ |
|-------------------|----------|----------|---------|-------|------------|
| KC                | 80%      | 20%      | 5%      | 400   | 800        |
| BR Mk01 - BR Mk07 | 70%      | 30%      | 1%      | 2000  | 1400       |
| BR Mk08           | 70%      | 30%      | 1%      | 2000  | 800        |
| BR Mk09           | 70%      | 30%      | 1%      | 2000  | 800        |
| BR Mk10           | 70%      | 30%      | 1%      | 2000  | 800        |

Tabela 20 – SEDA - Parâmetros

| Problema          | $POP$ | $MAX\_GEN$ | $EA(\%)$ |
|-------------------|-------|------------|----------|
| KC $4 \times 5$   | 200   | 120        | 70       |
| KC $8 \times 8$   | 500   | 400        | 70       |
| KC $10 \times 7$  | 400   | 400        | 70       |
| KC $10 \times 10$ | 400   | 400        | 70       |
| KC $15 \times 10$ | 800   | 400        | 70       |
| BR Mk01           | 1000  | 250        | 70       |
| BR Mk02           | 1000  | 250        | 70       |
| BR Mk03           | 1000  | 300        | 70       |
| BR Mk04           | 1000  | 250        | 70       |
| BR Mk05           | 1000  | 250        | 70       |
| BR Mk06           | 1000  | 350        | 70       |
| BR Mk07           | 1000  | 250        | 70       |
| BR Mk08           | 1000  | 350        | 70       |
| BR Mk09           | 1000  | 350        | 70       |
| BR Mk10           | 1000  | 350        | 70       |

coluna ‘Total de SND’ contém a quantidade total de Soluções Não Dominadas (SND) do processo realizado com os conjuntos de aproximação. A coluna referente a cada algoritmo de ponto de parada refere-se à relação da quantidade de soluções não dominadas que tal algoritmo apresentou em seu conjunto de aproximação pela quantidade de soluções apresentadas. Assim, na Tabela 24, referente ao comparativo do algoritmo DIPSO, o  $OCD-S_3$  encontrou vinte soluções para o problema BR Mk07 e, dessas, quatorze foram consideradas soluções não dominadas, em relação às soluções dos conjuntos de aproximação apresentados pelos outros algoritmos. Outra informação na tabela é o destaque em negrito para cada problema, que representa o melhor resultado do comparativo, conjunto de aproximação com maior número de soluções não dominadas.

A Tabela 24 com o resultado do comparativo de DIPSO, mostra o destaque para o  $OCD-S_3$  que, dos quinze problemas, em doze apresentou o melhor conjunto de aproximação.  $OCD-HV$  apresentou cinco destaques,  $OCD-IGD$  apresentou quatro e PFA, dois. Também vale destacar que o  $OCD-HV$  não conseguiu identificar a convergência em seis problemas. Observando os problemas KC, nota-se os bons resultados de  $OCD-S_3$  e PFA,

Tabela 21 – SEDASI - Parâmetros

| Problema          | <i>POP</i> | <i>MAX_GEN</i> |
|-------------------|------------|----------------|
| KC $4 \times 5$   | 200        | 120            |
| KC $8 \times 8$   | 500        | 400            |
| KC $10 \times 7$  | 400        | 400            |
| KC $10 \times 10$ | 400        | 400            |
| KC $15 \times 10$ | 800        | 400            |
| BR Mk01           | 1000       | 250            |
| BR Mk02           | 1000       | 250            |
| BR Mk03           | 1000       | 300            |
| BR Mk04           | 1000       | 250            |
| BR Mk05           | 1000       | 250            |
| BR Mk06           | 1000       | 350            |
| BR Mk07           | 1000       | 250            |
| BR Mk08           | 500        | 300            |
| BR Mk09           | 500        | 300            |
| BR Mk10           | 500        | 300            |

Tabela 22 – NSGA2 - Parâmetros

| Problema          | Mutação | <i>POP</i> | <i>MAX_GEN</i> |
|-------------------|---------|------------|----------------|
| KC                | 5%      | 400        | 800            |
| BR Mk01 - BR Mk07 | 1%      | 1000       | 1400           |
| BR Mk08           | 1%      | 500        | 700            |
| BR Mk09           | 1%      | 500        | 700            |
| BR Mk10           | 1%      | 500        | 700            |

Tabela 23 – SPEA2 - Parâmetros

| Problema | Mutação | <i>POP</i> | <i>MAX_GEN</i> |
|----------|---------|------------|----------------|
| KC       | 5%      | 400        | 800            |
| BR       | 1%      | 1000       | 1400           |

sendo que foi com esses problemas que  $S_3$ , CR e DR foram projetados. Na generalização para os problemas BR, OCD- $S_3$  manteve o mesmo padrão e somente em dois problemas, BR Mk09 e BR Mk10, não apresentou o melhor conjunto de aproximação, o que foi feito pelo OCD-IGD. Em contrapartida, PFA não apresentou bons resultados com os problemas BR, sendo que somente no BR Mk02 que ele apresentou soluções não dominadas. Nos outros problemas, apesar de apresentar soluções, todas foram dominadas por soluções dos demais conjuntos. Em relação ao OCD-IGD, ele apresentou soluções não dominadas para todos os problemas, sendo destaque para os problemas KC  $4 \times 5$ , BR Mk08, BR Mk09 e BR Mk10.

Tabela 24 – DIPSO - FJSP - Comparação do uso de indicadores para parada do algoritmo

| Problema          | Total de SND | OCD-HV       | OCD-IGD      | OCD- $S_3$   | PFA        |
|-------------------|--------------|--------------|--------------|--------------|------------|
| KC $4 \times 5$   | 4            | <b>4/4</b>   | <b>4/4</b>   | <b>4/4</b>   | <b>4/4</b> |
| KC $8 \times 8$   | 3            | <b>3/3</b>   | 2/4          | <b>3/3</b>   | 2/2        |
| KC $10 \times 7$  | 3            | 2/4          | 2/4          | 2/4          | <b>3/3</b> |
| KC $10 \times 10$ | 3            | 2/2          | 1/2          | <b>3/3</b>   | 2/2        |
| KC $15 \times 10$ | 4            | 0/4          | 0/4          | <b>4/4</b>   | 3/3        |
| BR Mk1            | 10           | <b>10/10</b> | 3/13         | <b>10/10</b> | 0/24       |
| BR Mk2            | 8            | <b>8/8</b>   | 3/10         | <b>8/8</b>   | 7/7        |
| BR Mk3            | 15           | 11/11        | 9/10         | <b>15/15</b> | 0/18       |
| BR Mk4            | 20           | -            | 8/21         | <b>20/20</b> | 0/47       |
| BR Mk5            | 11           | -            | 10/12        | <b>11/11</b> | 0/28       |
| BR Mk6            | 92           | -            | 32/94        | <b>80/90</b> | 0/29       |
| BR Mk7            | 19           | -            | 13/16        | <b>14/20</b> | 0/22       |
| BR Mk8            | 9            | <b>9/9</b>   | <b>9/9</b>   | <b>9/9</b>   | 0/16       |
| BR Mk9            | 70           | -            | <b>48/64</b> | 30/86        | 0/55       |
| BR Mk10           | 71           | -            | <b>69/69</b> | 6/76         | 0/17       |

No comparativo da Tabela 25, referente ao SEDA, observamos o comportamento identificado no experimento da seção 4.2, em que OCD-HV apresentou bons resultados para identificar o ponto de parada do algoritmo. Analisando os valores da Tabela 25, OCD-HV apresenta o melhor conjunto de aproximação para quatorze problemas, sendo que não obteve o melhor resultado somente no BR Mk10, além de não ter identificado um ponto de parada para esse problema. O OCD- $S_3$  foi o melhor em dois problemas, KC  $10 \times 10$  e BR Mk10, sendo que, nesse último, ele foi o único algoritmo a apresentar soluções não dominadas. PFA apresentou o melhor conjunto de aproximação em um problema, KC  $10 \times 10$ . De modo geral, OCD- $S_3$  e PFA apresentaram soluções não dominadas para todos os problemas, exceto para o BR Mk10 em que todas as soluções de PFA foram dominadas pelas de OCD- $S_3$ . Tal frequência não foi observada com o OCD-IGD, que não obteve um

rendimento regular em SEDA como em DIPSO. A utilização do OCD-IGD com o SEDA não apresentou soluções não dominadas para oito problemas.

Tabela 25 – SEDA - FJSP - Comparação do uso de indicadores para parada do algoritmo

| Problema          | Total de SND | OCD-HV       | OCD-IGD | OCD- $S_3$   | PFA        |
|-------------------|--------------|--------------|---------|--------------|------------|
| KC $4 \times 5$   | 4            | <b>4/4</b>   | 3/4     | 3/4          | 2/2        |
| KC $8 \times 8$   | 5            | <b>5/5</b>   | 3/3     | 1/4          | 1/1        |
| KC $10 \times 7$  | 3            | <b>3/3</b>   | 2/3     | 2/2          | 2/2        |
| KC $10 \times 10$ | 3            | <b>3/3</b>   | 2/3     | <b>3/3</b>   | <b>3/3</b> |
| KC $15 \times 10$ | 2            | <b>2/2</b>   | 0/3     | 1/2          | 1/1        |
| BR Mk1            | 18           | <b>18/18</b> | 1/25    | 16/18        | 4/6        |
| BR Mk2            | 10           | <b>10/10</b> | 5/13    | 1/11         | 1/2        |
| BR Mk3            | 10           | <b>9/10</b>  | 0/14    | 8/8          | 5/6        |
| BR Mk4            | 20           | <b>19/20</b> | 5/33    | 3/18         | 5/6        |
| BR Mk5            | 15           | <b>15/15</b> | 0/21    | 8/16         | 6/6        |
| BR Mk6            | 78           | <b>69/79</b> | 0/64    | 11/69        | 27/55      |
| BR Mk7            | 17           | <b>17/17</b> | 0/31    | 9/13         | 3/6        |
| BR Mk8            | 11           | <b>11/11</b> | 0/24    | 3/4          | 6/7        |
| BR Mk9            | 64           | <b>42/53</b> | 0/31    | 11/81        | 27/55      |
| BR Mk10           | 86           | -            | 0/30    | <b>86/86</b> | 0/62       |

A análise do comparativo apresentado na Tabela 26, referente ao SEDASI, mostra o bom desempenho de OCD- $S_3$  e PFA. Apresentando dez melhores conjuntos de aproximação, OCD- $S_3$  é o que recebeu mais destaque neste comparativo, sendo que PFA identificou nove melhores conjuntos de aproximação. Porém, em quatro problemas, BR Mk4, BR Mk8, BR Mk9 e BR Mk10, OCD- $S_3$  não identificou um ponto de parada, o que não ocorreu com PFA, que apresentou soluções não dominadas para todos os problemas. Em relação ao OCD-HV, esse não identificou o ponto de parada em cinco problemas, sendo destaque para quatro problemas KC. OCD-IGD foi um pouco melhor que o OCD-HV, apresentando seis destaques de melhor conjunto de aproximação, porém não identificou um ponto de parada para três problemas.

Além de focar o estudo nos algoritmos implementados para o FJSP, incluímos neste experimento o NSGA2 e SPEA2, AEMO's clássicos da literatura, para observarmos o comportamento dos indicadores, associados aos seus respectivos algoritmos, na identificação do ponto de parada em AEMO's difundidos na literatura. A Tabela 27 mostra o resultado do comparativo do NSGA2. Dos quinze problemas, OCD-HV apresentou o melhor conjunto de aproximação em doze e foi o de melhor desempenho. OCD-IGD e PFA apresentaram melhores conjuntos de aproximação em três problemas e OCD- $S_3$ , dois. Em todos os problemas OCD-HV apresentou soluções não dominadas, enquanto OCD-IGD

Tabela 26 – SEDASI - FJSP - Comparação do uso de indicadores para parada do algoritmo

| Problema          | Total de SND | OCD-HV     | OCD-IGD      | OCD- $S_3$   | PFA          |
|-------------------|--------------|------------|--------------|--------------|--------------|
| KC $4 \times 5$   | 4            | <b>4/4</b> | <b>4/4</b>   | <b>4/4</b>   | <b>4/4</b>   |
| KC $8 \times 8$   | 5            | <b>5/5</b> | <b>5/5</b>   | <b>5/5</b>   | 4/4          |
| KC $10 \times 7$  | 3            | <b>3/3</b> | <b>3/3</b>   | <b>3/3</b>   | <b>3/3</b>   |
| KC $10 \times 10$ | 3            | <b>3/3</b> | <b>3/3</b>   | <b>3/3</b>   | <b>3/3</b>   |
| KC $15 \times 10$ | 2            | 1/2        | 1/2          | <b>2/2</b>   | <b>2/2</b>   |
| BR Mk1            | 18           | 3/17       | 13/19        | <b>16/18</b> | 13/16        |
| BR Mk2            | 11           | 6/9        | 10/11        | <b>11/11</b> | <b>11/11</b> |
| BR Mk3            | 8            | 2/6        | 3/11         | <b>7/8</b>   | <b>7/8</b>   |
| BR Mk4            | 15           | -          | 8/17         | -            | <b>14/15</b> |
| BR Mk5            | 12           | -          | 9/15         | <b>10/14</b> | <b>10/13</b> |
| BR Mk6            | 58           | -          | -            | <b>48/48</b> | 11/67        |
| BR Mk7            | 14           | 8/11       | <b>13/15</b> | 11/14        | 11/11        |
| BR Mk8            | 9            | 7/7        | <b>9/9</b>   | -            | <b>9/9</b>   |
| BR Mk9            | 37           | -          | -            | -            | 37/37        |
| BR Mk10           | 53           | -          | -            | -            | 53/53        |

não apresentou em quatro, OCD- $S_3$  em dois e PFA, em um problema.

Tabela 27 – NSGA2 - FJSP - Comparação do uso de indicadores para parada do algoritmo

| Problema          | Total de SND | OCD-HV       | OCD-IGD      | OCD- $S_3$ | PFA        |
|-------------------|--------------|--------------|--------------|------------|------------|
| KC $4 \times 5$   | 4            | <b>4/4</b>   | <b>4/4</b>   | <b>4/4</b> | <b>4/4</b> |
| KC $8 \times 8$   | 5            | <b>5/5</b>   | 3/4          | 3/3        | 3/3        |
| KC $10 \times 7$  | 6            | <b>5/5</b>   | 4/5          | 3/5        | 4/5        |
| KC $10 \times 10$ | 3            | <b>3/3</b>   | <b>3/3</b>   | 1/3        | 1/3        |
| KC $15 \times 10$ | 5            | 3/4          | 1/6          | <b>4/4</b> | <b>4/4</b> |
| BR Mk1            | 11           | <b>10/11</b> | 6/12         | 9/11       | 6/10       |
| BR Mk2            | 7            | <b>5/6</b>   | 0/15         | 3/9        | 4/6        |
| BR Mk3            | 7            | 2/7          | 2/6          | 4/7        | <b>5/5</b> |
| BR Mk4            | 22           | 10/13        | <b>20/23</b> | 4/19       | 3/3        |
| BR Mk5            | 15           | <b>11/15</b> | 7/21         | 10/18      | 4/4        |
| BR Mk6            | 13           | <b>9/27</b>  | 0/22         | 0/21       | 6/9        |
| BR Mk7            | 22           | <b>13/20</b> | 9/30         | 6/25       | 1/7        |
| BR Mk8            | 10           | <b>10/10</b> | 2/17         | 6/10       | 6/12       |
| BR Mk9            | 20           | <b>20/20</b> | 0/90         | 0/69       | 0/23       |
| BR Mk10           | 66           | <b>48/71</b> | 0/120        | 16/71      | 3/4        |

Em relação ao SPEA2, considerando os resultados apresentados pela Tabela 28, temos que o PFA apresentou sete destaques de melhor conjunto de aproximação, seguido pelo OCD- $S_3$  com cinco. OCD-HV apresentou quatro e OCD-IGD apenas um, sendo que em onze problemas, ele não foi capaz de apresentar soluções não dominadas e o OCD-HV em quatro. Em relação ao OCD- $S_3$  e ao PFA, o primeiro não apresentou soluções não dominadas em três problemas e o segundo em dois.

Tabela 28 – SPEA2 - FJSP - Comparação do uso de indicadores para parada do algoritmo

| Problema          | Total de SND | OCD-HV       | OCD-IGD    | OCD- $S_3$   | PFA          |
|-------------------|--------------|--------------|------------|--------------|--------------|
| KC $4 \times 5$   | 6            | <b>6/6</b>   | <b>6/6</b> | 5/5          | 5/5          |
| KC $8 \times 8$   | 7            | 1/6          | 0/14       | <b>7/7</b>   | 3/6          |
| KC $10 \times 7$  | 5            | 2/4          | 1/3        | <b>5/5</b>   | 1/3          |
| KC $10 \times 10$ | 1            | 0/3          | 0/8        | <b>1/1</b>   | 0/4          |
| KC $15 \times 10$ | 7            | 0/7          | 0/6        | <b>4/6</b>   | <b>4/5</b>   |
| BR Mk1            | 11           | 9/13         | 0/13       | <b>11/11</b> | <b>11/11</b> |
| BR Mk2            | 12           | 0/19         | 0/21       | 5/11         | <b>9/9</b>   |
| BR Mk3            | 5            | <b>3/10</b>  | 0/19       | 2/2          | 2/2          |
| BR Mk4            | 23           | 15/34        | 3/33       | 10/35        | <b>20/24</b> |
| BR Mk5            | 24           | 1/23         | 0/30       | 1/20         | <b>24/24</b> |
| BR Mk6            | 33           | <b>33/33</b> | 0/42       | 0/85         | 0/10         |
| BR Mk7            | 14           | 0/18         | 0/17       | 0/20         | <b>14/14</b> |
| BR Mk8            | 9            | 7/10         | 0/36       | <b>9/9</b>   | 8/8          |
| BR Mk9            | 70           | <b>70/70</b> | 0/89       | 28/35        | 28/51        |
| BR Mk10           | 114          | 48/78        | 0/86       | 1/92         | <b>69/94</b> |

Outro aspecto importante de ser observado neste experimento é a quantidade de execuções de cada algoritmo que não identificou um ponto de parada, ou seja, em que houve falha na identificação do ponto de parada. A Tabela 29 apresenta a relação da quantidade de execuções com falha na identificação pela quantidade de execuções, separando pelo algoritmo executado, pelo problema tratado e pelo algoritmo aplicado na identificação do ponto de parada. Portanto na linha de DIPS0 - BR Mk3 com a coluna OCD-HV indicando o valor 11/30 indica que de trinta execuções, o OCD-HV não identificou um ponto de parada para onze. As linhas em que, para determinado algoritmo com um problema em específico, indicavam que não houve falha em nenhum processo foram excluídas por questão de espaço. Em negrito, estão destacados o menor valor de falha em cada linha, representando o algoritmo que menos falhou para determinada execução.

De forma geral, OCD-IGD foi o algoritmo que menos apresentou falhas, seguido por OCD- $S_3$ , PFA e OCD-HV. O AEMO em que os algoritmos de ponto de parada mais falharam foi o SEDASI, oitocentas falhas, e, conseqüentemente, foi o que mais gerou dificuldade para os algoritmos encontrarem o ponto de parada, ou seja, todos tiveram

seu maior número de falhas com o SEDASI. O que falhou mais com o SEDASI foi o OCD-HV, com 286 falhas e o que falhou menos com o SEDASI foi o OCD-IGD, 143. O AEMO em que os algoritmos de ponto de parada apresentaram o menor número de falhas foi o NSGA2, 65 falhas. O OCD-HV não apresentou falhas somente com o SPEA2, enquanto o OCD-IGD não apresentou falhas com o SEDA, NSGA2 e SPEA2. Semelhante ao OCD- $S_3$ , que não apresentou falhas com o NSGA2 e com o SPEA2. No caso do PFA, apresentou falhas com todos os algoritmos, sendo com o SEDA seu menor número de falhas, 36.

