

3 Técnicas de Inteligência Computacional

3.1. Introdução

Este capítulo resume os principais conceitos sobre as técnicas de inteligência computacional empregadas neste trabalho. Primeiramente, fornece-se uma breve explicação sobre o princípio de funcionamento dos Algoritmos Genéticos (AG), descrevendo suas partes principais e seus parâmetros de evolução. Em seguida, apresenta-se uma introdução sobre os conceitos básicos de Números *Fuzzy*, descrevendo a teoria de aritmética intervalar que é a base para a definição dos operadores de números *fuzzy*.

3.2. Algoritmos Genéticos

Essencialmente, Algoritmos Genéticos é um método de busca e otimização [12] [13] [43] [44] [45] que tem sua inspiração nos conceitos da teoria de seleção natural das espécies proposta por Darwin.

Os sistemas desenvolvidos a partir deste princípio são utilizados para procurar soluções de problemas complexos ou com espaço de soluções muito grande (espaço de busca), o que os tornam problemas de difícil modelagem e solução quando se aplicam métodos de otimização convencionais.

Estes algoritmos são inspirados nos processos genéticos de organismos biológicos para procurar soluções ótimas ou sub-ótimas. Para tanto, procede-se da seguinte maneira: codifica-se cada possível solução de um problema em uma estrutura chamada de "cromossomo", que é composta por uma cadeia de bits ou símbolos. Estes cromossomos representam indivíduos, que são evoluídos ao longo de várias gerações, de forma similar aos seres vivos, de acordo com os princípios de seleção natural e sobrevivência dos mais aptos, descritos pela primeira vez por Charles Darwin em seu livro "A Origem das Espécies". Emulando estes

processos, os algoritmos genéticos são capazes de "evoluir" soluções de problemas do mundo real.

Os cromossomos, ou indivíduos são então submetidos a um processo evolucionário que envolve avaliação, seleção, recombinação (*crossover*) e mutação. Após vários ciclos de evolução a população deverá conter indivíduos mais aptos. Os algoritmos genéticos utilizam uma analogia direta deste fenômeno de evolução na natureza, onde cada indivíduo representa uma possível solução para um problema dado. A cada indivíduo atribui-se um valor de adaptação: sua aptidão, que indica quanto a solução representada por este indivíduo é boa em relação às outras soluções da população. Desta maneira o termo *População* refere-se ao conjunto de todas as soluções com as quais trabalha o sistema. Aos indivíduos mais adaptados é dada a oportunidade de se reproduzirem mediante cruzamentos com outros indivíduos da população, produzindo descendentes com características de ambas as partes. A mutação também tem um papel significativo, ao introduzir na população novos indivíduos gerados de maneira aleatória.

O processo de evolução começa com a criação aleatória dos indivíduos que formarão a população inicial. A partir de um processo de seleção baseado na aptidão de cada indivíduo, são escolhidos indivíduos para a fase de reprodução que cria novas soluções utilizando-se para isto um conjunto de operadores genéticos. Deste modo, a aptidão do indivíduo determina o seu grau de sobrevivência e, assim, a possibilidade de que o cromossomo possa fazer parte das gerações seguintes.

O procedimento básico de um algoritmo genético é resumido na Figura 5 [46].

```
Início  
 $t \leftarrow 1$   
inicializar população  $P(t)$   
avaliar população  $P(t)$   
enquanto (não condição_de_fim) faça  
 $t \leftarrow t+1$   
selecionar população  $P(t)$  a partir de  $P(t-1)$   
aplicar operadores genéticos  
avaliar população  $P(t)$   
fim enquanto  
fim
```

Figura 5 – Procedimento básico do algoritmo genético

Para determinar o final da evolução pode-se fixar o número de gerações, o número de indivíduos criados, ou ainda condicionar o algoritmo à obtenção de uma solução satisfatória, isto é, quando atingir um ponto ótimo. Outras condições para a parada incluem o tempo de processamento e o grau de similaridade entre os elementos numa população (convergência).

As seções seguintes apresentam em mais detalhes cada um dos componentes de um algoritmo genético.

3.2.1. Representação

A representação é um aspecto fundamental na modelagem de um algoritmo genético para a solução de um problema. Neste estágio define-se a estrutura do cromossomo, com os respectivos genes que o compõem, de maneira que este seja capaz de descrever todo o espaço de busca relevante do problema.

3.2.2. Codificação e Decodificação

A solução de um problema pode ser representada por um conjunto de parâmetros (genes), unidos para formar uma cadeia de valores (cromossomo); a este processo chama-se codificação. As soluções (cromossomos) são codificadas

através de uma seqüência formada por símbolos de um sistema alfabético. Originalmente, utilizou-se o alfabeto binário (0, 1), porém, novos modelos de AGs codificam as soluções com outros alfabetos, como, por exemplo, com números reais [14].

A decodificação do cromossomo consiste basicamente na construção da solução real do problema a partir do cromossomo. O processo de decodificação constrói a solução para que esta seja avaliada pelo problema.

