
Capítulo 2

Optimização Evolucionária Multi-Objectivo

1. Optimização mono-objectivo

Um problema de optimização mono-objectivo é composto por uma função objectivo (a minimizar ou a maximizar) e, normalmente, por várias restrições que todas as soluções admissíveis têm de satisfazer. De uma maneira geral, um problema de optimização mono-objectivo pode ser formulado da seguinte forma:

$$\begin{array}{ll} \text{Minimizar } f(x), & \\ \text{sujeito a } g_j(x) \geq 0, & j = 1, 2, \dots, J \end{array} \quad (1)$$

$$h_k(x) = 0, \quad k = 1, 2, \dots, K \quad (2)$$

$$x_i^{inf} \leq x_i \leq x_i^{sup}, \quad i = 1, 2, \dots, n. \quad (3)$$

O problema tem associado $J+K$ restrições funcionais, das quais J são de desigualdade, $g_j(x)$, e K de igualdade, $h_k(x)$. As restrições de desigualdade podem ser do tipo " \geq " ou " \leq ", apesar da forma geral apresentar apenas restrições do tipo " \geq ", o que não implica perda de generalidade.

Uma solução x é um vector de n variáveis de decisão: $x = (x_1, \dots, x_n)$. As restrições do último conjunto denominam-se por *limites das variáveis de decisão*, as quais obrigam a que cada variável de decisão x_i assuma um valor entre um limite inferior, x_i^{inf} , e um limite superior, x_i^{sup} . Estes limites constituem o *espaço das variáveis de decisão*, *espaço das decisões* ou *espaço das soluções*.

Uma solução x que satisfaça todas as $J+K$ restrições funcionais e todos os $2n$ limites das variáveis de decisão denomina-se por *solução admissível*. Ao conjunto de todas as soluções admissíveis dá-se o nome de *região admissível*, X .

Se a função objectivo e todas as restrições de um problema são funções lineares em relação a x , então este problema denomina-se de *linear*. Pelo contrário, se a função objectivo ou alguma das restrições são não lineares em relação a x , então este problema denomina-se de *não linear*.

Num problema de optimização contínua, a pesquisa da solução óptima é feita num conjunto infinito de pontos (o espaço de pesquisa é infinito). Neste tipo de problemas, as variáveis de decisão são normalmente reais.

Quando a pesquisa da solução óptima é feita num conjunto finito de pontos (o espaço de pesquisa é finito), então diz-se que se está perante um problema de optimização discreta ou combinatória.

Na resolução de um problema de optimização mono-objectivo, pretende-se determinar a *solução óptima*, ou seja, a solução admissível que optimize a função objectivo, cujo valor é único, mesmo que existam soluções óptimas alternativas.

Os dados do problema são todos os coeficientes e parâmetros associados à função objectivo e às restrições funcionais. Se todos estes dados são valores únicos, então o problema designa-se de *determinístico*. Se isto não acontece, então existe alguma incerteza a envolver alguns (ou todos) estes dados e o problema designa-se de *não determinístico*.

2. Optimização multi-objectivo

Um problema de optimização é do tipo multi-objectivo se existem duas ou mais funções objectivo associadas ao problema. Neste tipo de problemas, o conceito de solução óptima, característico de problemas de optimização mono-objectivo, não se pode aplicar, uma vez que uma solução admissível que optimize um dos objectivos, não otimiza, em geral, os restantes objectivos, quando estes estão em conflito.

2.1. Formulação de um problema de optimização multi-objectivo

Um problema de optimização multi-objectivo é composto por duas ou mais funções objectivo, que se pretendem maximizar ou minimizar e, normalmente, por várias restrições funcionais que todas as soluções admissíveis têm de satisfazer.

De uma maneira geral, um problema de optimização multi-objectivo pode ser formulado da seguinte forma:

$$\begin{aligned} & \text{Minimizar } f_m(x), \quad m = 1, 2, \dots, M \\ & \text{sujeito a } x \in X \quad ((1) - (3)). \end{aligned}$$

Cada uma das M funções objectivo, $f(x) = (f_1(x), f_2(x), \dots, f_M(x))^T$, com $M \geq 2$, pode ser a minimizar ou a maximizar. Como foi referido anteriormente, esta formulação não implica perda de generalidade.

Uma das diferenças fundamentais entre optimização mono-objectivo e optimização multi-objectivo é que, ao espaço das variáveis de decisão há que acrescentar um outro espaço multi-dimensional gerado pelas funções objectivo, denominado de *espaço das funções objectivo*, ou simplesmente, *espaço dos objectivos*. Para cada solução $x = (x_1, \dots, x_n)$ no espaço das variáveis de decisão, existe um ponto que lhe corresponde no espaço dos objectivos, denotado por $f(x) = z = (z_1, \dots, z_M)^T$, com $z_m = f_m(x)$ e $m = 1, \dots, M$.

A região admissível no espaço das funções objectivo (o conjunto de todas as imagens dos pontos em X), pode então ser definida da seguinte forma:

$$Z = \{ z \in \mathbb{R}^M : z = (f_1(x), f_2(x), \dots, f_M(x)), x \in X \}.$$

2.2. Princípios de optimização multi-objectivo

Na resolução de problemas multi-objectivo, pretende-se encontrar a "*melhor*" *solução de compromisso* para o AD, que possa constituir uma *solução final* do problema de decisão, ou então, fazer a caracterização, exhaustiva ou não, da região não dominada. Desta forma, a noção de solução óptima usada em problemas de optimização mono-objectivo é substituída pela noção de *solução não dominada* (também designada por *ótima de Pareto*, *eficiente* ou *não inferior*).

Uma solução não dominada é uma solução admissível para a qual não é possível melhorar simultaneamente todas as funções objectivo; isto é, a melhoria numa função objectivo apenas pode ser alcançada por degradação de pelo menos uma das outras. Por outro lado, uma solução admissível diz-se dominada por outra, se ao passar-se da primeira para a segunda existir melhoria de pelo menos uma função objectivo, permanecendo inalterados as restantes (Clímaco et al. (2003)).

No conjunto das soluções não dominadas, também denominado por frente óptima de Pareto ou apenas frente de Pareto, têm de ser satisfeitas as duas condições seguintes:

- 1) quaisquer duas soluções deste conjunto têm de ser não dominadas entre si;
- 2) qualquer solução que não pertença a este conjunto é dominada por, pelo menos, uma solução deste conjunto.

Quando as funções objectivo de um problema de optimização multi-objectivo estão em conflito, normalmente o conjunto de soluções não dominadas é enorme. Desta forma, é difícil escolher uma solução deste conjunto, sem dispor de informação adicional relativa ao problema a resolver. No entanto, na ausência de tal informação, as soluções não dominadas não são comparáveis entre si, o que motiva as abordagens a determinar o número máximo possível de soluções não dominadas do problema. Assim, pode-se caracterizar o problema de optimização multi-objectivo como a determinação de um conjunto de soluções não dominadas com as seguintes características:

- 1) esteja o mais próximo possível da frente óptima de Pareto real,
- 2) seja o mais diversificado possível.

A primeira característica é inerente a qualquer tarefa de optimização; também em optimização mono-objectivo se pretende determinar uma solução admissível que garanta o valor óptimo para o modelo matemático.

A segunda característica é específica da optimização multi-objectivo. Apenas com um conjunto de soluções dispersas é possível garantir a existência de um bom conjunto de soluções de compromisso entre os objectivos. Uma vez que a optimização multi-objectivo actua em dois espaços, das variáveis de decisão e dos objectivos, a diversidade das soluções pode ser definida nestes dois espaços. Apesar de, na maioria dos problemas, a diversidade num dos espaços significar, normalmente, a diversidade no outro espaço, isto pode não acontecer em alguns problemas.

2.3. Diferenças com a optimização mono-objectivo

Existem algumas diferenças fundamentais entre a optimização multi-objectivo e a optimização mono-objectivo, as quais estão relacionadas com as metas a atingir, com os números de espaços de pesquisa onde se actua e pela necessidade de considerar funções escalares substitutas nalguns processos de cálculo (ver também Deb (2001)).

Em optimização mono-objectivo existe apenas uma meta a atingir: determinar uma solução óptima global (apesar de no espaço de pesquisa poderem existir soluções óptimas locais, é sempre uma solução óptima global que se pretende). No entanto, no caso da optimização multi-modal pretende-se determinar vários óptimos locais e globais.

Em optimização multi-objectivo pretendem-se atingir duas metas: fazer convergir o conjunto de soluções não dominadas obtido em direcção à frente óptima de Pareto real e manter um conjunto diversificado de soluções na frente obtida. Utilizando diferentes formas de compensação dos objectivos pode obter-se, em princípio, uma maior variedade de soluções não dominadas, obtendo-se assim um conjunto diversificado de soluções próximo da frente óptima de Pareto real.

Em optimização mono-objectivo existe apenas um espaço de pesquisa multi-dimensional, o espaço das variáveis de decisão, onde o algoritmo actua, aceitando e rejeitando soluções de acordo com os respectivos valores da função objectivo.

Em optimização multi-objectivo, para além do espaço das variáveis de decisão, é necessário considerar também o espaço dos objectivos, os quais, embora estejam relacionados, muitas vezes as suas propriedades não são semelhantes. Por exemplo, o nível de proximidade existente entre duas soluções pode ser muito diferente nos dois espaços. Assim, para garantir a diversidade no conjunto de soluções obtidas é importante decidir em que espaço deve ser alcançada a diversidade.

Apesar da pesquisa de soluções, por parte de qualquer algoritmo de optimização, ser efectuada no espaço das variáveis de decisão, o seu comportamento neste espaço pode ser mapeado no espaço dos objectivos. Nalguns algoritmos, o comportamento definido no espaço dos objectivos é usado para guiar a pesquisa no espaço das variáveis de decisão. Neste caso, o comportamento do algoritmo deve ser coordenado em ambos os espaços, de tal modo que a geração de novas soluções no espaço das variáveis de decisão possa contribuir para a necessária diversidade no espaço dos objectivos.

A resolução de problemas multi-objectivo envolve, geralmente, a utilização de funções escalares substitutas que agregam (temporariamente) as múltiplas funções objectivo numa única dimensão, de tal modo que a solução óptima da função escalar é não dominada do problema multi-objectivo original (Clímaco et al. (2003)). As funções escalares substitutas mais usadas consistem em somas pesadas, selecção de uma das funções considerando as outras como restrições e distância a um ponto de referência.

No caso da soma pesada, é criada uma única função objectivo composta, resultante da soma ponderada dos vários objectivos. O segundo caso consiste em escolher uma das funções objectivo para optimização e transformar as restantes em restrições do problema, limitando cada uma destas restrições a certos limiares predefinidos. As funções escalares baseadas na distância a um ponto de referência permitem obter soluções não dominadas minimizando a distância, segundo uma determinada métrica, da região admissível a um qualquer ponto de referência do espaço dos objectivos.

2.4. Dominância e optimalidade de Pareto

A maioria dos algoritmos de optimização multi-objectivo usa, no seu processo de pesquisa, o conceito de dominância. Nesta secção, vão ser apresentados os conceitos de dominância e de alguns termos relacionados, assim como algumas técnicas para identificar soluções não dominadas numa população de soluções de tamanho finito.

2.4.1. Pontos especiais

Os algoritmos de optimização multi-objectivo usam, muitas vezes, conceitos associados a alguns pontos ("soluções") especiais no espaço dos objectivos: ponto ideal e ponto nadir.

O ponto definido no espaço dos objectivos cujas componentes são o valor óptimo de cada função objectivo na região admissível, quando optimizadas separadamente, designa-se por *ponto ideal*, z^* . Geralmente, o ponto ideal não corresponde a uma solução admissível, uma vez que a solução óptima para cada função objectivo não é a mesma (os M objectivos estão, em geral, em conflito). No entanto, o ponto ideal tem grande utilidade, pois é usado em muitos algoritmos de optimização multi-objectivo como solução de referência.

O *ponto nadir*, z^{nad} , representa os piores valores de cada função objectivo no conjunto óptimo de Pareto. No entanto, dada a dificuldade computacional associada ao cálculo deste ponto, muitas vezes é usado como substituto de z^{nad} o ponto definido pelos piores valores de cada função objectivo entre todas as soluções que optimizam individualmente cada função (a chamada tabela de "pay-off").

2.4.2. Conceitos de dominância e de optimalidade de Pareto

A maioria dos algoritmos de optimização multi-objectivo usa o conceito de dominância, em particular quando há necessidade de comparar duas soluções, para averiguar se há dominância de uma sobre a outra.

Uma solução $x^1 \in X$ *domina* uma outra solução $x^2 \in X$, se e só se x^1 não é pior do que x^2 para todos os objectivos e x^1 é estritamente melhor do que x^2 para pelo menos um dos M objectivos.

Apesar de serem diferentes os conceitos de solução eficiente e de solução não dominada, quando utilizados de forma genérica nesta dissertação não é feita qualquer distinção entre eles. Enquanto que o conceito de solução eficiência se refere, geralmente, a pontos do espaço de decisão, o conceito de solução não dominada é utilizada para pontos do espaço dos objectivos; isto é, uma solução não dominada é a imagem de uma solução eficiente.

Matematicamente, e considerando todas as funções objectivo a minimizar, tem-se (Clímaco et al. (2003)):

- a) Uma solução $x^1 \in X$ é *eficiente* se e só se não existe uma outra solução $x^2 \in X$ tal que $z_m(x^2) \leq z_m(x^1)$ para todo o m ($m = 1, 2, \dots, M$) e $z_m(x^2) < z_m(x^1)$ para pelo menos um m .

b) Um ponto do espaço dos objectivos $z^1 \in Z$, com $z^1 = (z_1(x^1), \dots, z_M(x^1))$ e $x^1 \in X$, diz-se *não dominado* se e só se x^1 é uma solução eficiente.