Os resultados apresentados mostram que para cada AEMO, um ou dois algoritmos de ponto de parada se destacam com os melhores resultados. O OCD- $S_3$  apresentou bons resultados para o DIPSO, SEDASI e, com um menor destaque, SPEA2, enquanto o PFA se destacou com o SEDASI e SPEA2. Em relação ao número de falhas, OCD- $S_3$  e PFA ficaram à frente do OCD-HV, algoritmo que utiliza o indicador que mais se destaca na literatura e, de modo geral, OCD- $S_3$  apresentou equivalência na detecção de resultados com o OCD-HV, sendo que quando associado ao DIPSO, o algoritmo que serviu de base para sua idealização, obteve o melhor desempenho.

Para melhor representar os resultados obtidos pelo experimento, os gráficos com as soluções de cada algoritmo, considerando todos algoritmos de ponto de parada, para os problemas KC  $10 \times 7$ , KC  $15 \times 10$ , BR Mk3 e BR Mk8 são apresentados nas Figuras 17, 18, 19 e 20, respectivamente. Em cada figura são apresentados cinco gráficos, sendo que cada gráfico refere-se a um algoritmo do experimento e mostram as soluções obtidas no ponto de parada que cada algoritmo determinou.

Os gráficos apresentados para o problema KC  $10 \times 7$ , Figura 17, apresentam DIPSO, SEDA, SEDASI e NSGA2 com soluções semelhantes, porém, no SPEA2, percebe-se diferenças nas soluções, sendo que o OCD- $S_3$  apresenta uma quantidade maior de soluções e essas dominam algumas soluções apresentadas pelos outros algoritmos. Os gráficos da Figura 18 são referentes ao problema KC  $15 \times 10$ . No gráfico de DIPSO percebe-se algumas soluções de todos algoritmos de ponto de parada muito próximas, além de algumas soluções do OCD-IGD que estão em posições que são dominadas por outras. No gráfico de SEDA há algumas soluções sobrepostas e não há como concluir se há dominância na relação entre elas e, além dessas, há duas soluções. Uma do OCD-HV que é não dominada e outra solução do OCD- $S_3$  que está sendo dominada. No gráfico do SEDASI, percebe-se as soluções agrupadas e no do NSGA2 é perceptível que algumas soluções apresentadas pelo OCD-IGD são dominadas por outras. No gráfico do SPEA2, as soluções se apresentam de forma que se identifique que soluções do OCD-HV e do OCD-IGD são dominadas.

Os gráficos dos problemas BR Mk3 e BR Mk8, Figuras 19 e 20, apresentam um grande número soluções quando comparados aos problemas KC. Essa é uma característica dos problemas BR, que possuem várias soluções. Nos gráficos de BR MK3, DIPSO apresenta todas as soluções agrupadas, exceto as do PFA, dominadas pelas demais. No gráfico de

Tabela 29 – Relação de falhas na identificação do ponto de parada

| Algoritmo - Problema       | OCD-HV      | OCD-IGD         | OCD- $S_3$   | PFA         |
|----------------------------|-------------|-----------------|--------------|-------------|
| DIPSO - KC $4 \times 5$    | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 14/30       |
| DIPSO - KC $8 \times 8$    | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 19/30       |
| DIPSO - KC $10 \times 7$   | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 18/30       |
| DIPSO - KC $10 \times 10$  | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 16/30       |
| DIPSO - KC $15 \times 10$  | <b>0/30</b> | <b>0/30</b>     | 8/30         | 10/30       |
| DIPSO - BR Mk3             | 11/30       | <b>0/30</b>     | 23/30        | <b>0/30</b> |
| DIPSO - BR Mk4             | 30/30       | <b>0/30</b>     | 12/30        | <b>0/30</b> |
| DIPSO - BR Mk5             | 30/30       | <b>0/30</b>     | 29/30        | 17/30       |
| DIPSO - BR Mk6             | 30/30       | <b>0/30</b>     | <b>0/30</b>  | <b>0/30</b> |
| DIPSO - BR Mk7             | 30/30       | <b>0/30</b>     | 28/30        | <b>0/30</b> |
| DIPSO - BR Mk8             | 21/30       | <b>0/30</b>     | 29/30        | <b>0/30</b> |
| DIPSO - BR Mk9             | 30/30       | 3/30            | 15/30        | <b>0/30</b> |
| DIPSO - BR Mk10            | 30/30       | 1/30            | 28/30        | <b>0/30</b> |
| SEDA - KC $10 \times 7$    | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 4/30        |
| SEDA - KC $10 \times 10$   | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 8/30        |
| SEDA - KC $15 \times 10$   | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 1/30        |
| SEDA - BR Mk3              | <b>0/30</b> | <b>0/30</b>     | 3/30         | <b>0/30</b> |
| SEDA - BR Mk6              | 1/30        | <b>0/30</b>     | 1/30         | 6/30        |
| SEDA - BR Mk7              | <b>0/30</b> | <b>0/30</b>     | 4/30         | <b>0/30</b> |
| SEDA - BR Mk8              | <b>0/30</b> | <b>0/30</b>     | 10/30        | <b>0/30</b> |
| SEDA - BR Mk9              | 12/30       | <b>0/30</b>     | 19/30        | 17/30       |
| SEDA - BR Mk10             | 30/30       | <b>0/30</b>     | 12/30        | <b>0/30</b> |
| SEDASI - KC $4 \times 5$   | 1/30        | 1/30            | <b>0/30</b>  | 10/30       |
| SEDASI - KC $8 \times 8$   | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 23/30       |
| SEDASI - KC $10 \times 7$  | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 9/30        |
| SEDASI - KC $10 \times 10$ | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 20/30       |
| SEDASI - KC $15 \times 10$ | <b>0/30</b> | <b>0/30</b>     | 1/30         | <b>0/30</b> |
| SEDASI - BR Mk1            | 29/30       | <b>1/30</b>     | 5/30         | 11/30       |
| SEDASI - BR Mk2            | 26/30       | 3/30            | <b>1/30</b>  | 7/30        |
| SEDASI - BR Mk3            | 24/30       | <b>0/30</b>     | 23/30        | 14/30       |
| SEDASI - BR Mk4            | 30/30       | <b>18/30</b>    | 30/30        | 20/30       |
| SEDASI - BR Mk5            | 30/30       | 22/30           | <b>10/30</b> | 17/30       |
| SEDASI - BR Mk6            | 30/30       | 30/30           | 29/30        | <b>0/30</b> |
| SEDASI - BR Mk7            | 27/30       | <b>2/30</b>     | 22/30        | 9/30        |
| SEDASI - BR Mk8            | 29/30       | <b>6/30</b>     | 30/30        | 20/30       |
| SEDASI - BR Mk9            | 30/30       | 30/30           | 30/30        | <b>0/30</b> |
| SEDASI - BR Mk10           | 30/30       | 30/30           | 30/30        | <b>0/30</b> |
| NSGA2 - BR Mk9             | 11/30       | <b>0/30</b>     | <b>0/30</b>  | 21/30       |
| NSGA2 - BR Mk10            | 10/30       | <b>0/30</b>     | <b>0/30</b>  | 23/30       |
| SPEA2 - KC $8 \times 8$    | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 7/30        |
| SPEA2 - KC $10 \times 7$   | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 19/30       |
| SPEA2 - KC $10 \times 10$  | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 21/30       |
| SPEA2 - KC $15 \times 10$  | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 18/30       |
| SPEA2 - BR Mk2             | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 16/30       |
| SPEA2 - BR Mk6             | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 20/30       |
| SPEA2 - BR Mk7             | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 20/30       |
| SPEA2 - BR Mk9             | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 14/30       |
| SPEA2 - BR Mk10            | <b>0/30</b> | <b>0/30</b>     | <b>0/30</b>  | 15/30       |
| Total                      | 562/2250    | <b>147/2250</b> | 432/2250     | 484/2250    |

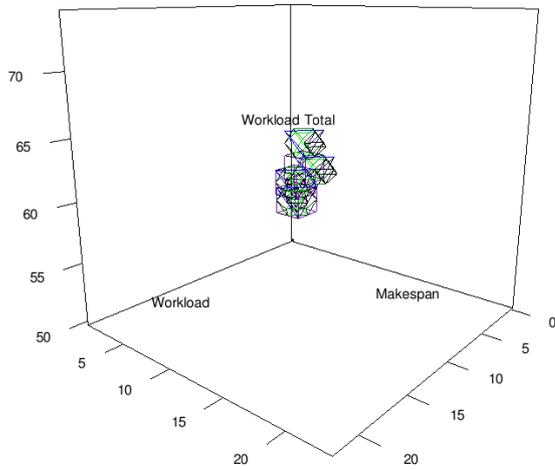
SEDA, consegue-se observar as soluções do OCD-IGD dominadas, sendo que o mesmo ocorre no SEDASI, junto com algumas soluções do OCD-HV. No gráfico do NSGA2, identifica-se soluções do OCD- $S_3$ , OCD-IGD e OCD-HV sendo dominadas e o destaque para o conjunto de soluções do PFA, que estão em uma região que dominam as outras. Por fim, no gráfico do SPEA2, é possível identificar o grande número de soluções de OCD-IGD dominadas. Assim como no gráfico do DIPSO com BR Mk3, no gráfico de DIPSO com BR Mk8, as soluções de PFA estão espalhadas em uma região que são dominadas pelas outras, que estão agrupadas. O mesmo ocorre para o gráfico do SEDA, porém as soluções que estão na região em que são dominadas por outras, são as soluções do OCD-IGD. No gráfico do SEDASI, as soluções estão espalhadas quase que nas mesmas posições e não há dominância entre elas. No gráfico do NSGA2, consegue-se identificar que soluções do PFA e do OCD-IGD são dominadas por outras, além de algumas do OCD- $S_3$ . Da mesma forma, no gráfico do SPEA2, visualiza-se as soluções do OCD-IGD na região em que são dominadas por outras soluções.

Dessa forma, os gráficos mostram as relações de dominância apresentadas nas tabelas anteriores (Tabelas 24, 25, 26, 27 e 28), além de apresentar as soluções dispersas pelo espaço de soluções. Por questões de organização do texto, foram selecionados apenas esses problemas para apresentar os gráficos nesta seção, sendo que todos os gráficos estão disponíveis no apêndice A.

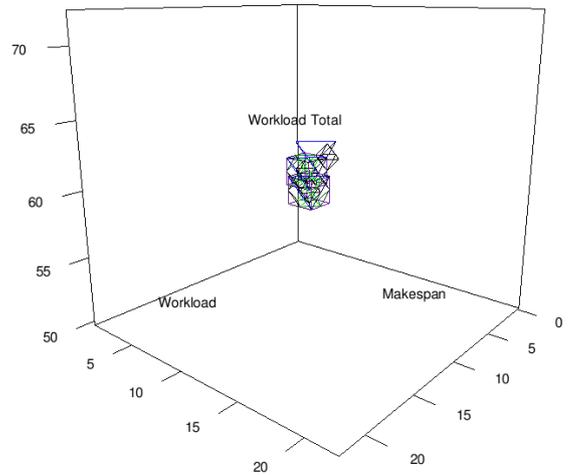
## 4.4 Experimentos de NSGA2 e SPEA2 com problemas ZDT's

Observado o comportamento dos indicadores, associados a algoritmos que identificam a parada de AEMO's, com algoritmos tratando o FJSP na seção anterior, este experimento tem objetivo de verificar o comportamento dos indicadores quando aplicados a AEMO's da literatura, que não são implementados para algum problema em específico, tratando problemas de um *benchmark* criado para realização de experimentos. Assim, utilizamos o NSGA2 e o SPEA2, por serem dois dos mais utilizados AEMO's da literatura e o *benchmark* ZDT (Zitzler; Deb; Thiele, 2000), um dos mais utilizados na literatura para problemas contínuos (Huband et al., 2006). De forma análoga ao experimento anterior, o objetivo foi identificar um ponto de parada para os AEMO's do experimento aplicando os algoritmos OCD, associado ao hipervolume, IGD e  $S_3$ , e PFA. Ambos os algoritmos foram configurados com uma população de cem indivíduos e duzentas gerações.

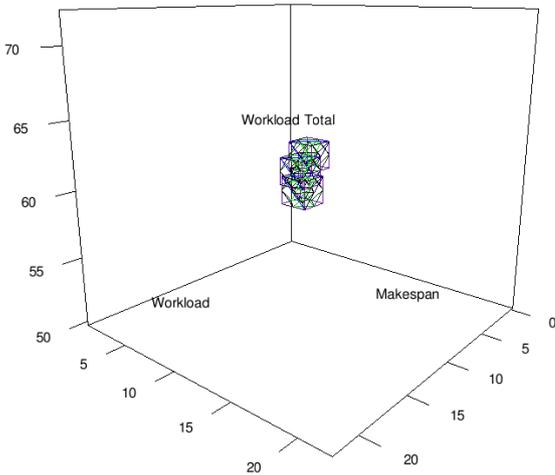
Os algoritmos foram executados quinze vezes para identificar  $CR_M$  e  $DR_m$  e trinta vezes para realizar o experimento, sendo que as soluções consideradas são as soluções não dominadas do conjunto de soluções de todas as execuções. O problema ZDT5 não foi considerado na análise, pois na literatura a maioria das análises não o considera por ser um problema binário (Huband et al., 2006). O parâmetro 'erro' do PFA foi ajustado com



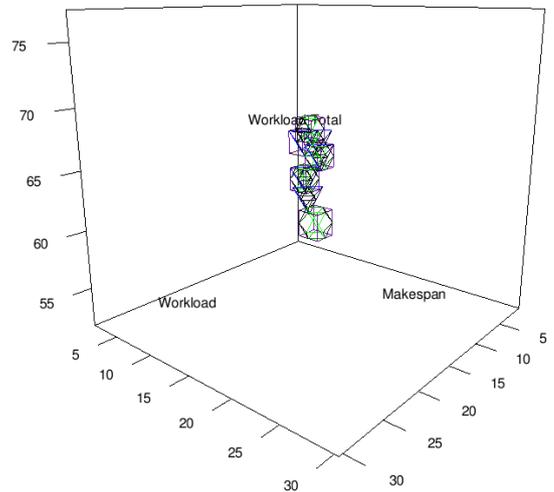
(a) DIPSO



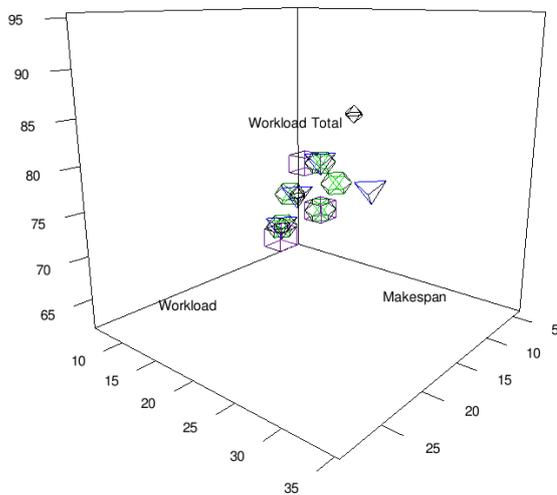
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

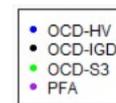
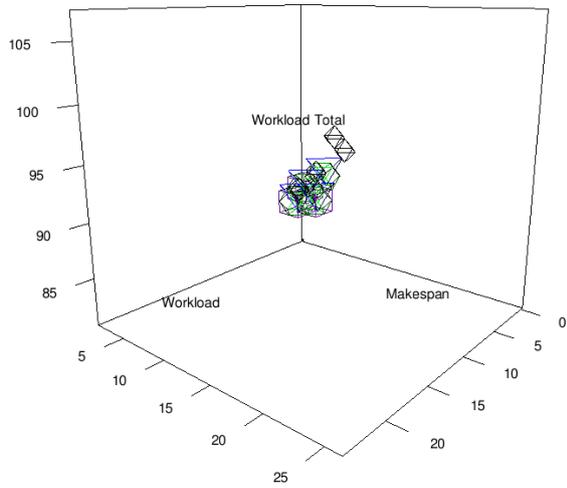
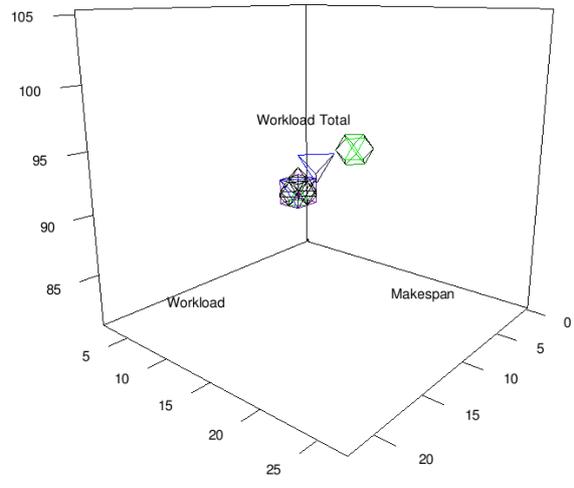


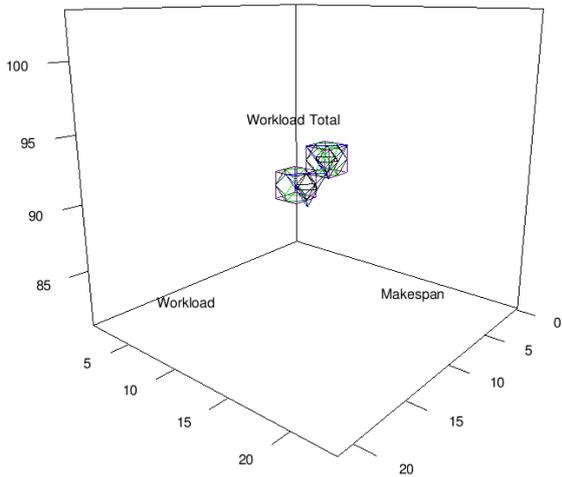
Figura 17 – Gráficos do problema KC10X07



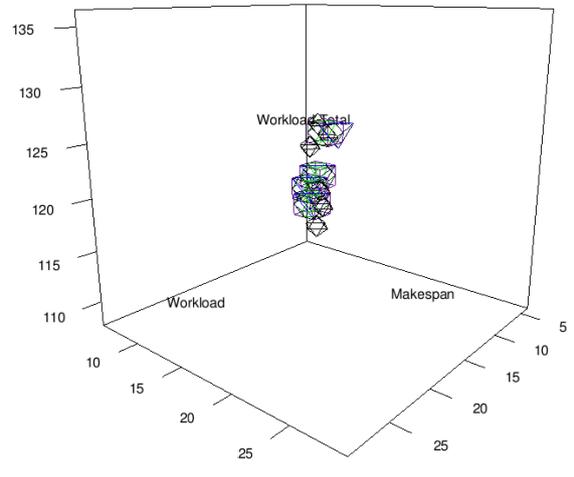
(a) DIPSO



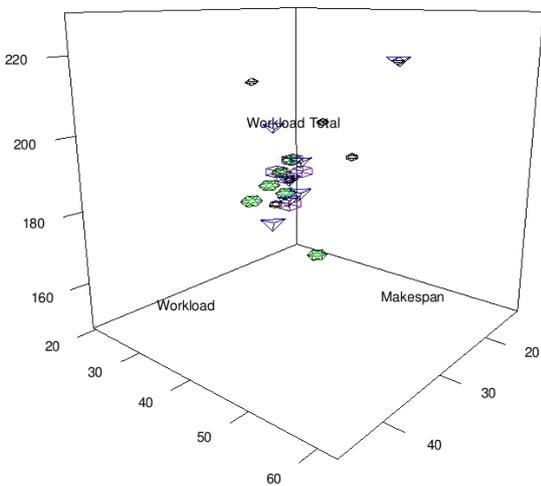
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