3.2.3. Avaliação

A avaliação permite ao algoritmo genético determinar sua proximidade à solução do problema. Ela é feita através de uma função que melhor representa o problema e tem por objetivo oferecer uma medida de aptidão de cada indivíduo na população corrente, que irá dirigir o processo de busca. Dado um cromossomo, a função de avaliação consiste em associar-se um valor numérico de “adaptação”, o qual supõe-se proporcional à sua "utilidade" ou "habilidade" do indivíduo representado em solucionar o problema em questão.

3.2.4. Operadores Genéticos

Os operadores mais conhecidos nos algoritmos genéticos são os de Reprodução, Cruzamento (*Crossover*) e Mutação.

Reprodução: refere-se ao processo de selecionar e copiar um determinado cromossomo para a população seguinte de acordo com sua aptidão. Isto significa que os cromossomos mais aptos, têm maior probabilidade de contribuir para a formação de um ou mais indivíduos da população seguinte. Existem basicamente os seguintes métodos: troca de toda população, troca de toda população com elitismo, onde todos os cromossomos são substituídos, sendo o cromossomo mais apto da população corrente copiado para população seguinte, e troca parcial da população (*steady state*), onde os M melhores indivíduos da população corrente são copiados para população seguinte [12] [13] [14].

Cruzamento: é um operador baseado na troca de partes dos cromossomos (pais), formando-se duas novas soluções (filhos). Este processo pode ser

observado no exemplo a seguir (Figura 6), onde a solução está codificada com alfabeto binário.

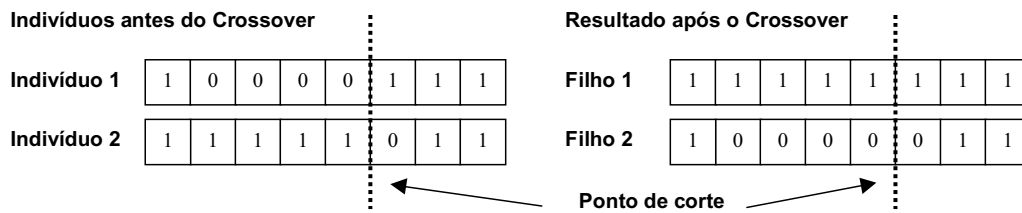


Figura 6 – Cruzamento de um ponto.

O ponto onde ocorre o corte para a realização do cruzamento é escolhido aleatoriamente; no exemplo da Figura 6 utilizou-se um único ponto, mas podem ser realizados cortes em mais de um ponto, caracterizando o *multi-point crossover* [12] [14] [15]. Para realizar o cruzamento, primeiro é necessária a escolha, por sorteio, dos cromossomos “pais”. Em seguida ocorre a realização ou não do cruzamento segundo um parâmetro, denominado taxa de cruzamento. Deste modo, de acordo com a taxa de cruzamento, pode ocorrer que os cromossomos “pais” sejam repassados sem modificação para a geração seguinte, criando “filhos” idênticos a eles.

A idéia do operador de Cruzamento é tirar vantagem (*exploit*) do material genético presente na população.

Mutação: é a troca aleatória do valor contido nos genes de um cromossomo por outro valor válido do alfabeto. No caso de alfabeto binário troca-se de 0 para 1 e vice-versa. Da mesma forma que para o cruzamento, utiliza-se uma taxa de mutação que, para cada bit da seqüência de caracteres, sorteia-se se ocorrerá ou não a mutação; no caso de ocorrência, o bit será trocado por outro valor válido pertencente ao alfabeto (Figura 7).

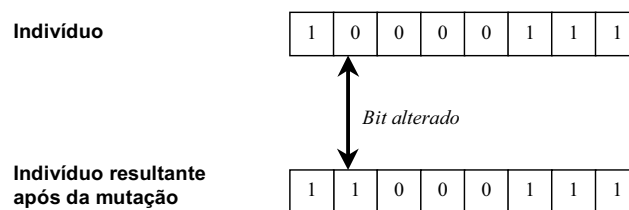


Figura 7 – Mutação.

A mutação garante a diversidade das características dos indivíduos da população e permite que sejam introduzidas informações que não estiveram presentes em nenhum dos indivíduos. Além disto, proporciona uma busca aleatória (*exploration*) no AG, oferecendo oportunidade para que mais pontos do espaço de busca sejam avaliados.

3.2.5. Parâmetros da Evolução

Os parâmetros que mais influenciam o desempenho do algoritmo genético são:

Tamanho da População: afeta o desempenho global e a eficiência dos Algoritmos Genéticos. Uma população muito pequena oferece uma pequena cobertura do espaço de busca, causando uma queda no desempenho. Uma população suficientemente grande fornece uma melhor cobertura do domínio do problema e previne a convergência prematura para soluções locais. Entretanto, com uma grande população tornam-se necessários recursos computacionais maiores, ou um tempo maior de processamento do problema. Logo, deve-se buscar um ponto de equilíbrio no que diz respeito ao tamanho escolhido para a população.