Uma solução de compromisso satisfatória para o problema multi-objectivo deverá ser não dominada, cujos valores das funções objectivo são satisfatórios para o AD e de tal modo que seja aceitável como solução final. Desta forma, é apenas sobre o conjunto das soluções não dominadas que deve recair a atenção do analista⁵ e do AD.

Entre quaisquer duas soluções não dominadas a uma melhoria em pelo menos um dos objectivos encontra-se sempre associado um sacrifício em pelo menos um dos outros objectivos. Isto é, verifica-se sempre uma compensação entre objectivos no conjunto das soluções não dominadas.

A relação de dominância entre duas soluções, tal como foi definida atrás, é muitas vezes referida como uma relação de dominância *fraca*: x^1 *domina fracamente* uma solução x^2 , se x^1 não é pior do que x^2 em todos os M objectivos e é estritamente melhor do que x^2 em pelo menos um dos M objectivos. A partir da definição de *dominância fraca*, pode-se obter a definição de *dominância estrita (forte)*, da forma que se segue.

Uma solução x^1 *domina estritamente (fortemente)* uma solução x^2 , se a solução x^1 é estritamente melhor do que a solução x^2 em todos os M objectivos. Desta forma, se uma solução x^1 *domina estritamente (fortemente)* uma solução x^2 , a solução x^1 também *domina fracamente* a solução x^2 , mas o recíproco não é verdadeiro.

Tal como existem soluções óptimas globais e locais em optimização mono-objectivo, também poderão existir *frentes óptimas de Pareto globais e locais* em optimização multi-objectivo.

Ao conjunto de soluções não dominadas de toda a região admissível dá-se o nome de *frente óptima de Pareto global*. A frente óptima de Pareto global é referida simplesmente como *frente óptima de Pareto* ou *frente de Pareto*. Uma vez que as soluções deste conjunto são não dominadas em relação a qualquer solução da região admissível, elas são as melhores soluções do problema de optimização multi-objectivo.

Define-se, também, frente óptima de Pareto local da seguinte forma (Deb (1999) e Miettinen (1999)): se para cada solução x de um conjunto P'' não existe nenhuma solução y (na vizinhança de x , tal que $\|y-x\|_\infty \leq \varepsilon$, em que ε é um número positivo muito pequeno), que domine qualquer solução de P'' , então as soluções que pertencem ao conjunto P'' constituem um conjunto óptimo de Pareto local.

⁵ Entidade responsável pela análise ou estudo do problema, com conhecimentos técnicos sobre programação multi-objectivo, que serve de mediador entre o AD e as componentes metodológica e computacional do processo de apoio à decisão (Clímaco et al. (2003)).

De acordo com esta definição, uma frente óptima de Pareto global é também uma frente óptima de Pareto local.

A definição de dominância forte pode ser usada para definir *frente fracamente não dominada*. Dado um conjunto de soluções, P , a frente fracamente não dominada, P' , é formada pelas soluções que não são fortemente dominadas relativamente a qualquer outra solução do conjunto P (Deb (2001)).

As definições de frente fracamente não dominada global e local podem também ser feitas de forma semelhante, a partir da definição de frente fracamente não dominada.

2.4.3. Determinação de uma frente não dominada

Existem vários métodos para determinar uma frente não dominada, a partir de uma dada população de soluções. A diferença entre estes métodos reside essencialmente na complexidade computacional de cada um deles.

Um método consiste em comparar cada solução x de uma população com todas as outras soluções desta população, para verificar se x é dominada por alguma delas.

Um outro método consiste em comparar cada solução da população com as soluções de uma outra população parcialmente ocupada com soluções não dominadas entre si. O processo começa por inserir a primeira solução de P num conjunto vazio P' . Depois, compara-se cada solução x de P com todas as soluções do conjunto P' , uma a uma. Se a solução x domina alguma solução de P' , então todas as soluções de P' que são dominadas por x são removidas de P' e x é inserida em P' . Pelo contrário, se a solução x é dominada por uma solução de P' , então a solução x é ignorada. Quando todas as soluções da população forem comparadas, as soluções do conjunto P' formam a frente não dominada.

2.4.4. Determinação de todas as frentes não dominadas

A maioria dos algoritmos de optimização multi-objectivo permite determinar apenas a frente não dominada de uma população. Estes algoritmos classificam a população de soluções apenas em dois conjuntos (frentes): o primeiro constituído pelas soluções não dominadas e o segundo pelas dominadas. Neste caso, todas as soluções dominadas são consideradas em pé de igualdade.

No entanto, existem outros algoritmos que classificam a população por níveis de dominância. As soluções da população são ordenadas crescentemente por níveis de dominância: no primeiro nível (menor índice) estão as melhores soluções e no último nível (maior índice) estão as piores soluções, em termos de dominância. Portanto, qualquer solução de um certo nível de dominância, é dominada por pelo menos uma

solução de cada nível mais baixo (Figura 2.1). Neste caso, as soluções mais importantes são as não dominadas (nível 1 na Figura 2.1), mas entre as dominadas podem existir vários níveis de dominância (níveis 2, 3, 4 e 5 na Figura 2.1). As soluções classificadas no nível 2 podem ser consideradas mais interessantes do que as do nível 3, e assim sucessivamente, até às do nível 5.

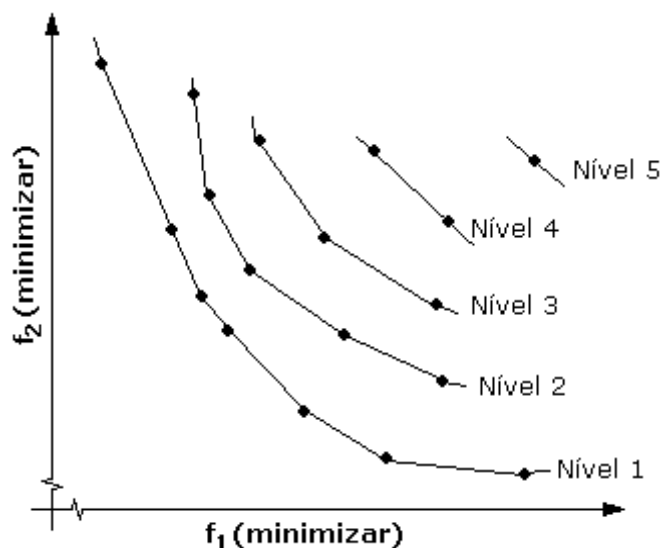


Figura 2.1: Classificação das soluções por níveis de dominância.

A diferença fundamental entre os vários algoritmos existentes para determinar as várias frentes não dominadas de uma população está na complexidade de cada um deles, quer em termos do espaço de armazenamento dos dados, quer em termos da velocidade de execução.

Um dos processos utilizados para determinar as várias frentes associadas aos vários níveis de dominância de uma população consiste em determinar primeiro a frente de nível 1 (composta pelas soluções não dominadas da população) e, depois, as restantes frentes para os restantes níveis (compostas pelas soluções dominadas da população).

Para determinar as soluções da frente de nível 1, basta tomar as soluções não dominadas da população. Para determinar as soluções de cada frente de nível $k > 1$, é necessário efectuar os dois passos seguintes:

- 1º) ignorar temporariamente todas as soluções das frentes dos níveis anteriores,
- 2º) determinar as soluções não dominadas entre as restantes soluções da população, as quais constituem a frente não dominada de nível k .

Estes dois passos repetem-se até todas as soluções da população estarem inseridas numa frente não dominada.

Um outro processo consiste em determinar, inicialmente, para cada solução x^i um contador n_i com o número de soluções que dominam x^i e um conjunto P_i com as soluções que são dominadas por x^i . No final desta fase do processo, todas as soluções que tenham os respectivos contadores a zero são aquelas que pertencem à frente de nível 1 (constituída pelas soluções não dominadas da população). Depois, para cada uma destas soluções x^i , com $n_i = 0$, analisa-se o respectivo conjunto P_i de tal forma que, para cada uma das soluções x^j deste conjunto sejam reduzidos os respectivos contadores n_j em uma unidade e transferidas as soluções cujos contadores se tornaram nulos para um conjunto P' . Desta forma, as soluções que estão em P' têm todos os respectivos contadores a zero, o que significa que estas soluções constituem a frente de nível 2. Este processo é repetido até todas as soluções da população estarem classificadas nos vários níveis de dominância (e distribuídas pelas diversas frentes).

Se na população também existirem soluções não admissíveis, então estas devem pertencer a um terceiro grupo de frentes. Desta forma, são determinados três grupos de frentes: o primeiro composto apenas por uma frente (com as soluções não dominadas da população), um segundo grupo que pode conter várias frentes (com soluções admissíveis dominadas), e um terceiro grupo que pode conter também várias frentes (com soluções não admissíveis).

3. Métodos para resolver problemas de optimização MO

Nos últimos anos tem sido dedicado um esforço considerável ao desenvolvimento de métodos para resolução de problemas de optimização multi-objectivo, tendo em atenção o contributo que o AD pode fornecer na procura das soluções não dominadas do problema, através da expressão do seu sistema de preferências. Estes métodos podem ser divididos em dois grupos: métodos clássicos e meta-heurísticos (onde se incluem os algoritmos evolucionários).

3.1. Métodos clássicos

Ao longo dos anos, alguns autores têm procurado classificar os métodos dedicados a problemas de optimização multi-objectivo de acordo com vários aspectos, como, por exemplo (Clímaco et al. (2003)): grau de intervenção do AD, tipo de modelação das preferências do AD, número de ADs, certeza/incerteza na determinação dos dados do modelo, e dados do modelo requeridos e/ou resultados obtidos.

No entanto, a classificação dos métodos baseada no grau de intervenção do AD é a que tem sido mais usada na literatura sobre os problemas de optimização multi-

objectivo, a qual agrupa os métodos em três tipos (Clímaco et al. (2003)): *articulação a posteriori das preferências (geradores)*, *articulação a priori das preferências* e *articulação progressiva das preferências (interactivos)*. Uma classificação deste tipo é também considerada por Cohon (1978) e Steuer (1986).

Nos *métodos a posteriori* são calculadas algumas, ou mesmo todas (quando isto é possível), as soluções não dominadas do problema, que são depois colocadas à disposição do AD para serem avaliadas. Estes métodos exigem um elevado esforço computacional, pois utilizam um cálculo exaustivo para caracterizar o conjunto de soluções não dominadas e não têm em atenção as preferências do AD que poderiam servir de apoio na resolução do problema (identificação de uma solução de compromisso aceitável como solução final).

Nos *métodos a priori*, o AD começa por indicar as suas preferências, a partir das quais é possível transformar o problema multi-objectivo inicial num problema mono-objectivo, por exemplo, através da construção de uma função utilidade. Desta forma, apesar de existir modelação multi-objectivo do problema, pretende-se otimizar a função utilidade, cuja solução óptima será a solução final. A maior dificuldade deste processo está em obter os parâmetros para construir uma função utilidade que agregue, numa única dimensão, todos os objectivos em análise.

Nos *métodos progressivos*, o AD expressa as suas preferências através de um processo de diálogo com a componente procedimental, de forma a conduzir a pesquisa para a zona da região admissível onde se localizam as soluções que melhor correspondem ao seu sistema de preferências (isto é, existe uma articulação progressiva das preferências). Desta forma, estes métodos, ao aproveitarem a intervenção do AD, reduzem o âmbito da pesquisa, de forma a minimizar o esforço quer computacional, quer do próprio AD no processamento da informação.

Nos métodos de articulação *a posteriori* das preferências (métodos geradores) podem, por exemplo, usar-se processos de cálculo de soluções não dominadas baseados na optimização de funções escalares soma ponderada das múltiplas funções objectivo ou na escolha de uma das funções para optimizar considerando as restantes M-1 como restrições. A intenção é sempre fazer uma caracterização o mais exaustiva possível do conjunto de soluções não dominadas, sem requerer a intervenção do AD. Por exemplo, em problemas de programação linear multi-objectivo a intenção é geralmente calcular todos os vértices (soluções básicas) não dominados, a partir dos quais podem ser caracterizadas as faces, de dimensão superior, não dominadas.

Nos métodos de articulação *a priori* das preferências podem, por exemplo, ser construídas funções de utilidade que se destinem a representar analiticamente as preferências do AD, que devem ser válidas em todo o espaço de pesquisa admissível,

baseadas na minimização da distância (de acordo com uma dada métrica) à solução ideal (ou outro ponto de referência), em processos lexicográficos ou em programação por metas.

A principal característica dos métodos interactivos é o envolvimento do AD no decurso do processo de optimização, usando a expressão das suas preferências face às soluções que lhe vão sendo apresentadas para orientar a pesquisa, alternando assim fases de cálculo de soluções não dominadas e fases de diálogo para recolha de informação que é usada na fase de cálculo seguinte. O facto de o AD ser envolvido no processo de optimização, tornou estes métodos populares para a resolução de problemas práticos. Em Miettinen (1999) e Clímaco et al. (2003) encontra-se informação mais detalhada sobre estes métodos.

3.2. Meta-heurísticas

Devido à dificuldade computacional associada à resolução de alguns problemas de optimização, particularmente de natureza combinatória e em que existem múltiplos óptimos locais, usando os métodos clássicos de programação matemática, foram desenvolvidas técnicas meta-heurísticas para tentar atenuar esta dificuldade.

Reeves (1995) definiu *heurística*, para problemas mono-objectivo, como uma técnica que procura obter boas soluções (isto é, quase óptimas), com um custo computacional razoável, sem ser capaz de garantir, quer a admissibilidade, quer a optimalidade, e até mesmo, em muitos casos, determinar o quanto uma solução admissível obtida está próxima do óptimo.

Ao contrário dos métodos clássicos, as heurísticas são "menos ambiciosas", na medida em que vão calculando o valor de uma função objectivo em pontos julgados promissores, os quais vão sendo obtidos ao longo de um processo iterativo. Terminado o processo de optimização (para o caso mono-objectivo) deverá ter-se encontrado uma solução "quase óptima" (esta solução até poderá ser o óptimo global, mas não há certeza disso). A solução obtida, apesar de possivelmente não ser a óptima, define um limite superior (problemas de minimização) ou inferior (problemas de maximização) para o valor óptimo.

