

NOVOS MÉTODOS BASEADOS EM RANKING PARA PROBLEMAS COM MUITOS OBJETIVOS APLICADOS À OTIMIZAÇÃO POR NUVEM DE PARTÍCULAS

André Britto de Carvalho e Aurora Pozo

Universidade Federal do Paraná

{andrebc,aurora}@inf.ufpr.br

Resumo – Algoritmos Evolucionários Multiobjetivo enfrentam vários problemas quando trabalham com um número grande de objetivos. Recentemente, algumas técnicas para muitos objetivos tem sido propostas para evitar a deterioração da busca desses algoritmos. Este trabalho propõe duas novas estratégias baseadas em ranking para trabalhar com problemas com muitos objetivos: uma nova relação de preferência chamada de *Balanced Ranking* e o método da Combinação de Rankings. Além disso, essas novas abordagens foram especialmente aplicadas na Otimização por Nuvem de Partículas. As novas estratégias são validadas através de uma análise empírica visando observar aspectos como convergência e diversidade em relação à fronteira de Pareto para problemas com muitos objetivos.

Palavras-chave – Otimização com Muitos Objetivos, Otimização Multiobjetivo, Otimização por Nuvem de Partículas.

Abstract – Multi-Objective Evolutionary Algorithms face several problems when dealing with a large number of objectives. Recently, some Many-Objective techniques have been proposed to avoid the deterioration of the search ability of these algorithms. This paper proposes two new rank-based strategies to deal with many-objectives problems: a new rank-based preference relation, called *Balanced Ranking*, and a ranking combination. Furthermore, these approaches are especially applied to Particle Swarm Optimization. The new strategies are validated through an empirical analysis, aiming to observe aspects like convergence and diversity in relation to the true Pareto front in many-objective problems.

Keywords – Many-Objective Optimization, Multi-Objective Optimization, Particle Swarm Optimization.

1. INTRODUÇÃO

O interesse pela Otimização com Muitos Objetivos, isto é, quando o número de objetivos otimizados é maior do que 3, tem crescido nos últimos anos [1]. Isto ocorre devido a limitação dos Algoritmos Evolucionários Multiobjetivo baseados em dominância de Pareto (MOEAs, do inglês *Multi-objective Evolutionary Algorithms*) em trabalhar com um número grande de objetivos. Uma dos principais problemas é a deterioração da busca destes MOEAs, pois quando o número de objetivos cresce, há um aumento no número de soluções não-dominadas e há a diminuição da pressão em direção às melhores soluções do problema.

Esse trabalho tem como objetivo propor novas estratégias para problemas com muitos objetivos. Um desafio é aplicar técnicas para muitos objetivos em algoritmos baseados na Otimização por Nuvem de Partículas Multiobjetivo (MOPSO, do inglês *Multi-objective Particle Swarm Optimization*). MOPSO é uma meta-heurística baseada em população utilizada para resolver diversos problemas de otimização. Ela utiliza a cooperação entre os indivíduos e possui características importantes para se trabalhar com a otimização multiobjetivo [2]. Além disso, existem poucos trabalhos na literatura que utilizam MOPSO com muitos objetivos [3].

Neste contexto, são propostas duas novas estratégias para problemas com muitos objetivos aplicadas ao PSO. A primeira técnica é uma nova relação de preferência baseada em ranking [4], chamada de *Balanced Ranking*. Este método busca evitar a geração de um resultado final muito próximo ao extremos da fronteira de Pareto e mais próximo à solução ideal do problema [5] (solução mais próxima aos melhores valores para todos os objetivos). O segundo método busca combinar diferentes métodos de ranking e utilizá-los como objetivos da busca com o intuito de juntar aspectos positivos de cada técnica. Os métodos propostos são aplicados à um algoritmo MOPSO.

Para a validação das estratégias propostas é feita uma análise empírica que visa observar aspectos como convergência e diversidade de cada algoritmo. Além disso, os resultados dos novos métodos são comparados com um algoritmo MOPSO utilizando a dominância de Pareto e com algoritmos MOPSO que utilizam técnicas da literatura baseadas em ranking, como *Average Ranking* (AR) e *Maximum Ranking* (MR) [4]. Para análise da convergência e diversidade e na comparação dos métodos é utilizado um conjunto de indicadores de qualidade: *Generational Distance* (GD), *Inverted Generational Distance* (IGD) e *Spacing*.

O resto trabalho está organizado da seguinte maneira: a Seção 2 apresenta os principais conceitos da otimização com muitos objetivos. Na Seção 3 a meta-heurística da Otimização por Nuvem de Partículas é discutida. Em seguida a Seção 4 apresenta as estratégias propostas. Por fim, a análise empírica dos métodos é discutida na Seção 5 e as conclusões são apresentadas na Seção 6.

2. OTIMIZAÇÃO COM MUITOS OBJETIVOS

Diversos problemas reais costumam possuir múltiplos critérios que devem ser satisfeitos ao mesmo tempo. Além disso, nesses problemas os objetivos (ou critérios) que devem ser otimizados normalmente estão em conflito, ou seja, quando um valor de objetivo é aumentado os valores dos outros objetivo é diminuído. Assim, normalmente não há somente uma melhor solução, mas sim um conjunto de soluções que apresentam um melhor compromisso entre os objetivos.

O problema multiobjetivo geral, de maximização e sem restrições, pode ser definido através da Equação (1).

$$\text{Maximizar } f(x) = (f_1(x), f_2(x), \dots, f_m(x)) \quad (1)$$

Nesse problema busca-se encontrar um conjunto de variáveis de decisão, representado por um vetor solução $x \in \Omega$, onde Ω é a região de todas as soluções possíveis, de forma que o vetor objetivo de x , $f(x)$, definido por m valores de objetivos $f_1(x), f_2(x), \dots, f_m(x)$, tenha o maior valor possível.