Taxa de Cruzamento: probabilidade de um indivíduo ser recombinado com outro. Quanto maior for esta taxa, mais rapidamente novas estruturas serão introduzidas na população. Entretanto, isto pode gerar um efeito indesejável, pois a maior parte da população será substituída, causando assim perda de variedade genética, podendo ocorrer perda de estruturas de alta aptidão e convergência a uma população com indivíduos extremamente parecidos, indivíduos estes de solução boa ou não. Com um valor baixo, o algoritmo pode-se tornar muito lento para oferecer uma resposta aceitável.

Taxa de Mutação: probabilidade do conteúdo de um gene do cromossomo ser alterado. A taxa de mutação previne que uma dada população fique estagnada em um valor, além de possibilitar que se chegue em qualquer ponto do espaço de busca. Porém deve-se evitar uma taxa de mutação muito alta, uma vez que esta

pode tornar a busca essencialmente aleatória, prejudicando fortemente a convergência para uma solução ótima.

Intervalo de Geração: controla a porcentagem da população que será substituída durante a próxima geração (substituição total, substituição com elitismo, substituição dos piores indivíduos da população atual, substituição parcial da população sem duplicatas). Esse número de indivíduos substituídos também é conhecido como GAP.

Número de gerações: representa o número total de ciclos de evolução de um Algoritmo Genético, sendo este um dos critérios de parada do algoritmo genético. Um número de gerações muito pequeno causa uma queda no desempenho: um valor grande faz necessário um tempo maior de processamento, mas fornece uma melhor cobertura do domínio do problema, evitando a convergência para soluções locais.

3.3. Teoria de Conjuntos *Fuzzy*

Dois dos principais aspectos da imperfeição da informação são a imprecisão e a incerteza. As teorias mais conhecidas para tratar de imprecisão e incerteza são, respectivamente, a teoria dos conjuntos e a teoria de probabilidades. Estas teorias, embora muito úteis, nem sempre conseguem captar a riqueza da informação fornecida por seres humanos. A teoria dos conjuntos não é capaz de tratar o aspecto vago da informação fornecida por seres humanos e a teoria de probabilidades é mais adequada para o tratamento de informações coletadas através de repetidas medições do que aquelas fornecidas por seres humanos

A teoria dos conjuntos *fuzzy* foi inicialmente introduzida por Zadeh [47] quando ele observou a impossibilidade de modelar sistemas com fronteiras mal definidas através das abordagens matemáticas rígidas e precisas dos métodos clássicos.

A transformação entre probabilidade e possibilidade vem sendo estudada e discutida há vários anos [48] [49] [50] [51]. Estes estudos examinam os princípios que devem ser cumpridos para satisfazer essa transformação, buscando estabelecer relações entre o grau de pertinência e a probabilidade. Uma visão mais ampla deste tema pode ser encontrada em [48] [49] [52] [53] [54] [55].

A teoria de conjuntos *fuzzy* proporciona uma estrutura matemática que permite trabalhar com a imprecisão e a incerteza da informação fornecida por seres humanos.

Esta teoria tem sido cada vez mais usada em sistemas que utilizam informações fornecidas por seres humanos e tem produzido bons resultados nas mais variadas aplicações [16] [17] [18].

A teoria dos conjuntos *fuzzy*, quando utilizada juntamente com conceitos de lógica, resulta nos chamados sistemas de inferência *fuzzy*. No entanto, quando usada para efetuar operações aritméticas, os conjuntos *fuzzy* são conhecidos como números *fuzzy*.

3.3.1. Conjunto *Fuzzy*

Conjunto *fuzzy* é a ponte que liga o conceito impreciso à sua modelagem numérica, atribuindo-se a cada elemento do universo um valor entre 0 e 1, que representa o grau de pertinência deste indivíduo ao conjunto *fuzzy*.

Um conjunto *fuzzy* F definido no universo de discurso U pode ser representado como um conjunto de pares ordenados de um elemento genérico x e seu grau de pertinência, μ . Este valor de pertinência é obtido mediante uma função de pertinência que mapeia os elementos de U para o intervalo $[0, 1]$, conforme descrito pela equação (20).

$$F = \{(x, \mu(x)) / x \in U\} \quad (20)$$

A Figura 8 ilustra as componentes de um conjunto *fuzzy*. O eixo x corresponde aos números reais que constituem o domínio do conjunto *fuzzy*, o eixo y , com valores entre 0 e 1, representa o grau de pertinência ao conjunto, e a curva representa a função de pertinência do conjunto, que conecta cada elemento do domínio com o seu grau de pertinência [16] [17] [56].

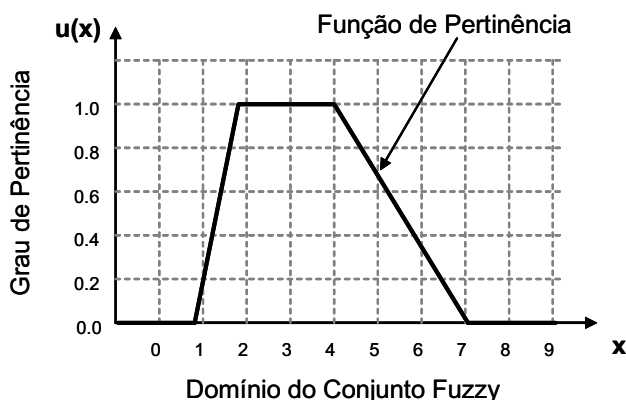


Figura 8 - Componentes de um conjunto *fuzzy*

3.3.2. Conjunto Singleton

Um conjunto *fuzzy* é chamado de singleton se seu suporte é um único ponto em U , com grau de pertinência igual a 1, $\mu(x) = 1$.