As heurísticas são técnicas em que, embora a exploração seja feita de forma algorítmica, o progresso é obtido pela avaliação puramente empírica do resultado. O uso de técnicas heurísticas é vantajoso em certas circunstâncias, tais como quando os dados do problema são escassos e incertos, quando não existem métodos exactos ou quando existindo requerem um elevado esforço computacional, ou quando o AD precisa de obter rapidamente um vasto conjunto de soluções para escolher uma delas.

A maioria das heurísticas surgiu para resolver tipos específicos de problemas, mas deixam de ser aplicáveis logo que algum elemento particular perturbe o modelo a que se destinam. Desta forma, a necessidade de existirem técnicas sólidas e flexíveis, que pudessem ser facilmente adaptadas a um vasto conjunto de problemas, levou ao aparecimento das técnicas *meta-heurísticas*.

O conceito de *meta-heurística* foi introduzido por Glover (1986), em simultâneo com o conceito de *pesquisa tabu* (que é uma das meta-heurísticas mais utilizadas). Uma *meta-heurística* é um processo iterativo de geração de soluções, que utiliza uma ou mais heurísticas subordinadas, combinando diferentes formas de exploração do espaço de pesquisa (a geração de novas soluções em regiões ainda não testadas — “exploration” — ou a concentração na vizinhança de soluções já conhecidas — “exploitation”).

Uma *meta-heurística* pode também ser definida como um conjunto de técnicas que podem ser usadas para definir heurísticas, as quais podem ser aplicadas a um vasto conjunto de problemas diferentes. Por outras palavras, uma *meta-heurística* pode ser vista como um esquema algorítmico geral que pode ser aplicado a diferentes problemas de optimização, em que são necessárias relativamente poucas modificações para que se adapte a um problema específico (Sörensen (2003)). Ou seja, uma meta-heurística é um processo de mais alto nível que fornece linhas directrizes no desenvolvimento de métodos para a resolução de um problema específico.

Alguns das características que favorecem a utilização de *meta-heurísticas* são as seguintes:

- enquanto que as heurísticas são dependentes da especificidade do problema que tenta resolver, o domínio de aplicação das *meta-heurísticas* é mais amplo;
- enquanto que as heurísticas assentam em processos iterativos que geralmente terminam quando não for encontrada uma solução que melhore a anterior, as *meta-heurísticas* incorporam estratégias de exploração do espaço das soluções para além da optimalidade local;
- as meta-heurísticas inspiram-se em processos estudados em áreas tão diversas como as ciências sociais, a física, a biologia, etc..

Existem vários tipos de *meta-heurísticas*, dependendo das técnicas de exploração do espaço de pesquisa, e que, segundo Sörensen (2003), podem ser divididas em três grandes classes: baseadas na vizinhança, baseadas em populações e híbridas.

As *meta-heurísticas baseadas na vizinhança*, também denominadas por *meta-heurísticas de pesquisa local*, usam uma operação chamada *movimento* para, iterativamente, se deslocarem de uma solução para uma outra. Ao conjunto de todas as

soluções que podem ser atingidas pelo movimento de uma dada solução x , dá-se o nome de *vizinhança*, $V(x)$, daquela solução. Estas soluções podem ser geradas aleatoriamente.

Geralmente, um operador de *pesquisa local* tenderá a encontrar apenas soluções óptimas locais, isto é, soluções em que na sua vizinhança não há soluções melhores. O principal objectivo do esquema meta-heurístico é dar a possibilidade de o operador de pesquisa local se afastar de um óptimo local, pois, geralmente, este óptimo não é o óptimo global.

Alguns exemplos de *meta-heurísticas* de pesquisa local são as seguintes: *pesquisa tabu*, *arrefecimento simulado*⁶, GRASP ("Greedy Randomized Adaptive Search Procedure") e *pesquisa iterativa local*.

As *meta-heurísticas baseadas em populações* mantêm um conjunto (população) de soluções. Estas meta-heurísticas tentam melhorar as soluções de uma população, permitindo que elas se combinem entre si. Estas combinações são efectuadas até que seja satisfeito um certo critério de paragem.

Como exemplos deste tipo de meta-heurísticas podem-se referir os seguintes: *algoritmos evolucionários*, *colónias de formigas*, *pesquisa por dispersão*⁷ e *algoritmos meméticos* (também podem ser considerados como meta-heurísticas híbridas).

As meta-heurísticas híbridas utilizam técnicas das meta-heurísticas baseadas na vizinhança e em populações no mesmo processo de pesquisa; por exemplo, um algoritmo genético que utiliza a pesquisa tabu para melhorar uma solução da sua população. Alguns trabalhos publicados que usam meta-heurísticas híbridas são: algoritmo genético com pesquisa local (Prins (2004)), algoritmo genético com arrefecimento simulado (Mahfoud e Goldberg (1995)), e algoritmo genético com pesquisa local e pesquisa tabu (Fleurent e Ferland (1994)).

4. Algoritmos Evolucionários

Os Algoritmos Evolucionários são procedimentos meta-heurísticos, que tentam abstrair e imitar alguns dos mecanismos da evolução natural, de forma a resolverem problemas que exijam adaptação, pesquisa e optimização. Estes métodos são estocásticos e iterativos, não garantindo a convergência para a melhor solução, uma vez que se baseiam em heurísticas. Estes algoritmos operam sobre um conjunto de indivíduos (população), em que cada indivíduo representa uma potencial solução do problema de optimização.

⁶ Tradução do termo inglês "simulated annealing".

⁷ Tradução do termo inglês "scatter search".

O processo evolucionário começa por gerar, aleatoriamente ou usando técnicas específicas para esse efeito, a primeira população (também denominada por população inicial), seguindo-se a avaliação de cada solução desta população usando uma função de aptidão. A função de aptidão atribui um valor a cada uma das soluções da população (valor de aptidão — “fitness”), o qual é uma medida da qualidade da solução do problema em estudo e é utilizado para orientar na pesquisa de novas soluções.

O processo evolucionário prossegue, aplicando às soluções da população principal (pode ou não ser a inicial), um procedimento para gerar uma nova população constituído pelas quatro etapas principais: selecção, cruzamento, mutação e substituição. A selecção consiste em criar uma população provisória com algumas das soluções da população principal. As soluções com maior valor de aptidão têm maior probabilidade de estarem presentes na população provisória do que as soluções com menor valor de aptidão. A estas soluções é então aplicado o operador genético cruzamento seguido de mutação, de forma a gerar uma nova população. A seguir, as soluções da população principal são substituídas pelas novas soluções, fazendo com que, normalmente, as soluções com melhores valores de aptidão se mantenham e as com os piores valores de aptidão sejam removidas. Este processo só termina, geralmente, quando for atingido um número predefinido de gerações, ou quando não for detectada uma melhoria significativa nas últimas gerações.

Existem vários tipos de Algoritmos Evolucionários, devendo-se a classificação destas técnicas a factores históricos relacionados com o aparecimento de diferentes linhas de investigação. No entanto, as técnicas mais representativas são as Estratégias Evolucionárias, a Programação Evolucionária e os Algoritmos Genéticos.

Tem sido amplamente reconhecida a vocação das abordagens evolucionárias para tratar problemas multi-objectivo (Deb (2001), Coello et al. (2002) e Osyczka (2002)), pela sua capacidade de trabalhar com uma população de soluções (não dominadas).

4.1. Estratégias Evolucionárias

As Estratégias Evolucionárias surgiram a partir do trabalho de Rechenberg (1964) (citado em Knowles e Corne (2000) e Costa e Oliveira (2002)), tendo sido desenvolvidas, inicialmente, por Rechenberg (1973) (citado em Bäck et al. (1997)) com a finalidade de resolver problemas de optimização. Posteriormente foram desenvolvidos novos esquemas evolucionários, mas seguindo os mesmos princípios, por Schwefel (1981). Estas técnicas foram desenvolvidas para serem aplicadas a problemas de engenharia.

A primeira versão desta técnica usa apenas duas soluções, uma progenitora e uma descendente, as quais são codificadas usando uma representação decimal. Esta versão considera apenas o operador genético mutação e um mecanismo de selecção (não

considera o operador cruzamento). Este tipo de Estratégias Evolucionárias designa-se por EE-(1+1), na actual notação de Estratégias Evolucionárias (Bäck et al. (1991)).

Surgiram, entretanto, algumas generalizações desta técnica, nomeadamente a sua aplicação a mais do que uma solução, nas denominadas EE-($\mu+1$). Nesta variante consideram-se μ soluções progenitoras, das quais é seleccionada apenas uma delas para a geração de uma descendente.

Mais tarde, desenvolveram-se outras variantes, designadas por EE-($\mu+\lambda$) e EE-(μ,λ), nas quais são geradas por mutação λ descendentes a partir de μ progenitoras e aplicados esquemas de selecção diferentes para escolher as μ melhores para a geração seguinte. No método EE-($\mu+\lambda$) as soluções da população para a geração seguinte são escolhidas entre as μ progenitoras e as λ descendentes. No método EE-(μ,λ) as soluções são escolhidas unicamente das λ descendentes (assumindo que $\lambda > \mu$).

Apesar das Estratégias Evolucionárias originais usarem apenas o operador genético mutação para gerar novos indivíduos, foi introduzido, posteriormente, o operador genético cruzamento, o qual foi aplicado juntamente com a mutação (Schwefel (1995)).

As Estratégias Evolucionárias revelaram-se algoritmos de optimização sólidos e eficientes, não exigindo nenhuma condição relativa à continuidade e convexidade do espaço de pesquisa, ao contrário de outros algoritmos de optimização (Schwefel (1995)).

4.2. Programação Evolucionária

A Programação Evolucionária foi introduzida por Fogel et al. (1966), com a finalidade de gerar máquinas de estados finitos para tarefas de pré-edição de sequências de símbolos. Actualmente, estes algoritmos são utilizados sobretudo em problemas de optimização numérica.

A principal característica da Programação Evolucionária está relacionada com a forma como os descendentes são gerados: o operador genético mutação é o único responsável pela geração de descendentes. O operador mutação é definido consoante o problema e está sujeito a adaptação durante o processo evolucionário.

Como não existe operador genético cruzamento, é possível usar qualquer representação para as soluções (está dependente do tipo de problema que se queira resolver), desde que se defina um operador mutação adequado ao problema. Na versão original, por exemplo, cada solução era uma máquina de estados finitos representada por um grafo. Em problemas de optimização numérica, as soluções são representadas por vectores de valores reais e para o problema do caixeiro-viajante, por exemplo, são utilizadas listas ordenadas.

A Programação Evolucionária simula a evolução natural como um processo baseado nas soluções, realçando o comportamento na ligação entre os progenitores e os descendentes, em vez da ligação genética, como acontece com os restantes Algoritmos Evolucionários.

4.3. Algoritmos Genéticos

Os Algoritmos Genéticos (AGs) são o grupo dos Algoritmos Evolucionários mais utilizados na resolução de problemas de pesquisa e de optimização. Estes algoritmos são métodos adaptativos inspirados nos mecanismos de evolução de populações de seres vivos, as quais evoluem de acordo com os princípios da selecção natural, descritos pela primeira vez por Charles Darwin⁸.

Os princípios básicos dos AGs foram introduzidos por Holland (1975) e desenvolvidos por ele e pelos seus colaboradores (alunos e colegas) na Universidade de Michigan, nas décadas de 1960 e 1970. Os objectivos da pesquisa de Holland foram os seguintes (Goldberg (1989)):

- a) abstrair e explanar com rigor os processos adaptativos dos sistemas naturais, e
- b) construir *software* para sistemas artificiais que retenham os mecanismos importantes dos sistemas naturais.

No entanto, foram os trabalhos realizados por um dos alunos de Holland, David Goldberg, que popularizaram estes algoritmos, quando este aplicou esta técnica para resolver um problema complexo de controlo no transporte de gás natural em gasodutos (no âmbito da sua tese de doutoramento). O algoritmo genético original, proposto por Holland, designa-se por AG canónico ou simples (Goldberg (1989)).

Os AGs são diferentes dos mecanismos clássicos de optimização e de pesquisa, nos seguintes aspectos (Goldberg (1989)):

- a) usam uma codificação do conjunto de parâmetros e não os próprios parâmetros;
- b) pesquisam a partir de uma população de pontos e não de um único ponto;
- c) utilizam a informação relativa vinda da função objectivo, e não de derivações ou de outro tipo de conhecimento auxiliar;
- d) utilizam regras de transição probabilísticas, e não determinísticas.

É normal, na terminologia dos AGs, usar-se vários sinónimos para os mesmos elementos (Eiben e Smith (2003)). No contexto do problema original, pode-se usar qualquer um dos termos *solução*, *fenótipo* ou *indivíduo* para designar os pontos do espaço das soluções (também designado por *espaço do fenótipo*). No contexto dos AGs, pode-se usar qualquer um dos termos *genótipo*, *cromossoma* e também *indivíduo* para

⁸ Charles Darwin (1809-1882) escreveu a obra "A Origem das Espécies", publicada em 1859.

designar os pontos do espaço onde se realiza a pesquisa evolucionária (também designado por *espaço do genótipo*). Também os elementos dos indivíduos (lugar e objecto) têm vários sinónimos: um *lugar* denomina-se normalmente por *variável*, *locus* (plural: loci), *posição*, ou *gene*; um *objecto* num lugar pode-se chamar de *valor* ou *alelo*. É vulgar usarem-se os termos *solução* e *indivíduo* (no espaço dos genótipos) como sinónimos, uma vez que um indivíduo é uma representação da solução.

Nesta dissertação os termos *solução* e *indivíduo* são usados como sinónimos, em que cada solução/indivíduo é composta por um conjunto de *variáveis* ou *genes*, cada uma com um certo *valor*.

