ADAPTAÇÃO DO ALGORITMO DE FORD-MOORE-BELLMAN PARA O PROBLEMA DE CAMINHO MÍNIMO *FUZZY* COM RESTRIÇÕES DE TEMPO

Fábio Hernandes
Departamento de Ciência da Computação - DECOMP
Universidade Estadual do Centro-Oeste – UNICENTRO
Caixa Postal 3010 – 85.015-430 – Guarapuava – PR
hernandes@unicentro.br

Wagner Santos de Oliveira[1]
Departamento de Ciência da Computação - DECOMP
Universidade Estadual do Centro-Oeste – UNICENTRO
Caixa Postal 3010 – 85.015-430 – Guarapuava – PR
darkghost199@hotmail.com

RESUMO

A teoria dos grafos é uma importante área da programação matemática, pois possui aplicações nas mais diversas áreas da Engenharia e da Computação, tais como: telecomunicações, transportes, manufaturas e redes de computadores. Neste trabalho é estudado o problema de caminho mínimo com restrições de tempo aplicado em grafos com parâmetros incertos, sendo as incertezas abordadas utilizando a teoria dos conjuntos *fuzzy*. Propõe-se um algoritmo, baseado no algoritmo clássico de Ford-Moore-Bellman, cuja solução é um conjunto *fuzzy* ordenado de caminhos não-dominados que satisfazem as restrições de tempo.

PALAVARAS CHAVE: teoria dos grafos; problema de caminho mínimo; teoria dos conjuntos *fuzzy*.

ABSTRACT

The graphs theory is an important area of mathematical programming, it has applications in several areas of Engineering and Computation, such as: telecommunications, transportation, manufacturing and computer networks. In this paper is studied the shortest path problem with time constraints applied in graphs with uncertain parameters, where the uncertainties are discussed using the fuzzy set theory. We present an algorithm, which is based on the classic Ford-Moore-Bellman algorithm, whose solution is a fuzzy ordered set of non-dominated paths to satisfy the time constraints.

KEYWORDS: graphs theory; shortest path problem; fuzzy set theory**.**

---

[1] Bolsista - Fundação Araucária-PR.

1. Introdução

Sendo *u* e *v* dois nós de um grafo orientado *G=(N,A)*, o *caminho mínimo* entre *u* e *v* é uma sequência de arcos que, passando por nós distintos, une *u* a *v* de forma a acumular o menor comprimento (distância). Para que haja um caminho mínimo entre dois nós é necessário que exista uma conexão entre eles (Goldbarg e Luna, 2005).

Na teoria dos grafos clássica existe um grande número de algoritmos eficientes para resolver este problema, pois os custos dos arcos têm valores precisos e com isso a comparação entre os custos/comprimentos dos caminhos é exata. Porém, em muitas aplicações reais da Engenharia e da Computação estes parâmetros (custos, capacidades, etc) não são naturalmente precisos, muitas vezes, por falta de informações. Fatores estes (incertezas, imprecisões e ambigüidades) que nem sempre podem ser ignorados na resolução de problemas.

Introduzida por Zadeh em 1965 (Zadeh 1965; 1968; 1978), a teoria de conjuntos *fuzzy* é uma ferramenta frequentemente utilizada na modelagem desses fatores. Diversos trabalhos nas áreas de Engenharia e Computação (problemas de controle, sistemas de suporte à decisão, reconhecimento de padrões e programação matemática) utilizam esta teoria (Shaw e Simões, 1988; Bellman e Zadeh, 1970; Bedzek, 1981; Takahashi e Yamakami, 2004; Hernandes, 2007).

Na literatura há diversos problemas que envolvem grafos com incertezas, dentre as quais destacam-se: caminho mínimo (Klein 1991; Lin e Chern,1993; Okada e Soper, 2000; Blue *et al*, 2002; Hernandes e Takahashi, 2004; Okada, 2004; Hernandes *et al* 2007; Hernandes *et al*, 2009); fluxo máximo (Chanas *et al*, 1995; Hernandes *et al*, 2006); fluxo de custo mínimo (Kim e Roush, 1982; Shih e Lee, 1999; Takahashi, 2004; Hernandes, 2007); dentre outros.

Dentre os problemas da programação matemática supracitados, o de *caminho mínimo com incertezas* (*PCM-fuzzy*) foi um dos primeiros a merecer atenção (Dubois e Prade, 1980; Klein, 1991; Lin e Chern, 1993, etc), haja vista que pode ser utilizado em modelos de telecomunicações, transporte, transmissão de energia, dentre outras aplicações.

Neste trabalho, propõe-se um algoritmo aplicado ao problema de caminho mínimo com restrições de tempo, sendo que os parâmetros dos arcos (custos e tempo) e as restrições de tempo nos nós são incertos (*fuzzy*), tratando-se portanto de uma complementação do trabalho de Hernandes *et al* (2009). Este algoritmo, baseado no algoritmo clássico de Ford-Moore-Bellman (Bellman,1958), tem como finalidade encontrar e ordenar um conjunto solução de caminhos não-dominados entre um par de nós do grafo que satisfazem as restrições de tempo dos nós. A fim de detectar os caminhos não-dominados, utilizou-se a definição de dominância aplicada em Okada e Soper (2000).