A Figura 9 ilustra um conjunto singleton de domínio igual a 4.

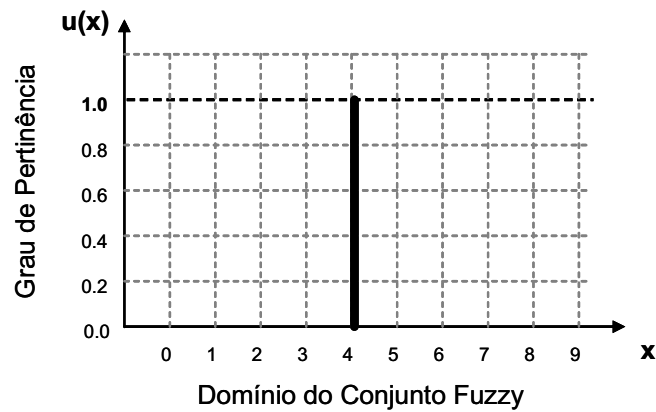


Figura 9 – Conjunto singleton

3.3.3. α -Cut de um Conjunto *Fuzzy*

O α -cut de um conjunto *fuzzy* A é o conjunto A_α que contém todos os elementos do universo de discurso U que possuem graus de pertinência $\mu_A(x)$ acima de um certo valor de α .

A equação (21) mostra a definição deste conjunto.

$$A_\alpha = \{x \in U \mid \mu_A(x) \geq \alpha\}, \quad \alpha \geq 0 \quad (21)$$

A Figura 10 ilustra o α -cut de um conjunto *fuzzy* com $\alpha = 0.2$.

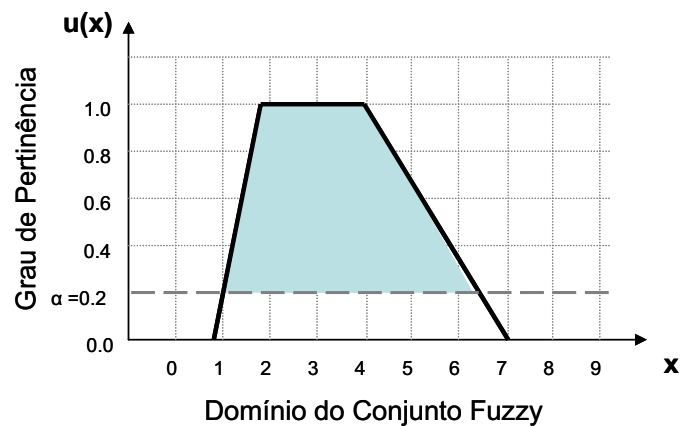


Figura 10 – α -Cut de um conjunto *fuzzy*.

3.4. Números *Fuzzy*

O número *fuzzy* [16] [17] [57] [58] é um conjunto *fuzzy* definido em um universo de discurso sobre os números reais, $U \in R$.

A operação de α -cut é também aplicada para os números *fuzzy*. Denota-se como A_α o intervalo α -cut para um número *fuzzy* A . A_α é um intervalo crisp definido como:

$$A_\alpha = [a_1^{(\alpha)}, a_3^{(\alpha)}] \quad \forall \alpha \in [0,1], \quad a_1, a_3, a_1^{(\alpha)}, a_3^{(\alpha)} \in R \quad (22)$$

Com esta definição é possível conhecer qualquer intervalo crisp dentro do número *fuzzy*, correspondente a um α -cut qualquer, Figura 11.

Para que um conjunto *fuzzy* seja definido como um número *fuzzy*, este deve cumprir as seguintes condições:

- Estar definido nos números reais;
- A função de pertinência deve ser contínua;
- O conjunto *fuzzy* deve ser normalizado;
- O conjunto *fuzzy* deve ser convexo.

Logo, um número *fuzzy* deverá ser normalizado e convexo, onde a condição de normalização implica que o máximo valor de pertinência é 1.

$$\exists x \in R, \quad \mu_A(x) = 1 \quad (23)$$

A condição de convexidade significa que a linha traçada por um α -cut é contínua e o intervalo α -cut satisfaz a relação a seguir. Seja o intervalo A_α definido pela equação (24):

$$A_\alpha = [a_1^{(\alpha)}, a_3^{(\alpha)}] \quad (24)$$

Logo:

$$(\alpha' < \alpha) \Rightarrow (a_1^{(\alpha')} \leq a_1^{(\alpha)}, \quad a_3^{(\alpha')} \geq a_3^{(\alpha)}) \quad (25)$$