Os AGs usam uma população de indivíduos (que representam as soluções do problema), sobre a qual aplicam mecanismos de selecção e de substituição, e operadores genéticos cruzamento (recombinação) e mutação, criando gerações sucessivas de indivíduos cada vez mais aptos. A pesquisa é orientada apenas pelo valor de aptidão de cada indivíduo da população: aqueles com maior valor de aptidão terão maiores probabilidades de serem seleccionados para reprodução (passando, assim, a sua informação genética para os descendentes nas gerações futuras), do que os com menor valor de aptidão (cujas informações genéticas tendem a desaparecer nas gerações futuras). As características essenciais dos AGs são as seguintes:

- usam uma população de soluções do problema;
- cada solução tem associado um valor de aptidão que está ligado à sua capacidade de sobrevivência e de reprodução;
- as soluções com os melhores valores de aptidão têm mais oportunidades de se combinarem com outras soluções da população;
- na reprodução são geradas novas soluções que herdam algumas das características de cada um dos seus progenitores;
- a combinação de algumas boas características dos progenitores permite obter uma geração com melhores soluções.

A estrutura básica de um AG, considerando uma população principal de tamanho POP, pode ser descrita da seguinte forma:

- 1) $t = 0$;
- 2) **Início**: gerar a primeira população principal (inicial), P_t ;
- 3) **Avaliação**: calcular o valor de aptidão de cada solução da população P_t ;
- 4) Construir a nova população (associada à próxima geração), P_{t+1} :
 - a) **Seleccionar** POP soluções (progenitores) da população P_t para reprodução (estas POP soluções constituem uma população provisória);

- b) Aplicar o operador genético cruzamento às POP soluções seleccionadas (aplicado a pares de soluções), donde resultam POP novas soluções (descendentes);
 - c) Aplicar o operador genético mutação às POP novas soluções (descendentes);
 - d) Inserir estas POP novas soluções na nova população P_{t+1} ;
- 5) $t = t + 1$;
- 6) Se o critério de paragem é verificado, então terminar; senão regressar ao passo 3).

No desenvolvimento de um AG para resolver um certo problema real é normalmente necessário considerar a representação (codificação) das soluções e especificar a função de aptidão, já que estas são as únicas entidades dependentes do problema. Desta forma, é necessário considerar no processo evolucionário os seguintes aspectos: representação (codificação) das soluções, medida de aptidão, início do processo, população, mecanismos de selecção, operadores genéticos (cruzamento e mutação), mecanismos de substituição e critérios de paragem.

4.3.1. Representação (codificação) dos indivíduos

O primeiro passo na definição de um AG é estabelecer uma ponte entre o contexto do problema original e o espaço de resolução do problema onde ocorre o processo evolucionário. Este primeiro passo é geralmente designado por *representação*, que significa especificar uma correspondência entre um conjunto de soluções do problema original (espaço dos fenótipos) e um conjunto de indivíduos (no espaço dos genótipos) (Eiben e Smith (2003)). Ou seja, a *representação* consiste em estabelecer uma codificação adequada para as possíveis soluções do problema (*fenótipos*) ao nível do material genético de cada indivíduo (*genótipos*). A escolha de um tipo de representação que facilite o tratamento das soluções do problema por parte do AG requer, normalmente, um grande conhecimento das características do problema.

No entanto, o termo *representação* pode ser usado nos dois sentidos (Eiben e Smith (2003)): como codificação e como descodificação. A *codificação* está relacionada com a representação de um fenótipo no espaço dos genótipos (por exemplo, pode-se falar em representação ou codificação binária das soluções). A *descodificação* está relacionada com a correspondência inversa, isto é, do genótipo para o fenótipo, a qual deve exigir que para cada indivíduo exista no máximo uma solução correspondente.

A representação mais comum, e que coincide com a ideia original de Holland, é aquela que utiliza o alfabeto binário para codificar os valores possíveis de cada variável do indivíduo. No entanto, outras representações podem ser usadas, as quais devem ser apropriadas às características do problema em estudo, tais como: números reais, números inteiros e até mesmo caracteres. De facto, se a resolução de um problema

necessitar de valores com elevada precisão, então a utilização da codificação binária implica a utilização de um elevado número de *bits*, o que exige um maior esforço computacional para explorar o espaço das soluções, que pode ser enorme. Desta forma, uma representação mais natural para este tipo de problemas, é a utilização de valores reais para representar as soluções. Este tipo de representação é usado em trabalhos apresentados por Michalewicz (1996).

No entanto, o tipo de codificação mais utilizada é a binária, quer seja na sua versão tradicional, quer numa versão designada por código *gray*. O código *gray* é uma representação que se caracteriza pela propriedade de adjacência entre dois valores inteiros consecutivos; isto é, dois números inteiros consecutivos ou adjacentes têm codificações binárias que diferem apenas em um *bit* (ver Tabela 1). A vantagem do código *gray*, relativamente à codificação binária tradicional, está em que, devido à propriedade de adjacência, quando um dado indivíduo é alterado em apenas um dos genes (de 0 para 1 ou vice-versa), o indivíduo resultante não é muito diferente do anterior, o que traduz uma pequena perturbação na solução associada ao indivíduo.

Número inteiro	Tradicional	Gray
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

Tabela 1: Comparação entre as codificações binárias tradicional e *gray*.

Os AGs usados na resolução de alguns problemas de optimização combinatoria apresentam dificuldades em usar a representação binária, como é o caso do problema do caixeiro-viajante. Nestes problemas, as representações mais adequadas são aquelas que se baseiam em permutações, nas quais cada indivíduo é definido como uma sequência ordenada de todos os símbolos, sem repetições de símbolos.

4.3.2. Início do processo

A primeira etapa do processo evolucionário consiste em criar a primeira população (também denominada de população inicial), cujos indivíduos são, normalmente, gerados aleatoriamente.

No entanto, é importante que já exista nesta população um certo nível de diversidade entre os seus indivíduos, de forma a permitir uma exploração mais alargada no espaço de pesquisa. Para tal, pode-se usar um gerador de valores aleatórios de distribuição uniforme, ou então, utilizar algumas metodologias construídas para o efeito (ver Poles et al. (2006)).

A maioria dos algoritmos evolucionários multi-objectivo gera as suas populações iniciais de forma aleatória. No entanto, se o espaço de pesquisa contiver poucas soluções admissíveis, é de esperar que a população inicial, quando gerada desta forma, possa ser composta apenas por soluções não admissíveis. Por isso, a finalidade do processo evolucionário não é apenas convergir para a frente óptima de Pareto real, mas também conduzir a pesquisa para a região admissível. Desta forma, pode ser tão complicado encontrar a região admissível como melhorar uma solução admissível, no sentido de determinar a frente óptima de Pareto.

Podem também ser usadas heurísticas específicas do problema para se chegar a uma população inicial de indivíduos com elevada aptidão (Eiben e Smith (2003)). No entanto, apesar desta estratégia implicar um elevado esforço computacional, poderá trazer vantagens, se acelerar a convergência do algoritmo ou evitar que esta seja atingida prematuramente.

Affenzeller e Wagner (2004) demonstraram que a geração da população inicial é um aspecto muito importante na construção de um AG, uma vez que uma população inicial mal construída pode levar o AG a convergir prematuramente, ficando retido em óptimos locais. Isto acontece se a informação genética guardada nos indivíduos de uma população não contiver a informação necessária para melhorar a qualidade da solução. Uma das razões para que parte desta informação genética essencial (informação genética associada à solução óptima global) esteja perdida (o que leva à convergência prematura do AG), pode ser por esta informação não estar representada na população inicial do AG. É frequente acontecer, especialmente na fase inicial do processo de pesquisa, que a informação genética essencial esteja escondida em indivíduos com maus valores de aptidão, os quais são eliminados no processo de selecção.

Por outro lado, também Haubelt et al. (2005) e Hill e Hiremath (2005) provaram que a construção de uma população inicial bem distribuída acelera a convergência para a frente óptima de Pareto real, independentemente do AG e do problema a resolver. Este resultado é muito importante, tendo em conta que o processo de optimização pode consumir um elevado tempo de execução. Haubelt et al. (2005) propuseram a incorporação de um método para construção de populações iniciais em algoritmos evolucionários multi-objectivos já existentes, baseado na abordagem denominada de

“Pareto-Front-Arithmetics” (PFA), a qual permite uma rápida aproximação à frente óptima de Pareto real (Haubelt e Teick (2003)).

4.3.3. Função de avaliação (aptidão)

O papel da função de avaliação é representar as condições de adaptabilidade dos indivíduos da população, constituindo o suporte do mecanismo de selecção e facilitando, assim, o melhoramento da população (ou seja, definindo o que significa melhoramento). Tecnicamente, a função de avaliação é uma função ou procedimento que atribui uma medida de qualidade aos indivíduos (genótipos). Tipicamente, esta função é composta por uma medida de qualidade no espaço das soluções possíveis (dos fenótipos) e pela representação inversa (descodificação): por exemplo, o valor de aptidão do genótipo 10010 é $f(18)$, em que f é a função a otimizar (Eiben e Smith (2003)).

A função da aptidão atribui um valor de aptidão a cada indivíduo (genótipo), o qual é o elemento fundamental para o processo de selecção. Para se calcular o valor de aptidão dos indivíduos, utiliza-se, normalmente, uma função de avaliação específica do problema em estudo. A função de avaliação devolve, para cada indivíduo, um valor numérico que reflecte o respectivo mérito. Note-se que um valor de aptidão pode estar associado a mais do que uma solução (fenótipo), e um indivíduo (genótipo) está associado a apenas uma solução (fenótipo) e um só valor de aptidão.

Normalmente, as noções de avaliação e de aptidão são iguais, mas nalguns casos isto não acontece. A função de avaliação fornece uma medida de desempenho relativamente a um conjunto particular de parâmetros, em que a avaliação de um indivíduo é independente das avaliações dos outros. A função de aptidão transforma aquela medida de desempenho na atribuição de oportunidades de reprodução, sendo, desta forma, dependente dos outros indivíduos da população.

Ao definir-se uma função de aptidão, é de extrema importância que esta seja adequada ao problema para uma execução correcta do AG, uma vez que a função de aptidão representa o ambiente do problema. O ideal seria a função de aptidão comportar-se de um modo suave e regular, no sentido de indivíduos semelhantes e com as mesmas características (que tenham em comum uma parte importante do material genético), terem valores de aptidão muito próximos.

4.3.4. População

Uma população é um conjunto de indivíduos (genótipos), com a particularidade de permitir a existência de elementos repetidos. A definição de uma população pode consistir apenas em especificar o número máximo de indivíduos que suporta, isto é, o seu tamanho.

O tamanho das populações é um aspecto muito importante na implementação de AG, uma vez que este valor vai afectar, quer a qualidade das soluções, quer o tempo de processamento do algoritmo. Populações pequenas têm como consequência uma fraca diversidade genética por parte dos seus elementos, o que leva a uma menor cobertura do espaço das soluções e, conseqüentemente, à convergência prematura (obtenção de soluções de menor qualidade). Grandes populações garantem uma maior diversidade de soluções na população, devido a uma maior cobertura do espaço das soluções, e permitem prevenir a convergência prematura, mas tal é conseguido à custa de um maior esforço computacional. Os AGs podem usar populações de quaisquer tamanhos, sendo normal este tamanho ser constante (não se alterando durante o processo evolucionário).

Um aspecto essencial para o bom funcionamento de um AG é a existência de uma boa *diversidade* entre os indivíduos da população. A *diversidade populacional* é uma medida associada à diferenciação das soluções no espaço das soluções e/ou no espaço dos objectivos. Se o nível de diversidade populacional for baixo, o que significa que os indivíduos são muito semelhantes, o operador genético cruzamento perde muito a capacidade de troca de informações úteis entre os indivíduos da população, o que faz com que a pesquisa possa progredir muito lentamente ou praticamente estacionar. Se o nível de diversidade populacional for alto, isto permitirá explorar melhor o espaço de pesquisa, o que é importante para escapar à optimalidade local.

A necessidade de controlar a diversidade populacional implica que as populações tenham tamanhos finitos e não muito grandes. Se isto não acontecer, o AG poderia tornar-se não aplicável na prática, visto que controlar uma população absurdamente grande poderia ocasionar um esforço computacional muito elevado.

Não existe uma medida única para a *diversidade populacional* pois, geralmente, pode referir-se à gama dos valores de aptidão, ao número de soluções (fenótipos) diferentes ou ao número de indivíduos (genótipos) diferentes (Eiben e Smith (2003)).

4.3.5. Mecanismos de selecção

Os mecanismos de selecção baseiam-se no princípio Darwiniano da "sobrevivência dos mais aptos": os indivíduos com melhores valores de aptidão têm maiores probabilidades de serem escolhidos para reprodução. Portanto, para além destes

mecanismos permitirem determinar que indivíduos (progenitores) são escolhidos para cruzamento, também permitem escolher que indivíduos (descendentes) devem sobreviver para a geração seguinte.

Existem basicamente dois tipos de mecanismos de selecção: baseados na proporção dos valores de aptidão e baseados na ordenação dos valores de aptidão. Na selecção baseada na proporção, os indivíduos são escolhidos de acordo com os seus valores de aptidão e da relação com os valores de aptidão dos outros indivíduos da população. Na selecção baseada na ordenação, os indivíduos são escolhidos de acordo com as respectivas posições na população ordenada e não de acordo com os seus valores de aptidão.

Alguns dos mecanismos de selecção baseados na proporção mais utilizados em problemas de optimização, são o método da roleta e a amostra universal estocástica.

O *método da roleta* (Goldberg (1989)) é um método estocástico que consiste em associar os indivíduos a porções contíguas de uma roleta, em que cada porção é proporcional à aptidão do indivíduo que lhe está associado (os indivíduos com maior valor de aptidão têm maiores probabilidades de serem escolhidos). São então realizados vários lançamentos da roleta, sendo seleccionados os indivíduos associados às porções atingidas por cada um destes lançamentos. Este mecanismo apresenta essencialmente duas desvantagens: redução da diversidade e convergência prematura (os mais aptos podem ser seleccionados muitas vezes), e estagnação da população (ao fim de algumas gerações, o valor médio da aptidão pode ser muito próximo dos melhores valores de aptidão).