Os principais trabalhos de caminho mínimo *fuzzy* encontrados na literatura são: Dubois e Prade (1980), Okada e Soper (2000), Okada (2004), Nayeem e Pal (2005), Hernandes *et al* (2007) e Hernandes *et al* (2009). Um dos primeiros trabalhos da literatura, Dubois e Prade (1980) propõem a extensão dos algoritmos clássicos de Floyd e de Ford-Moore-Bellman, sendo verificado que ambos os algoritmos podem retornar comprimentos sem um caminho associado, problema este contornado, com o uso de dominância, por Klein (1991). Já o trabalho de Okada e Soper (2000) se destaca pela obtenção da solução final, que é composta por um conjunto de caminhos não-dominados, onde cada elemento da solução é um caminho não-dominado. Okada (2004) estudou a interatividade entre caminhos *fuzzy* e propôs um algoritmo que utiliza a teoria de possibilidade, introduzindo o conceito de grau de possibilidade de um arco pertencer a um caminho mínimo. No trabalho de Nayeem e Pal (2005) é proposto um algoritmo baseado no índice de aceitabilidade de Sengupta e Pal (2000), onde o usuário, segundo seu ponto de vista (otimista/pessimista), deve escolher o melhor caminho. Hernandes *et al* (2007) apresentam um algoritmo que trabalha com as incertezas até a obtenção da solução final, apresentando um conjunto solução de caminhos não-dominados, semelhante ao de Okada e Soper (2000), com a vantagem de poder ser aplicado em redes com parâmetros negativos, haja vista que é baseado no algoritmo de Ford-Moore-Bellman. Um dos trabalhos mais recentes da literatura, Hernandes *et al* (2009) é o único que aborda o problema de caminho mínimo *fuzzy* com restrições de tempo nos nós, sendo proposto um algoritmo, baseado no Dijkstra, que tem como solução final

um conjunto ordenado de caminhos não-dominados que satisfazem as restrições de tempo dos nós.

Os trabalhos supracitados apresentam algumas desvantagens que necessitam ser contornadas, tais como: encontram custos sem caminhos associados; apresentam um conjunto solução de caminhos não-dominados sem ordená-los; dentre outras. Considerando o PCM-*fuzzy* com restrições de tempo, o trabalho de Hernandes *et al* (2009) contorna tais desvantagens, além de resolver o problema com restrições de tempo. Porém, a desvantagem deste está no algoritmo base, neste caso o algoritmo de Dijkstra, pois não pode ser aplicado em redes com parâmetros, (p.e.: custos), negativos. Logo, como citado no início desta seção, este trabalho tem por objetivo apresentar um algoritmo para o problema de caminho mínimo *fuzzy* com restrições de tempo *fuzzy*, contornando a desvantagem do trabalho de Hernandes *et al* (2009).

Este trabalho está organizado da seguinte forma: na Seção 2 são apresentados alguns conceitos da teoria dos conjuntos *fuzzy*. A Seção 3 aborda o problema estudado e apresenta o algoritmo proposto. Os resultados computacionais estão na Seção 4. Conclusões e propostas futuras são discutidas na Seção 5.

2. Conceitos da Teoria dos Conjuntos *Fuzzy*

A teoria dos conjuntos *fuzzy* (Zadeh, 1965; Zadeh, 1968; Zadeh, 1978) foi introduzida com o objetivo de dar um tratamento matemático a termos linguísticos subjetivos, como, por exemplo, "aproximadamente" e "em torno de".

Nesta seção são apresentados alguns conceitos utilizados neste trabalho (Bassanezi e Barros, 2006; Pedrycz e Gomide, 1998).

2.1. Conceitos Básicos

Definição 1: Seja U um conjunto clássico. Um conjunto *fuzzy F* de *U* é caracterizado por uma função $\mu_F : U \to [0,1]$, denominada *função de pertinência* do conjunto *fuzzy*. O valor $\mu_F(x) \in [0,1]$ indica o grau com que o elemento *x* de *U* pertence ao conjunto *fuzzy F*; $\mu_F(x) = 0$ e $\mu_F(x) = 1$ indicam, respectivamente, a não pertinência e a pertinência completa de *x* ao conjunto *fuzzy F*.

Analisando formalmente, a definição de conjuntos *fuzzy* foi obtida simplesmente ampliando-se o contradomínio da função característica clássica, que é o conjunto $\{0,1\}$, para o intervalo [0,1]. Nesse sentido, pode-se dizer que um conjunto clássico é um caso particular de conjunto *fuzzy* cuja função de pertinência μ_F é a sua função característica *X*. Um conjunto clássico, na linguagem *fuzzy*, costuma ser denominado *conjunto crisp*.