A condição de convexidade também pode ser escrita como:

$$(\alpha' < \alpha) \Rightarrow (A_\alpha \subset A_{\alpha'}) \quad (26)$$

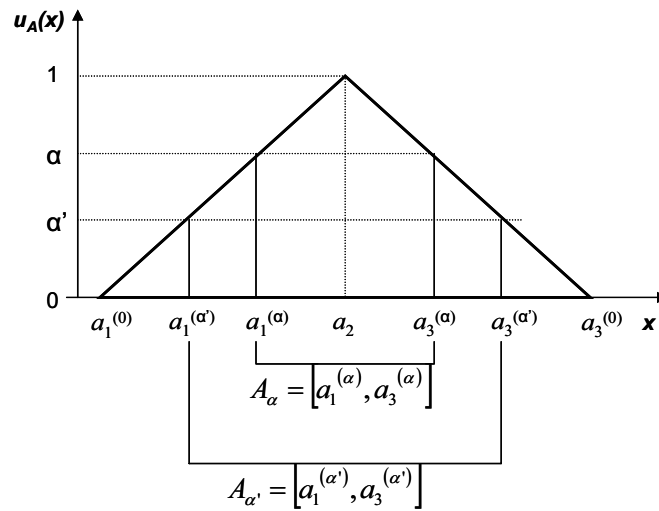


Figura 11 – Intervalo α -Cut de um número *fuzzy* ($\alpha' < \alpha$) $\rightarrow (A_\alpha \subset A_{\alpha'})$

3.4.1. Número *Fuzzy* Triangular

Neste número *fuzzy* a função de pertinência é triangular e definida por:

$$\mu_A(x) = \begin{cases} 0, & x < a_1 \\ \frac{x - a_1}{a_2 - a_1}, & a_1 \leq x \leq a_2 \\ \frac{a_3 - x}{a_3 - a_2}, & a_2 \leq x \leq a_3 \\ 0, & x > a_3 \end{cases} \quad (27)$$

A Figura 12 ilustra um número *fuzzy* triangular: no eixo x estão os parâmetros que definem o triângulo (a_1 , a_2 e a_3) e no eixo y está representado o grau de pertinência para cada valor de x .

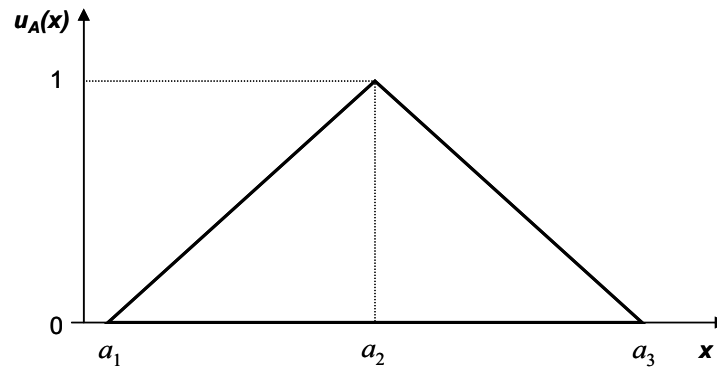


Figura 12 – Número *fuzzy* triangular $A = [a_1, a_2, a_3]$

A seguir define-se o intervalo α -cut para um número *fuzzy*.

Seja A_α um intervalo crisp de um número *fuzzy* obtido através de uma operação α -cut, $\forall \alpha \in [0,1]$. Da equação (27) tem-se:

$$\frac{a_1^{(\alpha)} - a_1}{a_2 - a_1} = \alpha, \quad \frac{a_3 - a_3^{(\alpha)}}{a_3 - a_2} = \alpha$$

Assim:

$$a_1^{(\alpha)} = (a_2 - a_1)\alpha + a_1$$

$$a_3^{(\alpha)} = -(a_3 - a_2)\alpha + a_3$$

Logo:

$$A_\alpha = [a_1^{(\alpha)}, a_3^{(\alpha)}]$$

$$A_\alpha = [(a_2 - a_1)\alpha + a_1, -(a_3 - a_2)\alpha + a_3] \quad (28)$$

3.4.2. Número *Fuzzy* Trapezoidal

Para definir este número *fuzzy* são necessários quatro parâmetros (a_1, a_2, a_3, a_4) e sua função de pertinência é dada pela equação (29):

$$\mu_A(x) = \begin{cases} 0, & x < a_1 \\ \frac{x - a_1}{a_2 - a_1}, & a_1 \leq x \leq a_2 \\ 1, & a_2 \leq x \leq a_3 \\ \frac{a_4 - x}{a_4 - a_3}, & a_3 \leq x \leq a_4 \\ 0, & x > a_4 \end{cases} \quad (29)$$

A Figura 13 ilustra um número *fuzzy* trapezoidal. No eixo x estão apresentados os parâmetros que definem o número *fuzzy* trapezoidal (a_1, a_2, a_3 e a_4), onde a_2 e a_3 correspondem aos valores de maior possibilidade, e no eixo y está representado o grau de pertinência para cada valor de x . O número *fuzzy* trapezoidal é utilizado quando a variável em análise possui uma faixa de variação onde todos os números dentro desta faixa possuem a mesma possibilidade de ocorrência.