No mecanismo denominado por *amostra universal estocástica* (Baker (1987)), os indivíduos são associados a porções proporcionais aos seus valores de aptidão e contíguas de uma roleta, tal como no método da roleta. A diferença é que aqui a roleta tem N ponteiros equidistantes — N é o número de indivíduos a seleccionar — e gira apenas uma vez, sendo escolhidos os indivíduos marcados pelos N ponteiros (assim, os indivíduos menos aptos têm mais possibilidades de serem escolhidos do que no método anterior).

Entre os mecanismos de selecção baseados na ordenação mais usados, estão a ordenação linear, a selecção por truncatura e a selecção por torneio.

No mecanismo designado por *ordenação linear* (Baker (1985)), os indivíduos da população são ordenados de acordo com os seus valores da função de aptidão, sendo atribuído a cada indivíduo um valor que corresponde à sua posição na população ordenada. Desta forma, ao pior indivíduo é atribuído o valor 1 e ao melhor o valor POP (em que POP é o tamanho da população). Depois, a cada indivíduo é atribuída uma probabilidade de selecção calculada com base numa dada distribuição (as mais usuais

são a linear e a exponencial). Esta técnica trava a convergência prematura do algoritmo (não há favorecimento dos melhores indivíduos) e evita, em gerações avançadas, a estagnação da população.

No mecanismo de *selecção por truncatura*, os indivíduos são ordenados de acordo com os seus valores de aptidão e, de seguida, são seleccionados aqueles indivíduos cujos valores de aptidão sejam maiores do que um limiar predefinido (só são seleccionados os melhores indivíduos).

O mecanismo denominado por *selecção por torneio* (Goldberg (1989)) consiste em escolher aleatoriamente um certo número de indivíduos da população (designado por dimensão do torneio) e fazer um torneio entre eles. Cada torneio consiste em comparar os valores de aptidão dos indivíduos envolvidos, sendo o vencedor (e o seleccionado) aquele com melhor valor de aptidão. O número de torneios realizados é igual ao número de indivíduos a serem seleccionados, ou seja, igual ao tamanho da população. Esta técnica não conduz à convergência prematura (desde que a dimensão dos torneios seja pequena), combate a estagnação da população, é simples de implementar e não requer grande esforço computacional. Este é talvez o mecanismo de selecção mais utilizado na resolução de problemas de optimização.

Uma técnica que está intimamente ligada ao mecanismo de selecção é o *elitismo*. O *elitismo* foi introduzido por De Jong (1975) e consiste em reter na população os seus melhores indivíduos, os quais passam directamente para a próxima geração. Muitos investigadores têm encontrado no *elitismo* vantagens significativas para o desempenho dos AG (Mitchell (1997)). Com esta técnica, pretende-se, por um lado, garantir que os melhores indivíduos de cada geração não sejam destruídos pela acção dos operadores genéticos cruzamento e mutação (a definir mais à frente), e por outro, acelerar a convergência do algoritmo.

A desvantagem do *elitismo* está em quando os melhores indivíduos encontrados são os mesmos durante um certo número de gerações consecutivas, o que pode implicar a presença de várias cópias dos mesmos indivíduos em cada geração. Desta forma, pode haver perda de diversidade entre os indivíduos da população e forçar a pesquisa na direcção de um óptimo local. Assim sendo, e para combater este efeito negativo, o elitismo não deve ser utilizado de forma constante, mas sim esporadicamente ao longo do processo evolucionário.

O número de indivíduos que formam a elite constitui um dos parâmetros mais importantes dos AGs, pois se este valor não for adequado pode influenciar negativamente a convergência do AG. Em geral, este é um número baixo (no máximo 10% da população) e a sua amostragem pode ser directa (os E melhores) ou por sorteio (os E melhores entre os E' melhores da população).

4.3.6. Operador genético cruzamento

O operador genético *cruzamento* (ou recombinação) consiste em efectuar trocas de genes entre dois indivíduos. Neste processo são gerados dois novos indivíduos (descendentes), resultantes da combinação de informação contida num par de indivíduos (progenitores). O sucesso do AG está apoiado na expectativa de que o resultado do cruzamento entre indivíduos (progenitores) com melhores valores de aptidão gere novos indivíduos (descendentes) ainda de melhor qualidade (relativamente aos progenitores).

É importante que o operador cruzamento possibilite a propagação, ao longo das gerações, da informação genética dos melhores indivíduos, para conseguir explorar bem o espaço das soluções, assim como reunir a melhor informação genética num indivíduo que representará uma solução final do problema de optimização em questão.

Existem vários tipos de cruzamento, os quais dependem do tipo de representação usada na codificação dos indivíduos. Quando os indivíduos são representados por cadeias de dimensão fixa (por exemplo, alfabeto binário), os tipos de cruzamento usados com mais frequência são os que a seguir se descrevem (Baker (1985)).

O *cruzamento com um ponto de corte* consiste em determinar aleatoriamente uma posição do indivíduo (um ponto de corte), separando em dois, e no mesmo ponto, cada um dos dois indivíduos seleccionados. Desta forma, são formadas quatro sequências de genes dos indivíduos seleccionados, as quais aparecerão cruzadas nos descendentes, que recebem uma sequência de cada um dos progenitores. Se $A = (A1, A2)$ e $B = (B1, B2)$, então $A + B = \{(A1, B2), (B1, A2)\}$.

O *cruzamento com dois pontos de corte* consiste em determinar aleatoriamente duas posições do indivíduo (dois pontos de corte), separando em três, e nos mesmos pontos, cada um dos dois indivíduos seleccionados. Desta forma, são formadas seis sequências de genes dos indivíduos seleccionados, em que as sequências do meio aparecerão trocadas nos descendentes, que recebem as duas sequências dos extremos de um dos progenitores e a do meio do outro. Se $A = (A1, A2, A3)$ e $B = (B1, B2, B3)$, então $A + B = \{(A1, B2, A3), (B1, A2, B3)\}$.

O *cruzamento com vários pontos de corte* é semelhante ao anterior, mas com mais do que dois pontos de corte, em que cada um dos descendentes recebe de um dos progenitores as sequências de genes de índice ímpar e do outro as sequências de genes de índice par, e o outro descendente recebe as restantes — se $A = (A1, A2, A3, A4, \dots)$ e $B = (B1, B2, B3, B4, \dots)$, então $A + B = \{(A1, B2, A3, B4, \dots), (B1, A2, B3, A4, \dots)\}$.

O *cruzamento uniforme* consiste na utilização de uma máscara binária, gerada aleatoriamente e de comprimento igual ao dos indivíduos. Depois, um dos progenitores herda os genes de um dos pais para as posições da máscara com valor 0, e do outro pai para as posições da máscara com valor 1; o mesmo acontece com o outro descendente,

mas agora para os valores da máscara trocados. Este tipo de cruzamento pode não ser aplicável nalgumas representações, como por exemplo nas baseadas em permutações (usadas normalmente em problemas como o caixeiro-viajante). Nestes casos, é necessário definir outros tipos de cruzamento mais específicos e que se enquadrem no alfabeto escolhido.

O desempenho obtido por cada um dos tipos de cruzamento depende muito da especificidade dos problemas a resolver, uma vez que nem sempre é o mesmo tipo de cruzamento que permite o melhor desempenho de um algoritmo quando este é aplicado na resolução de diferentes problemas.

Um aspecto importante a ter em conta quando se utiliza este operador está na atribuição de uma *probabilidade de cruzamento*, a qual se define como a medida da possibilidade de aplicação do operador cruzamento a um dado par de indivíduos. Quanto maior for este valor, maior é a possibilidade de entrarem novos indivíduos na população e em maior número. Os valores mais usuais para a probabilidade de cruzamento variam entre 0.6 e 1.0.

4.3.7. Operador genético mutação

O operador genético mutação consiste em alterar aleatoriamente o valor de um dos genes de um indivíduo. A utilização deste operador genético nos AGs serve para, por um lado, fazer regressar à população os valores dos genes perdidos durante o processo de selecção, de modo a que possam ser testados num novo contexto, e por outro, proporcionar a entrada de novos genes que não estavam presentes nas populações anteriores.

Quando são utilizadas as representações com conjuntos de valores inteiros (por exemplo, o binário), o operador mutação consiste em determinar aleatoriamente uma posição do indivíduo, para depois substituir o valor que se encontra nessa posição por um dos outros valores do conjunto associado à representação usada (se existir mais do que um valor possível para o substituir, então este é escolhido aleatoriamente).

Quando as representações usadas são baseadas em permutações, uma forma simples para este operador consiste em trocar de posição dois elementos.

No entanto, outros tipos de mutação têm sido propostos, como a *mutação múltipla* que consiste em aplicar este operador genético a todos os elementos do indivíduo. Com este tipo de mutação, podem existir diferenças significativas entre o indivíduo antes e depois da aplicação da mutação, no espaço de pesquisa, o que é vantajoso no sentido de escapar à optimalidade local.

Um outro tipo de *mutação* é a *localizada*, onde só são afectados pela mutação os genes menos significativos do indivíduo, o que se traduz em pequenas perturbações nos indivíduos.

Um dos aspectos importantes a considerar quando se utiliza o operador genético mutação em AGs, está na sua taxa de ocorrência sobre o indivíduo (*probabilidade de mutação*). Alguns estudos indicam que uma elevada probabilidade tenderá a tornar o AG num algoritmo essencialmente aleatório. Desta forma, o valor da probabilidade de mutação deve ser baixo, o suficiente para diversificar os indivíduos da população e não prejudicar a convergência do algoritmo. Nalguns casos, a probabilidade de mutação pode variar de acordo com uma função matemática monotonamente decrescente, fazendo com que a mutação ocorra com maior frequência nas primeiras gerações (quando é necessário uma pesquisa mais alargada) e mais raramente nas últimas gerações (para possibilitar a convergência do algoritmo) (Reeves (1995) e Gomes et al. (2004)). De qualquer forma, a probabilidade de mutação deve ser diferente de zero, para se poder escapar à optimalidade local.

4.3.8. Mecanismos de substituição

Depois de terem sido gerados e avaliados (através da função de aptidão), os indivíduos devem ser introduzidos na população. Deverá ser definido quantos e quais os indivíduos descendentes que serão inseridos na população da nova geração, e que indivíduos da população actual devem ser substituídos. Para tal, define-se uma taxa de substituição, que indica a proporção de indivíduos da população a ser substituída em cada geração. Quanto menor for este valor, menor será a diferenciação genética entre gerações, o que implicará uma convergência mais lenta do algoritmo.

Existem vários mecanismos de substituição, sendo o mais comum aquele que consiste em gerar tantos descendentes como o número de indivíduos da população, para depois substituir todos estes indivíduos pelos descendentes gerados. No entanto, existem outras alternativas como, por exemplo, determinar apenas um ou dois indivíduos, determinar uma certa percentagem para reprodução, ou usar elitismo.

No primeiro caso, denominado de *AG de estado estacionário*, a etapa básica do processo evolucionário consiste em seleccionar apenas dois progenitores da população corrente para reprodução (cruzamento seguido de mutação), da qual resulta apenas um ou dois novos indivíduos (Whitley (1989)). Estes novos indivíduos gerados vão substituir outros indivíduos da população corrente, os quais são escolhidos de acordo com os seus valores de aptidão (escolhem-se os com piores valores de aptidão) ou aleatoriamente. No entanto, por vezes, a substituição só é efectuada caso os novos indivíduos tenham valores de aptidão melhores do que os indivíduos da população corrente.

O segundo caso é um mecanismo de evolução intermédio (está entre os dois mecanismos anteriores), que consiste em escolher uma certa percentagem da população corrente para reprodução (De Jong e Sarma (1992)).

No terceiro caso é feito uma recolha dos melhores indivíduos da população (elitismo – ver secção 4.3.5 deste capítulo), os quais passarão imediatamente, e sem qualquer tipo de competição, para a população da geração seguinte.

De notar que o primeiro mecanismo difere dos outros três essencialmente no facto de nestes três mecanismos existir a possibilidade de competição entre indivíduos de gerações diferentes, uma vez que numa geração existem sempre indivíduos de gerações anteriores (o que não acontece no primeiro mecanismo, em que todos os indivíduos da população corrente são substituídos por novos indivíduos).

4.3.9. Critérios de paragem

A conclusão do processo evolucionário associado a um AG depende, quer das características do problema em estudo, quer do tempo de execução do algoritmo (esforço computacional). Desta forma, pode-se definir a qualidade da solução procurada de acordo com o tempo e os recursos disponíveis, e o fim a que se destina aquela solução. O critério de paragem do processo evolucionário deve reflectir todos estes aspectos. Desta forma, os critérios mais usados para este fim são geralmente os seguintes (Eiben e Smith (2003)):

- definir um tempo de CPU máximo;
- definir o número máximo de gerações (muito frequente);
- definir um valor mínimo para o desvio padrão do valor de aptidão das soluções da população;
- não se verificarem melhorias significativas das soluções durante um número de gerações consecutivas (muito frequente);
- definir um limiar mínimo de diversidade da população;
- encontrar uma "boa" solução, caso seja possível avaliar a qualidade das soluções.

5. Algoritmos genéticos multi-objectivo

Os métodos de programação matemática multi-objectivo calculam, em geral, em cada iteração uma única solução não dominada, através da optimização de funções escalares substitutas. Estas funções agregam temporariamente numa única dimensão as múltiplas funções objectivo, de tal modo que a solução óptima de uma função escalar substituta é uma solução não dominada do problema multi-objectivo. Assim, se for necessário caracterizar com alguma exaustividade o conjunto das soluções não

dominadas, estas abordagens podem exigir um esforço computacional apreciável. As características das soluções calculadas, em particular a respectiva diversidade, estão bastante dependentes dos parâmetros de informação de preferências incluídas nas funções escalares substitutas (Clímaco et al. (2003)).