A maneira mais utilizada para se tratar problemas multiobjetivo é através da dominância de Pareto [6]. Através desse método, são definidos importantes conceitos com a Fronteira de Pareto e Otimalidade de Pareto. A dominância de Pareto é definida como:

Dadas duas soluções $x \in \Omega$ e $y \in \Omega$, para um problema de maximização, a solução x domina y se:

$$\forall i \in \{1, 2, \dots, m\} : f_i(x) \geq f_i(y), \text{ e } \exists i \in \{1, 2, \dots, m\} : f_i(x) > f_i(y)$$

x domina y se ela é pelo menos igual para todos objetivos, porém deve ser melhor em ao menos um objetivo. x é uma solução não-dominada se não existe nenhuma solução y que domina x .

O objetivo da otimização multiobjetivo é descobrir soluções que não são dominadas por nenhuma outra no espaço de objetivos. O conjunto de todas as soluções não dominadas é chamado conjunto ótimo de Pareto e o conjunto de todos os vetores objetivos não-dominados é chamado de fronteira de Pareto. Na maioria das aplicações, a busca pelo ótimo de Pareto é NP-Difícil e então foca-se em encontrar um conjunto de aproximação ao ótimo de Pareto. Algoritmos Evolucionários Multiobjetivo, MOEAs, tem sido aplicados com sucesso para vários problemas multiobjetivo [7]. MOEAs são particularmente adequados para estas tarefas pois eles evoluem simultaneamente uma população de soluções para o problema obtendo um conjunto de soluções que aproximam a fronteira de Pareto em uma só execução do algoritmo.

Apesar da aplicação com sucesso dos MOEAs, a maioria dos estudos em problemas multiobjetivo trabalham com um número pequeno de objetivos, embora problemas práticos envolvem um número grande de critérios. No entanto, o esforço dos pesquisadores vem sendo direcionando para investigar a escalabilidade destes algoritmos com relação ao número de objetivos [1]. Problemas multiobjetivo com mais de 3 objetivos são chamados de problemas com Muitos Objetivos, ou *Many-Objective problems*.

Os MOEAs encontram diversos problemas quando aplicados à problemas com muitos objetivos [1]. A principal razão para isto é que a o número de soluções não-dominadas cresce muito quando o número de objetivos cresce. Como consequência, a habilidade de busca é deteriorada porque não é possível impor um preferência na seleção das soluções. Além disso o número de soluções para aproximar a fronteira de Pareto também aumenta e há uma maior dificuldade para observar as soluções geradas. Atualmente, pesquisadores enfrentam esse problemas utilizando principalmente três abordagens. A primeira abordagem é a adaptação das relações de preferência para induzir uma ordem mais fina no espaço de objetivos [3] [8]. Outra abordagem é a redução da dimensão do espaço de objetivos, discutida em [9]. A terceira abordagem se refere a estratégias de decomposição, que decompõe os problemas com muitos objetivos em diferentes problemas escalares e então algoritmos evolutivos são aplicados [10].

Porém, apesar das propostas de novas soluções, existem diversas dificuldades que motivam o nosso trabalho. Uma delas é relacionada à metaheurística utilizada. A maioria dos trabalhos utilizam algoritmos genéticos, normalmente o NSGA-II. Neste trabalho, estamos interessados em estudar o comportamento da Otimização por Nuvem de Partículas, ou *Particle Swarm Optimization* (PSO). Alguns trabalhos lidam com problemas com muitos objetivos e PSO.

Dentre esses trabalhos, podemos destacar [11] que estudou a influência do Controle da Área de Dominância das Soluções (CADS) [8] em diferentes algoritmos PSO. Este trabalho mostrou que a técnica CADS melhora os resultados do PSO para problemas com muitos objetivos. Além deste, em [3] é apresentado um amplo estudo que aplica a técnica AR e o CADS em uma algoritmo PSO buscando identificar aspectos como convergência e diversidade na busca do PSO. Para a utilização do PSO com o AR foi proposto um novo algoritmo chamado de AR-SMPSO, que substitui a dominância de Pareto por uma nova relação de preferência, que utiliza o ranqueamento do AR para definição das melhores soluções. Baseado nesses trabalhos da anteriores, este artigo visa produzir novas estratégias para problemas com muitos objetivos aplicadas em algoritmos PSO. As estratégias desenvolvidas são baseadas em ranking, pois são de fácil implementação e já são validadas em diversos trabalhos da literatura.

3. OTIMIZAÇÃO POR NUVEM DE PARTÍCULAS

A Otimização por Nuvem de Partículas, PSO, do inglês *Particle Swarm Optimization*, é uma metaheurística baseada em população inspirada por pássaros em busca de alimento. PSO executa uma busca cooperativa entre as soluções. O conjunto com as soluções possíveis é um conjunto de partículas, chamado de enxame, que se move pelo espaço de busca através de um operador de velocidade que é baseado na melhor posição das partículas vizinhas (componente social), chamado de líder, e na melhor posição da partícula (componente local). A Otimização por Nuvem de Partículas Multiobjetivo, MOPSO, usa conceitos da dominância de Pareto para definir os líderes. O conjunto de soluções não dominadas, que representam os líderes, é guardado

num arquivo externo. Cada partícula do enxame pode ter diferentes líderes, porém somente um pode ser escolhido para a atualização da velocidade.