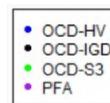
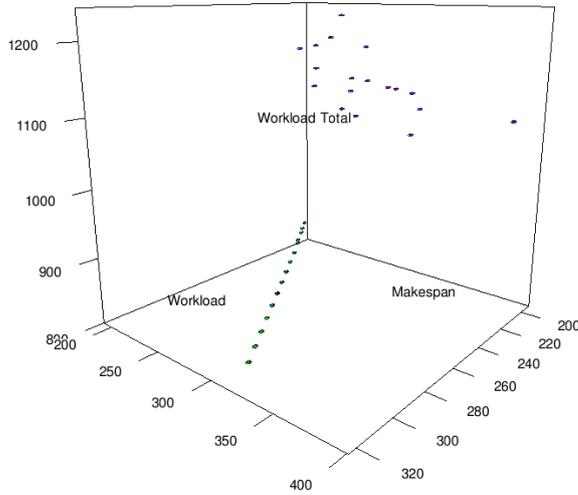
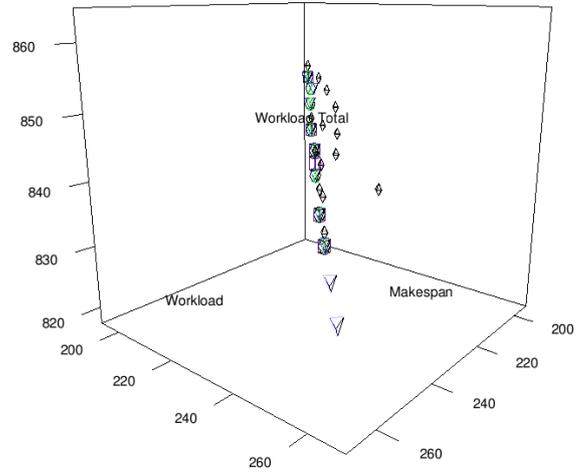


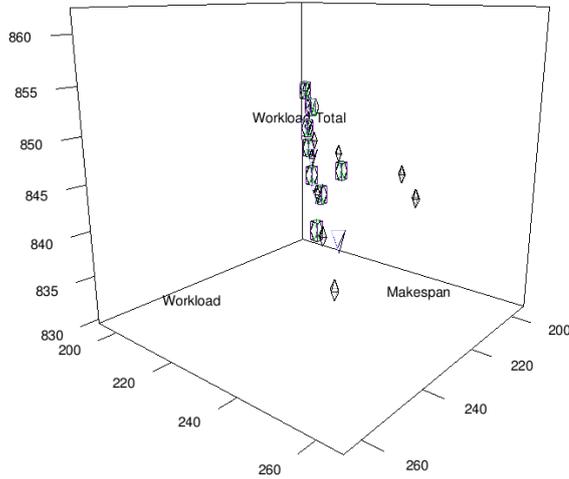
Figura 18 – Gráficos do problema KC15X10



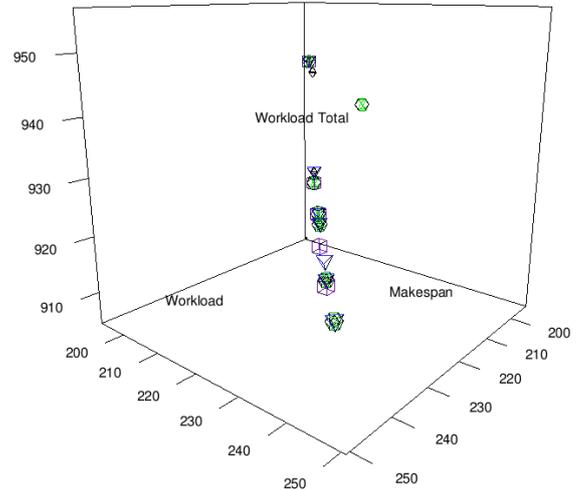
(a) DIPSO



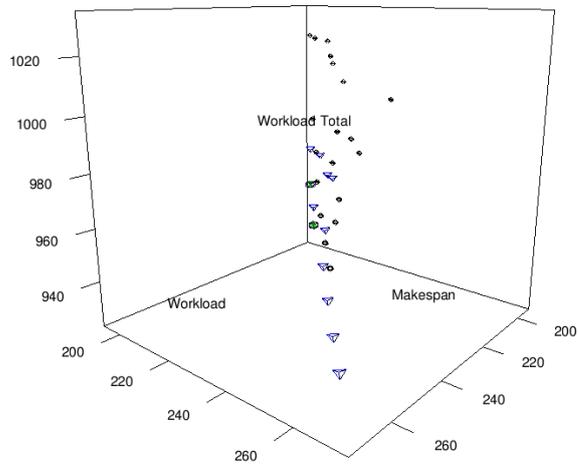
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

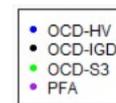
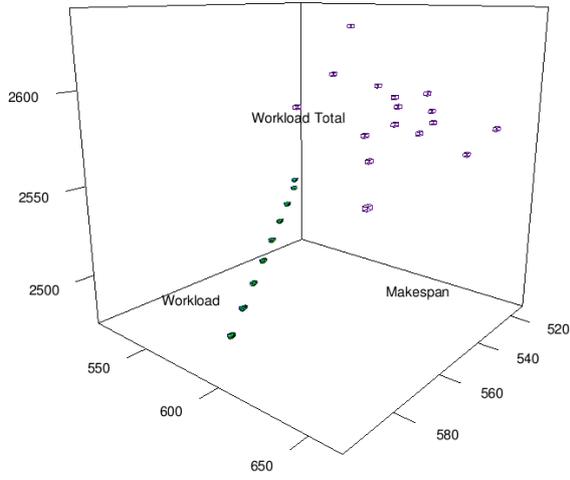
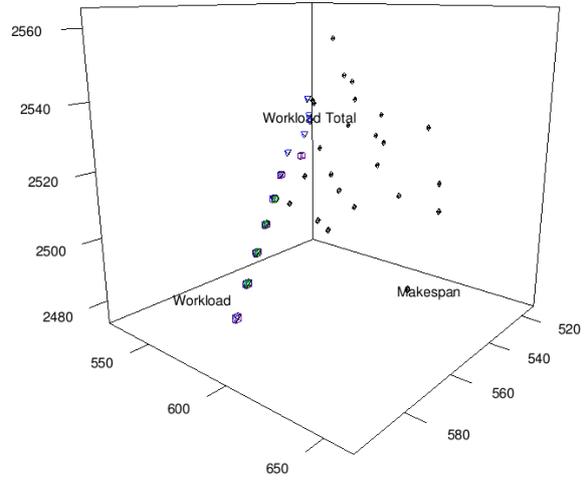


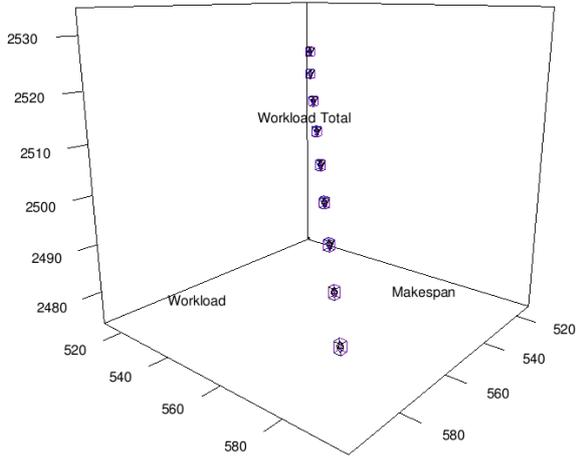
Figura 19 – Gráficos do problema BR MK03



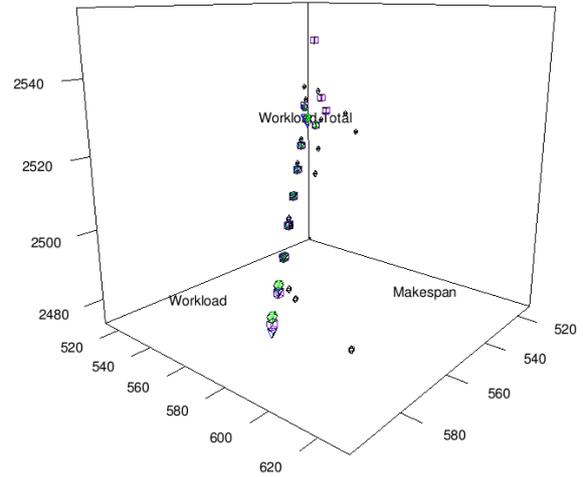
(a) DIPSO



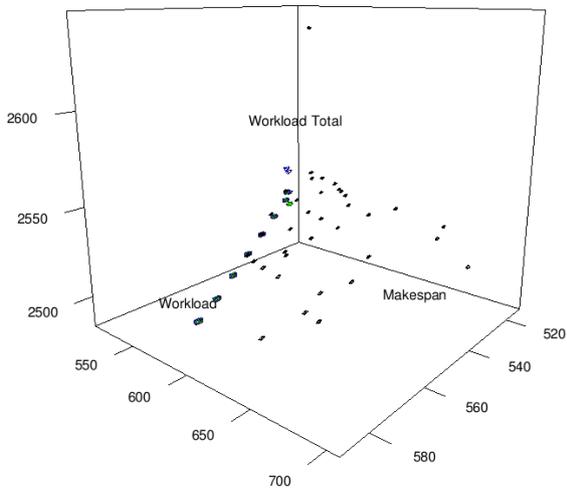
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

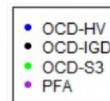


Figura 20 – Gráficos do problema BR MK08

o valor 0,09, pois nos problemas ZDT, que são de domínio contínuo, a comparação dos valores máximos de  $CR$  e mínimos de  $DR$  entre cada execução apresentou uma diferença maior do que no experimento da seção anterior. Por esse motivo, o parâmetro ‘erro’ foi incrementado para que PFA pudesse indicar um ponto de parada para um número maior de execuções. A variância (parâmetro  $varLimit$ ) utilizada pelo OCD-HV e OCD-IGD foi de 0,0005 e para o OCD- $S_3$  foi de 0,0002. Assim como no experimento anterior, para todas as implementações de OCD  $nPreGen = 5$ . Os pontos de referência utilizados para o hipervolume foram obtidos com o mesmo experimento descrito na Seção 4.2 e os do IGD são as soluções conhecidas do *benchmark*, que foram obtidas em (Nebro; Durillo, 2015).

As Tabelas 30 e 31 apresentam os resultados do experimento para os algoritmos NSGA2 e SPEA2, respectivamente, sendo que são análogas as tabelas de resultado do experimento anterior (Tabelas 24, 25, 26, 27 e 28). Em ambas as tabelas, a predominância de melhores resultados para o PFA é observada, exceto para o problema ZDT4 executado pelo NSGA2. Nos outros casos, o ponto de parada apontado pelo PFA possui melhor qualidade e maior diversidade, ou seja, várias soluções dominam as apresentadas pelos outros pontos de parada e a quantidade de soluções é maior.

Tabela 30 – NSGA2 - ZDT - Comparação do uso de indicadores para parada do algoritmo

| Problema | Total de SND | OCD-HV         | OCD-IGD | OCD- $S_3$ | PFA            |
|----------|--------------|----------------|---------|------------|----------------|
| ZDT1     | 366          | 0/250          | 0/114   | 0/111      | <b>366/452</b> |
| ZDT2     | 371          | 3/261          | 0/116   | 0/89       | <b>368/371</b> |
| ZDT3     | 569          | 0/231          | 0/101   | 0/141      | <b>569/569</b> |
| ZDT4     | 101          | <b>100/104</b> | 0/172   | 1/67       | 0/190          |
| ZDT6     | 346          | 1/231          | 0/84    | 0/22       | <b>345/348</b> |

Tabela 31 – SPEA2 - ZDT - Comparação do uso de indicadores para parada do algoritmo

| Problema | Total de SND | OCD-HV  | OCD-IGD | OCD- $S_3$ | PFA            |
|----------|--------------|---------|---------|------------|----------------|
| ZDT1     | 354          | 0/167   | 0/99    | 0/96       | <b>354/435</b> |
| ZDT2     | 463          | 122/176 | 0/104   | 0/59       | <b>341/472</b> |
| ZDT3     | 564          | 0/152   | 0/108   | 0/147      | <b>564/564</b> |
| ZDT4     | 193          | 78/105  | 15/104  | 0/79       | <b>100/102</b> |
| ZDT6     | 318          | 1/154   | 0/77    | 0/9        | <b>317/317</b> |

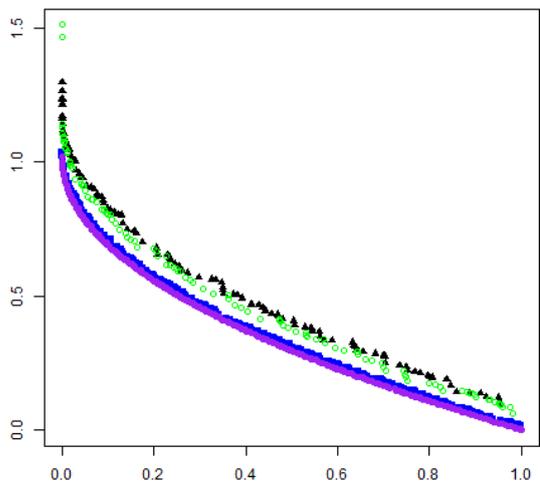
As Figuras 21 e 22 apresentam os gráficos dos resultados obtidos no experimento, para o NSGA2 e o SPEA2, respectivamente. Nos gráficos, os resultados das Tabelas 30 e 31 são visualmente percebidos. Em todos, a curva de soluções de PFA se mostra com pontos que dominam as demais, exceto para o problema ZDT4 executado pelo NSGA2, conforme apresentado nos resultados da respectiva tabela. De qualquer forma, na maioria

dos problemas, todas as curvas estão próximas, sendo o PFA com melhores resultados. Os resultados apresentados pelo OCD- $S_3$  para os problemas ZDT3 e ZDT4 se mostram próximos do PFA, no ZDT1 se mostra melhor do que o OCD-IGD e no ZDT2 é semelhante ao do OCD-IGD. Generalizando, os melhores resultados são do PFA, seguido pelo OCD-HV, OCD- $S_3$  e OCD-IGD.

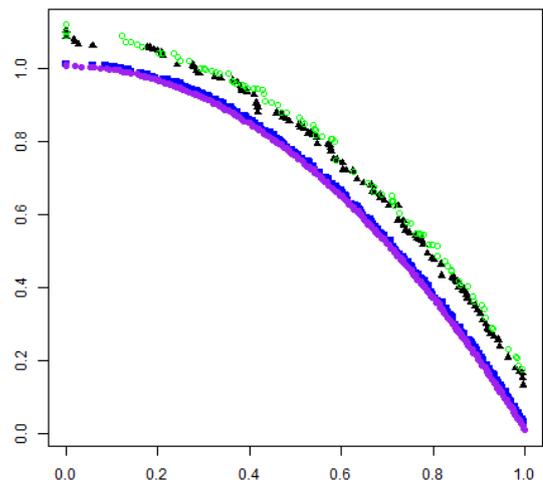
## 4.5 Considerações Finais

Neste capítulo, apresentamos os experimentos realizados em que os resultados indicam:

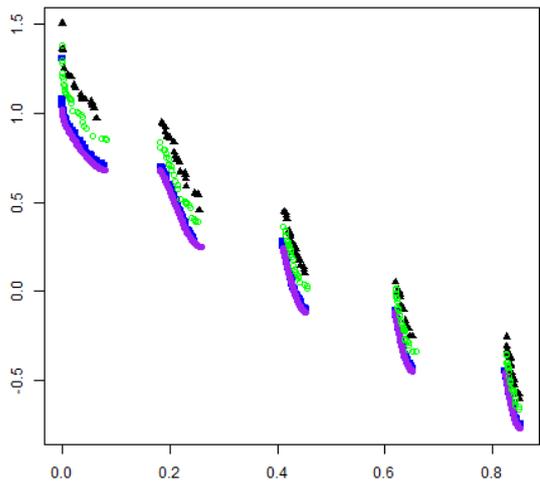
- 1) a efetividade dos algoritmos SEDA e SEDASI aplicados ao problema *Job Shop* Flexível;
- 2) a necessidade de indicadores que apresentem dados comportamentais de um AEMO;
- 3) a efetividade dos indicadores CR e DR, integrando o PFA, e do  $S_3$ , associado ao OCD, para indicar um ponto de parada para AEMO's que tratam o FJSP e para problemas experimentais da literatura (ZDT). No próximo capítulo, as conclusões deste trabalho são apresentadas.