Ao trabalharem com populações de soluções, os AGs têm a potencialidade de determinar várias soluções não dominadas numa só execução do algoritmo. O processo de optimização multi-objectivo deve ter em conta os três aspectos seguintes:

- minimizar a distância da frente de Pareto obtida à frente óptima de Pareto real;
- as soluções obtidas devem ter uma boa distribuição (de preferência uniforme);
- maximizar a extensão da frente de Pareto obtida, isto é, as soluções não dominadas obtidas devem abranger a maior gama de valores possíveis para cada objectivo.

Atendendo a estes três aspectos, quando se implementa um AG em optimização multi-objectivo, devem ter-se sempre em atenção as duas questões seguintes:

- 1) como aperfeiçoar a função de aptidão e a selecção, de forma a conduzir a pesquisa em direcção à frente óptima de Pareto real;
- 2) como manter uma população diversificada de soluções, de forma a prevenir a convergência prematura e alcançar uma frente de Pareto extensa e bem distribuída.

5.1. Função de aptidão e mecanismo de selecção

Ao contrário da optimização mono-objectivo, em que a função de aptidão é muitas vezes a função objectivo, na optimização multi-objectivo a função de aptidão e o mecanismo de selecção devem usar os vários objectivos do problema. Normalmente, podem-se distinguir três tipos de AGs multi-objectivo, de acordo com o tipo de função de aptidão e o mecanismo de selecção usados (Zitzler (1999)):

- os objectivos são considerados em separado;
- baseados em técnicas clássicas de agregação;
- usam directamente o conceito de dominância de Pareto.

Na *selecção por alternância dos objectivos*, durante a fase de selecção escolhe-se alternadamente um dos objectivos para determinar os valores de aptidão dos indivíduos da população. Sempre que for necessário seleccionar um indivíduo da população para reprodução, será escolhido um dos objectivos do problema, que poderá ser diferente do usado anteriormente, o qual será usado para definir que indivíduo irá para uma população temporária (composta pelos indivíduos seleccionados para reprodução).

Na *selecção por agregação com variação dos parâmetros*, os vários objectivos são agregados numa única função objectivo parametrizada, em que para cada indivíduo da população é atribuído um conjunto diferente de parâmetros desta função. Algumas

abordagens usam o método da soma pesada para parametrizar os objectivos do problema. Uma vez que cada indivíduo é avaliado usando uma combinação particular de parâmetros, isto significa que cada indivíduo da população é avaliado por uma função objectivo diferente. Desta forma, a optimização é realizada em várias direcções, em simultâneo.

A *selecção baseada no conceito de dominância*, introduzida por Goldberg (1989), consiste em determinar vários níveis de frentes (de soluções) não dominadas (ver secção 2.4.4 deste capítulo). Esta ideia foi aproveitada por vários investigadores, resultando em várias abordagens evolucionárias onde os mecanismos para atribuir valores de aptidão às soluções são baseados na dominância de Pareto (por exemplo, Fonseca e Fleming (1993), Horn et al. (1994) e Srinivas e Deb (1994)). No entanto, Fonseca e Fleming (1995), fizeram notar que o desempenho das abordagens que utilizam este tipo de selecção pode ser influenciado pela dimensão do espaço de pesquisa. Mesmo assim, estas técnicas parecem ser as mais populares no campo da optimização evolucionária multi-objectivo (Veldhuizen e Lamont (1998)).

5.2. Diversidade da população

A preservação da diversidade dos indivíduos de uma população é crucial para a eficiência de qualquer abordagem evolucionária multi-objectivo. De forma a controlar a diversidade populacional, deve existir um controlo sobre o mecanismo de determinação das probabilidades de sobrevivência dos indivíduos mais aptos. Este mecanismo denomina-se por *pressão selectiva* e está associado à probabilidade dos melhores indivíduos serem seleccionados.

Uma elevada pressão selectiva significa que a pesquisa se concentra nos indivíduos mais aptos (porque têm uma elevada probabilidade de serem seleccionados), o que implica que a população tende a ficar rapidamente homogénea e com os seus indivíduos com elevada aptidão. Uma baixa pressão selectiva permite a exploração de uma maior diversidade de soluções, pois as probabilidades dos indivíduos da população serem seleccionados são muito próximas.

A aplicação ideal da pressão selectiva seria: menor pressão selectiva no início do processo (para favorecer a diversidade) e maior pressão selectiva na etapa final do processo (para favorecer a convergência para um óptimo global). Portanto, como o aumento da *pressão selectiva* diminui a *diversidade populacional*, é importante um equilíbrio entre estes dois factores, o que é conseguido com a aplicação dos operadores genéticos.

Para tratar esta questão da diversidade da população, foram desenvolvidos vários métodos, descrevendo-se seguidamente alguns dos mais utilizados em optimização evolucionária multi-objectivo.

5.2.1. Pré-selecção

Cavichio (1970) foi o primeiro a introduzir um mecanismo explícito de preservação da diversidade num AG, que designou por *pré-selecção*. A ideia base da *pré-selecção* é substituir um indivíduo por um semelhante. Este mecanismo consiste no seguinte: quando é gerado um descendente, este é comparado com os seus progenitores; dos três indivíduos (descendente e progenitores) os dois com melhores valores de aptidão transitam para a próxima geração.

5.2.2. Técnica de multidões

A técnica de multidões desencoraja a aglomeração de soluções em qualquer zona do espaço de pesquisa, introduzindo, desta forma, diversidade entre as soluções de uma população. No AG apresentado por De Jong (1975), o modelo de multidões utiliza o conceito de população parcialmente semelhante e uma estratégia de multidões. Neste AG apenas é permitido que uma proporção LG (denominada de desnível de geração) da população seja reproduzida em cada geração. Além disso, antes de um descendente ser introduzido na população parcialmente semelhante, são escolhidas aleatoriamente FM soluções (designado por factor de multidão) desta população, as quais são comparadas com o descendente para que seja substituída a mais *semelhante* a ele (o autor usou LG = 0.1 e FM = 2/3). Como uma solução é substituída por um descendente semelhante, este processo preserva a diversidade da população.

5.2.3. Partilha do valor de aptidão

Goldberg e Richardson (1987) sugeriram a técnica de partilha do valor de aptidão, que consiste, basicamente, em degradar o valor de aptidão de indivíduos semelhantes (mais próximos). Esta técnica é a mais usada pelos AGs para a resolução de problemas de optimização multi-objectivo, e baseia-se na ideia de que os indivíduos que se concentram numa mesma zona (nicho) têm de partilhar os recursos disponíveis nessa zona. Quanto mais indivíduos estiverem localizados na vizinhança de um certo indivíduo, maior degradação sofrerá o valor de aptidão deste indivíduo. Esta vizinhança é definida em termos de uma medida de distância (no espaço das soluções ou dos objectivos) entre dois indivíduos x e y da população, $d(x,y)$, e especificada pelo *parâmetro de partilha* (*raio do nicho*), σ_{share} . O valor de aptidão de um indivíduo x é então dividido pelo número

de indivíduos que estão a uma distância de x inferior a σ_{share} , isto é, ao número de indivíduos que pertencem ao nicho de centro x e raio σ_{share} .

O processo para determinar o valor de aptidão de um indivíduo x numa população P , é o seguinte (Deb (2001)):

1. Calcular o *valor da função de partilha*, $sh(d(x,y))$, para todos os indivíduos da população, da seguinte forma:

$$Sh(d(x,y)) = \begin{cases} 1 - \left(\frac{d(x,y)}{\sigma_{\text{share}}} \right)^\alpha, & \text{se } d(x,y) \leq \sigma_{\text{share}}, \\ 0, & \text{caso contrário} \end{cases}$$

em que $\alpha > 0$ e assume tipicamente o valor 1 ou 2 (Deb (1989)).

2. Calcular o *contador de nichos*, $nc(x)$, para cada indivíduo x da população, da seguinte forma:

$$nc(x) = \sum_{y \in P} Sh(d(x,y))$$

3. Calcular o *valor de aptidão partilhado*, da seguinte forma:

$$F'(x) = \frac{f(x)}{nc(x)}, \text{ em que } f(x) \text{ é o } \textit{valor de aptidão} \text{ do indivíduo } x.$$

Depois de calculado o valor de aptidão partilhado para todos os indivíduos da população, é aplicado, normalmente, um método de selecção proporcional ao valor de aptidão partilhado para seleccionar os indivíduos para competição.

O parâmetro de partilha, σ_{share} , corresponde à distância máxima entre duas soluções, para que estas sejam consideradas como pertencentes ao mesmo nicho. No contexto da optimização multi-objectivo, o valor para o parâmetro σ_{share} deve ser escolhido de forma a obter uma distribuição uniforme das soluções da frente óptima de Pareto.

Se os nichos forem considerados no espaço dos objectivos, então usa-se a distância Euclidiana entre duas soluções. Se a frente óptima de Pareto for conhecida, então pode-se partir do princípio que as POP (tamanho da população) soluções estão distribuídas uniformemente pela frente óptima de Pareto. Desta forma, conhecendo-se o perímetro L da frente óptima de Pareto, deve-se usar $\sigma_{\text{share}} = L/POP$. Assim, cada nicho conterà apenas uma solução, para uma distribuição quase-uniforme das soluções óptimas de Pareto. No entanto, normalmente, em particular em problemas reais, a frente óptima de Pareto real não é conhecida *a priori*.

Fonseca e Fleming (1993) sugeriram um procedimento para calcular o valor de σ_{share} em cada geração. Este procedimento consiste em calcular o hiper-volume limitado pelos valores mínimos (\min_m) e máximos (\max_m) de cada função objectivo $f_m(x)$ ($m = 1,$

..., M), onde se encontram todas as POP soluções da população: $V = \prod_{m=1}^M (\max_m - \min_m)$.

Para cada objectivo, o respectivo termo é acrescido em σ_{share} e calcula-se um novo hiper-volume: $V' = \prod_{m=1}^M (\max_m - \min_m + \sigma_{\text{share}})$. A diferença entre estes dois hiper-volumes é $\Delta V = V' - V$ e corresponde ao hiper-volume onde idealmente cada uma das POP soluções deveria estar. Como cada nicho ocupa um hiper-volume de $(\sigma_{\text{share}})^M$ e existem POP hiper-volumes destes em ΔV , então para calcular σ_{share} usa-se a seguinte equação:

$$\Delta V = \prod_{m=1}^M (\max_m - \min_m + \sigma_{\text{share}}) - \prod_{m=1}^M (\max_m - \min_m) = \text{POP} (\sigma_{\text{share}})^M \quad (1).$$

Para problemas com duas funções objectivo ($M = 2$), esta equação reduz-se à seguinte:

$$\sigma_{\text{share}} = \frac{(\max_2 - \min_2) + (\max_1 - \min_1)}{\text{POP} - 1}.$$

Se forem usados os valores normalizados das funções objectivo (isto é, valores entre 0 e 1), a equação (1) pode ser escrita de forma a normalizar o parâmetro de partilha σ'_{share} da seguinte forma:

$$(1 + \sigma'_{\text{share}})^M - 1 = \text{POP} (\sigma'_{\text{share}})^M, \text{ pois } \max_m - \min_m = 1, \text{ para } m = 1, \dots, M.$$

Logo, para $M = 2$, esta última equação reduz-se a:

$$\sigma'_{\text{share}} = \frac{2}{\text{POP} - 1}.$$

5.2.4. Técnica de agrupamentos

Esta técnica de preservação da diversidade apenas é aplicada em abordagens que usam populações externas/secundárias, pois só são aplicadas quando o número de soluções candidatas a pertencerem à população externa é superior ao seu tamanho máximo. Esta técnica consiste em determinar os agrupamentos que melhor se distribuem pela frente não dominada, em número igual ao tamanho máximo da população externa.

Este processo começa por considerar tantos agrupamentos como o número total de soluções candidatas, para de seguida calcular as distâncias entre todos os pares de agrupamentos; por exemplo, pode-se considerar a distância d_{ij} entre dois agrupamentos C_i e C_j como a distância Euclidiana média de todos os pares de soluções ($x \in C_i, y \in C_j$):

$$d_{ij} = \frac{1}{|C_i| |C_j|} \sum_{x \in C_i, y \in C_j} d(x, y).$$

A distância entre x e y , $d(x, y)$, pode ser calculada nos espaços das soluções ou dos objectivos. Depois de calcular as distâncias entre todos os pares de agrupamentos, aqueles que estão associados à menor distância obtida são agrupados, formando um só

agrupamento (o número de agrupamentos é decrementado em uma unidade). Este processo (calcular as distâncias entre todos os pares de agrupamentos e juntar o par que está à menor distância) repete-se, até ficarem apenas tantos agrupamentos como o tamanho máximo da população externa, os quais são disjuntos entre si. Por fim, para cada agrupamento determina-se a solução que detém a menor distância média para as restantes soluções do agrupamento, a qual será a representante do agrupamento.

5.3. Principais abordagens com Algoritmos Genéticos

A primeira verdadeira implementação de um algoritmo evolucionário multi-objectivo foi sugerido por Schaffer (1984) com o AG designado por VEGA.

Mais tarde, Goldberg (1989) apresentou um procedimento de classificação das soluções por níveis de dominância. A sugestão de Goldberg foi no sentido de usar o conceito de dominância para inserir mais cópias de soluções não dominadas numa população. Para manter a diversidade, sugeriu a utilização de uma estratégia de nichos entre as soluções da mesma classe de não dominância. A partir das sugestões de Goldberg foram desenvolvidos, alguns anos depois, alguns algoritmos evolucionários multi-objectivo, dos quais se destacam os seguintes: WBGA (Hajela e Lin (1992)), MOGA (Fonseca e Fleming (1993)), NSGA (Srinivas e Deb (1994)), NPGA (Horn et al. (1994)) e NPGA 2 (Erickson et al. (2001)).