Definição 2: Seja A um conjunto *fuzzy* de U e $\alpha \in [0,1]$. Um α*-corte* de A é o conjunto clássico de U definido por $[A]^\alpha = \{x \in U : \mu_A(x) \geq \alpha\}$ para $0 \leq \alpha \leq 1$. Se $[A]^\alpha = \{x \in U : \mu_A(x) > \alpha\}$, $[A]^\alpha$ é dito α*-corte forte*.

O nível zero de um conjunto *fuzzy* A é definido como sendo o menor conjunto (clássico) fechado de U que contém o conjunto suporte de A. Matematicamente, $[A]^0$ é o *fecho do suporte de* A e é indicado por *supp(A)*.

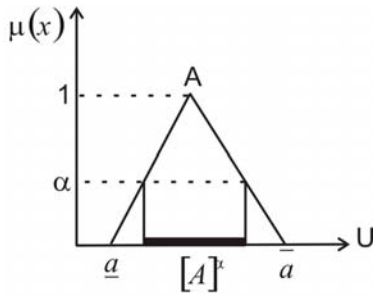

Figura 1 - Exemplo de α - corte.

2.2 Números *Fuzzy*

Definição 3: Um *número fuzzy triangular,* denotado por $\tilde{a} = (m_1, \delta, \beta)$, tem sua *função de pertinência* $\Re \rightarrow [0,1]$, $\mu_{\tilde{a}}(x)$, definida pela expressão:

$$\mu_{\tilde{a}}(x) = \begin{cases} 0, & \text{se } x < (m_1 - \delta) \\ \dfrac{x - (m_1 - \delta)}{\delta}, & \text{se } (m_1 - \delta) \leq x \leq m_1 \\ \dfrac{(m_1 + \beta) - x}{\beta}, & \text{se } m_1 < x \leq (m_1 + \beta) \\ 0, & \text{se } x > (m_1 + \beta) \end{cases}$$

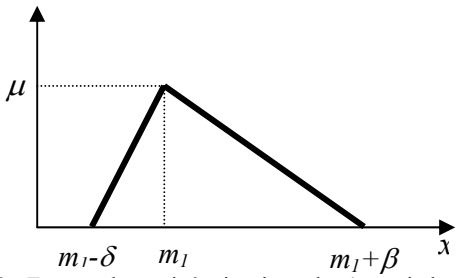

Figura 2 - Função de pertinência triangular (associada a $\tilde{a}$)

tal que:

m_1: valor modal (valor máximo da função de pertinência);
δ: espalhamento à esquerda; e
β: espalhamento à direita.

Notação: Os valores (m_1-δ) e (m_1+β) são denominados de limitante inferior e limitante superior de $\tilde{a}$, respectivamente.

Definição 4: Sejam $\tilde{a}$ e $\tilde{b}$ números *fuzzy* triangulares, $\tilde{a} = (m_1, \delta_1, \beta_1)$ e $\tilde{b} = (m_2, \delta_2, \beta_2)$, a *soma fuzzy* é denotada por $\tilde{a} \oplus \tilde{b}$ e definida pela expressão:

$$\tilde{a} \oplus \tilde{b} = (m_1, \delta_1, \beta_1) \oplus (m_2, \delta_2, \beta_2) = (m_1 + m_2, \delta_1 + \delta_2, \beta_1 + \beta_2)$$

Definição 5: Seja um grafo *G=(N,A)* com custos $\tilde{c} \in \Re^n$, $\tilde{c} = \{\tilde{c}_{ij}\}$, associados aos seus arcos. Sejam dois subgrafos T^1 e T^2, $T^1 \neq T^2$. Diz-se que T^1 tem um *grau de possibilidade* de ser menor do que T^2 dado por (Okada, 2004):

$$\tilde{w} = Poss\left(\sum_{ij \in T^1} \tilde{c}_{ij} \leq \sum_{ij \in T^2} \tilde{c}_{ij} \right) = \sup_{u \leq v} \min\{\mu_{\tilde{T}^1}(u), \mu_{\tilde{T}^2}(v)\}$$

sendo que $\tilde{T}^1$ e $\tilde{T}^2$ representam as somas dos custos dos arcos pertencentes aos caminhos T^1 e T^2, respectivamente.

Definição 6: Sejam $\tilde{a} = (m_1, \delta_1, \beta_1)$ e $\tilde{b} = (m_2, \delta_2, \beta_2)$ dois números *fuzzy* triangulares, então $\tilde{a}\ \pi\ \tilde{b}$ ($\tilde{a}$ *domina* $\tilde{b}$) se, e somente se, $m_1 \leq m_2$, $m_1 - \delta_1 \leq m_2 - \delta_2$, $m_1 + \beta_1 \leq m_2 + \beta_2$ e $\tilde{a} \neq \tilde{b}$ (Okada e Soper, 2000).