Os passos do algoritmo básico do MOPSO são: inicialização das partículas, cálculo da velocidade, atualização da velocidade, mutação e atualização do arquivo dos líderes. Cada partícula p_i , num tempo t , possui uma posição $x(t) \in R^n$ (Equação 2), que representa uma possível solução. A posição de uma partícula, no tempo $t + 1$, é obtida adicionando a velocidade $v(t) \in R^n$ (Equação 3), a $x(t)$:

$$\vec{x}(t+1) = \vec{x}(t) + \vec{v}(t+1) \quad (2)$$

A velocidade de uma partícula p_i é baseada na melhor posição já alcançada pela partícula, $\vec{p}_{best}(t)$, e a melhor posição já alcançada pelos vizinhos de p_i , $\vec{R}_h(t)$, isto é um líder do repositório. A velocidade é definida como:

$$\vec{v}(t+1) = \varpi \cdot \vec{v}(t) + (C_1 \cdot \phi_1) \cdot (\vec{p}_{best}(t) - \vec{x}(t)) + (C_2 \cdot \phi_2) \cdot (\vec{R}_h(t) - \vec{x}(t)) \quad (3)$$

As variáveis ϕ_1 e ϕ_2 , em (3), são coeficientes que determinam a influência dos líder local e do líder global e são definidos aleatoriamente em cada iteração. As variáveis C_1 e C_2 indicam o quanto cada componente influencia a velocidade. O coeficiente ϖ é a inércia da partícula e controla o quanto a velocidade anterior afeta a atual. \vec{R}_h é uma partícula do repositório. O repositório com as melhores soluções é preenchido após todas as partículas terem sido atualizadas. Existem diversas maneiras para escolher o líder, tal como o método Sigma, torneio binário [2], entre outros. Após atualização da velocidade um operador de mutação é aplicado. No final da execução o conjunto de partículas presentes no repositório é a solução do problema.

3.1 SMPSO

O algoritmo SMPSO é um algoritmo PSO multiobjetivo e foi apresentado em [12]. Este algoritmo possui a característica de limitar a velocidade das partículas por um fator de restrição χ , que varia baseado nas variáveis C_1 e C_2 . O SMPSO introduz um mecanismo que limita o valor acumulado de cada variável da velocidade de cada partícula, assim restringe o espaço da busca e aumenta o poder de convergência de um algoritmo PSO. Os valores superior e inferior do limite são definidos pelo usuário. Após a atualização da velocidade, é efetuada a mutação, aplicada em 15% da população. O líder de cada partícula é escolhido através de um torneio binário. O repositório com as soluções não-dominadas é limitado, com o mesmo tamanho da população, e é utilizada a *Crowding Distance* (CD) [13] para introduzir diversidade na busca.

3.2 AR-SMPSO

O algoritmo AR-SMPSO [3] é baseado no algoritmo SMPSO. Nesse algoritmo é utilizada a relação de preferência \prec_{AR} para definir se uma solução é melhor que a outra, ao invés da dominância de Pareto. Essa nova relação de preferência utiliza o método *Average Ranking* (AR) [4] para definir se uma solução é melhor que outra. A nova relação define que uma solução x é melhor que y , se o AR de x é menor que o AR de y . Este algoritmo segue os passos básicos do SMPSO discutidos anteriormente, com algumas modificações. Primeiro, na inicialização, para efetuar uma pressão inicial em direção às melhores soluções, somente 10% (melhores soluções de acordo com a relação \prec_{AR}) da população inicial preenche o repositório. No laço evolutivo, a atualização do repositório é a principal diferença entre o AR-SMPSO e o SMPSO. Nesta etapa, inicialmente é calculado o AR de cada solução e então uma solução entra no repositório se e somente se ela domina qualquer outra do repositório de acordo com a relação \prec_{AR} . No procedimento de poda, o número de soluções no repositório é no máximo igual ao tamanho da população. Somente as partículas com melhor AR ficam no repositório, se houver empate é utilizada a CD, para manter a diversidade na busca. Para evitar que soluções dominadas pela dominância de Pareto influenciem a busca para regiões ruins do espaço de objetivos, essas soluções são retiradas, ao final de cada iteração.

4. NOVAS ESTRATÉGIAS PROPOSTAS

Essa seção apresenta as novas estratégias propostas. Inicialmente, o método *Balanced Ranking* (BR) é apresentado na Seção 4.1 e em seguida a *Combinação de Rankings* é apresentada na Seção 4.2

4.1 Balanced Ranking

A primeira estratégia proposta por este trabalho é uma nova relação de preferência chamada de *Balanced Ranking* (BR). Essa técnica é uma relação de preferência que induz uma ordem de preferência para um conjunto de soluções. Ela tem como base o mesmo princípio adotado pelos métodos AR e MR propostos por [4], porém busca evitar a geração de soluções nos extremos da fronteira de Pareto [14]. A Figura 1 mostra um exemplo de uma fronteira de Pareto em um espaço de objetivos bi-dimensional, para um problema de maximização. A linha tracejada indica a fronteira de Pareto, os pontos próximos aos eixos indicam as soluções com valores extremos, ou seja, possuem o valor de um objetivo muito maior que os demais. O ponto no centro da linha indica a solução ideal [5], ou seja, solução mais próxima aos melhores valores para todos os objetivos.