No entanto, e apesar do sucesso da aplicação das abordagens referidas, as quais não incorporam o elitismo nos seus AG, a diversos problemas de optimização multi-objectivo, alguns autores, como Rudolph (1999) e Zitzler et al. (2000), sugeriram que o elitismo pode melhorar o desempenho dos AG e, até mesmo, impedir que algumas soluções não dominadas obtidas durante o processo de pesquisa se percam. Algumas das abordagens elitistas em optimização multi-objectivo são: SPEA (Zitzler e Thiele (1998)), NSGA-II (Deb et al. (2000)) e SPEA-2 (Zitzler et al. (2001)).

Existem na literatura várias publicações que apresentam descrições de algumas das várias abordagens existentes para optimização evolucionária multi-objectivo, incluindo discussões detalhadas dos seus pontos fortes e fracos. Alguns destes artigos ou livros são: Fonseca e Fleming (1995), Veldhuizen e Lamont (1998), Veldhuizen e Lamont (2000), Fonseca e Fleming (1998-a), Fonseca e Fleming (1998-b), Deb (1999), Veldhuizen (1999), Coello (2000), Deb (2001) e Kicinger et al. (2005).

Existem também na literatura algumas publicações que, para além de descreverem alguns dos métodos existentes nesta área, apresentam também alguns dos problemas a que estas abordagens foram aplicadas: Coello (1999) e Coello et al. (2002).

Para além disto, existe também um sítio na internet que contém uma lista com muitas publicações nesta área: <http://www.lania.mx/~ccoello/EMOO/EMOObib.html>.

5.3.1. Abordagens Não Elitistas

Apesar de até à data terem sido sugeridas várias abordagens não elitistas em optimização evolucionária multi-objectivo, neste texto apenas serão descritas as seguintes: VEGA, WBGA, MOGA, NSGA e NPGA.

5.3.1.1. Vector Evaluated Genetic Algorithm (VEGA)

O primeiro AG para determinar um conjunto de soluções não dominadas de um problema de optimização multi-objectivo foi implementado por Schaffer (1984), ao qual deu o nome de "Vector Evaluated Genetic Algorithm" (VEGA). Posteriormente, Schaffer (1985) comparou este AG com uma técnica de pesquisa adaptativa aleatória, tendo observado um melhor desempenho por parte do VEGA.

Para tratar os M objectivos do problema, em cada geração a população é dividida, aleatoriamente, em M sub-populações de tamanhos iguais, em que a cada uma destas sub-populações é associado um único objectivo.

Depois, às soluções de cada uma das sub-populações é atribuído um valor de aptidão, de acordo com a correspondente função objectivo (*selecção por alternância dos objectivos* — ver secção 5.1 deste capítulo). É então aplicado um mecanismo de selecção (proporcional à aptidão), mas apenas entre soluções da mesma sub-população.

Por fim, são aplicados os operadores genéticos cruzamento e mutação a toda a população, para gerar as soluções da nova população. Neste algoritmo, não é usada qualquer técnica específica para controlar a diversidade da população.

Esta abordagem é particularmente útil no tratamento de problemas em que as funções objectivo tomam valores de diferentes ordens de magnitude. Uma vez que todas as soluções de uma sub-população têm associado um valor de aptidão baseado numa função objectivo particular, a restrição do mecanismo de selecção apenas às soluções desta população leva a que sejam realçadas as melhores soluções para cada função objectivo. Apesar de serem reconhecidas algumas falhas importantes (Schaffer (1985), Fonseca e Fleming (1995)), esta abordagem é um ponto de referência nesta área.

5.3.1.2. Weight-Based Genetic Algorithm (WBGA)

Hajela e Lin (1992) introduziram uma abordagem, que designaram por "Weight-Based Genetic Algorithm" (WBGA), em que cada função objectivo $f_m(x)$ é multiplicada por um peso w_m (com $m = 1, \dots, M$), isto é as funções objectivo são ponderadas.

Para determinar os valores de aptidão dos indivíduos da população, esta abordagem baseia-se na *selecção por agregação com variação dos parâmetros* (ver secção 5.1 deste capítulo). Para tal, é usado o método da soma pesada, em que cada indivíduo da

população tem associado uma combinação de pesos normalmente diferente. Desta forma, em vez de se determinar uma única solução não dominada correspondente a uma combinação de pesos específica, a população mantém em simultâneo várias combinações de pesos, determinando, assim, várias soluções não dominadas em cada iteração do AG.

A preservação da diversidade das combinações de pesos é conseguida através de duas técnicas: usando um método baseado em nichos apenas sobre cada combinação de pesos (através da técnica de partilha do valor de aptidão — secção 5.2.3 deste capítulo — aplicada no espaço dos objectivos), ou avaliando sub-populações escolhidas adequadamente e associadas a diferentes combinações de pesos predefinidas (semelhante ao VEGA) (Deb (2001)).

O valor de aptidão de uma solução x , $F(x)$, é igual à soma pesada dos valores normalizados das funções objectivo, de acordo com a equação seguinte:

$$F(x) = \sum_{m=1}^M w_m \frac{f_m(x) - f_m^{\min}}{f_m^{\max} - f_m^{\min}},$$

em que $w = (w_1, \dots, w_M)$ é a combinação de pesos normalizados a partir da qual se obteve a solução x , f_m^{\max} e f_m^{\min} são os valores máximo e mínimo de $f_m(x)$, respectivamente.

Na técnica de partilha do valor de aptidão, para calcular o valor de aptidão partilhado, $F'(x)$, é necessário calcular as distâncias entre todos os pares de soluções da população, $d(x,y)$, o que é feito usando uma seriação das combinações de pesos associadas às soluções. Por exemplo, se forem usadas CP combinações de pesos a partir das quais se obtiveram POP soluções, a cada uma destas CP combinações é associado um valor inteiro entre 1 e CP. Desta forma, a distância entre duas soluções x e y é igual ao valor absoluto da diferença dos inteiros associados às combinações de pesos de que resultaram as soluções x e y (para mais detalhes, ver Deb (2001)). Uma vez que o valor de aptidão é degradado pela aplicação da técnica de partilha do valor de aptidão, é necessário usar-se o mecanismo de selecção proporcional ao valor de aptidão partilhado. Os operadores genéticos (cruzamento e mutação) são aplicados normalmente.

Na técnica semelhante ao VEGA, o AG começa por escolher um conjunto de CP combinações diferentes de pesos. A seguir, cada uma destas combinações de pesos é usada para determinar a aptidão pesada normalizada de todos os POP indivíduos da população, dos quais se escolhem, de acordo com estes valores de aptidão, os POP/CP melhores indivíduos para formarem uma sub-população. Portanto, são constituídas CP sub-populações, cada uma contendo os respectivos melhores POP/CP indivíduos de acordo com os valores da função de aptidão pesada normalizada associada a cada uma

das combinações de pesos. O mecanismo de selecção e os operadores genéticos (cruzamento e mutação) são restringidos às soluções de cada sub-população.

5.3.1.3. Multiple Objective Genetic Algorithm (MOGA)

Em 1993 surgiu um outro algoritmo, denominado "Multiple Objective Genetic Algorithm" (MOGA), da autoria de Fonseca e Fleming (1993). Estes autores foram os primeiros a sugerir um AG multi-objectivo que realça explicitamente a importância das soluções não dominadas, ao mesmo tempo que preserva a diversidade das soluções na população.

Para determinar os valores de aptidão das soluções da população de tamanho POP, este algoritmo utiliza o esquema de classificação das soluções da população do AG por níveis de dominância (ver secção 5.1 deste capítulo): a classificação de um certo indivíduo da população é proporcional ao número de indivíduos da mesma população que o dominam. Esta técnica começa por determinar, para cada solução x , o número de soluções que a domina, $nd(x)$, obtendo-se a sua classificação de acordo com o seguinte valor: $r(x) = 1 + nd(x)$. Desta forma, as soluções classificadas com valor 1, são as soluções não dominadas da população.

As soluções são então ordenadas de acordo com as respectivas classificações, das melhores (com $r = 1$) para as piores (com $r \leq POP$). De seguida, a cada solução é atribuído um valor de *aptidão bruta*, através de uma função linear (ou outra adequada), de forma a atribuir às soluções valores entre POP (para as melhor classificadas) e 1 (para as pior classificadas). Determina-se então a média dos valores de *aptidão bruta* entre as soluções com a mesma classificação, a qual é atribuída como *valor de aptidão* a cada solução com a mesma classificação.

Finalmente, é aplicada a técnica de partilha do valor de aptidão (secção 5.2.3 deste capítulo), para determinar o valor de aptidão partilhado das soluções da população. Desta forma, embora as soluções com a mesma classificação tenham o mesmo valor de aptidão, as soluções que pertencem a um nicho com menos indivíduos têm melhor valor de aptidão partilhado.

Para garantirem a diversidade entre as soluções não dominadas da população, os autores introduziram um mecanismo baseado num esquema de nichos entre soluções com a mesma classificação. Para tal, usaram a técnica de partilha do valor de aptidão (ver secção 5.2.3 deste capítulo), com as seguintes adaptações:

- a distância entre soluções é medida no espaço dos objectivos e é normalizada;
- para cada solução x , $d(x, y)$ é calculada para todas as soluções y (incluindo x) com a mesma classificação;

- para calcular o valor da função de partilha de uma solução x , é usada a equação para calcular $Sh(x)$, com $\alpha = 1$ (ver secção 5.2.3 deste capítulo);
- o contador de nicho, $nc(x)$, é calculado somando os valores da função de partilha, envolvendo todas as soluções com a mesma classificação de x .

Finalmente, são aplicados o mecanismo de selecção universal estocástica (de acordo com os valores de aptidão partilhado) e os operadores genéticos cruzamento com um ponto de corte e mutação binária simples, de forma a gerar uma nova população.

5.3.1.4. Non-dominated Sorting Genetic Algorithm (NSGA)

A ideia de Goldberg em usar o conceito de ordenação por não dominância (Goldberg (1989)), foi implementada na totalidade por Srinivas e Deb (1994), na abordagem denominada por "Non-dominated Sorting Genetic Algorithm" (NSGA). Esta abordagem utiliza a selecção baseada no conceito de dominância para classificar as soluções da população por níveis de dominância (ver secção 5.1 deste capítulo) e uma estratégia de partilha dos valores de aptidão para preservar a diversidade entre as soluções de cada frente não dominada.

O primeiro passo deste algoritmo consiste em aplicar o conceito de dominância de Pareto às soluções de uma população P , classificando-as em várias frentes de soluções não dominadas (ver secção 2.4.4 deste capítulo).

O processo para atribuir valores de aptidão às soluções da população P começa pela primeira frente não dominada (a que contém as melhores soluções, isto é, as não dominadas), até à última (que contém as piores soluções). A cada solução da primeira frente não dominada é atribuído um valor de aptidão igual ao tamanho da população, uma vez que todas estas soluções têm o mesmo nível de importância, em termos de proximidade à frente óptima de Pareto.

Para manter a diversidade da população, os valores de aptidão das soluções são degradados, de acordo com os respectivos números de soluções vizinhas. Para tal, é usado o método de partilha do valor de aptidão (ver secção 5.2.3 deste capítulo), que começa por calcular, para cada solução x de uma frente P_1 , a distância Euclidiana normalizada para uma outra solução desta frente, no espaço das soluções. Depois, usando esta distância, calcula-se a função de partilha, Sh , com $\alpha = 2$. Qualquer solução da população que esteja a uma distância superior a σ_{share} (que é um valor fixo) de x não contribui para o valor da função de partilha desta solução. Depois de estarem calculados os valores de aptidão partilhados de todas as soluções da frente P_1 , calculam-se os contadores de nicho de cada solução $nc(x)$ da frente P_1 ($nc(x)$ traduz o número de

soluções vizinhas de x , incluindo ela própria). Por fim, para se obter o valor de aptidão partilhado da solução x , divide-se o seu valor de aptidão pelo seu contador de nicho.

Depois de calculados os valores de aptidão partilhados das soluções da primeira frente, determinam-se estes valores para as soluções da segunda frente, começando-se por atribuir a cada uma destas soluções um valor de aptidão partilhado ligeiramente inferior ao valor de aptidão partilhado mínimo associado a uma solução da primeira frente. Desta forma, qualquer solução da primeira frente tem atribuído um valor de aptidão partilhado superior a qualquer solução da segunda frente. Depois, o processo continua até estar atribuído um valor de aptidão partilhado a todas as soluções desta frente. Finalmente, e depois de estar atribuído um valor de aptidão partilhado às soluções da segunda frente, o processo repete-se para as restantes frentes não dominadas.

Uma vez que o método de partilha do valor de aptidão é usado com um operador de selecção proporcional, o NSGA usa como mecanismo de selecção o método da roleta (Goldberg (1989)), que divide a roleta em tantas porções como o número de soluções da população e em que cada porção é proporcional ao valor de aptidão partilhado da respectiva solução (ver secção 4.3.5 deste capítulo). Os operadores genéticos cruzamento e mutação são aplicados a toda a população normalmente.

5.3.1.5. Niche Pareto Genetic Algorithm (NPGA)

Horn et al. (1994) propuseram uma abordagem, designada por "Niche Pareto Genetic Algorithm" (NPGA), a qual difere das abordagens anteriores, essencialmente, no tipo de mecanismo de selecção que usa.