3. O Problema do Caminho Mínimo com Incertezas e Restrições de Tempo

O problema a ser considerado neste trabalho, além de possuir os dois parâmetros dos arcos (custo e tempo) incertos, também possui uma restrição de tempo em cada um dos nós, isto é, a soma dos tempos dos arcos que pertencem ao caminho, entre os nós 1 e *j*, não pode ultrapassar o tempo máximo estipulado na restrição do referido nó *j*. Considerando que os parâmetros dos arcos e as restrições dos nós são números *fuzzy* triangulares, as verificações das restrições são satisfeitas utilizando a medida de possibilidade, definida da seguinte forma:

Definição 7: Seja $\tilde{t}_{i(1,j)}$ o tempo associado ao caminho com o *i-ésimo* rótulo (etiqueta), entre os nós 1 e *j*, dentre todos os existentes. Sejam $\gamma \in [0,1]$ e Rt_j a restrição de tempo do nó *j*, então a restrição de tempo é satisfeita se $Pt = Poss\left(\tilde{t}_{i(1,j)} \leq Rt_j\right) > \gamma$.

Obs: Vale ressaltar que a restrição de tempo da Definição 7 é satisfeito no nível de γ de acordo com a Definição 2, ficando a cargo do usuário decidir qual é este nível desejável.

Com relação à ordenação do conjunto solução é utilizado o critério proposto por Hernandes *et al* (2009), que trata de uma combinação convexa entre as medidas de possibilidades do caminho com relação ao tempo P_t e com relação ao custo P_c ($\alpha P_c + (1-\alpha) P_t$), tal que $\alpha \in [0,1]$.

3.1 Algoritmo Proposto

O algoritmo proposto tem por objetivo encontrar e ordenar todos os caminhos não-dominados entre os nós 1 e *i* (*i=2,3,...,n*) que satisfazem as restrições de tempo de cada nó. Tal algoritmo trata de uma adaptação do algoritmo de Ford-Moore-Bellman (Passos 1, 2 e 3).

3.1.1 Algoritmo

<u>Informações sobre o algoritmo</u>:
N: conjunto dos nós;
it: contador de iterações;
$(m+\beta)^i$: espalhamento à direita ($m+\beta$) do custo do arco (*i,j*) - limitante superior;
$\tilde{c}_{ji}$: custo do arco *(j,i)*;
$\tilde{t}_{ji}$: tempo do arco *(j,i)*;
Rt_j : restrição de tempo do nó *j;*
$\tilde{c}^{it}_{(i,k)}$: custo do caminho entre o nó 1 e *i,* com o rótulo *k*, na iteração *it;*
$\tilde{t}^{it}_{(i,k)}$: tempo do caminho entre o nó 1 e *i,* com o rótulo *k*, na iteração *it;*
M: número grande que substitui o infinito; e
Γ_i^{-1} : conjunto dos nós predecessores de *i*.

Algoritmo Proposto

PASSO 0: *Início.*

i) $\tilde{c}^0_{(1,1)} \leftarrow (0,0,0)$.

ii) $\tilde{c}^0_{(j,1)} \leftarrow (M+2,1,1),\ j = 2,3,...,n;$

tal que:

- *n*: número de nós;

- $M = \sum_{i=1}^{E} \left| (m+\beta)^i \right|$; e
- E: número de arcos.

iii) $it \leftarrow 1$.

PASSO 1: *Detecção dos caminhos, verificação da dominância e eliminação de caminhos.*

1) *Determinação dos caminhos*

i) $\tilde{c}^{it}_{(1,1)} \leftarrow (0,0,0)$; $\tilde{t}^{it}_{(1,1)} \leftarrow (0,0,0)$

ii) $\forall j \in \Gamma_i^{-1},\ i = 1,2,3,...,n,$ faça:

- $\tilde{c}^{it}_{(i,k1)} \leftarrow \tilde{c}^{it-1}_{(j,k2)} \oplus \tilde{c}_{ji}$ (cálculo dos custos dos caminhos, sendo *k1* e *k2* os rótulos dos caminhos)
- $\tilde{t}^{it}_{(i,k1)} \leftarrow \tilde{t}^{it-1}_{(j,k2)} \oplus \tilde{t}_{ji}$ (cálculo dos tempos dos caminhos)

2) *Verificação da dominância (Definição 6) e eliminação de caminhos*

Para todos os rótulos do nó *i* faça:

- se $\tilde{c}^{it}_{(i,k1)}\ \phi\ \tilde{c}^{it}_{(i,k2)} \Rightarrow$ elimine o *k1-ésimo* rótulo
- se $\tilde{c}^{it}_{(i,k1)}\ \pi\ \tilde{c}^{it}_{(i,k2)} \Rightarrow$ elimine o *k2-ésimo* rótulo

3) *Eliminação de caminhos que não satisfazem as restrições de tempo*

Para todos os caminhos não-dominados existentes entre os nós 1 e *i,* faça:

- $Pt_{(i,k)} = Poss(\tilde{t}^{it}_{(i,k)} \le Rt_j)$
- se $Pt_{(i,k)} < \gamma \Rightarrow$ elimine o caminho *j* com rótulo *k*.