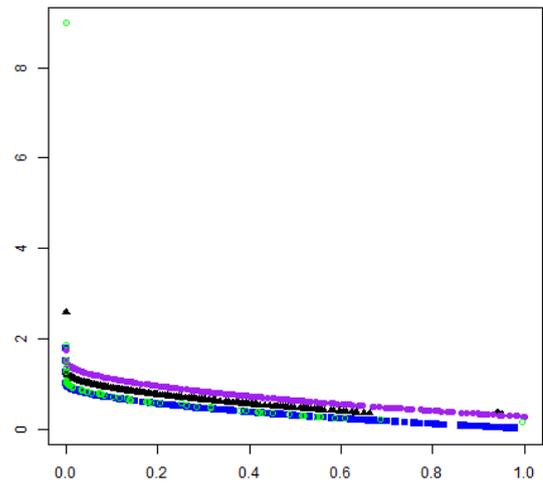
(a) ZDT1



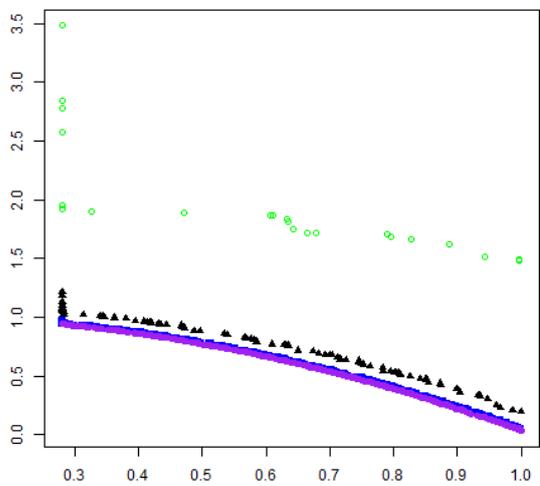
(b) ZDT2



(c) ZDT3



(d) ZDT4



(e) ZDT6

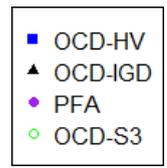
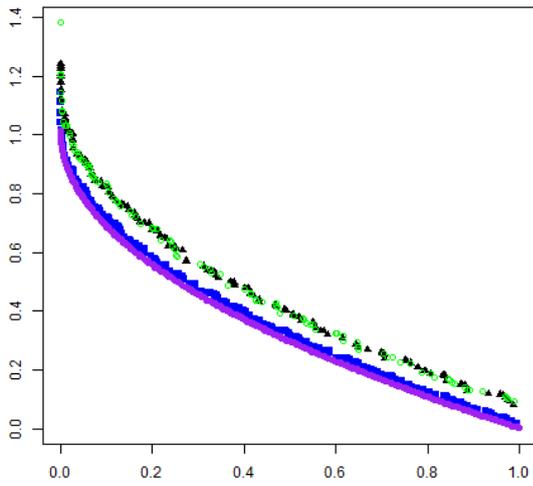
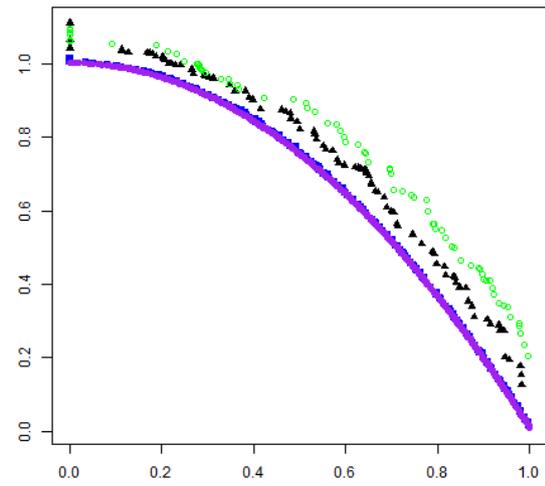


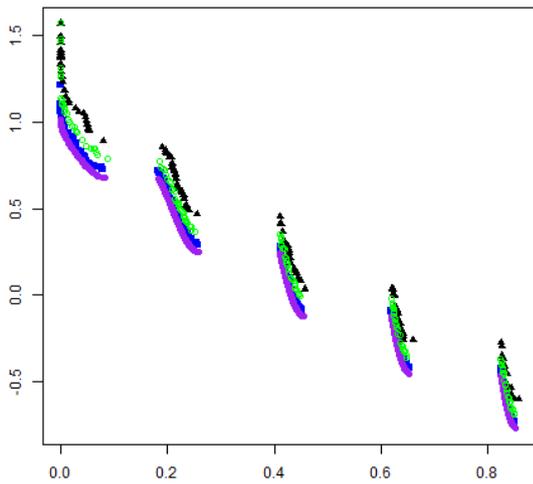
Figura 21 – Gráficos dos problemas ZDT's do algoritmo NSGA2



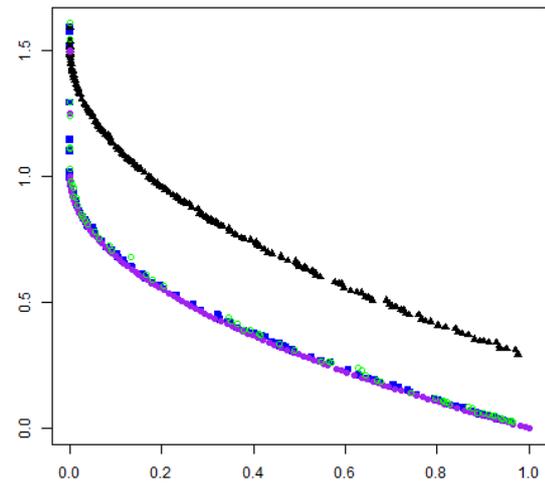
(a) ZDT1



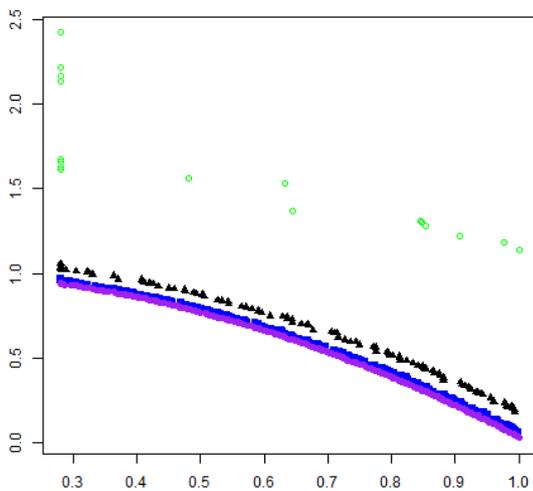
(b) ZDT2



(c) ZDT3



(d) ZDT4



(e) ZDT6

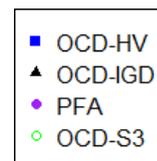


Figura 22 – Gráficos dos problemas ZDT's do algoritmo SPEA2

---

## Conclusão

Neste trabalho, foram propostos dois indicadores, CR e DR, para apresentar informações comportamentais dos AEMO's, além de um indicador,  $S_3$ , para determinar o ponto de convergência, sendo que foram associados a algoritmos para identificar um ponto de parada para os AEMO's. Para analisar o comportamento de tais indicadores, dois AEMO's foram propostos, SEDA e SEDASI, baseados no EDA e, no caso do SEDASI, na inteligência de enxame.

Os indicadores CR e DR têm a finalidade de avaliar as condições que um AEMO possui para melhorar o conjunto de aproximação em um determinado instante da evolução. O indicador CR apresenta o quão concentradas estão as soluções candidatas na fronteira 1 (soluções referenciadas no conjunto de aproximação), enquanto DR mostra a efetividade do AEMO em encontrar novas posições do espaço de busca. Ambos foram utilizados para apresentar evidências de que o OCD-HV não apresenta o ponto de convergência de melhor resultado quando aplicado ao DIPSO.

A utilização dos indicadores para determinar um ponto de parada de AEMO's foi proposta por meio dos algoritmos PFA e OCD- $S_3$ . O algoritmo PFA foi implementado para utilizar CR e DR e estimar um ponto de parada em que o AEMO está com a menor probabilidade de melhorar o resultado. A associação de  $S_3$  com o OCD permitiu definir um ponto de parada para AEMO's através da identificação do ponto de convergência a partir de observações no espaço de soluções. Portanto, a utilização de PFA e OCD- $S_3$  para identificação de um ponto de parada, além do uso de CR e DR para observar o comportamento de um AEMO, respondem a questão de pesquisa QP1, pois a utilização desses permite fazer observações sobre convergência, PFA e OCD- $S_3$ , e diversidade, DR, através do espaço de soluções.

Os AEMO's SEDA e SEDASI foram propostos para tratar o FJSP multiobjetivo sendo que SEDA é um EDA e SEDASI é um EDA híbrido com características de inteligência de enxame. Os experimentos realizados mostraram que ambos os algoritmos apresentaram resultados equivalentes àqueles encontrados na literatura. Porém, a comparação dos resultados com algoritmos semelhantes mostra que o conjunto de soluções não dominadas

de SEDA e SEDASI possuem uma diversidade maior do que o observando nos outros.

Foram realizados experimentos para verificar o comportamento das propostas de identificação do ponto de parada com o FJSP e com um *benchmark* da literatura, o ZDT (Zitzler; Deb; Thiele, 2000), utilizando para comparação o OCD-HV e o OCD-IGD.

O experimento realizado com o FJSP foi elaborado com os AEMO's: SEDA, SEDASI, DIPSO, NSGA2 e SPEA2, utilizando os *benchmarks* de Kacem, Hammadi e Borne (2002) (KC) e Brandimarte (1993) (BR). Tal experimento apresentou resultados promissores para ambas as propostas, PFA e OCD- $S_3$ , sendo que para os AEMO's DIPSO e SEDASI, o OCD- $S_3$  apresentou os melhores resultados e para o SPEA2, PFA apresentou os melhores resultados. Apesar disto, para todos os algoritmos, PFA e OCD- $S_3$  apresentaram resultados competitivos, exceto os resultados do PFA com o DIPSO nos problemas do *benchmark* BR. Nesse caso, PFA encontrou soluções não dominadas somente em um problema, BR Mk2, no comparativo com os outros algoritmos de ponto de parada.

O experimento realizado com os problemas do *benchmark* ZDT confirmaram a competitividade de PFA e OCD- $S_3$ , sendo que em todos os problemas e algoritmos, PFA apresentou o melhor resultado, exceto para o problema ZDT4 com o NSGA2.

Esses experimentos respondem às questões de pesquisa QP2 e QP3, pois, em ambos os experimentos, os algoritmos PFA e OCD- $S_3$  apresentam pontos de parada para AEMO's, com um conjunto de soluções não dominadas diversificado (QP2) e com ênfase para a competitividade que apresentam, sendo que, no experimento com o FJSP, o destaque maior é do OCD- $S_3$  e, no experimento com o ZDT, o destaque maior é do PFA (QP3).

Portanto, os resultados desses experimentos, o primeiro com algoritmos tratando um problema de domínio discreto, FJSP, e o segundo tratando problemas de domínio contínuo e com AEMO's conhecidos da literatura, indicam a efetividade em utilizar PFA e OCD- $S_3$ , para identificar um ponto de parada para AEMO's.

## 5.1 Principais Contribuições

O trabalho desenvolvido apresentou as seguintes contribuições: 1) CR e DR, indicadores que podem ser utilizados para analisar o comportamento evolutivo de um AEMO, estimando se o algoritmo pode melhorar seu conjunto de aproximação; 2)  $S_3$ , indicador que, mede a distância entre as soluções no espaço de soluções, apresentando uma média do quão espalhadas elas estão, e, ao ser associado ao algoritmo OCD (OCD- $S_3$ ), permite determinar um ponto de convergência para AEMO's, observando a estabilidade do conjunto de aproximação no respectivo espaço; 3) PFA, um algoritmo que utiliza CR e DR para estimar um ponto de parada para AEMO's; e 4) SEDA e SEDASI, dois AEMO's para tratar o FJSP multiobjetivo.

## 5.2 Trabalhos Futuros

A fim de estender este trabalho, temos a possibilidade de propor novos indicadores que observem outras características comportamentais dos AEMO's, combinar os indicadores apresentados, além de aplicar a definição de indicadores de progresso aos que foram propostos, de modo que se tenha um novo ponto de vista. Os AEMO's, SEDA e SEDASI, podem ser modificados, utilizando observações que possam ser realizadas a partir de CR e DR para que explorem de uma outra forma o espaço de busca, como por exemplo, aplicando mais diversidade na população. Além disso, ambos os algoritmos podem ser aplicados a outros problemas de domínio discreto, por exemplo, problemas de escalonamento diferentes do FJSP.

## 5.3 Contribuições em Produção Bibliográfica

O trabalho desenvolvido resultou em três publicações em conferências da área:

- ❑ L. C. F. Carvalho and M. A. Fernandes, “Simple Estimation of Distribution Algorithm for the Flexible Job-Shop Problem” 2017 IEEE Congress on Evolutionary Computation (CEC), 2017, pp. 2233-2239, doi: 10.1109/CEC.2017.7969575.
- ❑ L. C. F. Carvalho and M. A. Fernandes, “Convergence analysis of evolutionary algorithms solving the Flexible Job Shop Problem,” 2018 IEEE Congress on Evolutionary Computation (CEC), 2018, pp. 1-7, doi: 10.1109/CEC.2018.8477685.
- ❑ L. C. F. Carvalho and M. A. Fernandes, “An EDA with swarm intelligence for the multi-objective flexible job-shop problem,” In Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion (GECCO '20), 159–160, doi: 10.1145/3377929.3389918.



---

## Referências

- Abouhawwash, M.; Seada, H.; Deb, K. Towards faster convergence of evolutionary multi-criterion optimization algorithms using karush kuhn tucker optimality based local search. **Comput. Oper. Res.**, Elsevier Science Ltd., Oxford, UK, UK, v. 79, n. C, p. 331–346, 2017. ISSN 0305-0548.
- Audet, C. et al. Performance indicators in multiobjective optimization. **European Journal of Operational Research**, v. 292, n. 2, p. 397–422, 2021. ISSN 0377-2217. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0377221720309620>>.
- Bansal, J. C.; Gopal, A.; Nagar, A. K. Stability analysis of artificial bee colony optimization algorithm. **Swarm and Evolutionary Computation**, v. 41, p. 9 – 19, 2018. ISSN 2210-6502.
- Boyd, S.; Vandenberghe, L. **Convex Optimization**. [S.l.]: Cambridge University Press, 2004. ISBN 9780521833783.
- Brandimarte, P. Routing and scheduling in a flexible job shop by tabu search. **Annals of Operations Research**, Baltzer Science Publishers, Baarn/Kluwer Academic Publishers, v. 41, n. 3, p. 157–183, 1993. ISSN 0254-5330. Disponível em: <<http://dx.doi.org/10.1007/BF02023073>>.
- Carvalho, L.; Fernandes, M. Multi-objective flexible job-shop scheduling problem with DIPSO: More diversity, greater efficiency. In: **Proceedings of the IEEE Congress on Evolutionary Computation**. Beijing, China: [s.n.], 2014. p. 282 – 289. ISBN 0-7803-8515-2.
- Carvalho, L. C. F. **Algoritmo híbrido multiobjetivo para o problema Flexible Job Shop**. Tese (Master's program) — Graduate Program in Computer Science - Federal University of Uberlândia, 2015.
- Chen, Y.; Zou, X.; Xie, W. Convergence of multi-objective evolutionary algorithms to a uniformly distributed representation of the pareto front. **Information Sciences**, v. 181, n. 16, p. 3336–3355, 2011. ISSN 0020-0255. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0020025511001721>>.
- Cheng, R.; Gen, M.; Tsujimura, Y. A tutorial survey of job-shop scheduling problems using genetic algorithms - i. representation. **Computers & Industrial Engineering**, v. 30, n. 4, p. 983 – 997, 1996.

- Chiang, T.-C.; Lin, H.-J. A simple and effective evolutionary algorithm for multiobjective flexible job shop scheduling. **International Journal of Production Economics**, v. 141, n. 1, p. 87 – 98, 2013. ISSN 0925-5273. Meta-heuristics for manufacturing scheduling and logistics problems. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0925527312001429>>.
- Coello, C. A. C.; Sierra, M. R. A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm. In: MONROY, R. et al. (Ed.). **MICAI 2004: Advances in Artificial Intelligence**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. p. 688–697. ISBN 978-3-540-24694-7.
- Cuiyu, W.; Yang, L.; Xinyu, L. Solving flexible job shop scheduling problem by a multi-swarm collaborative genetic algorithm. **Journal of Systems Engineering and Electronics**, v. 32, n. 2, p. 261–271, 2021.
- Darwin, C. **On the Origin of Species by Means of Natural Selection: Or, The Preservation of Favoured Races in the Struggle for Life**. [S.l.]: J. Murray, 1860.
- Deb, K. **Multi-Objective Optimization Using Evolutionary Algorithms**. USA: John Wiley & Sons, Inc., 2001. ISBN 047187339X.
- Deb, K.; Jain, H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. **IEEE Transactions on Evolutionary Computation**, v. 18, n. 4, p. 577–601, 2014.
- Deb, K.; Jain, S. **Running performance Metrics for evolutionary multi-objective optimization**. [S.l.], 2002.
- Deb, K. et al. A fast and elitist multiobjective genetic algorithm: Nsga-ii. **IEEE Transactions on Evolutionary Computation**, v. 6, n. 2, p. 182 – 197, 2002.
- Derrac, J. et al. Analyzing convergence performance of evolutionary algorithms: A statistical approach. **Information Sciences**, v. 289, p. 41 – 58, 2014. ISSN 0020-0255.
- Engelbrecht, A. P. **Computational Intelligence: An Introduction**. 2nd. ed. [S.l.]: Wiley Publishing, 2007. ISBN 0470035617.
- Falcón-Cardona, J. G.; Coello, C. A. C. Convergence and diversity analysis of indicator-based multi-objective evolutionary algorithms. In: **Proceedings of the Genetic and Evolutionary Computation Conference**. New York, NY, USA: Association for Computing Machinery, 2019. (GECCO '19), p. 524–531. ISBN 9781450361118. Disponível em: <<https://doi.org/10.1145/3321707.3321718>>.
- Fonseca, C. M.; Fleming, P. J. Genetic algorithms for multiobjective optimization: Formulation discussion and generalization. In: **Proceedings of the International Conference on Genetic Algorithms**. [S.l.: s.n.], 1993. p. 416–423. ISBN 1-55860-299-2.
- Gao, F.; Zhou, A.; Zhang, G. An estimation of distribution algorithm based on decomposition for the multiobjective tsp. In: **Natural Computation (ICNC), 2012 Eighth International Conference on**. [S.l.: s.n.], 2012. p. 817–821. ISSN 2157-9555.

- Gen, M.; Tsujimura, Y.; Kubota, E. Solving job-shop scheduling problems by genetic algorithm. In: **IEEE International Conference on Systems, Man, and Cybernetics**. [S.l.: s.n.], 1994. v. 2, p. 1577 – 1582.
- González-Álvarez, D. L.; Vega-Rodríguez, M. A.; Rubio-Largo, Á. Convergence analysis of some multiobjective evolutionary algorithms when discovering motifs. **Soft Computing**, v. 18, n. 5, p. 853–869, 2014. ISSN 1433-7479.
- Guimarães, K. F. **Escalonamento Genético FJSP com tempo de configuração dependente de sequência**. Dissertação (Mestrado) — Faculdade de Computação - Universidade Federal de Uberlândia, 2007.
- Hao, X. et al. An effective markov network based eda for flexible job shop scheduling problems under uncertainty. In: **2014 IEEE International Conference on Automation Science and Engineering (CASE)**. [S.l.: s.n.], 2014. p. 131–136. ISSN 2161-8070.
- Hauschild, M.; Pelikan, M. An introduction and survey of estimation of distribution algorithms. **Swarm and Evolutionary Computation**, v. 1, n. 3, p. 111 – 128, 2011. ISSN 2210-6502.
- He, J.; Lin, G. Average convergence rate of evolutionary algorithms. **IEEE Transactions on Evolutionary Computation**, v. 20, n. 2, p. 316–321, 2016. ISSN 1089-778X.
- Huband, S. et al. A review of multiobjective test problems and a scalable test problem toolkit. **IEEE Transactions on Evolutionary Computation**, v. 10, n. 5, p. 477–506, 2006.
- Ishibuchi, H. et al. Modified distance calculation in generational distance and inverted generational distance. In: . [S.l.: s.n.], 2015. p. 110–125. ISBN 978-3-319-15891-4.
- Izmailov, A.; Solodov, M. **Otimização - volume 2. Métodos computacionais**. [S.l.]: Associação Instituto Nacional de Matemática Pura e Aplicada - IMPA, 2010. ISBN 9788524402685.
- Jain, H.; Deb, K. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: Handling constraints and extending to an adaptive approach. **IEEE Transactions on Evolutionary Computation**, v. 18, n. 4, p. 602–622, 2014.
- Jia, Z.; Chen, H.; Tang, J. A new multi-objective fully-informed particle swarm algorithm for flexible job-shop scheduling problems. In: **Computational Intelligence and Security Workshops, 2007. CISW 2007. International Conference on**. [S.l.: s.n.], 2007. p. 191 – 194.
- Kacem, I.; Hammadi, S.; Borne, P. Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic. **Mathematics and Computers in Simulation**, v. 60, n. 3 - 5, p. 245 – 276, 2002.
- Kennedy, J.; Eberhart, R. Particle swarm optimization. In: **Neural Networks, 1995. Proceedings., IEEE International Conference on**. [S.l.: s.n.], 1995. v. 4, p. 1942–1948 vol.4.