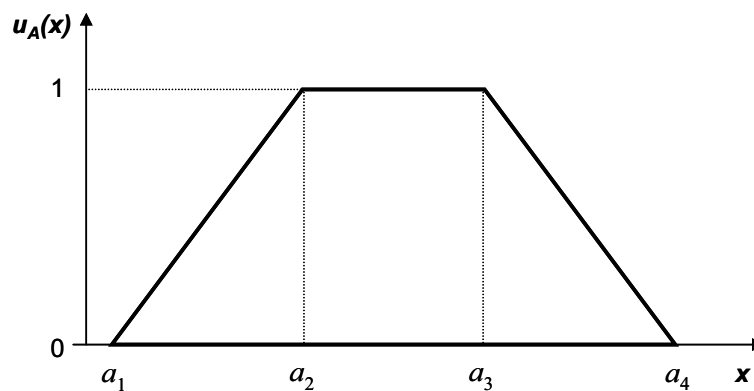


Figura 13 – Número *fuzzy* trapezoidal $A = (a_1, a_2, a_3, a_4)$

A equação (30) descreve o intervalo α -cut para este número *fuzzy*.

$$\forall \alpha \in [0,1]$$

$$\frac{a_1^{(\alpha)} - a_1}{a_2 - a_1} = \alpha, \quad \frac{a_4 - a_4^{(\alpha)}}{a_4 - a_3} = \alpha$$

Assim:

$$a_1^{(\alpha)} = (a_2 - a_1)\alpha + a_1$$

$$a_4^{(\alpha)} = -(a_4 - a_3)\alpha + a_4$$

Logo:

$$A_\alpha = [a_1^{(\alpha)}, a_4^{(\alpha)}]$$

$$A_\alpha = [(a_2 - a_1)\alpha + a_1, -(a_4 - a_3)\alpha + a_4] \quad (30)$$

Observe que quando $a_2 = a_3$, o número *fuzzy* trapezoidal coincide com um número *fuzzy* triangular.

3.4.3. Aritmética Intervalar

O conceito de números *fuzzy* pode ser apresentado de diversas maneiras. Neste trabalho, um número *fuzzy* é considerado como uma extensão do conceito de intervalo de confiança. Esta extensão baseia-se em uma idéia natural e simples: ao invés de se considerar o intervalo de confiança em um único nível, ele é considerado em vários níveis entre 0 e 1. O máximo intervalo de confiança é considerado igual a 1 e o mínimo igual a 0. O nível de pertinência α , para $\alpha \in [0,1]$, fornece um intervalo de confiança $A_\alpha = [a_1^{(\alpha)}, a_3^{(\alpha)}]$, que é uma função monotônica decrescente de α , isto é:

para todo $\alpha, \alpha' \in [0,1]$

se $(\alpha' > \alpha) \Rightarrow (A_{\alpha'} \subset A_\alpha)$

ou

$(\alpha' > \alpha) \Rightarrow [a_1^{(\alpha')}, a_3^{(\alpha')}] \subset [a_1^{(\alpha)}, a_3^{(\alpha)}]$

Desta forma, pode-se aplicar a aritmética intervalar para definir as operações com números *fuzzy*, onde cada intervalo possibilístico definido por um α -cut pode ser tratado independentemente.

A seguir são apresentadas brevemente as definições das principais operações intervalares, considerando A e B como números expressados como intervalos [59] [60] [61], de modo que:

$$\forall a_1, a_3, b_1, b_3 \in R$$

$$A = [a_1, a_3] \quad B = [b_1, b_3]$$

1. **Adição:** a adição de dois intervalos definidos nos números reais é:

$$A + B = [a_1, a_3] \quad (+) \quad [b_1, b_3] = [a_1 + b_1, \quad a_3 + b_3] \quad (31)$$

2. **Subtração:**

$$A - B = [a_1, a_3] \quad (-) \quad [b_1, b_3] = [a_1 - b_3, \quad a_3 - b_1] \quad (32)$$

3. **Multiplicação:**

$$A \bullet B = [a_1, a_3] \quad (\bullet) \quad [b_1, b_3]$$

$$A \bullet B = [\min\{a_1 \bullet b_1, a_1 \bullet b_3, a_3 \bullet b_1, a_3 \bullet b_3\}, \quad \max\{a_1 \bullet b_1, a_1 \bullet b_3, a_3 \bullet b_1, a_3 \bullet b_3\}] \quad (33)$$

4. **Divisão:**

$$A / B = [a_1, a_3] \quad (/) \quad [b_1, b_3]$$

$$A / B = [\min\{a_1 / b_1, a_1 / b_3, a_3 / b_1, a_3 / b_3\}, \quad \max\{a_1 / b_1, a_1 / b_3, a_3 / b_1, a_3 / b_3\}] \quad (34)$$

excluindo o caso de $b_1 = 0$ ou $b_3 = 0$

5. **Inversa de um intervalo:**

$$A^{-1} = 1/A = [a_1, a_3]^{-1}$$

$$A^{-1} = [\min\{1/a_1, 1/a_3\}, \quad \max\{1/a_1, 1/a_3\}] \quad (35)$$

excluindo o caso de $a_1 = 0$ ou $a_3 = 0$

6. **Multiplicação de um intervalo por um escalar:**

$$\forall \lambda \in R$$

se $\lambda > 0$

$$\lambda A = \lambda [a_1, a_3] = [\lambda a_1, \quad \lambda a_3] \quad (36)$$

$$\begin{aligned} & \text{se } \lambda < 0 \\ & \lambda A = \lambda[a_1, a_3] = [\lambda a_3, \lambda a_1] \end{aligned} \quad (37)$$

3.4.4. Operações com Números *Fuzzy*

A seguir define-se brevemente a forma de realizar as operações com números *fuzzy*.