A ideia do método proposto é privilegiar soluções em que os objetivos possuam valores semelhantes, ou seja, privilegiar soluções que não contenham valores extremos em relação aos demais e sejam mais próximas à solução ideal. Os passos iniciais

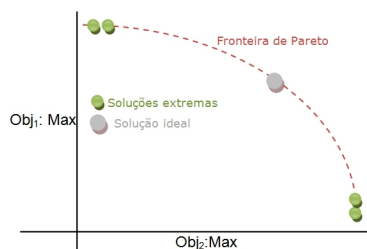


Figura 1: Exemplo de uma fronteira de Pareto com soluções com valores extremos e a solução ideal

Tabela 1: Exemplo do Balanced Ranking

#	(f_1, f_2, f_3)	ranking(f_1)	ranking(f_2)	ranking(f_3)	MAX	MIN	BR	AR	MR
A	(9, 1, 3)	5	1	2	5	1	5,3	8	1
B	(4, 2, 6)	3	2	4	4	2	3	9	2
C	(1, 7, 7)	1	4	5	5	1	6,66	10	1
D	(2, 8, 1)	2	5	1	5	1	5,33	8	1
E	(7, 5, 8)	4	3	6	6	3	7	14	3
F	(10, 9, 4)	6	6	3	6	3	7,5	15	3

para o cálculo do BR são os mesmos necessários para o cálculo do AR e do MR. Assim, dado um conjunto S de soluções, o BR de uma solução s_i é calculado da seguinte maneira. Primeiro, para cada objetivo, é calculado um ranking entre todas as soluções analisadas. Para o cálculo do ranking é tomada a convenção de quanto melhor é a solução, mais baixo é o ranking. Após, é feita a soma de todos desses rankings. Em seguida, é calculada uma ponderação que é obtida através da diferença entre o maior e o menor ranking de uma solução. Essa diferença, é normalizada pelo tamanho de S (o maior ranking possível), assim, quanto maior a diferença entre os rankings, pior será o BR da solução e quanto menor a diferença, melhor será o BR. O BR pode ser definido através Equação (4).

$$BR(s_i, S) = \frac{MAX_{ranking}(s_i) - MIN_{ranking}(s_i)}{|S|} * \sum_{1 < j < m} ranking(f_j(s_i)) \quad (4)$$

onde m é o número de objetivos, s_i a solução a ser rankeada, S o conjunto de soluções e $ranking(f_j(S))$ é o ranking para o j -ésimo objetivo. A Tabela 1 apresenta um exemplo do funcionamento do BR. É apresentado também o cálculo do AR e do MR. O AR faz a soma de todos os rankings e o MR define como valor o melhor ranking encontrado. Primeiro, cada solução é rankeada para cada objetivo, por exemplo, a solução C possui o melhor ranking (ranking 1) para o primeiro objetivo. Após, é obtido para cada solução os valores máximo e mínimo. Por fim, é feita a soma dos rankings e essa soma é ponderada pela diferença dos rankings normalizada. Através das duas primeiras soluções pode-se perceber a diferença entre o AR e o BR. Utilizando o AR a melhor solução é a A , pois possui um AR igual a 8. Porém, percebe-se que ela possui o melhor ranking para um objetivos (ranking 1 para o segundo objetivo), mas apresenta um ranking ruim para outros (ranking 5 para o primeiro objetivo). Utilizando o BR essa diferença é penalizada e a melhor solução passa ser a B , que apesar de não ser melhor em nenhum objetivo possui rankings semelhantes em todos.

4.2 Combinação de Rankings

A outra estratégia proposta é chamada de Combinação de Rankings e visa juntar aspectos positivos de diferentes esquemas de ranking. Como discutido anteriormente, métodos como o AR e o MR tendem a gerar soluções com um objetivo com melhores valores que os demais, assim, o conjunto de aproximação gerado tende a ser localizado nos extremos da fronteira de Pareto. Em contrapartida, o método proposto na seção anterior, BR, privilegia soluções com valores dos objetivos semelhantes e assim evita a geração do conjunto de aproximação nos extremos. Porém, os dois métodos apresentam deficiências, pois é preferível que o conjunto de aproximação fique próximo à solução ideal, mas também é importante obter diversidade sobre toda a fronteira de Pareto.

A Combinação de Rankings utiliza uma ideia de reduzir um número grande de objetivos para poucos objetivos, normalmente 2 ou 3. No método proposto, os objetivos de uma solução passam a ser o valores de diferentes rankings escolhidos. Os passos da Combinação de Rankings são: Dada uma solução x e o seu vetor de objetivos $f(x) = (f_1(x), f_2(x), \dots, f_m(x))$, onde m é o número de objetivos; calcular o valor de cada ranking $r_j = ranking_j(x, S)$, onde S é o conjunto de soluções, $ranking_j$ é um método de ranking e $2 \leq j \leq 3$ é o número de rankings escolhidos. É importante que o número de rankings não seja maior do que 3 para que não haja deterioração na busca do algoritmo; definir os valores dos rankings como objetivos de x , $f'(x) = (r_1, \dots, r_j)$; executar iteração do algoritmo evolutivo multiobjetivo com os novos valores $f'(x)$.

Por exemplo, utilizando os rankings AR e BR, pode-se executar um MOEA tradicional (NSGA-II [13], SMP SO [12]) e otimizar tanto um objetivo que gere soluções nos extremos, quanto um objetivo que gere soluções no centro da fronteira de Pareto. A hipótese da Combinação de Rankings é que juntando diferentes rankings como objetivo de uma solução pode-se obter um melhor conjunto de aproximação do que a aplicação de cada método separado.