PASSO 2: *Critério de parada.*

i) Se $\left(\tilde{c}^{it}_{(i,k1)} = \tilde{c}^{it-1}_{(i,k1)},\ \forall i \in N\right)$ ou (it=n) faça:

- se $it = n$ e $\tilde{c}^{it}_{(i,k1)} \ne \tilde{c}^{it-1}_{(i,k1)},\ \forall i \in N \Rightarrow$ Passo 5 (circuito negativo)
- senão $\Rightarrow$ Passo 3

ii) Senão: $it \leftarrow it+1 \Rightarrow$ volte ao Passo 1.

PASSO 3: *Composição dos caminhos não-dominados.*

Se existir mais de um caminho não-dominado entre os nós 1 e *i*, faça:

i) Para cada caminho *k=1,...,r*, (*r: número de caminhos*) calcule:

$$Pc_{(i,k1k2)} \leftarrow \min\left(Poss(\tilde{t}^{it}_{(i,k1)} \le \tilde{t}^{it}_{(i,k2)}),\ \ k1,k2 = 1,...,\ r \ \text{ e } \ k1 \ne k2\right).$$

ii) $Pc_{(i,k1)} = \min\{Pc_{(i,k1k2)},\ \ k2 = 1,2,..,r\}$

PASSO 4: *Ordenação dos caminhos não-dominados.*

Após o cálculo da possibilidade de cada caminho ser o menor, faça:

i) Ordem (i) ← 0;

ii) Enquanto todos os caminhos não-dominados não estiverem ordenados faça:

1. Se um caminho *k* possuir os valores de $Pt_{(i,k)}$ e $Pc_{(i,k)}$ maiores que os dos demais caminhos, então:
   - Ordem (i) ← caminho *k*;
   - $i \leftarrow i+1$;
   - Descarte o *caminho k* da lista dos não ordenados e volte ao início da lista.
2. Senão, os caminhos não-dominados são ordenados da seguinte forma:

   Para cada caminho não-ordenado faça:

   Ordenação = $(\alpha.Pc_{(i,k)} + (1-\alpha).Pt_{(i,k)})$

tal que $\alpha \in [0,1]$, sendo uma combinação convexa entre o custo e o tempo, cabendo ao usuário a decisão sobre qual dos dois parâmetros é mais importante. Esta equação tem por finalidade ordenar os caminhos de acordo com os parâmetros decididos pelo usuário.

3. Após calculada a ordenação faça:
   Repita esta etapa até a ordenação de todos os caminhos.
   Encontre o caminho *k* que possui maior valor de ordenação
   - Ordem (*i*) ← caminho *k*;
   - *i* ← *i*+1;
   - Descarte o *caminho k* da lista dos não ordenados e volte ao início desta etapa 3.

PASSO 5: *FIM.*

3.1.2. Convergência e complexidade computacional

Como o algoritmo de Ford-Moore-Bellman, em caso de inexistência de circuito negativo, converge em no máximo *n-1* iterações (*n*: número de nós), este algoritmo também irá convergir em até *n-1* iterações, visto a semelhança na análise de convergência dos dois algoritmos.

Quanto ao estudo da complexidade, tem-se que no Passo 1 são efetuadas no máximo nV_{max} somas para o cálculo dos custos e nV_{max} somas para o tempo, sendo $V_{max} = max\{V_1, V_2, \ldots, V_n\}$ e V_j o número de rótulos permanentes do nó *j*. No Passo 2 são efetuadas no máximo nV^2_{max} comparações para dominância. Para cada iteração temos a complexidade $O(nV^2_{max})=O(nV_{max}+nV_{max}+nV^2_{max})$. Considerando que o algoritmo proposto é uma adaptação de Ford-Moore-Bellman e que converge em *r-1* iteração, tem-se que a complexidade do algoritmo proposto é limitada por $O((n-1)(nV^2_{max}))=O(n^2V^2_{max})$ (Hernandes *et al*, 2007).

4. Resultados Computacionais

O algoritmo proposto foi implementado em *Java Standard Edition*, versão 5.0, e a ferramenta *BlueJ*, versão 2.1.3, sendo executado em uma rede com custos negativos e outra utilizada em Hernandes *et al* (2009). Nos testes trabalhados os valores considerados para γ (Definição 7) foram iguais a zero.

4.1 Testes Computacionais

Exemplo 1. Rede com custos positivos e negativos.

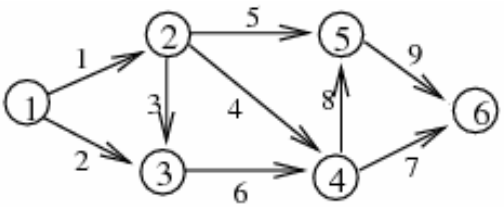

Figura 3 – Rede com custo negativo.

Na Tabela 1 encontram-se as restrições de tempo dos nós e na Tabela 2 os dados referentes à rede da Figura 3.

Tabela 1 – Restrições de tempo dos nós.

Nós	Restrições de tempo
1	(0 0 0)
2	(0.9 0.1 0.1)
3	(1.3 0.2 0.2)
4	(1.4 0.1 0.3)
5	(2 0.4 0.5)
6	(2.1 0.1 0.1)

Tabela 2 - Dados da rede da Figura 3.