- Lamarck, J. de Monet de. **Philosophie zoologique, ou Exposition des considérations relatives à l'histoire naturelle des animaux...** [S.l.]: Dentu, 1809. (Philosophie zoologique, ou Exposition des considérations relatives à l'histoire naturelle des animaux).
- Larrañaga, P. An introduction to probabilistic graphical models. In: \_\_\_\_\_. **Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation**. Boston, MA: Springer US, 2002. p. 27–56. ISBN 978-1-4615-1539-5. Disponível em: <[https://doi.org/10.1007/978-1-4615-1539-5\\_2](https://doi.org/10.1007/978-1-4615-1539-5_2)>.
- Li, J. qing; Pan, Q. ke; Liang, Y.-C. An effective hybrid tabu search algorithm for multi-objective flexible job-shop scheduling problems. **Computers & Industrial Engineering**, v. 59, n. 4, p. 647 – 662, 2010. ISSN 0360-8352. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0360835210002056>>.
- Li, M.; Yao, X. Quality evaluation of solution sets in multiobjective optimisation: A survey. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 52, n. 2, mar. 2019. ISSN 0360-0300. Disponível em: <<https://doi.org/10.1145/3300148>>.
- Li, Y.; Zhou, A.; Zhang, G. A decomposition based estimation of distribution algorithm for multiobjective knapsack problems. In: **Natural Computation (ICNC), 2012 Eighth International Conference on**. [S.l.: s.n.], 2012. p. 803–807. ISSN 2157-9555.
- Ling-li, Z.; Feng-Xing, Z.; Xiao-hong, X. Mathematical model and hybrid particle swarm optimization for flexible job-shop scheduling problem. **Genetic and Evolutionary Computation Conference**, p. 731 – 736, 2009.
- Liu, H. et al. Investigating the effect of imbalance between convergence and diversity in evolutionary multiobjective algorithms. **IEEE Transactions on Evolutionary Computation**, v. 21, n. 3, p. 408–425, 2017. ISSN 1089-778X.
- Liu, Y.; Zhang, L.; Sun, T. An improved nondominated sorting genetic algorithm-ii for multi-objective flexible job-shop scheduling problem considering worker assignments. In: **2021 International Conference on Communications, Information System and Computer Engineering (CISCE)**. [S.l.: s.n.], 2021. p. 837–843.
- Martí, L. et al. A stopping criterion for multi-objective optimization evolutionary algorithms. **Inf. Sci.**, Elsevier Science Inc., USA, v. 367, n. C, p. 700–718, nov. 2016. ISSN 0020-0255. Disponível em: <<https://doi.org/10.1016/j.ins.2016.07.025>>.
- Martí, L. et al. A progress indicator for detecting success and failure in evolutionary multi-objective optimization. In: **IEEE Congress on Evolutionary Computation**. [S.l.: s.n.], 2010. p. 1–8.
- Mühlenbein, H.; Paaß, G. From recombination of genes to the estimation of distributions i. binary parameters. In: VOIGT, H.-M. et al. (Ed.). **Parallel Problem Solving from Nature — PPSN IV**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996. p. 178–187. ISBN 978-3-540-70668-7. Disponível em: <[https://doi.org/10.1007/3-540-61723-X\\_982](https://doi.org/10.1007/3-540-61723-X_982)>.

Naujoks, B.; Hochstrate, N.; Emmerich, M. Multi-objective optimisation using s-metric selection: application to three-dimensional solution spaces. In: . [S.l.: s.n.], 2005. v. 2, p. 1282 – 1289 Vol. 2. ISBN 0-7803-9363-5.

Nebro, A. J.; Durillo, J. J. **Problems included in jMetal**. 2015. Disponível em: <<http://jmetal.sourceforge.net/problems.html>>.

Niu, Q.; Jiao, B.; Gu, X. Particle swarm optimization combined with genetic operators for job shop scheduling problem with fuzzy processing time. **Applied Mathematics and Computation**, v. 205, n. 1, p. 148 – 158, 2008.

Shao, X. et al. Hybrid discrete particle swarm optimization for multi-objective flexible job-shop scheduling problem. **The International Journal of Advanced Manufacturing Technology**, v. 67, n. 9, p. 2885–2901, Aug 2013. ISSN 1433-3015. Disponível em: <<https://doi.org/10.1007/s00170-012-4701-3>>.

Sheskin, D. J. **Handbook of Parametric and Nonparametric Statistical Procedures**. 4. ed. [S.l.]: Chapman & Hall/CRC, 2007. ISBN 1584888148, 9781584888147.

Srinivas, N.; Deb, K. Multiobjective optimization using nondominated sorting in genetic algorithms. **Evolutionary Computation**, MIT Press, Cambridge, MA, USA, v. 2, n. 3, p. 221–248, 1994. ISSN 1063-6560.

Trautmann, H. et al. A convergence criterion for multiobjective evolutionary algorithms based on systematic statistical testing. In: **Parallel Problem Solving from Nature**. [S.l.: s.n.], 2008. p. 825–836.

Trautmann, H. et al. Statistical methods for convergence detection of multi-objective evolutionary algorithms. **Evol. Comput.**, MIT Press, Cambridge, MA, USA, v. 17, n. 4, p. 493–509, dez. 2009. ISSN 1063-6560. Disponível em: <<https://doi.org/10.1162/evco.2009.17.4.17403>>.

Trelea, I. C. The particle swarm optimization algorithm: convergence analysis and parameter selection. **Information Processing Letters**, v. 85, n. 6, p. 317 – 325, 2003. ISSN 0020-0190. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0020019002004477>>.

Veldhuizen, D. A. V. **Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations**. Tese (Doutorado), USA, 1999.

Wagner, T.; Trautmann, H. Online convergence detection for evolutionary multi-objective algorithms revisited. In: **IEEE Congress on Evolutionary Computation**. [S.l.: s.n.], 2010. p. 1–8. ISSN 1089-778X.

Wagner, T.; Trautmann, H.; Martí, L. A taxonomy of online stopping criteria for multi-objective evolutionary algorithms. In: Takahashi, R. H. C. et al. (Ed.). **Evolutionary Multi-Criterion Optimization**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 16–30. ISBN 978-3-642-19893-9.

Wagner, T.; Trautmann, H.; Naujoks, B. Ocd: Online convergence detection for evolutionary multi-objective algorithms based on statistical testing. In: **International Conference on Evolutionary Multi-Criterion Optimization**. [S.l.: s.n.], 2009. p. 198–215. ISBN 978-3-642-01020-0.

- Wang, L.; Wang, S.; Liu, M. A pareto-based estimation of distribution algorithm for the multi-objective flexible job-shop scheduling problem. **International Journal of Production Research**, v. 51, n. 12, p. 3574–3592, 2013. Disponível em: <<http://dx.doi.org/10.1080/00207543.2012.752588>>.
- Wang, S. et al. An estimation of distribution algorithm for the multi-objective flexible job-shop scheduling problem. In: **Computational Intelligence in Scheduling (SCIS), 2013 IEEE Symposium on**. [S.l.: s.n.], 2013. p. 1–8.
- Wang, S. et al. An estimation of distribution algorithm for the flexible job-shop scheduling problem. In: HUANG, D.-S. et al. (Ed.). **Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. p. 9–16. ISBN 978-3-642-25944-9.
- Wang, X. et al. A multi-objective genetic algorithm based on immune and entropy principle for flexible job-shop scheduling problem. **The International Journal of Advanced Manufacturing Technology**, v. 51, n. 5 - 8, p. 757 – 767, 2010.
- Xiao-hong, X.; Ling-li, Z.; Yue-wen, F. Hybrid particle swarm optimization for flexible job-shop scheduling problem and its implementation. In: **Information and Automation (ICIA), 2010 IEEE International Conference on**. [S.l.: s.n.], 2010. p. 1155 – 1159.
- Xing, L.-N.; Chen, Y.-W.; Yang, K.-W. An efficient search method for multi-objective flexible job shop scheduling problems. **Journal of Intelligent Manufacturing**, v. 20, n. 3, p. 283–293, Jun 2009. ISSN 1572-8145. Disponível em: <<https://doi.org/10.1007/s10845-008-0216-z>>.
- Zhang, G. et al. An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem. **Computers & Industrial Engineering**, v. 56, n. 4, p. 1309 – 1318, 2009.
- Zhang, Q.; Hu, S. An improved hybrid quantum particle swarm optimization algorithm for fjsp. In: **Proceedings of the 2019 11th International Conference on Machine Learning and Computing**. New York, NY, USA: Association for Computing Machinery, 2019. (ICMLC '19), p. 246–252. ISBN 9781450366007. Disponível em: <<https://doi.org/10.1145/3318299.3318359>>.
- Zhang, Q.; Li, H. Moea/d: A multiobjective evolutionary algorithm based on decomposition. **Transactions on Evolutionary Computation**, v. 11, n. 6, p. 712–731, 2007. ISSN 1089-778X.
- Zitzler, E.; Deb, K.; Thiele, L. Comparison of multiobjective evolutionary algorithms: Empirical results. **Evol. Comput.**, MIT Press, Cambridge, MA, USA, v. 8, n. 2, p. 173–195, jun. 2000. ISSN 1063-6560. Disponível em: <<https://doi.org/10.1162/106365600568202>>.
- Zitzler, E.; Laumanns, M.; Thiele, L. **SPEA2: Improving the Strength Pareto Evolutionary Algorithm**. [S.l.], 2001.
- Zitzler, E.; Thiele, L. Multiobjective optimization using evolutionary algorithms - a comparative case study. In: **Proceedings of the International Conference**

---

on **Parallel Problem Solving from Nature**. [S.l.: s.n.], 1998. p. 292–304. ISBN 3-540-65078-4.

Zitzler, E.; Thiele, L. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. **IEEE Transactions on Evolutionary Computation**, v. 3, n. 4, p. 257–271, 1999.

Zitzler, E. et al. Performance assessment of multiobjective optimizers: an analysis and review. **IEEE Transactions on Evolutionary Computation**, v. 7, n. 2, p. 117–132, 2003. ISSN 1089-778X.



# Apêndices

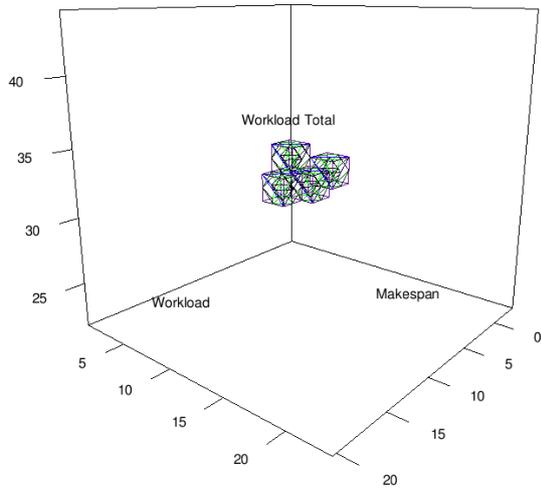


# APÊNDICE **A**

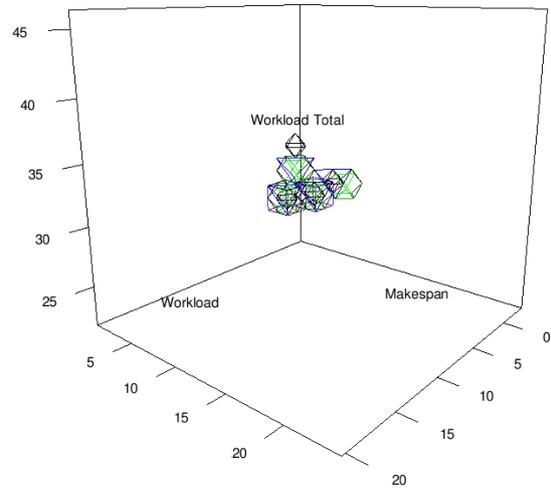
---

## **Gráficos**

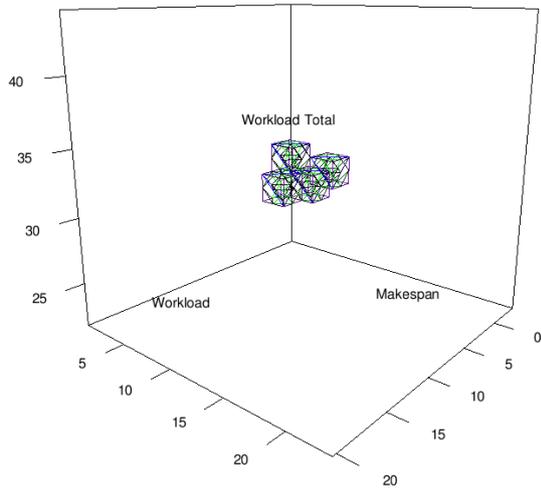
Conforme dinâmica do texto, apenas alguns gráficos foram apresentados na seção 4.3. Enfim, todos os gráficos de cada algoritmo, para cada problema, são aqui apresentados.



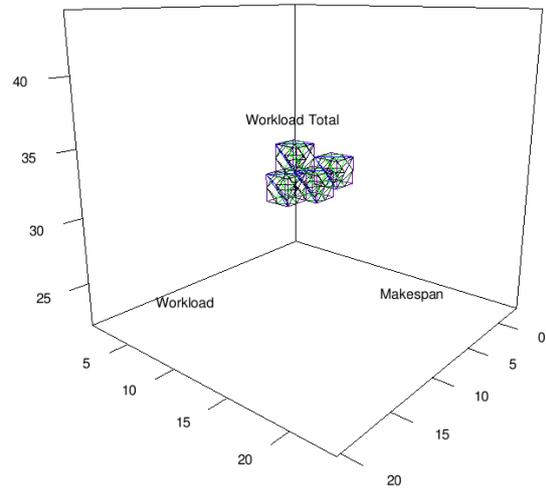
(a) DIPSO



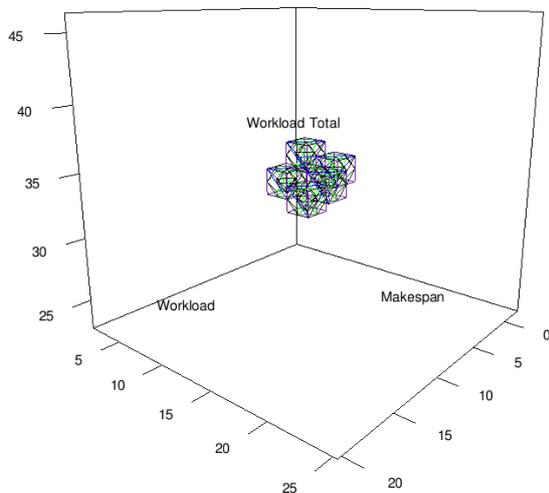
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

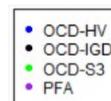
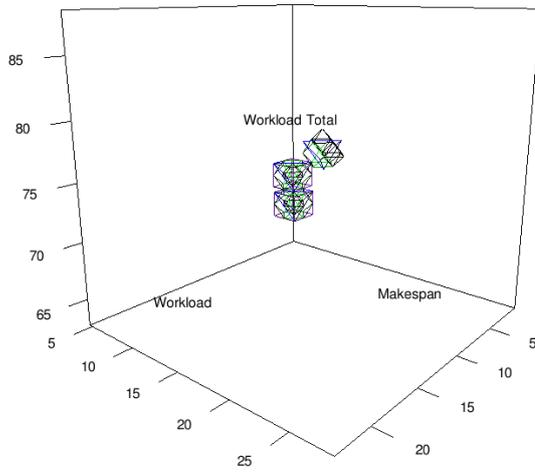
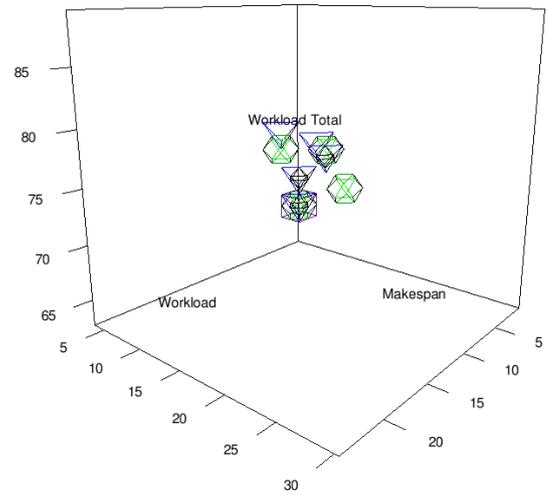


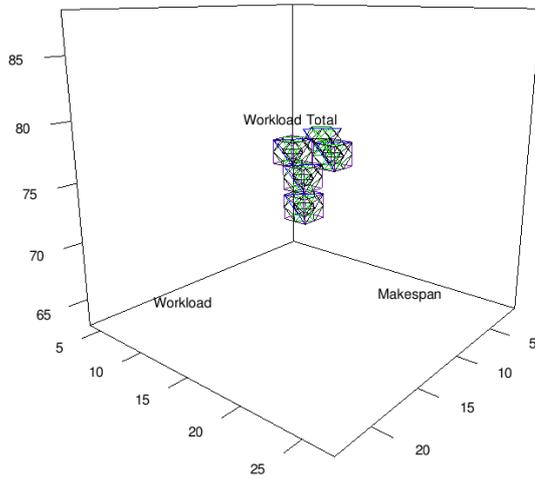
Figura 23 – Gráficos do problema KC04X05



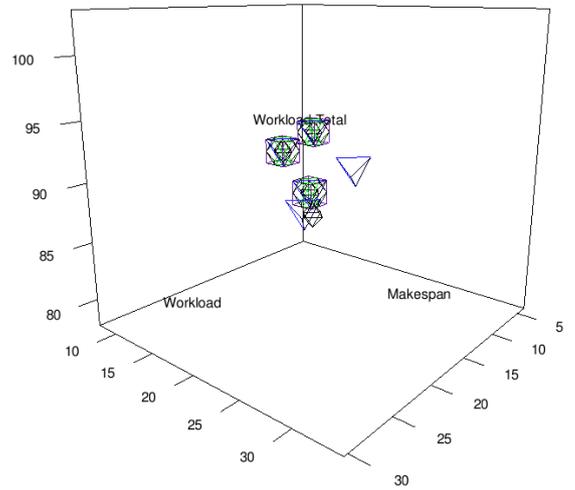
(a) DIPSO



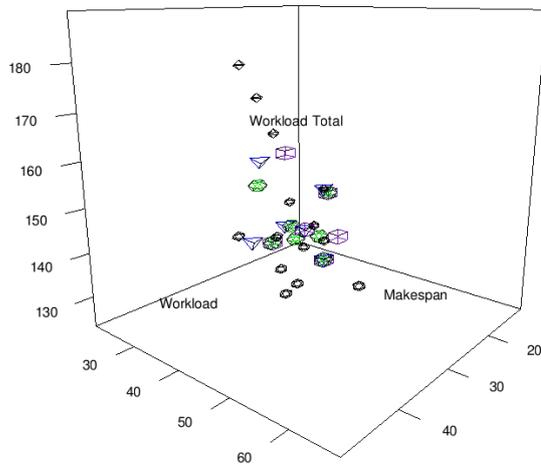
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

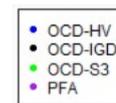
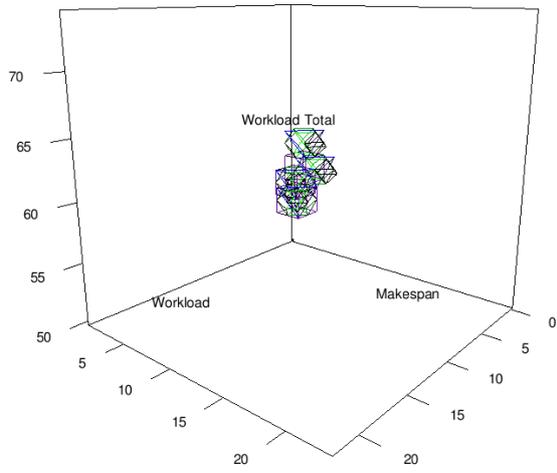
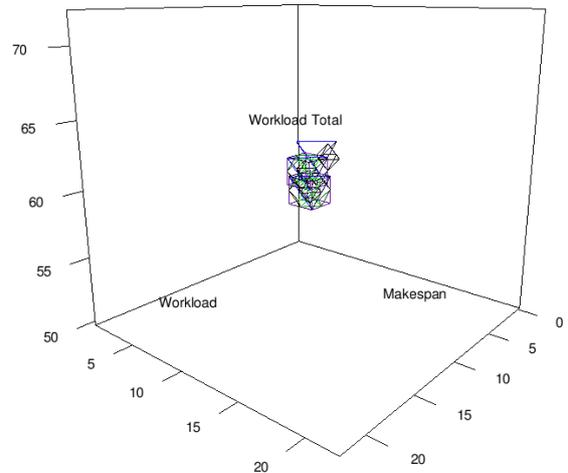