Tabela 2: Melhores métodos para cada objetivo e cada problema de acordo com o teste de Friedman

Prob	Obj	Melhor método		
		GD	IGD	Spacing
DTLZ2	2	DP, AR_BR, BR_MR	DP, AR_BR, BR_MR	DP, AR, MR, AR_BR, BR_MR
	3	DP, AR, AR_BR	DP, AR_BR, BR_MR	AR, BR
	5	AR, BR	AR_BR, BR_MR	AR, BR
	10	AR_BR, BR_MR	AR_BR, BR_MR	MR, AR_BR
	15	MR, AR_BR	MR, AR_BR, BR_MR	MR, AR_BR
	20	MR, AR_BR	MR, AR_BR, BR_MR	MR, AR_BR
DTLZ4	2	BR	BR, AR_BR	BR, MR
	3	Todos	Todos	AR, BR, MR, AR_BR, BR_MR
	5	AR, BR, AR_BR	DP, AR, BR, AR_BR	AR, BR, AR_BR
	10	AR, BR, AR_BR	AR, BR, AR_BR	AR, BR, AR_BR
	15	AR, BR, AR_BR	AR, BR, AR_BR	AR, BR, AR_BR
	20	AR, BR, AR_BR	AR, BR, AR_BR	AR, BR, AR_BR

5. ANÁLISE EMPÍRICA

Esta seção apresenta a análise empírica que tem como objetivo comparar as estratégias propostas com os métodos de ranking da literatura AR e MR em termos da convergência e diversidade da busca. Para essa validação são utilizados dois problemas com muitos objetivos da família de problemas DTLZ, DTLZ2 e DTLZ4, comumente utilizada na análise de MOEAs [15]. Para cada problema, uma variável k representa a complexidade da busca, onde $k = n - m + 1$, onde n é o número de variáveis de decisão e m o número de objetivos. O problema DTLZ2 é utilizado para investigar a habilidade do algoritmo escalar sua performance para um número grande de objetivos. O problema DTLZ4 é utilizado para investigar a habilidade dos algoritmos em manter uma boa diversidade das soluções. Como estamos interessados em analisar as abordagens propostas para muitos objetivos, os problemas foram dimensionados com 2, 3, 5, 10, 15 e 20 objetivos

A relação BR é aplicada ao algoritmo AR-SMPSO e a combinação de rankings é utilizada com algoritmo SMPSO. Logo, os parâmetros utilizados para as duas abordagens são os mesmos: o critério de parada foi definido como o número máximo de gerações, que foi definido como 100 e a população possui 250 partículas. ω varia no intervalo de $[0, 0.8]$ e ambos ϕ_1 e ϕ_2 variam no intervalo de $[0, 1]$. C_1 e C_2 variam no intervalo $[1.5, 2.5]$. O tamanho do repositório é o mesmo tamanho da população. É aplicada a mutação polinomial com probabilidade $p_{mut} = 1/n$. Cada variável da velocidade é limitada no intervalo de $[-5, +5]$. Todos os parâmetros foram definidos com os valores apresentados em [12], que utilizou os mesmo problemas da família DTLZ nos experimentos.

Neste conjunto de experimentos foram consideradas 5 diferentes abordagens: A dominância de Pareto (DP), através da execução do algoritmo SMPSO, dois algoritmos com rankings da literatura, AR e MR, ambos executados através do algoritmo AR-SMPSO e as duas novas abordagens propostas neste artigo, BR e duas combinações dos rankings. Para combinação dos rankings buscou-se combinar o ranking BR com o método AR e com MR, visando combinar estratégias que gerem soluções próximas à ideal e que privilegiam soluções extremas. Em resumo, os métodos utilizados são: DP, AR, MR, BR, AR com BR (AR_BR) e BR com MR (BR_MR). Neste trabalho só serão considerados técnicas para muitos objetivos baseadas em ranking, trabalhos futuros irão abordar a comparação das novas estratégias com diferentes abordagens da literatura.

Para a medição de aspectos como a convergência e diversidade em relação à fronteira de Pareto real de cada problema foi utilizado um conjunto de indicadores de qualidade. O primeiro é *General Distance* (GD) que mede a distância entre o conjunto aproximação gerado por cada algoritmo (chamado de PF_{approx}) e a fronteira de Pareto real, PF_{true} . Com o GD podemos observar se a busca do algoritmo converge para alguma região da fronteira real. O segundo indicador é o *Inverted General Distance* (IGD), que mede a distância mínima de cada ponto da fronteira de Pareto real em relação ao conjunto de aproximação gerado. Com o IGD podemos observar se o PF_{approx} converge para a fronteira real e também observar se ele é diversificado e se espalha por toda fronteira, isto é, mede convergência e diversidade. O outro indicador utilizado é o *Spacing*. Essa medida indica o intervalo de variação entre as soluções do PF_{approx} . Não será utilizado o indicador Hipervolume [16] que, apesar de ser amplamente utilizado em trabalhos multiobjetivo na literatura, apresenta algumas limitações. Primeiro, o cálculo do hipervolume é caro computacionalmente e cresce exponencialmente em relação ao número de objetivos. Como estamos interessados em utilizar um número grande de objetivos o seu uso torna-se inviável. Além disso, segundo [14], esse indicador favorece conjuntos da aproximação mais próximos aos extremos, ao invés de soluções mais concentradas próximas à solução ideal [5].

Nesta análise, cada algoritmo foi executado 30 vezes. Para medir se existem diferenças entre os algoritmos para cada métrica utilizada foi utilizado o teste estatístico de Friedman [17], com um nível de significância de 5%. O teste de Friedman é um teste estatístico não-paramétrico usado para detectar diferenças entre um conjunto de algoritmos. Para cada problema e número de objetivos é apresentado qual algoritmo foi o melhor, segundo o teste de Friedman. A Tabela 2 apresenta um resumo com as melhores técnicas encontradas pelo teste de Friedman. Na tabela, se mais de uma técnica é escolhida para um cenário, isto representa que não há diferença estatística entre elas. Para auxiliar a interpretação do teste de Friedman, alguns gráficos *boxplot* são apresentados para identificar quais configurações apresentaram os melhores resultados. Devido ao limite de espaço do artigo os resultados estatísticos são omitidos e só são apresentados os *boxplots* para as medidas GD e IGD.