Arcos	Custos	Tempo dos arcos
1	(2 1 1)	(0.6 0.1 0.1)
2	(7 2 2)	(1.2 0.2 0.1)
3	(4 3 5)	(0.3 0.1 0.1)
4	(11 1 1)	(0.7 0.2 0.1)
5	(6 1 1)	(0.5 0.1 0.3)
6	(9 1 1)	(0.4 0.1 0.1)
7	(13 2 1)	(0.8 0.1 0.4)
8	(-8 1 1)	(0.3 0.1 0.1)
9	(9 1 1)	(0.2 0.1 0.1)

Levando em consideração que o algoritmo de Hernandes *et al* (2009) não trabalha com redes que possuem custos negativos, este exemplo foi executado somente no algoritmo proposto. Os resultados apresentados estão na Tabela 3.

Tabela 3 - Resultados da rede da Figura 3

Nó Destino		Caminho Não-dominado	Custo do caminho	Tempo do caminho	Pt*	Pc**
2		1→2	(2 1 1)	(0.6 0.1 0.1)	-	-
3	p_1	1→3	(7 2 2)	(0.9 0.2 0.2)	1	0.875
	p_2	1→2→3	(6 4 6)	(1.2 0.2 0.1)	0.25	1
4	p_1	1→2→4	(13 2 2)	(1.3 0.3 0.2)	1	1
	p_2	1→2→3→4	(15 5 7)	(1.6 0.3 0.2)	0,4	0.7143
5	p_1	1→2→4→5	(5 3 3)	(1.6 0.4 0.3)	1	1
	p_2	1→2→3→4→5	(7 6 8)	(1.9 0.4 0.3)	0,572	0.777
6	p_1	1→2→4→5→6	(14 4 4)	(1.8 0.5 0.4)	1	1
	p_2	1→2→3→4→5→6	(16 7 9)	(2.1 0.5 0.4)	0.666	0.82

$* \; Pt = Poss(\tilde{t}_1 \leq Rt) \qquad ** \; Pc = Poss(\tilde{c}_1 \leq \tilde{c}_2)$

Na Tabela 3 verificou-se que as medidas de possibilidade de todos os Caminhos de número 1 possuem *Pt* maiores do que as dos caminhos de número 2, já para os valores dos *Pc* isso não ocorreu.

Para efeitos de ilustração, foram feitas as variações de α para os caminhos entre os nós 1 e 6 (Tabela 4).

Tabela 4 - Resultados da rede da Figura 3.

Etiqueta	Caminho	Valor da ordenação (α*Pc+(1-α)*Pt)								
		α = 0,1	α = 0,2	α = 0,3	α = 0,4	α = 0,5	α = 0,6	α = 0,7	α = 0,8	α = 0,9
1	1→2→4→5→6	1	1	1	1	1	1	1	1	1
2	1→2→3→4→5→6	0,804	0.788	0.772	0.756	0.74	0.724	0.708	0.692	0.676
Ordem		1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1,2	1,2

Exemplo 2. Este exemplo (Figura 4) se refere à rede Européia COST239, sendo o mesmo utilizado em Hernandes *et al* (2009). Os nós da rede e suas restrições de tempo estão definidos na Tabela 5.

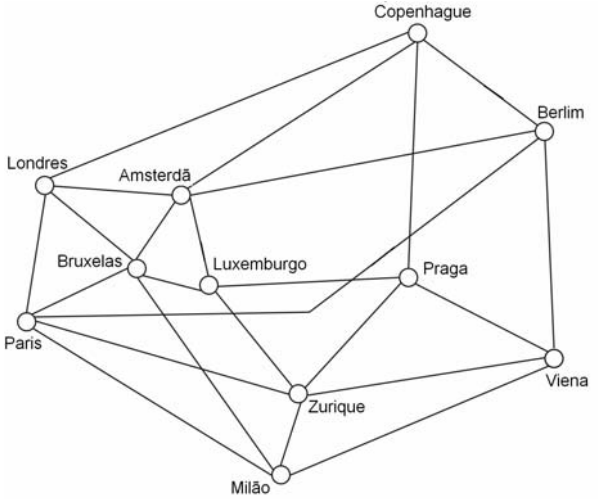

Figura 4 - Rede Européia – COST239.

Tabela 5 - Restrição de tempo dos nós da rede COST239.

Nós	Cidades	Restrições - Tempo	Nós	Cidades	Restrições - Tempo
1	Paris	(0 0 0)	7	Amsterdã	(0,8 0,1 0,1)
2	Milão	(1,7 0,1 0,3)	8	Luxemburgo	(1,0 0,1 0,1)
3	Zurique	(1,0 0,1 0,3)	9	Bruxelas	(0,45 0,15 0,05)
4	Praga	(1,7 0,3 0,2)	10	Londres	(1,2 0,2 0,1)
5	Viena	(2,2 0,2 0,3)	11	Copenhague	(1,95 0,3 0,05)
6	Berlim	(1,0 0,2 0,1)			

Na Tabela 6 estão os nós origem e destino de cada arco, bem como os seus respectivos custos e tempos, enquanto na Tabela 7 são apresentados os resultados finais.