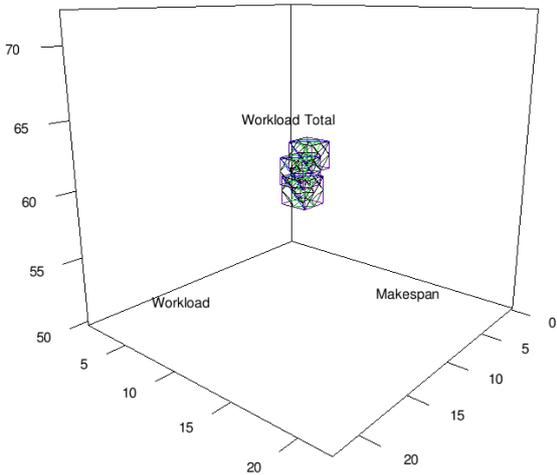
Figura 24 – Gráficos do problema KC08X08



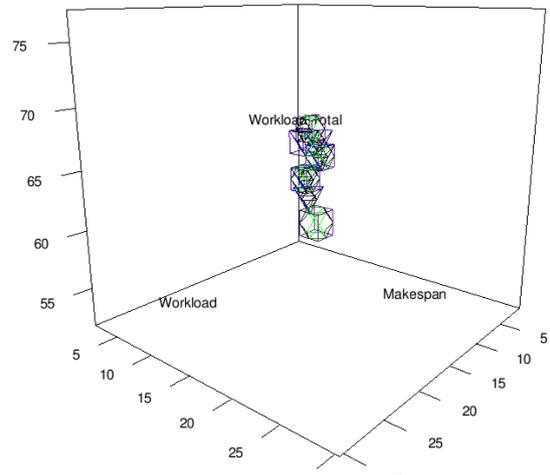
(a) DIPSO



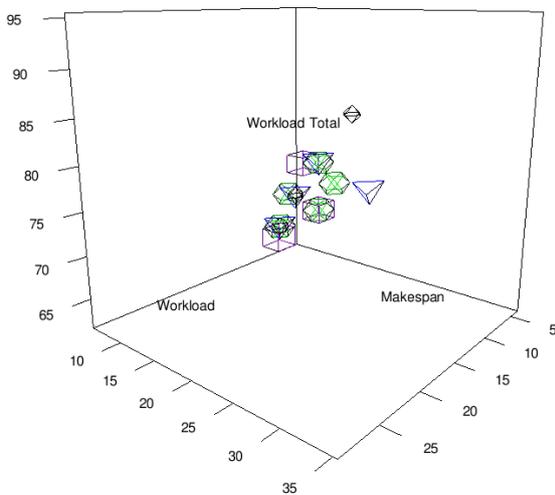
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

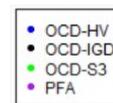
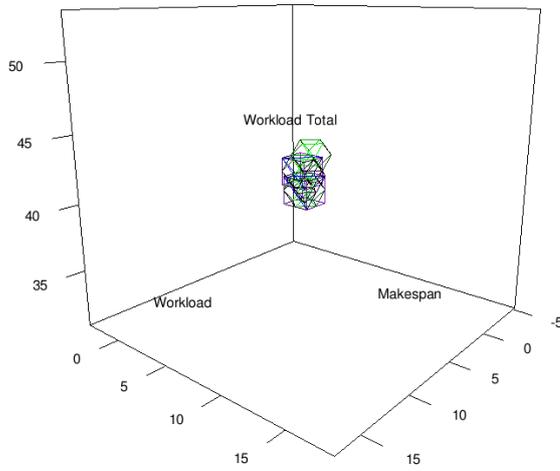
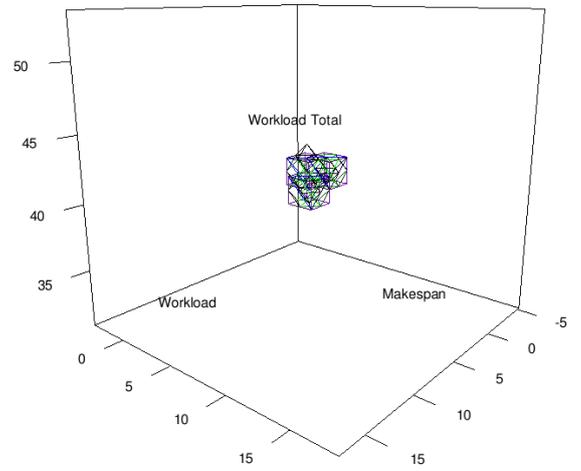


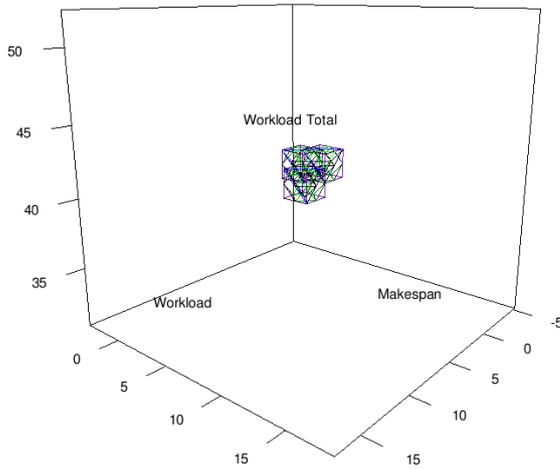
Figura 25 – Gráficos do problema KC10X07



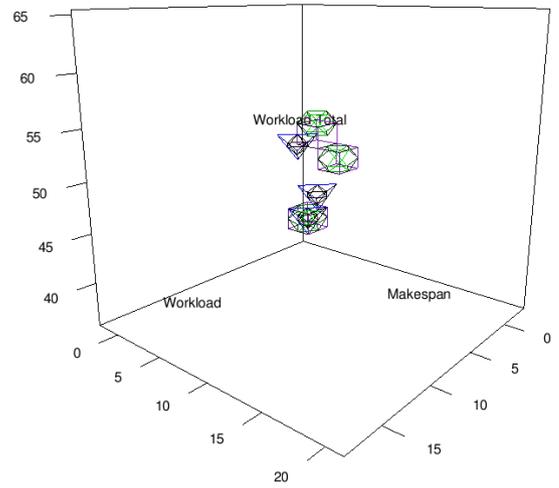
(a) DIPSO



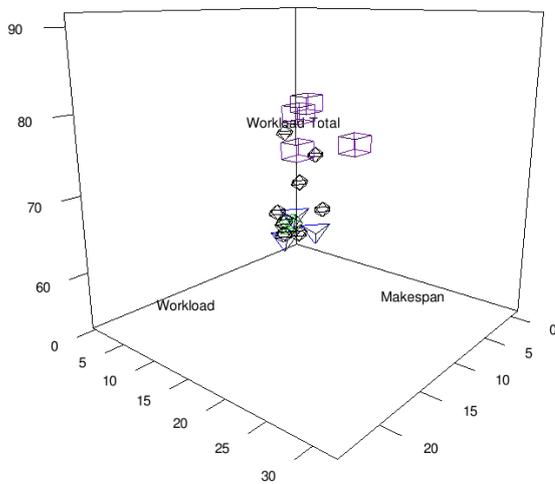
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

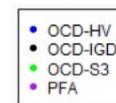
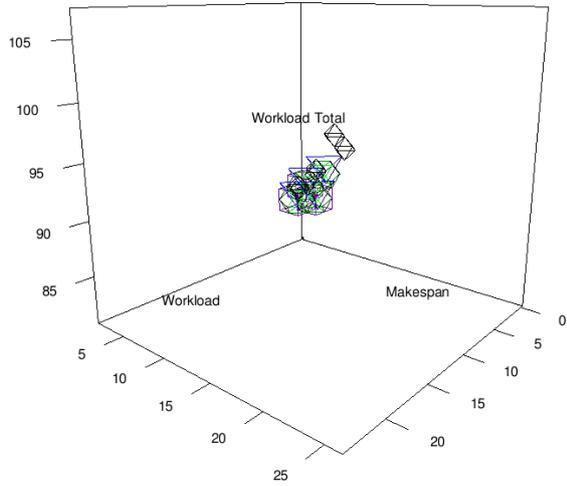
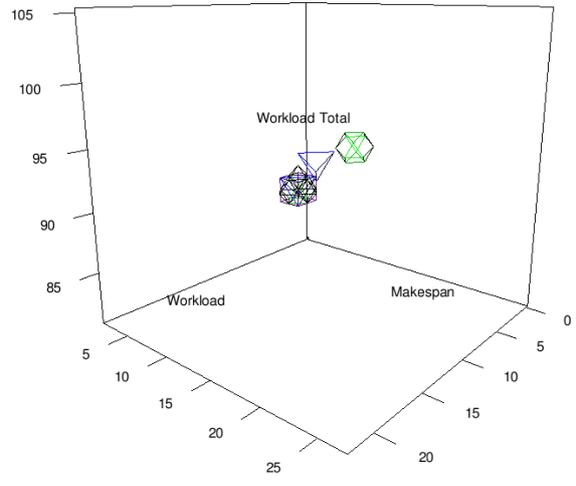


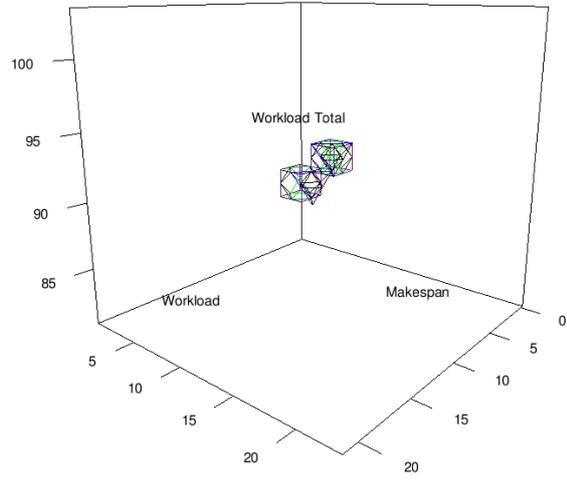
Figura 26 – Gráficos do problema KC10X10



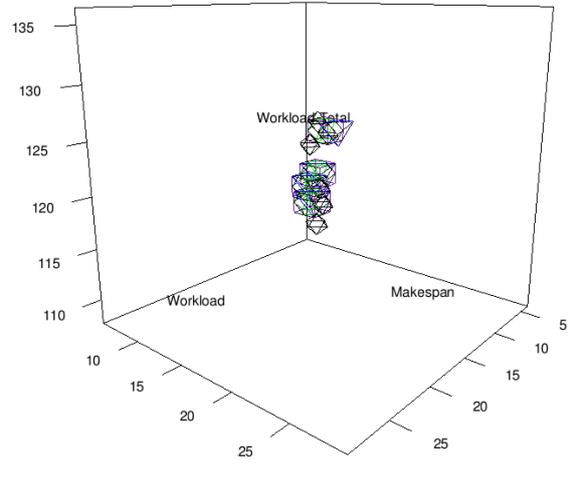
(a) DIPSO



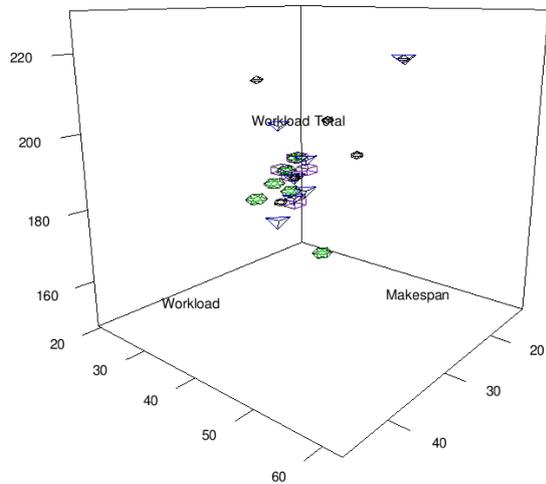
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

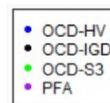
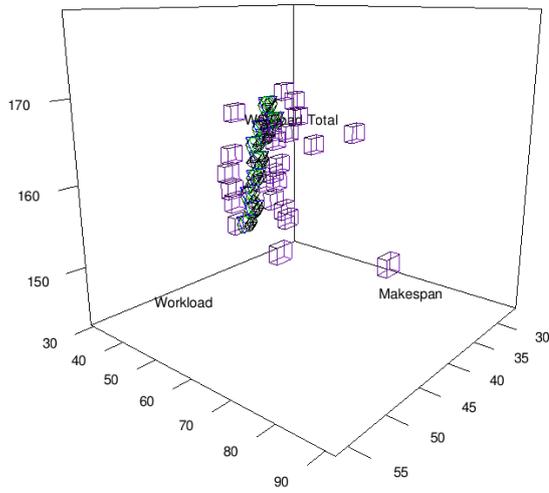
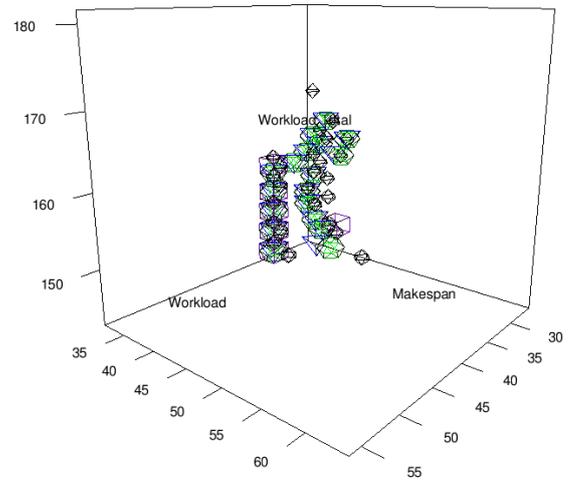


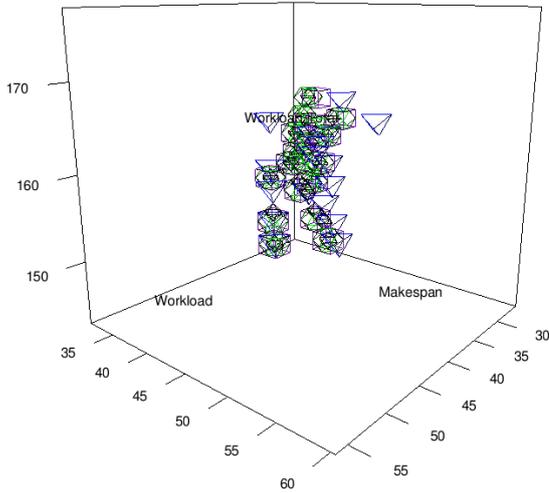
Figura 27 – Gráficos do problema KC15X10



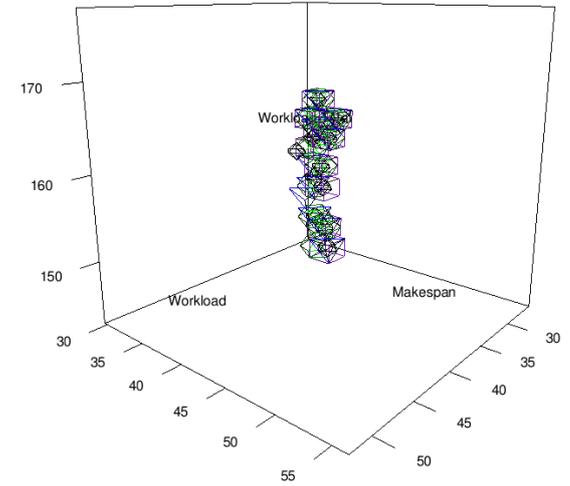
(a) DIPSO



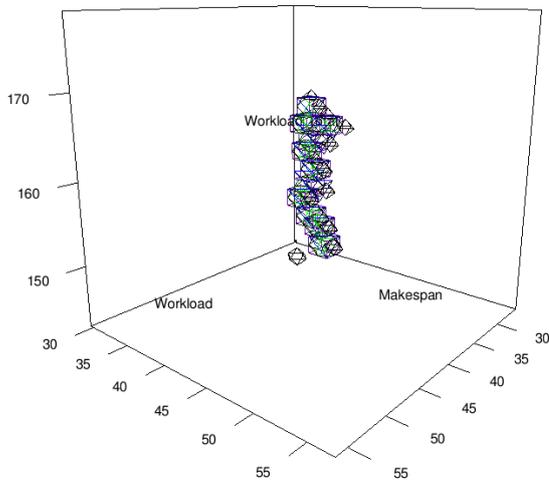
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

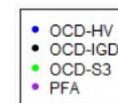
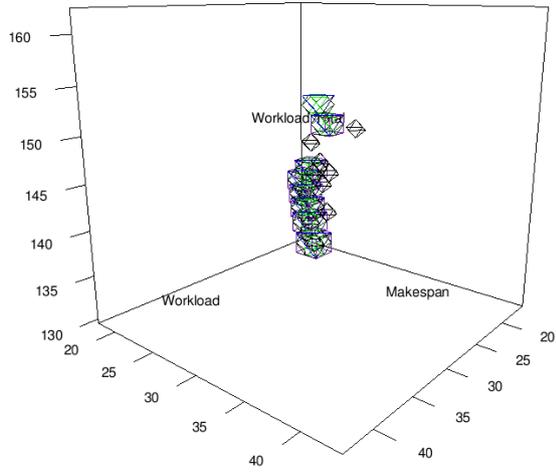
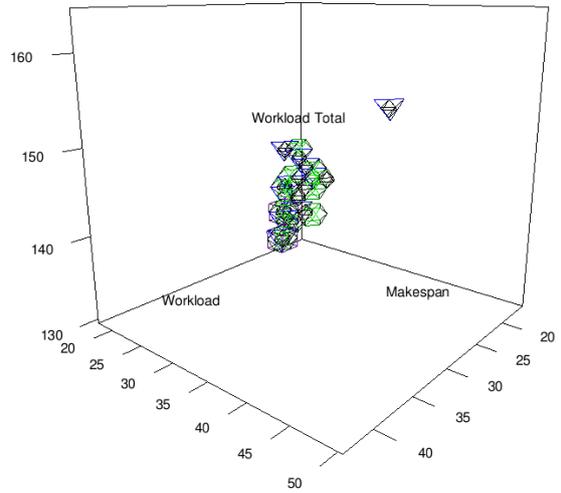


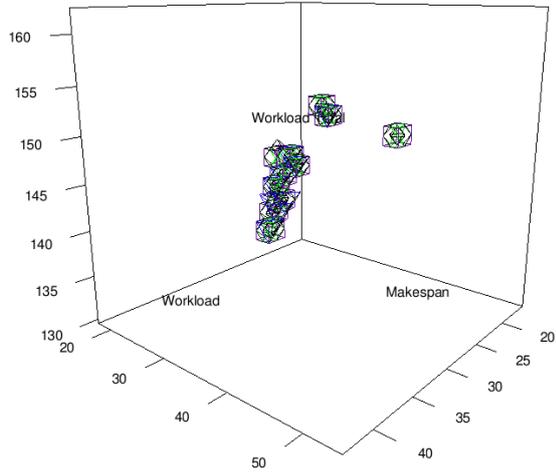
Figura 28 – Gráficos do problema BR MK01