A Tabela 2 e as Figuras 2 e 4 apresentam um resumo das comparações efetuadas. As Figuras mostram para cada método o valor médio de cada indicador para cada objetivo analisado. Nas Figuras 2 e 4, cada curva representa um algoritmo e cada ponto representa o valor do indicador para cada objetivo.

O primeiro problema analisado é o DTLZ2. Analisando o GD, pode-se perceber na Tabela 2 que a combinação de rankings obteve o melhor resultado para quase todos os números de objetivos. Tanto a combinação AR_BR quanto a BR_MR obtiveram bons resultados, sendo que a primeira foi melhor para um número maior de objetivos. Outro método que destaca-se é o MR,

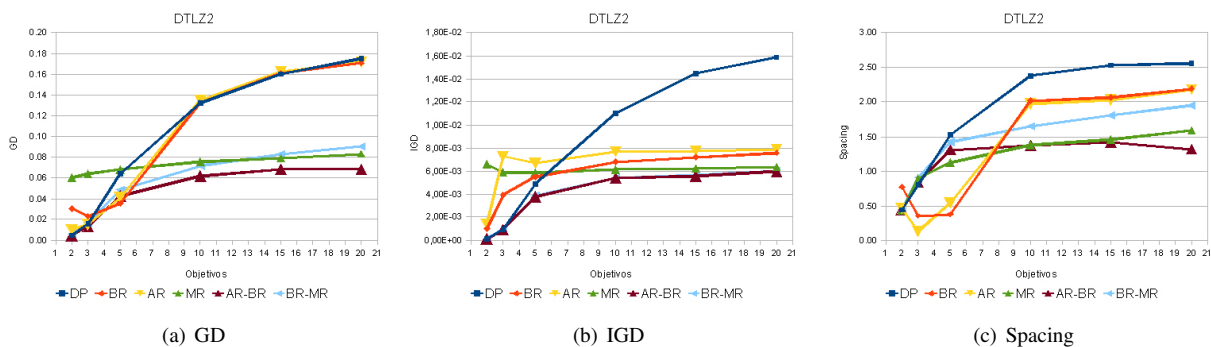


Figura 2: Média das medidas GD, IGD e Spacing para o problema DTLZ2

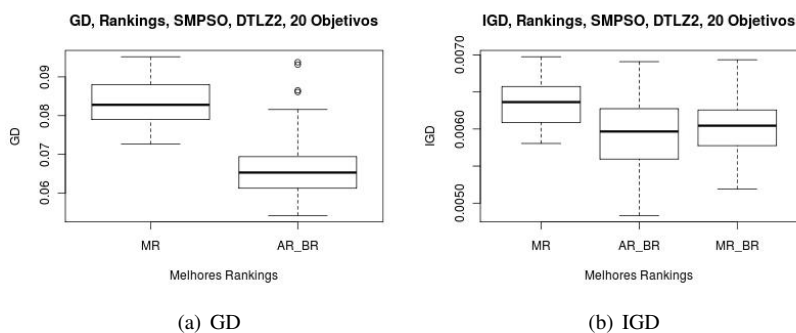


Figura 3: Boxplot com os melhores rankings para o GD e IGD. Problema DTLZ2, 20 objetivos

que obteve bons resultados para muitos objetivos, 15 e 20. O boxplot da Figura 3(a) mostra que para 20 objetivos essas técnicas tiveram resultados equivalentes. O BR não obteve bons resultados para muitos objetivos, somente foi melhor para 5 objetivos. O baixo GD no problema DTLZ2 indica que o método proposto tem dificuldade em convergir para a fronteira de Pareto real quando o número de objetivos cresce. O algoritmo PSO com a dominância de Pareto (DP) consegue obter os melhores resultados para 2 e 3 objetivos, mas não consegue obter bons resultados quando o número de objetivos cresce. A Figura 2(a) mostra um resumo desses comportamentos. As curvas que representam os métodos de combinação de rankings e do MR tem um crescimento pouco acentuado quando o número de objetivos aumenta, obtendo bons valores de GD para todos os objetivos. Para os demais métodos há uma inclinação acentuada quando o número de objetivos cresce, ou seja, os algoritmos tem a busca deteriorada quando o número de objetivos cresce e não conseguem convergir para a fronteira de Pareto real.

Para o IGD, o resultado é semelhante. A combinação de rankings (AR_BR e BR_MR) e o MR obtiveram os melhores resultados quando o número de objetivos cresce (ver boxplot na Figura 3(b)). Novamente o BR não conseguiu bons resultados e não atingiu o melhor IGD em nenhum objetivo. O DP apresentou o mesmo comportamento da análise do GD, bons valores somente com poucos objetivos. Observando a Figura 2(b), nota-se o comportamento discutido anteriormente. Um crescimento pouco acentuado para os métodos de AR_BR, BR_MR e MR e uma alta deterioração para o DP. Apesar de os métodos AR e BR não apresentarem os melhores resultados, o IGD desses métodos não deteriora muito quando o número de objetivos cresceu. Assim, através do IGD, pode-se concluir que os métodos da combinação de rankings e o MR, além de terem uma boa convergência para a fronteira de Pareto real, conseguem produzir um conjunto de aproximação diversificado em torno dessa fronteira, quando o número de objetivos cresce. Para o método BR, não há uma deterioração da convergência e da diversidade quando o número de objetivos cresce, porém em todos os casos o algoritmo tem dificuldade para convergir. A DP ainda consegue convergir para a fronteira de Pareto real quando o número de objetivos é baixo, porém quando esse número cresce os resultados apresentados são ruins.