1. **Operações com o intervalo α -cut:** os intervalos α -cut de um número *fuzzy* $A = [a_1, a_3]$ podem ser referenciados como um conjunto crisp.

A_α é um intervalo crisp definido por:

$$A_\alpha = [a_1^{(\alpha)}, a_3^{(\alpha)}] \quad \forall \alpha \in [0,1], \quad a_1, a_3, a_1^{(\alpha)}, a_3^{(\alpha)} \in R$$

Logo, as operações de intervalos vistas na seção anterior podem ser aplicadas para o intervalo α -cut A_α .

Se o intervalo α -cut B_α de um número *fuzzy* B é definido por:

$$B = [b_1, b_3], \quad b_1, b_3 \in R$$

$$B_\alpha = [b_1^{(\alpha)}, b_3^{(\alpha)}] \quad \forall \alpha \in [0,1], \quad b_1^{(\alpha)}, b_3^{(\alpha)} \in R$$

então, as operações entre A_α e B_α podem ser descrita da seguinte forma:

$$A_\alpha + B_\alpha = [a_1^{(\alpha)}, a_3^{(\alpha)}] \quad (+) \quad [b_1^{(\alpha)}, b_3^{(\alpha)}] = [a_1^{(\alpha)} + b_1^{(\alpha)}, a_3^{(\alpha)} + b_3^{(\alpha)}]$$

$$A_\alpha - B_\alpha = [a_1^{(\alpha)}, a_3^{(\alpha)}] \quad (-) \quad [b_1^{(\alpha)}, b_3^{(\alpha)}] = [a_1^{(\alpha)} - b_3^{(\alpha)}, a_3^{(\alpha)} - b_1^{(\alpha)}]$$

Isto é, para cada grau de pertinência do número *fuzzy* estabelece-se um α -cut, criando-se intervalos α -cut. Logo, as operações com o número *fuzzy* serão realizadas para cada nível (cada α -cut) segundo a teoria intervalar. Esta metodologia é também aplicável às operações de multiplicação, divisão e outras.

Pode-se concluir que as operações com números *fuzzy* seguem as mesmas propriedades das operações intervalares. A diferença é que, com os números *fuzzy*, as operações são realizadas para cada nível de pertinência, como se o número *fuzzy* fosse “fatiado” em diversos intervalos [58] [62] [63].

2. **Operação de números *fuzzy* - Solução Analítica:** na solução analítica de operações com números *fuzzy*, a função de pertinência do resultado é obtida da seguinte forma:

- Obtém-se os intervalos *fuzzy* dos operandos;
- Com estes intervalos *fuzzy* realiza-se a operação matemática desejada;
- Com o resultado da operação matemática (em intervalo *fuzzy*) escreve-se a função de pertinência do número *fuzzy* resultante.

A solução analítica será apresentada com um exemplo numérico para a soma de dois números *fuzzy*. A Figura 14 apresenta a soma de dois números *fuzzy* na forma analítica ($C=A+B$).

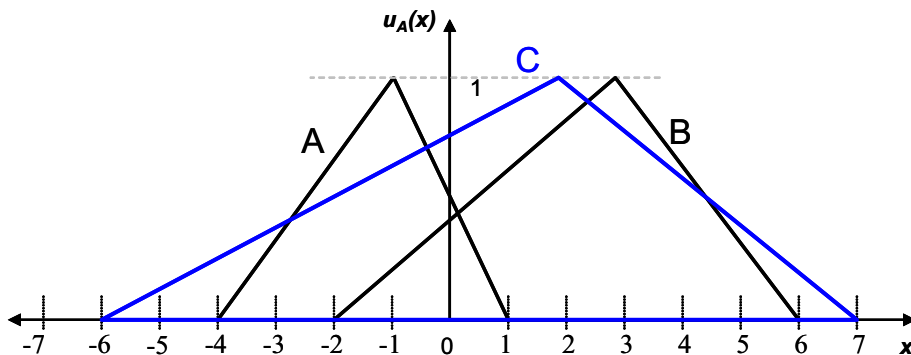


Figura 14 - Solução analítica da soma *fuzzy*

Os números **A** e **B** são definidos pelas seguintes funções:

$$\forall x \in R,$$

$$\mu_A(x) = \begin{cases} 0, & x \leq -4 \\ x/3 + 4/3, & -4 \leq x \leq -1 \\ -x/2 + 1/2, & -1 \leq x \leq 1 \\ 0, & x \geq 1 \end{cases}$$

$$\mu_B(x) = \begin{cases} 0, & x \leq -2 \\ x/5 + 2/5, & -2 \leq x \leq 3 \\ -x/3 + 6/3, & 3 \leq x \leq 6 \\ 0, & x \geq 6 \end{cases}$$