Tabela 6 - Dados da rede COST239.

Arco	Origem	Destino	Custos	Tempos
1	1	2	(820 20 20)	(1,7 0,2 0,2)
2	1	3	(361 11 20)	(0,9 0,1 0,1)
3	1	6	(677 27 6)	(1,05 0,1 0,1)
4	1	9	(300 10 50)	(0,52 0,12 0,08)
5	1	10	(450 30 20)	(1,0 0,05 0,3)
6	2	3	(186 6 7)	(0,45 0,1 0,05)
7	2	5	(510 15 15)	(1,5 0,25 0,4)
8	2	9	(930 30 30)	(2,0 0,3 0,1)
9	3	4	(667 17 196)	(0,98 0,08 0,22)
10	3	5	(748 18 22)	(1,3 0,15 0,15)
11	3	8	(443 18 22)	(0,6 0,1 0,1)
12	4	5	(199 9 11)	(0,3 0,1 0,1)
13	4	6	(340 30 20)	(0,65 0,15 0,05)
14	4	11	(740 30 30)	(1,15 0,1 0,05)
15	5	6	(660 50 30)	(0,85 0,1 0,1)
16	6	11	(242 12 18)	(0,9 0,1 0,1)
17	7	6	(410 20 30)	(0,85 0,1 0,11)
18	7	11	(472 22 18)	(1,16 0,16 0,14)
19	8	4	(730 20 5)	(1,2 0,15 0,05)

20	8	7	(242 12 13)	(0,43 0,13 0,1)
21	9	8	(137 7 8)	(0,3 0,07 0,05)
22	9	7	(130 10 20)	(0,28 0,03 0,04)
23	9	10	(242 12 18)	(0,42 0,12 0,11)
24	10	7	(342 12 8)	(0,65 0,15 0,05)
25	10	11	(1310 60 120)	(2,45 0,15 0,15)

Tabela 7 - Resultados da rede COST239.

Origem	Destino	Caminho	Custo	Tempo	Pt *	Pc**
1	2	1 → 2	(820 20 20)	(1,7 0,2 0,2)	-	-
1	3	1 → 3	(361 11 9)	(0,9 0,1 0,1)	-	-
1	4	1 → 3 → 4	(1028 28 205)	(1,88 0,18 0,32)	0,5263	1,0000
		1 → 9 → 8 → 4	(1167 37 63)	(2,02 0,34 0,18)	0,4074	0,4256
1	5	1 → 3 → 5	(1109 29 31)	(2,2 0,25 0,25)	-	-
1	6	1 → 6	(677 27 6)	(1,05 0,1 0,1)	-	-
1	7	1 → 9 → 7	(430 20 70)	(0,8 0,15 0,12)	-	-
1	8	1 → 9 → 8	(437 17 58)	(0,82 0,19 0,13)	-	-
1	9	1 → 9	(300 10 50)	(0,52 0,12 0,08)	-	-
1	10	1 → 10	(450 30 20)	(1,0 0,05 0,3)	-	-
1	11	1 → 6 → 11	(919 29 24)	(1,95 0,2 0,2)	1,0000	0,8661
		1 → 9 → 7 → 11	(902 42 88)	(1,96 0,31 0,26)	0,9722	1,0000

$* \; Pt = Poss(\tilde{t}_1 \leq Rt)$ $\quad$ $** \; Pc = Poss(\tilde{c}_1 \leq \tilde{c}_2)$

Na Tabela 6 verifica-se que existem dois caminhos não-dominados entre os nós 1 e 4 (1→ 3→ 4; 1→ 9→ 8→ 4) e entre os nós 1 e 11 (1→ 6→ 11; 1→ 9→ 7→ 11) que satisfazem as restrições de tempo. Para os demais nós existe apenas um caminho não-dominado. Também na Tabela 6 verifica-se que os custos e os tempos dos caminhos são os mesmos obtidos por Hernandes *et al* (2009), o qual é justificado porque os parâmetros dos arcos da rede são todos positivos.

As Tabelas 8 e 9 apresentam as variações de α e ordens dos respectivos caminhos.

Tabela 8 – Ordenação dos caminhos entre os nós 1 e 4.

Rótulo	Caminho	Valor da ordenação (α*Pc+(1-α)*Pt)								
		α = 0,1	α = 0,2	α = 0,3	α = 0,4	α = 0,5	α = 0,6	α = 0,7	α = 0,8	α = 0,9
1	1 → 3 → 4	0,5737	0,0611	0,6684	0,7158	0,7632	0,8105	0,8579	0,9053	0,9526
2	1 → 9 → 8 → 4	0,4092	0,4110	0,4129	0,4147	0,4165	0,4183	0,4202	0,4220	0,4238
Ordem dos caminhos		1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2

Tabela 9 – Ordenação dos caminhos entre os nós 1 e 11.