Na análise do *Spacing*, os resultados também são semelhantes. A combinação de rankings e o MR obtiveram os melhores resultados para muitos objetivos. Quando o número de objetivos é baixo, quase todos os métodos são competitivos, em especial os métodos AR e BR. A Figura 2(c) mostra um resumo deste comportamento. A partir de 10 objetivos, os métodos que possuíam bons valores de *Spacing* apresentam uma deterioração, enquanto os métodos MR e AR_BR apresentam valores constantes de *spacing* para todos os objetivos.

O segundo problema, DTLZ4, é utilizado para observar a habilidade dos algoritmos em manter uma boa distribuição das soluções. Neste problema as medidas IGD e *Spacing* são mais importantes, pois medem diversidade. Para o GD os métodos AR, BR e a combinação AR_BR obtiveram os melhores resultados. O *boxplot* da Figura 5(a) mostra o resultado equivalente dessas técnicas para 15 objetivos. Diferentemente do DTLZ2, neste problema os métodos AR e BR não tiveram sua convergência deteriorada quando número de objetivos cresceu. A Figura 4(a) mostra o resumo da evolução do GD quando o número de objetivos cresce. Novamente nota-se um bom resultado da combinação de rankings através do método AR_BR, que mantém um

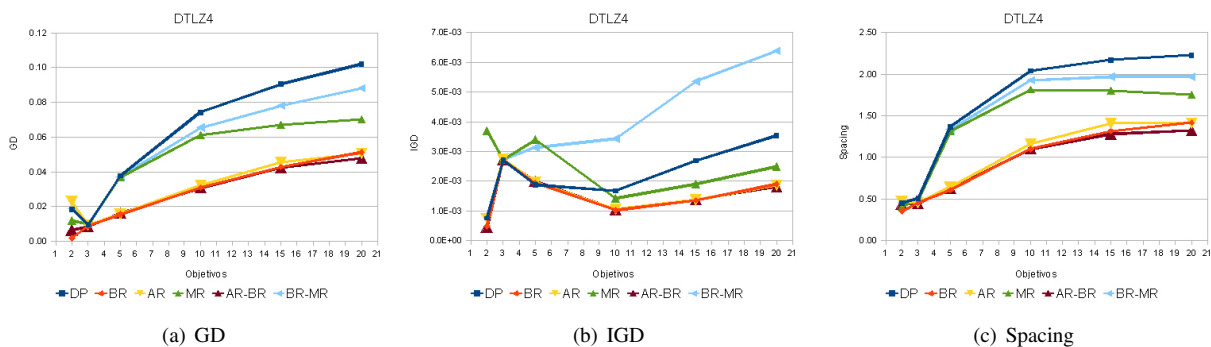


Figura 4: Média das medidas GD, IGD e Spacing para o problema DTLZ4

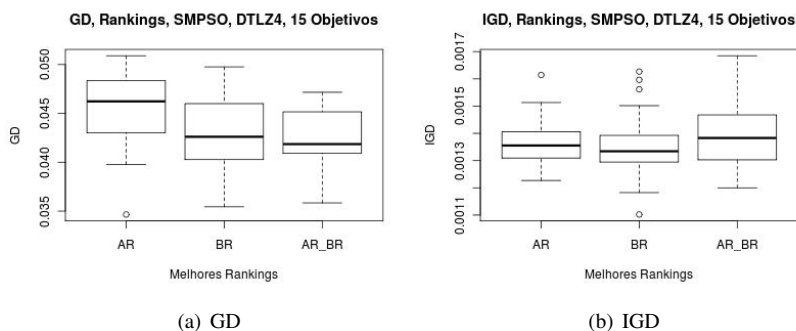


Figura 5: Boxplot com os melhores rankings para o GD e IGD. Problema DTLZ2, 15 objetivos

aumento suave quando o número de objetivos cresce. Os métodos AR e BR apresentam um comportamento semelhante. Para o DP, novamente, a uma deterioração grande na convergência quando o número de objetivos é alto.

O IGD apresenta um resultado bastante semelhante ao GD para o DTLZ4. Os melhores resultados para muitos objetivos foram obtidos pelos métodos AR, BR e a combinação AR_BR (ver boxplot na Figura 5(b)). Observando a Figura 4(b), esses métodos apresentam valores semelhantes de IGD para quase todos os números de objetivos, ou seja, conseguiram controlar a diversificação das soluções mesmo trabalhando com muitos objetivos. É interessante notar, que o método BR_MR obteve os piores resultados, sendo pior que a execução dos métodos em separado e até pior que a relação da dominância de Pareto.

Para os *Spacing* os resultados são semelhantes aos demais indicadores: AR, BR e AR_BR com os melhores valores para muitos objetivos. Na Figura 4, pode-se observar que as curvas que representam estes algoritmos estão mais próximas de zero e que quando o número de objetivos cresce, os valores de *Spacing* não pioram muito. Para os demais métodos é possível perceber que há uma maior deterioração da diversificação quando o número de objetivos cresce, em especial para a dominância de Pareto.