Esta abordagem combina a técnica de partilha do valor de aptidão com um mecanismo de selecção baseado na ordenação dos valores de aptidão, a selecção por torneio (ao contrário das abordagens VEGA, NSGA e MOGA, que aplicam a selecção proporcional à aptidão). Este algoritmo começa por seleccionar aleatoriamente duas soluções x e y da população P de tamanho POP , as quais são comparadas com uma sub-população P_{dom} (de dominação), de tamanho POP_{dom} ($\ll POP$), escolhida aleatoriamente da população P (geralmente $POP_{dom} = 0.1 \times POP$). Se uma destas soluções domina todas as soluções de P_{dom} e a outra solução é dominada por pelo menos uma solução de P_{dom} , então a vencedora do torneio é a primeira. Se esta situação não acontecer, então as soluções x e y são testadas na sub-população (incompleta) de descendentes já gerados Q e se $Q \neq \emptyset$ (se $Q = \emptyset$, então escolhe-se uma delas aleatoriamente): calculam-se os contadores de nicho das soluções x e y considerando apenas a sub-população de descendentes Q , em que o contador de nicho de x , $nc(x)$, é calculado como o número de descendentes ($z \in Q$) que estão a uma distância $d(x,z)$ de x inferior a σ_{share} , em que $d(x,z)$ é a distância Euclidiana entre as soluções x e z no espaço

dos objectivos. Das soluções x e y a que tiver menor contador de nicho (com menos vizinhos) é a vencedora do torneio.

5.3.2. Abordagens Elitistas

A utilização de elitismo em AG levanta algumas questões importantes, como sejam (Zitzler (1999)):

- Que indivíduos guardar na elite e durante quanto tempo?
- Quando e como os indivíduos guardados devem ser reinseridos na população?

Uma forma de implementar o elitismo consiste em utilizar uma população externa (também denominada por secundária), na qual são mantidas as soluções não dominadas obtidas durante o processo de pesquisa. Neste caso, é necessário definir o mecanismo de actualização (de forma a garantir a diversidade das soluções na população), o tamanho (número máximo de soluções que pode guardar) e a estrutura da população externa, assim como o modo como as soluções são reinseridas na população.

Este tipo de abordagem (elitista com população externa) poderá ser vantajosa em problemas com muitas soluções não dominadas. Também é possível parametrizar o elitismo, fazendo variar, por exemplo, o mecanismo de actualização da população externa e o nível de elitismo. No entanto, a actualização da população externa exige um esforço computacional extra.

Apesar de existirem várias abordagens elitistas em optimização evolucionária multi-objectivo, neste texto apenas serão descritas as três abordagens mais referidas na literatura: SPEA, NSGA-II e SPEA-2.

5.3.2.1. Strength Pareto Evolutionary Algorithm

Zitzler e Thiele (1998) propuseram um AG elitista, que designaram por "Strength Pareto Evolutionary Algorithm" (SPEA), em que o elitismo é tratado através de uma população externa (ver também Zitzler e Thiele (1999)). Esta população guarda um número fixo de soluções não dominadas que vão sendo obtidas ao longo das gerações (iterações) do AG. Em cada geração, as novas soluções não dominadas obtidas são comparadas com as da população externa, sendo guardadas as soluções não dominadas resultantes. Este AG não se preocupa apenas em preservar as elites, pois estas também participam no mecanismo de selecção e na aplicação dos operadores genéticos, conjuntamente com a população corrente, na esperança de influenciar a condução da população na direcção das regiões mais interessantes do espaço de pesquisa.

O SPEA é semelhante a outras abordagens em optimização evolucionária multi-objectivo, nos seguintes aspectos (Zitzler (1999)):

- guarda algumas soluções externamente, as quais representam uma frente não dominada entre todas as soluções consideradas até então;
- usa o conceito de dominância no cálculo dos valores de aptidão das soluções;
- determina agrupamentos de soluções não dominadas para reduzir o número destas soluções guardadas na população externa, sem destruir as características da frente óptima de Pareto;

e é diferente nos seguintes aspectos:

- combina os três aspectos mencionados antes num único algoritmo;
- os valores de aptidão das soluções das populações externa e corrente são determinados usando apenas as soluções da população externa;
- todas as soluções da população externa participam no mecanismo de selecção;
- introduz um novo método baseado em nichos, de forma a preservar a diversidade da população.

O SPEA começa por gerar aleatoriamente uma população P_0 de tamanho POP e uma população externa $PE_0 = \emptyset$ de tamanho máximo POP_E . Em cada geração t , as soluções não dominadas da população P_t (nova elite) são copiadas para a nova população externa PE_t , que é actualizada. Esta actualização consiste em remover as soluções dominadas da população externa PE_t .

No entanto, continuando este processo ao longo das gerações, pode-se chegar a uma situação em que o número de soluções da população externa, depois de actualizada, é superior a seu tamanho máximo, POP_E . Desta forma, é necessário aplicar um mecanismo para escolher as "melhores" POP_E soluções da população externa PE_t para constituírem uma população externa apenas com POP_E soluções (as "melhores" soluções são as que estão menos agrupadas). Para determinar as POP_E "melhores" soluções, o SPEA utiliza um método de agrupamentos (ver secção 5.2.4 deste capítulo), que consiste em determinar os POP_E agrupamentos que melhor se distribuem pela frente não dominada. Por fim, a população externa da próxima geração, PE_{t+1} , é construída à custa das soluções que se encontram em cada um dos POP_E agrupamentos, em que cada agrupamento contribui com apenas uma solução.

Depois de obtida a nova população externa (a da próxima geração), a fase seguinte consiste em avaliar as soluções desta população e também da população corrente, o que é feito de modos diferentes. O valor de aptidão de uma solução x da população externa, $F(x)$, é um valor proporcional ao número $n(x)$ de soluções da população corrente que são

dominadas pela solução x : $F(x) = n(x) / (POP+1)$ (divide-se por $POP+1$ para garantir que $F(x) < 1$); desta forma, as soluções que dominam mais soluções têm maior valor de $F(x)$. O valor de aptidão de uma solução y da população corrente é obtido através da soma dos valores de aptidão das soluções da população externa que dominam a solução y : $F(y) = 1 + \sum_{x \in PE_t} F(x)$, em que x domina y ($F(y) > 1$); desta forma, os valores de aptidão das soluções da população corrente são sempre superiores aos valores de aptidão das soluções da população externa. Com este método de cálculo dos valores de aptidão, as soluções com menor valor de aptidão são as “melhores”.

Por fim, aplica-se o mecanismo de selecção por torneio usando os valores de aptidão determinados antes e os operadores genéticos cruzamento e mutação às soluções da população resultante da combinação das duas populações, $PE_t \cup P_t$, de tamanho $POP + POP_E$, para criar a população associada à geração seguinte, P_{t+1} , de tamanho POP .

5.3.2.2. Elitist Non-dominated Sorting Genetic Algorithm (NSGA-II)

Deb et al. (2000) sugeriram a abordagem denominada por “Elitist Non-dominated Sorting Genetic Algorithm” (NSGA-II), a qual usa explicitamente um mecanismo de preservação e diversificação das soluções (ver também Deb et al. (2002)).

O NSGA-II começa por gerar, aleatoriamente, uma população inicial P_0 de tamanho POP , a qual é depois classificada em diferentes níveis de dominância, donde resulta a identificação de várias frentes não dominadas. Desta forma, a cada solução é atribuído um valor de aptidão igual ao seu nível de dominância (1 para o melhor nível — ver secção 5.1 deste capítulo). A seguir, aplica-se um mecanismo de selecção por torneio baseado na técnica de multidões (a descrever à frente) de forma a preservar a diversidade da população e os operadores genéticos cruzamento e mutação, para se construir uma população de descendentes, Q_0 , de tamanho POP .

É depois aplicado um processo comum a cada geração t , que começa por combinar as duas populações, de progenitores e de descendentes, $R_t = P_t \cup Q_t$, de tamanho $2 \times POP$. As soluções de R_t são então classificadas por níveis de dominância, donde resulta a identificação de diferentes frentes não dominadas.

A nova população, P_{t+1} , é então construída a partir das várias frentes não dominadas obtidas (com $2 \times POP$ soluções), adicionando a primeira frente, depois a segunda, e assim sucessivamente, até que esta nova população, de tamanho POP , fique completa. Como o tamanho máximo de R_t é $2 \times POP$ e o tamanho da nova população é apenas POP , nem todas as frentes podem ser adicionadas a esta população, o que implica que as “piores” frentes sejam simplesmente ignoradas. Por outro lado, ao considerar-se a

última frente a adicionar, o número de soluções desta frente pode ser maior do que o número de soluções necessárias para completar a nova população, o que significa que nem todas as soluções desta frente podem ser adicionadas à nova população. Desta forma, para completar a nova população, o NSGA-II usa uma estratégia baseada em nichos para escolher as soluções da última frente que se localizam em regiões de menor densidade populacional nesta frente (a descrever à frente).

Por fim, é então construída a população de descendentes, Q_{t+1} , de tamanho POP, a partir da população P_{t+1} , usando selecção por torneio baseada na técnica de multidões e os operadores genéticos cruzamento e mutação.

O mecanismo de selecção por torneio baseado na técnica de multidões consiste em comparar duas soluções e declarar como vencedora aquela com melhor classificação (em termos de ordenação das frentes não dominadas — ver secção 2.4.4 deste capítulo) ou, tendo a mesma classificação, a que tiver maior distância de multidões (aquela que se localiza na área de menor densidade).

A distância de multidões de uma solução x pode ser calculada de várias formas, como, por exemplo, o contador de nicho (ver secção 5.2.3 deste capítulo) ou o perímetro do cubóide cujos vértices são formados pelas soluções mais próximas da solução x para todas as funções objectivo.

5.3.2.3. Strength Pareto Evolutionary Algorithm 2

Zitzler et al. (2001) apresentaram uma versão melhorada da abordagem "Strength Pareto Evolutionary Algorithm", a que chamaram SPEA2. As principais diferenças entre esta nova versão e a versão anterior são as seguintes:

- o mecanismo para determinar o valor de aptidão de uma solução usa uma estratégia baseada no número de soluções que a dominam e no número de soluções que são dominadas por ela;
- aquele mecanismo incorpora uma técnica adicional para estimação da densidade para discriminar as soluções com características idênticas em termos de dominância;
- o tamanho da população externa é constante, o que significa que esta população pode conter soluções dominadas;
- usa um método de truncatura aplicado à população externa para preservar as soluções extremas desta população (as soluções com os melhores valores para cada uma das funções objectivo);
- apenas as soluções da população externa participam no processo de selecção.

Os valores de aptidão das soluções das populações corrente P e externa PE são calculados da mesma forma. Na geração t , a cada solução x de qualquer das populações é atribuído um valor $F(x) = n(x)$, em que $n(x)$ é o número de soluções que são

dominadas por x . É então calculado o valor de aptidão bruta de x , $FB(x)$, como a soma dos valores de $n(y)$ de todas as soluções y das populações P_t e PE_t que dominam x ($F(x) = \sum_{y \in P_t \cup PE_t} F(y)$, em que y domina x). Desta forma, para uma solução x , $FB(x) = 0$ significa que x é uma solução não dominada, e $FB(x)$ com valores elevados significa que x é uma solução dominada por muitas soluções. Este mecanismo incorpora ainda uma técnica de estimação da densidade das soluções baseada na distância para o vizinho mais próximo, de forma a discriminar as soluções com o mesmo valor de aptidão bruta. Esta técnica é uma adaptação do método do k -ésimo vizinho mais próximo (Silverman (1986)), e consiste em tomar o inverso da distância, no espaço dos objectivos, para o k -ésimo vizinho mais próximo como estimação da densidade; a densidade de uma solução x , é $D(x) = 1 / (\sigma^k(x)+2)$, em que $\sigma^k(x)$ é a distância da solução x para o k -ésimo vizinho mais próximo (segundo os autores, um valor comum é $k = \sqrt{POP + POP_E}$, em que POP e POP_E são os tamanhos das populações P e PE , respectivamente). Finalmente, o valor de aptidão da solução x é $F(x) = FB(x) + D(x)$.

O tamanho da população externa é constante, o que significa que esta deve estar sempre completa, nem que seja com soluções dominadas. Quando o número de soluções não dominadas for superior ao tamanho da população externa, aplica-se um método de truncatura, que é semelhante ao método baseado em nichos usado no SPEA, para evitar que sejam removidas as soluções não dominadas extremas.

Este método começa por inserir todas as soluções não dominadas x (com $F(x) < 1$) das populações corrente P_t e externa PE_t , na população externa da próxima geração, PE_{t+1} . Se PE_{t+1} ficar completa, então termina este processo; caso contrário, aquela população está incompleta ($|PE_{t+1}| < POP_E$) ou está sobrelotada ($|PE_{t+1}| > POP_E$).

No primeiro caso, a população fica completa com as melhores soluções dominadas y ($F(y) \geq 1$), de acordo com os seus valores de aptidão, das anteriores populações externa PE_t e corrente P_t . No segundo caso, é utilizado um procedimento para truncar a população externa, que consiste em remover soluções da população externa sobrelotada até que esta fique com o tamanho adequado ($|PE_{t+1}| = POP_E$). Em cada iteração deste processo, a solução a ser removida é aquela que detém a distância mínima para outra solução; se existir mais do que uma solução nestas circunstâncias, entre estas escolhe-se aquela que detém a segunda menor distância, e assim sucessivamente.

O AG associado ao SPEA2 começa por gerar uma população inicial P_0 e criar uma população externa $PE_0 = \emptyset$, na geração $t = 0$. A seguir, determinam-se os valores de aptidão das soluções das duas populações, utilizando o processo já descrito nesta secção.

O passo seguinte consiste em construir a população secundária da próxima geração, PE_{t+1} , usando o procedimento já descrito também nesta secção.

A etapa seguinte consiste em verificar se a condição de paragem é satisfeita. Se sim, então todas as soluções não dominadas da população externa, PE_{t+1} (esta população pode conter também soluções dominadas), são inseridas num conjunto A e o algoritmo termina. Caso contrário, aplica-se um mecanismo de selecção por torneio com substituição sobre a população PE_{t+1} , e de seguida aplicam-se os operadores genéticos cruzamento e mutação às soluções escolhidas pelo mecanismo de selecção. Por fim, repete-se todo o processo para a próxima geração ($t = t + 1$).