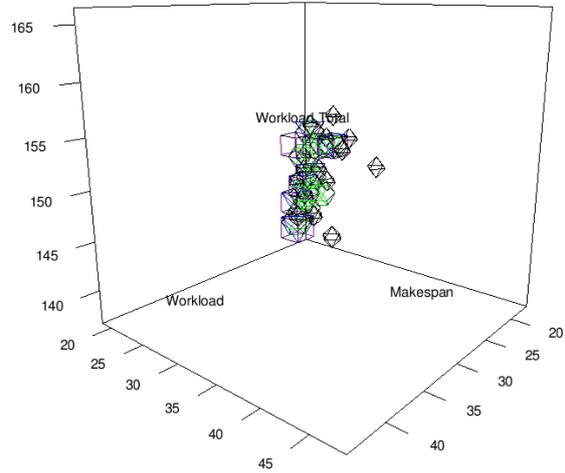
(a) DIPSO



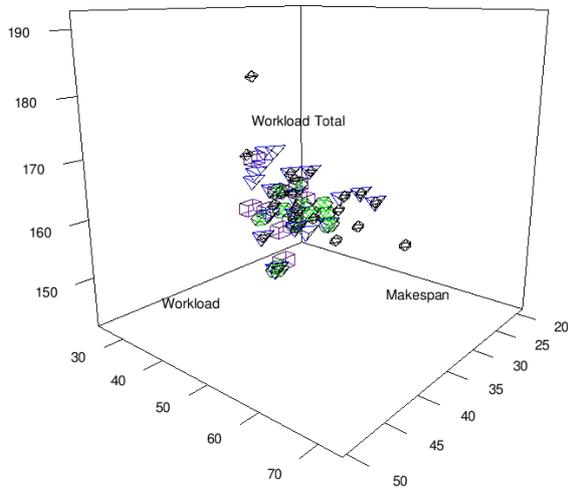
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

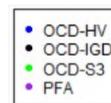
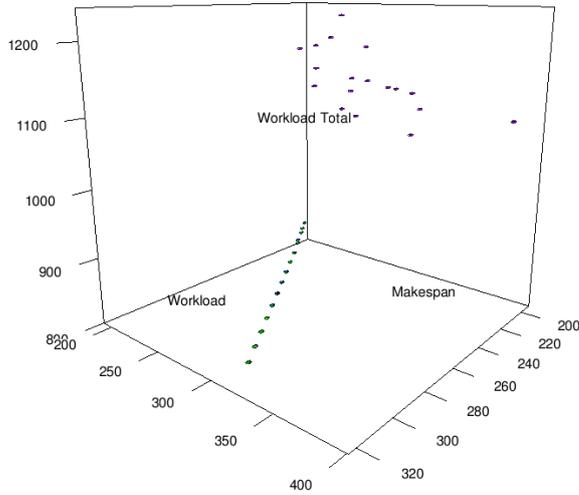
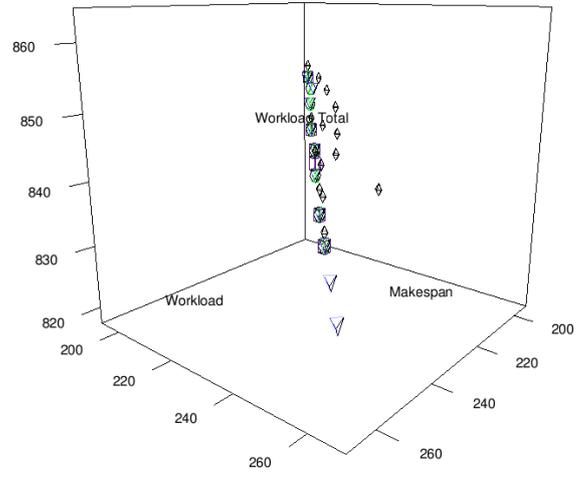


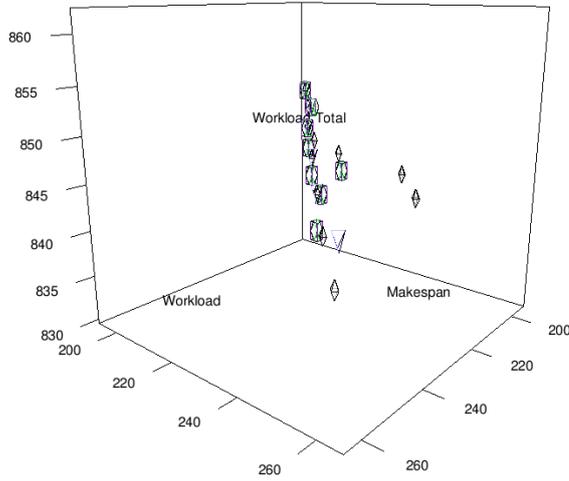
Figura 29 – Gráficos do problema BR MK02



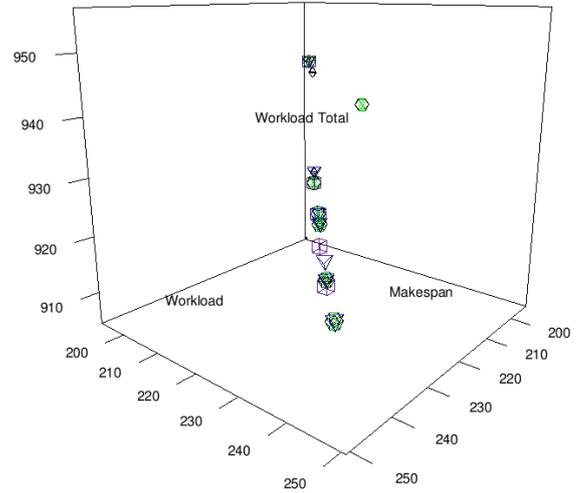
(a) DIPSO



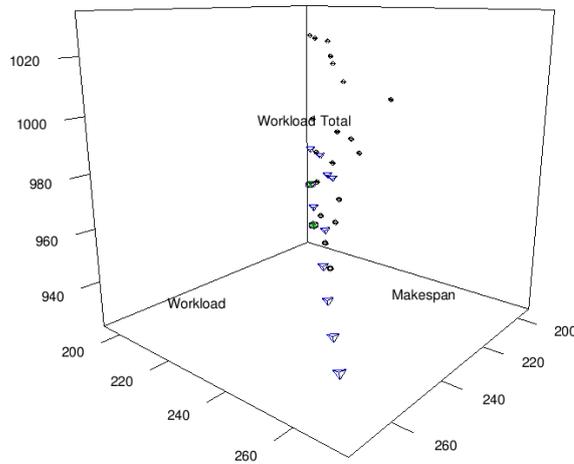
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

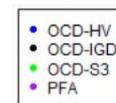
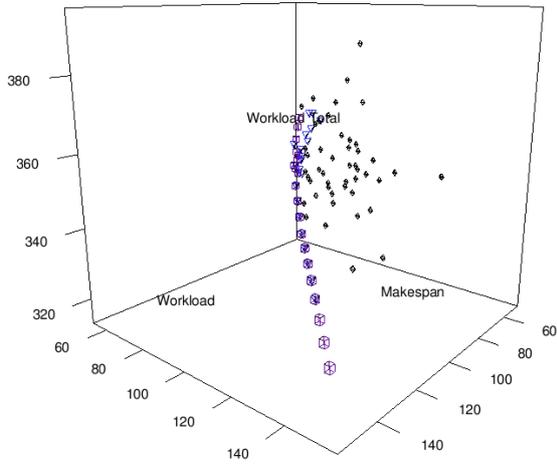
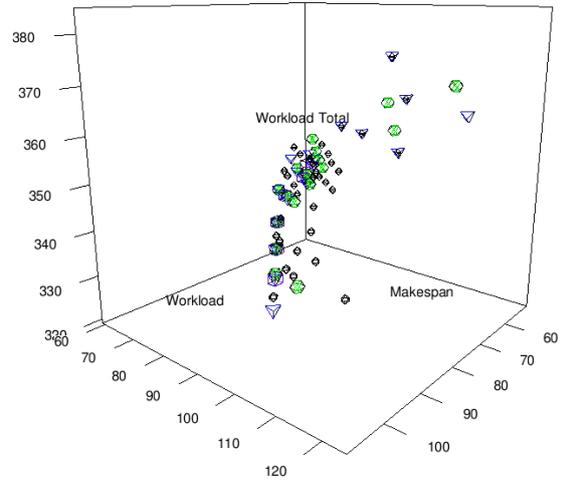


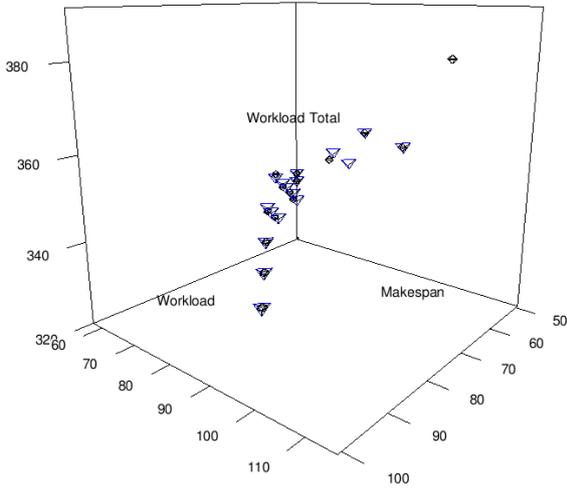
Figura 30 – Gráficos do problema BR MK03



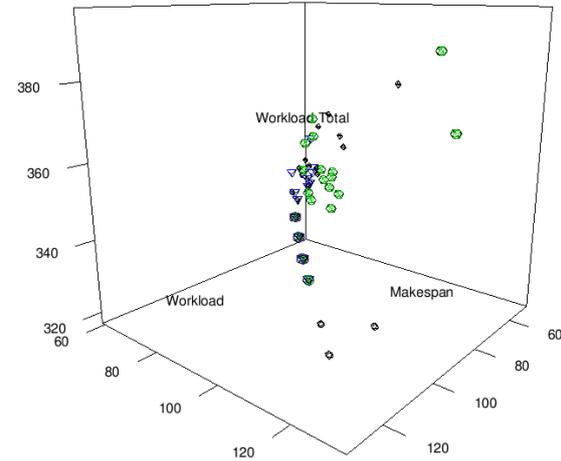
(a) DIPSO



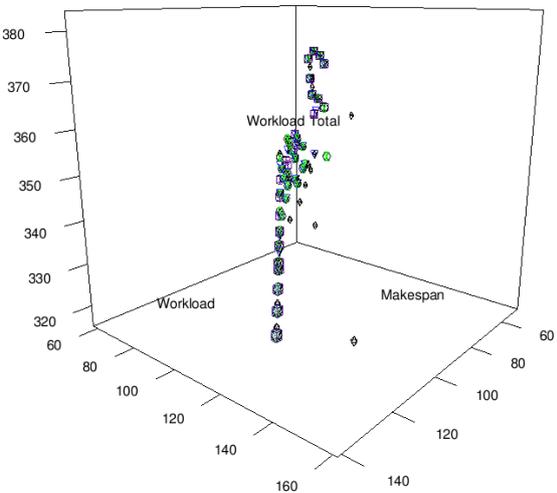
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

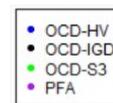
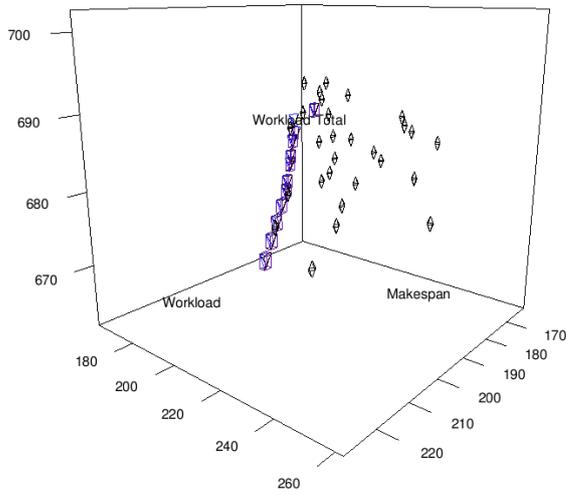
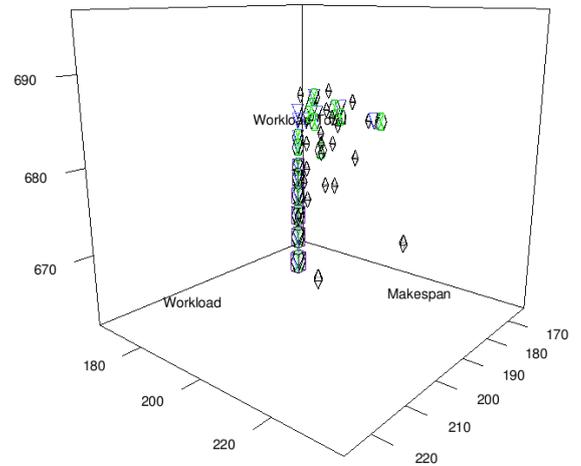


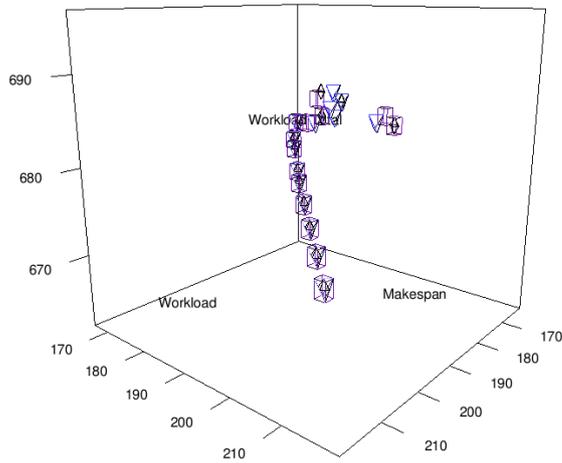
Figura 31 – Gráficos do problema BR MK04



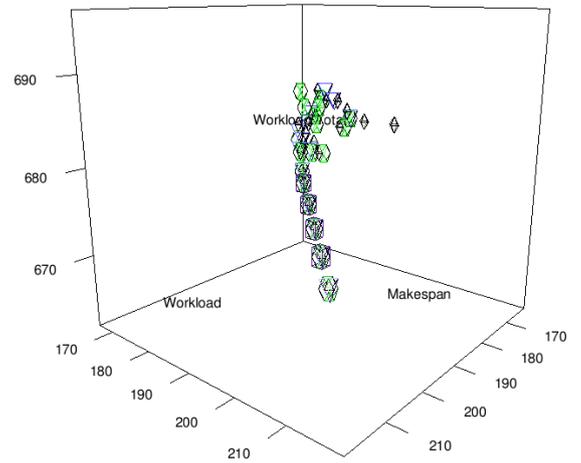
(a) DIPSO



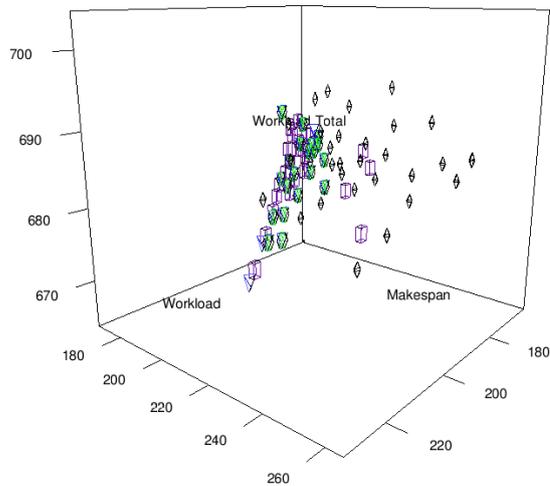
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

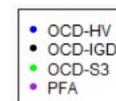
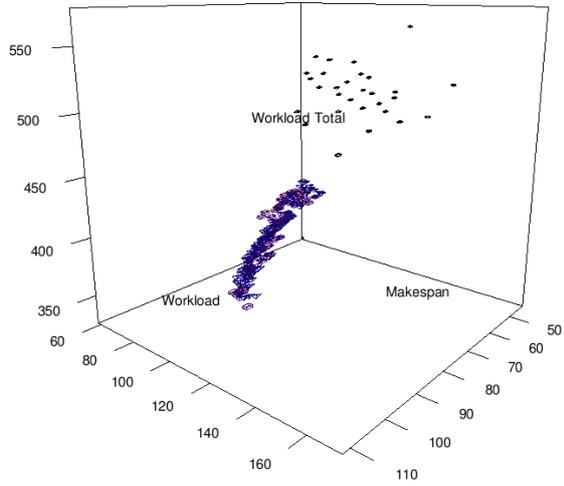
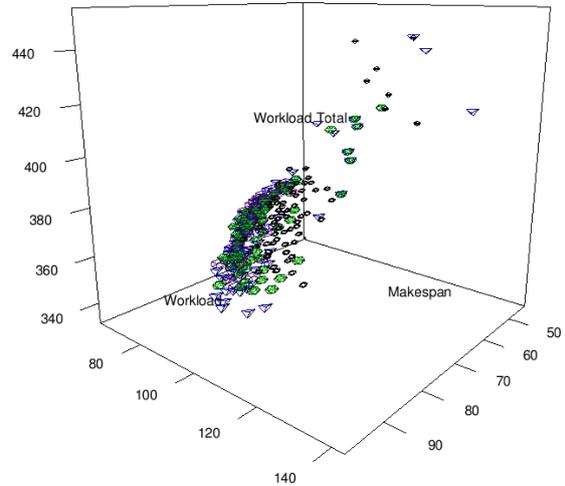


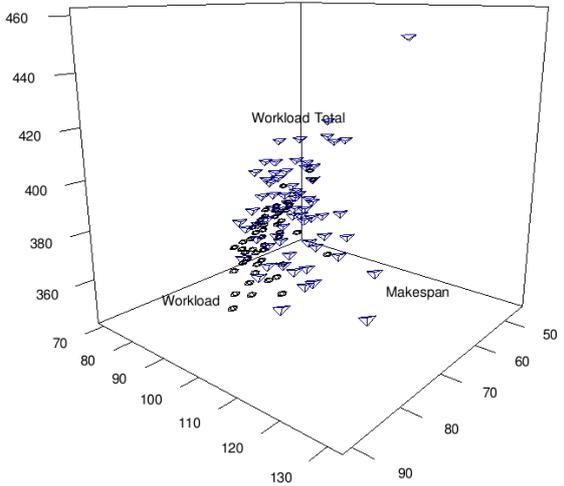
Figura 32 – Gráficos do problema BR MK05



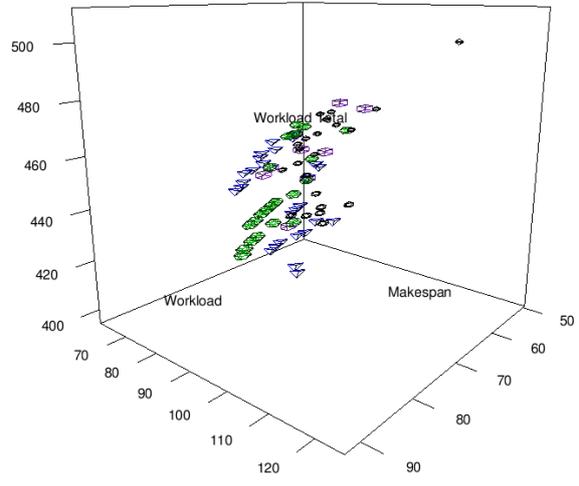
(a) DIPSO



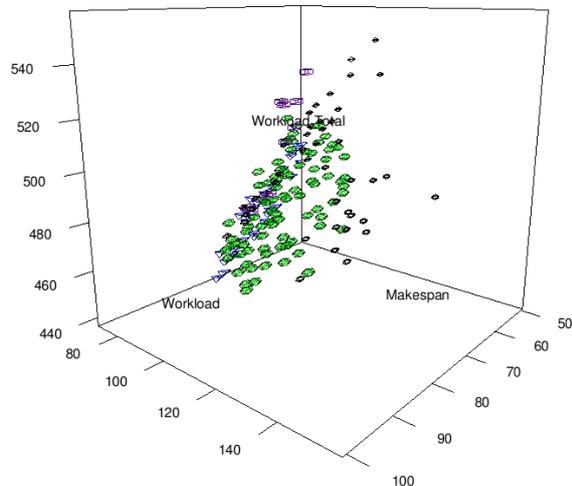
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