Em resumo, a análise empírica mostrou que as novas estratégias propostas obtiveram bons resultados, melhores que as técnicas AR e MR da literatura e que o uso da relação da dominância de Pareto. A combinação de rankings foi a melhor técnica dentre todas as analisadas, obtendo uma boa convergência em direção à fronteira de Pareto real e uma boa diversificação nas soluções geradas, tanto em torno da fronteira real quanto em relação a distribuição das soluções. O método BR não conseguiu bons resultados para o problema DTLZ2 e apresentou problemas para convergir para a fronteira real quando o número de objetivos cresceu, porém obteve bons resultados para o DTLZ4 e mostrou que gera um conjunto de aproximação com boa diversificação.

6. CONCLUSÃO

Este trabalho apresentou a proposta de duas novas estratégias para problemas com muitos objetivos, especialmente aplicadas a um algoritmo PSO. Foi proposto um novo ranking chamado de *Balanced Ranking* que visa evitar a criação de soluções com valores extremos em um dos objetivos, sacrificando os demais. Além disso, foi proposto uma nova estratégia que visa combinar diferentes rankings, com o objetivo de reduzir o número de total de objetivos do problema para o número de rankings escolhido e juntar as características de cada ranking.

Para a validação dos métodos propostos foi feita uma análise empírica para investigar aspectos como convergência e diversidade na execução dos métodos e compará-los com alguns métodos de ranking da literatura. Os experimentos foram conduzidos utilizando dois diferentes problemas com muitos objetivos, DTLZ2 e DTLZ4 e com diferentes números de objetivos: 2, 3, 5, 10, 15 e 20. A convergência e a diversidade foram analisadas através de indicadores de qualidade como os GD, IGD e *Spacing*.

Através dos resultados conclui-se que as novas abordagens proposta apresentam bons resultados. A combinação de rankings foi a técnica que obteve os melhores resultados, apresentando a melhor convergência em direção à fronteira de Pareto real e gerando um conjunto de aproximação mais diversificado. O BR também apresentou bons resultados, em especial obteve uma boa diversidade das soluções, porém não apresentou boa convergência quando o número de objetivos era grande.

Trabalhos futuros incluem a aplicação destas novas estratégias em diferentes metaheurísticas, a comparação com diferentes métodos para muitos objetivos e o desenvolvimento de novas estratégias para muitos objetivos.

REFERÊNCIAS

- [1] H. Ishibuchi, N. Tsukamoto and Y. Nojima. “Evolutionary many-objective optimization: A short review”. In *CEC 2008. IEEE Congress on Evolutionary Computation*, pp. 2419–2426, 2008.
- [2] M. Reyes-Sierra and C. A. C. Coello. “Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art”. *International Journal of Computational Intelligence Research*, vol. 2, no. 3, pp. 287–308, 2006.
- [3] A. B. Carvalho and A. Pozo. “Using Different Many-Objective Techniques in Particle Swarm Optimization for Many Objective Problems: An Empirical Study”. *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 3, pp. 096–107, 2011.
- [4] P. J. Bentley and J. P. Wakefield. “Finding acceptable solutions in the pareto-optimal range using multiobjective genetic algorithms”. In *Soft Computing in Engineering Design and Manufacturing*, edited by P. K. Chawdhry, R. Roy and R. K. Pant, pp. 231–240. Springer-Verlag, 1998.
- [5] J. Cochrane and M. Zeleny. *Multiple Criteria Decision Making*. University of South Carolina Press, Columbia, 1973.
- [6] V. Pareto. *Manuel D’Economie Politique*. Ams Pr, 1927.
- [7] C. A. C. Coello, G. B. Lamont and D. A. V. Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems (Genetic and Evolutionary Computation)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.
- [8] H. Sato, H. E. Aguirre and K. Tanaka. *Controlling Dominance Area of Solutions and Its Impact on the Performance of MOEAs*, pp. 5–20. Lecture Notes in Computer Science 4403: Evolutionary Multi-Criterion Optimization. Springer, Berlin, 2007.
- [9] P. Lindroth, M. Patriksson and A.-B. Strömberg. “Approximating the Pareto optimal set using a reduced set of objective functions”. *European Journal of Operational Research*, vol. 207, no. 3, pp. 1519–1534, 2010.
- [10] Q. Zhang and H. Li. “MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition”. *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.
- [11] A. B. Carvalho and A. Pozo. “The Control of Dominance Area in Particle Swarm Optimization Algorithms for Many-objective Problems”. In *2010 Eleventh Brazilian Symposium on Neural Networks*, pp. 140–145. IEEE Computer Society, 2010.
- [12] A. Nebro, J. Durillo, J. Garcia-Nieto, C. A. C. Coello, F. Luna and E. Alba. “SMPSO: A new PSO-based metaheuristic for multi-objective optimization”. In *IEEE symposium on Computational intelligence in multi-criteria decision-making, 2009. mcdm '09*, pp. 66–73, 2009.
- [13] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan. “A fast and elitist multiobjective genetic algorithm: NSGA-II”. *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, August 2002.
- [14] A. L. Jaimes and C. A. C. Coello. “Study of preference relations in many-objective optimization”. *Proceedings of the 11th Annual conference on Genetic and evolutionary computation - GECCO '09*, pp. 611–618, 2009.
- [15] K. Deb, L. Thiele, M. Laumanns and E. Zitzler. “Scalable Multi-Objective Optimization Test Problems”. In *Congress on Evolutionary Computation (CEC 2002)*, pp. 825–830, 2002.
- [16] D. A. V. Veldhuizen and G. Lamont. “On Measuring Multiobjective Evolutionary Algorithm Performance”. In *In 2000 Congress on Evolutionary Computation*, pp. 204–211. press, 2000.
- [17] J. Demšar. “Statistical Comparisons of Classifiers over Multiple Data Sets”. *The Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.