Para se obter o resultado da soma, o número *fuzzy* C , considera-se um nível de pertinência genérico α e escreve-se a função de pertinência de cada número para este nível genérico:

Número A:

O nível α (reta α) intercepta o número A em dois pontos, aqui chamados de a_1 e a_2 .

$$\alpha = a_1^\alpha / 3 + 4/3 \quad \text{e}$$

$$\alpha = -a_2^\alpha / 2 + 1/2$$

Resolvendo para $a_i^\alpha, i = 1, 2$, o intervalo de confiança de A para o nível α é dado por:

$$A_\alpha = [a_1^\alpha, a_2^\alpha] = [3\alpha - 4, -2\alpha + 1]$$

Número B:

O nível α (reta α) intercepta o número B em dois pontos, aqui chamados de b_1 e b_2 .

$$\alpha = b_1^\alpha / 5 + 2/5 \quad \text{e}$$

$$\alpha = -b_2^\alpha / 3 + 6/3$$

Resolvendo para $b_i^\alpha, i = 1, 2$, o intervalo de confiança para o nível α é dado por:

$$B_\alpha = [b_1^\alpha, b_2^\alpha] = [5\alpha - 2, -3\alpha + 6]$$

Realizando a operação de soma:

$$A + B = A_\alpha + B_\alpha = [a_1^\alpha, a_2^\alpha] + [b_1^\alpha, b_2^\alpha] = [a_1^\alpha + b_1^\alpha, a_2^\alpha + b_2^\alpha] =$$

$$A + B = A_\alpha + B_\alpha = [(3\alpha - 4) + (5\alpha - 2), (-2\alpha + 1) + (-3\alpha + 6)] = [8\alpha - 6, -5\alpha + 7]$$

Comparando termo a termo:

$$a_1^\alpha + b_1^\alpha = 8\alpha - 6$$

$$a_2^\alpha + b_2^\alpha = -5\alpha + 7$$

Fazendo as substituições de variáveis: $a_i^\alpha + b_i^\alpha = c_i^\alpha = x$ e $\alpha = \mu_C(x)$:

$$x = 8\mu_C(x) - 6$$

$$x = -5\mu_C(x) + 7$$

Os extremos, $\mu_C(x) = 0$, e os intervalos das funções são simplesmente somados. O resultado da operação final é o número *fuzzy* representado pela seguinte função de pertinência:

$$\mu_C(x) = \begin{cases} 0, & x \leq -6 \\ x/8 + 6/8, & -6 \leq x \leq 2 \\ -x/5 + 7/5, & 2 \leq x \leq 7 \\ 0, & x \geq 7 \end{cases}$$

Este procedimento é genérico e pode ser utilizado para qualquer tipo de operação com números *fuzzy*. Na prática, a solução analítica não é utilizada e sim a solução numérica exposta a seguir.

3. **Operação de números *fuzzy* - Solução Numérica:** na implementação computacional, o grau de pertinência $\alpha = [0,1]$ é discretizado em n níveis, por exemplo 10, e as operações de intervalo de confiança (aritmética intervalar) são realizadas para cada nível. A Figura 15 apresenta o número *fuzzy* como um conjunto de números intervalares, $[a_1^\alpha, a_3^\alpha]$, que são definidos para cada nível de pertinência α . Observe que, quando α cresce, isto é, o grau de confiança aumenta, a largura do intervalo diminui.

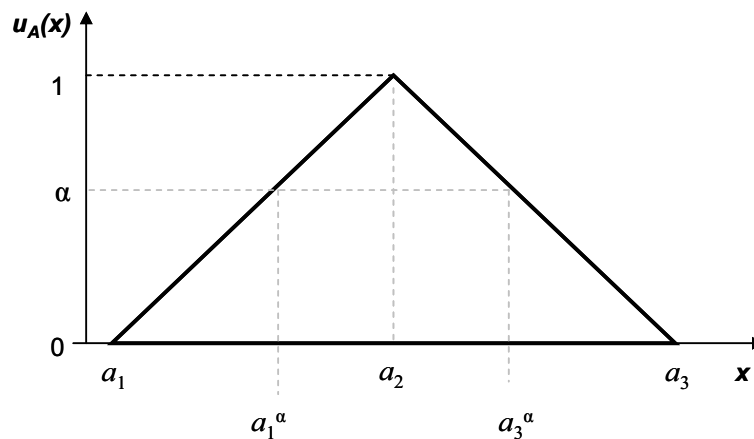


Figura 15 - Número *fuzzy* como intervalo de confiança

A associação entre intervalo de confiança e grau de pertinência forma o número *fuzzy*. Por exemplo, considere-se que um projeto devesse terminar entre os dias 15 e 25 e que foi efetivamente encerrado no dia 22. Ou seja, o intervalo de confiança é $[15,25]$, pode-se atribuir um grau de pertinência 0 para os extremos do intervalo (dias 15 e 25) e 1 para o dia correto (22), obtendo-se assim, por exemplo, um número *fuzzy* triangular. É importante ressaltar que a função de pertinência possui valores entre $[0,1] \subset R^+$, enquanto que a lógica booleana possui apenas os valores 0 e 1.