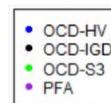
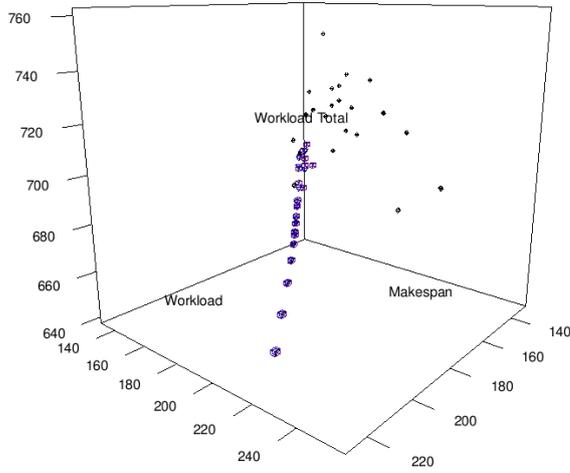
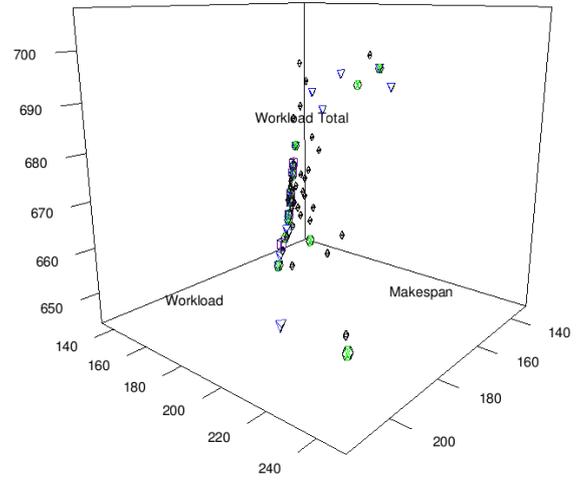


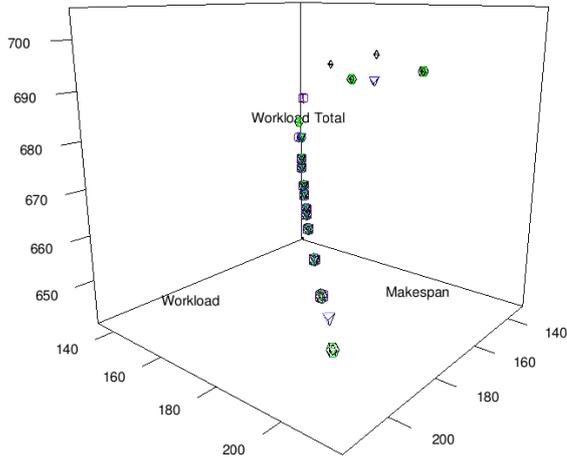
Figura 33 – Gráficos do problema BR MK06



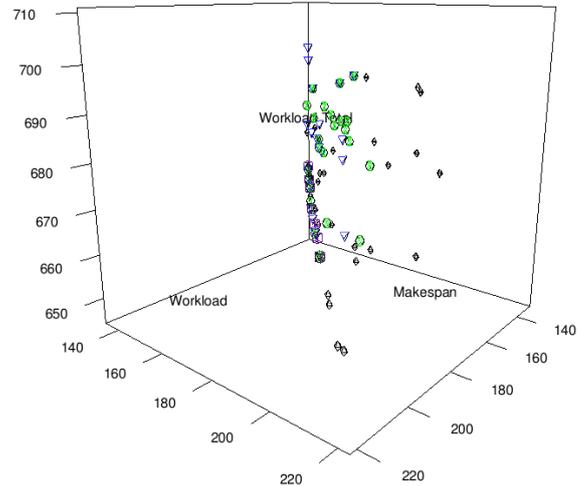
(a) DIPSO



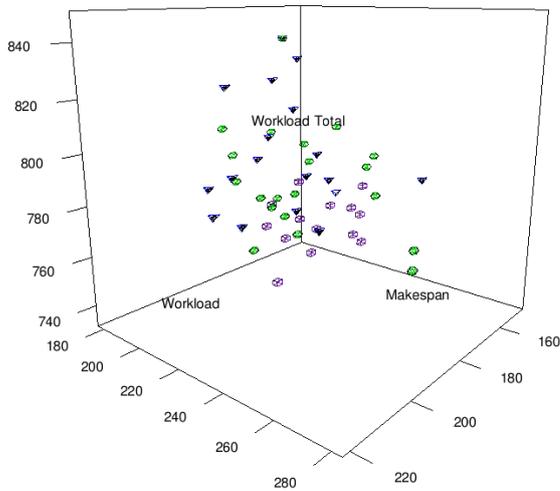
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

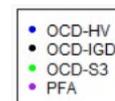
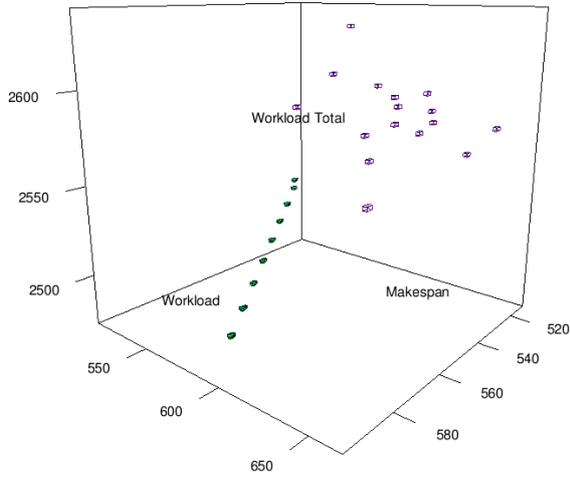
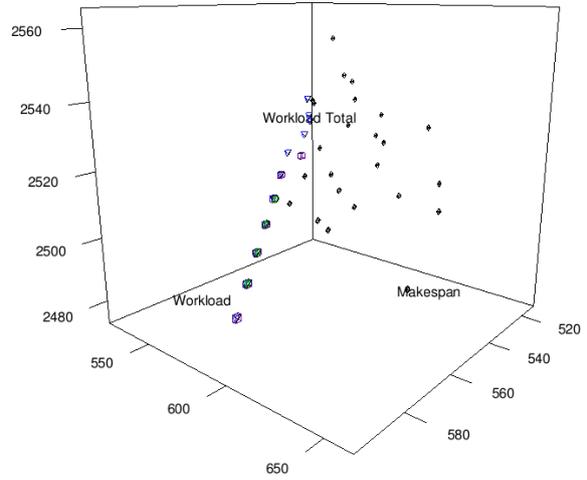


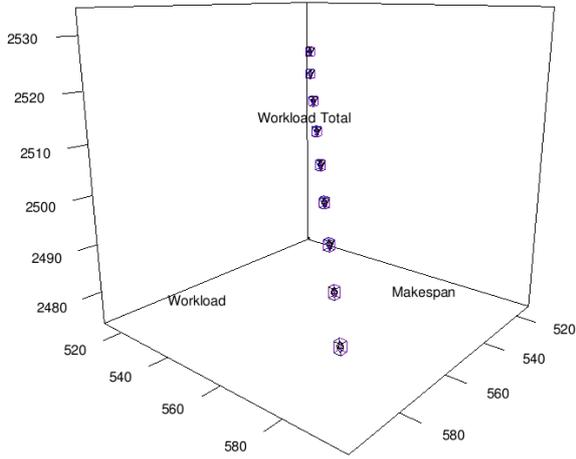
Figura 34 – Gráficos do problema BR MK07



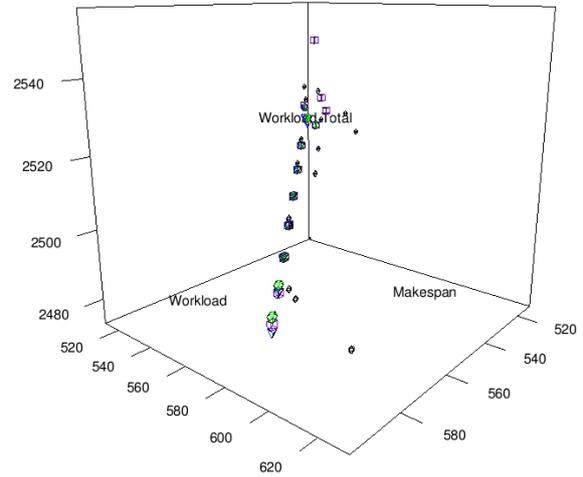
(a) DIPSO



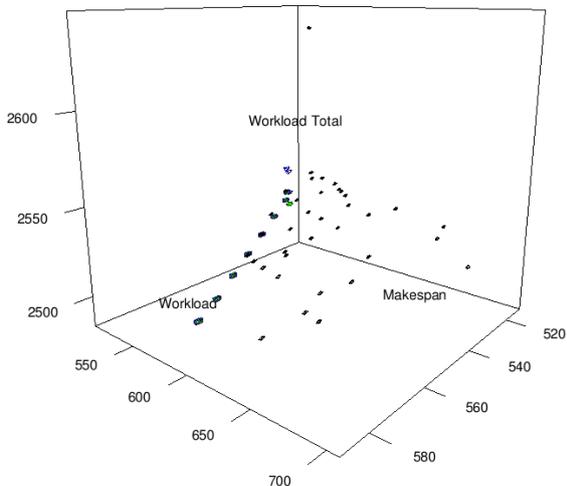
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

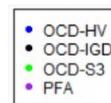
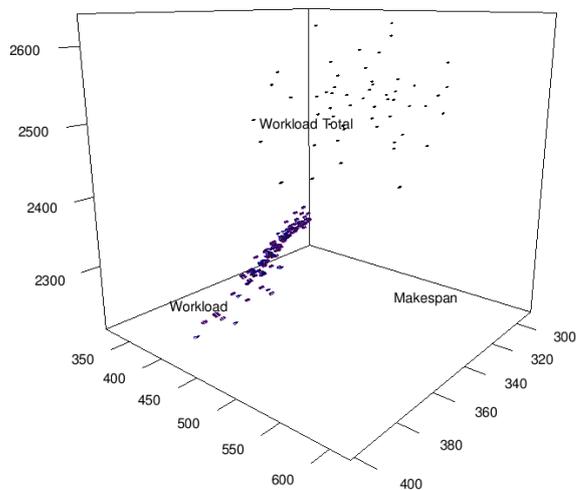
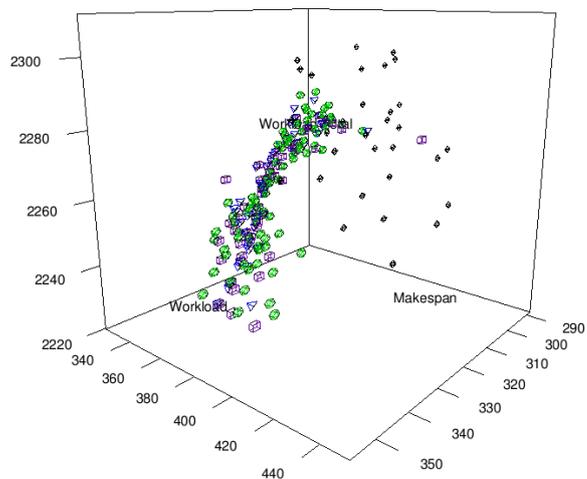


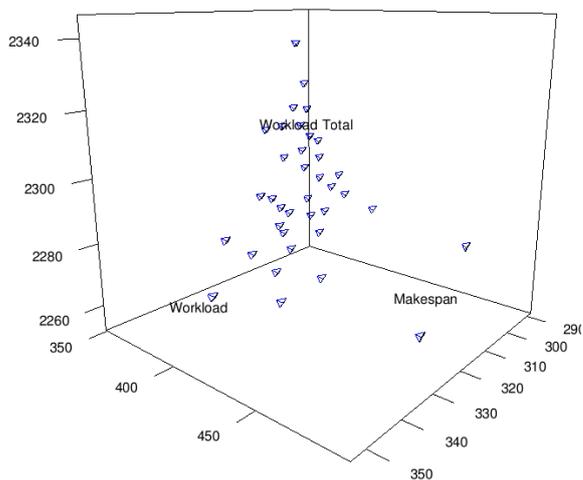
Figura 35 – Gráficos do problema BR MK08



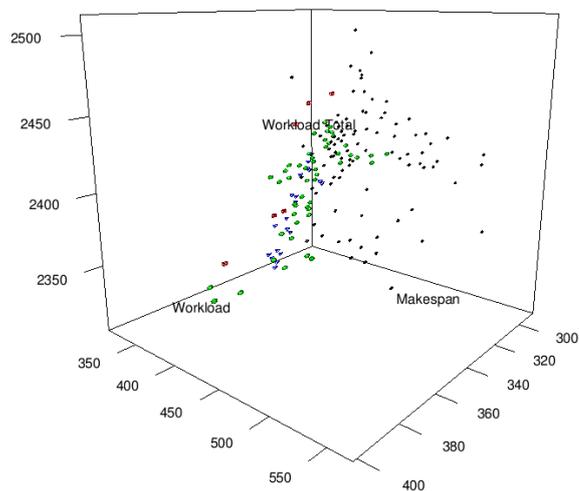
(a) DIPSO



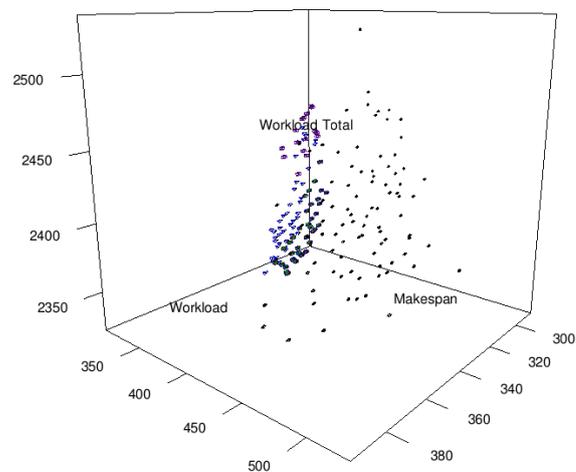
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

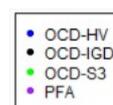
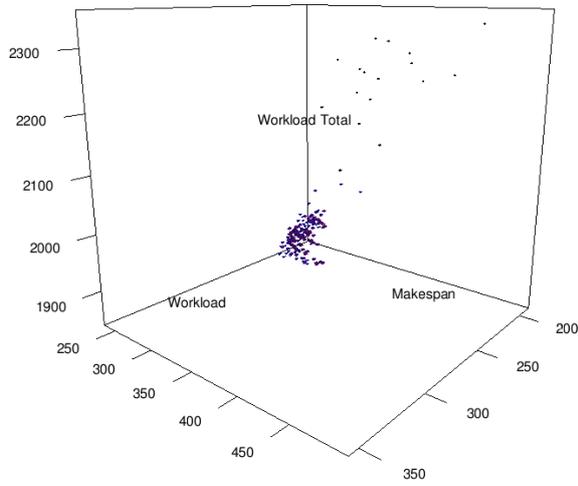
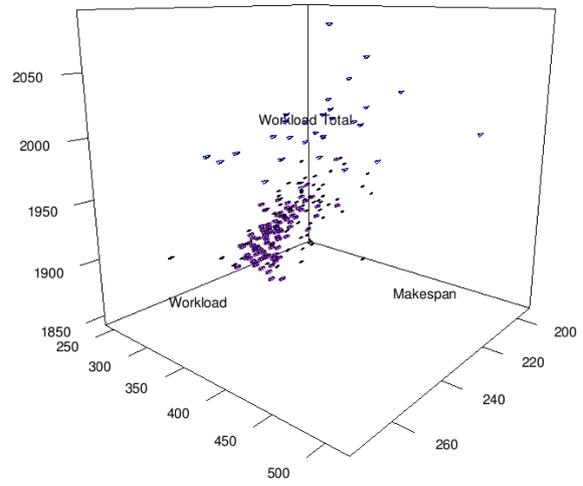


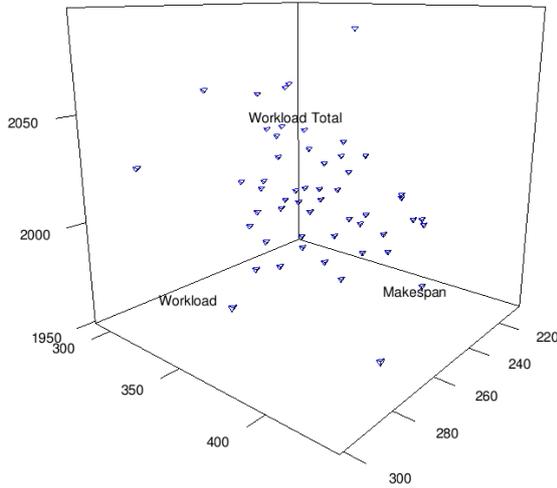
Figura 36 – Gráficos do problema BR MK09



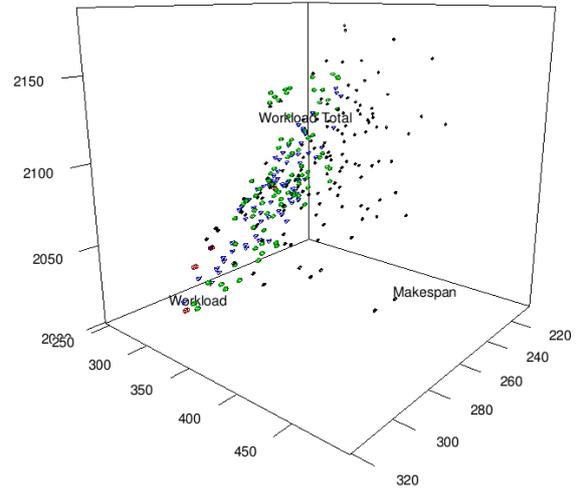
(a) DIPSO



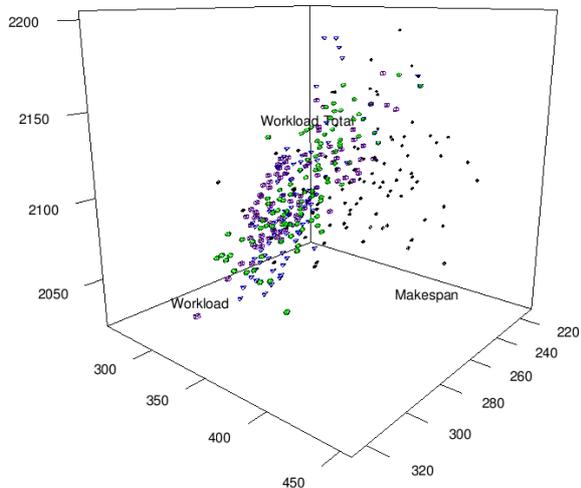
(b) SEDA



(c) SEDASI



(d) NSGA2



(e) SPEA2

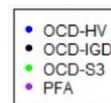


Figura 37 – Gráficos do problema BR MK10