Rótulo	Caminho	Valor da ordenação (α*Pc+(1-α)*Pt)								
		α = 0,1	α = 0,2	α = 0,3	α = 0,4	α = 0,5	α = 0,6	α = 0,7	α = 0,8	α = 0,9
3	1 → 6 → 11	0,9866	0,9732	0,9598	0,9465	0,9331	0,9197	0,9063	0,8929	0,8795
4	1 → 9 → 7 → 11	0,9750	0,9778	0,98	0,9833	0,9861	0,9889	0,9917	0,9944	0,9972
Ordem dos caminhos		3, 4	4, 3	4, 3	4, 3	4, 3	4, 3	4, 3	4, 3	4, 3

Em Hernandes *et al* (2009) este exemplo foi convertido em um problema clássico, sendo que foram considerados os valores modais dos custos e os tempos dos arcos.

4.2 Comparações dos resultados

Analisando os resultados obtidos na seção anterior pode-se afirmar que a diferença principal entre o algoritmo proposto e o de Hernandes *et al* (2009) está nos parâmetros dos arcos, haja vista que um aceita parâmetros *fuzzy* negativos e outro não. Com relação ao desempenho, pode-se afirmar que estes são equivalentes, pois os dois algoritmos foram executados na mesma máquina apresentando os mesmos tempos computacionais de execução.

5. Conclusões e Trabalhos Futuros

Neste trabalho foi proposto um algoritmo para o *Problema de Caminho Mínimo com Restrições de Tempo* aplicado em redes com parâmetros incertos nos arcos, sendo o custo e o tempo, além das restrições de tempo dos nós. O algoritmo proposto apresenta como solução um conjunto ordenado de caminhos não-dominados, similar ao apresentado no trabalho de Hernandes *et al* (2009).

Por se tratar de uma adaptação do algoritmo de Dijkstra, o trabalho de Hernandes *et al* (2009) não pode ser aplicado em redes com parâmetros negativos, logo, o algoritmo proposto possui a vantagem de ser aplicado em redes com parâmetros negativos, pois trata-se de uma adaptação do algoritmo de Ford-Moore-Bellman. Além disso, o mesmo apresenta um conjunto solução ordenado, diferente da maioria dos trabalhos da literatura citados na introdução deste trabalho.

Vale ressaltar que o algoritmo proposto possui complexidade computacional – $O(n^2V^2_{max})$ - menor do que a do trabalho de Hernandes *et al* (2009) - $O(n^3V^2_{max})$.

Como propostas futuras pretende-se adaptar este algoritmo em outros problemas de grafos que envolvem incertezas, tais como: fluxo máximo e fluxo de custo mínimo.

6. Referências

Bassanezi, R. C. e Barros, L. C., *Tópicos de lógica fuzzy e biomatemática*. Campinas, SP: UNICAMP / IMECC, 2006.

Bellman, R. E. (1958), On a routing problem, *Quarterly Applied Mathematics*, 16,87-90.

Bellman, R. E. e Zadeh, L. (1970), Decision making in a fuzzy environment, *Management Science*, 17.

Bezdek, J., *Pattern recognition with fuzzy objective function algorithms*, Plenum Press, 1981.

Blue, M., Bush, B. e Puckett, J. (2002), Unified approach to fuzzy graph problems, *Fuzzy Sets and Systems*, 125, 355-368.

Chanas, S., Delgado, M., Verdegay, J. L. e Vila, M. (1995), Fuzzy optimal flow on imprecise structures, *European Journal of Operational Research*, 83, 568-580.

Dubois, D. e Prade, H., *Fuzzy sets and systems: theory and applications*, Academic Press, New York, 1980.

Goldbarg, M. C. e Luna, H. P. L., *Otimização combinatória e programação linear – modelos e algoritmos,* Editora Campus Ltda, Rio de Janeiro, RJ, 2005.

Hernandes, F. e Takahashi, M. T. (2004), Adaptação de um algoritmo de caminho mínimo em grafos com custo nos arcos e restrições de tempo *fuzzy*, *Anais do XXXVI Simpósio Brasileiro de Pesquisa Operacional*.

Hernandes, F., Yamakami, A., Takahashi, M. T. e Verdegay, J. L. (2006), Um algoritmo para o problema de fluxo máximo em redes com incertezas, *Anais do XXXVIII Simpósio Brasileiro de Pesquisa Operacional*, 2272-2283.

Hernandes, F., *Algoritmos para grafos com incertezas*, Tese de doutorado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 2007.

Hernandes, F., Lamata, M. T., Verdegay, J. L e Yamakami, A. (2007), The shortest problem on networks with fuzzy parameters, *Fuzzy Sets and Systems*, 158, 1561-1570.

Hernandes, F., Berton, L. e Castanho, M. J. P. (2009), O problema de caminho mínimo com incertezas e restrições de tempo, *Pesquisa Operacional*, 29, 471-488.

Kim, K. e Roush, F. (1981), Fuzzy flows on networks, *Fuzzy Sets and Systems*, 8,35-38.

Klein, C. M. (1991), Fuzzy shortest paths, *Fuzzy Sets and Systems*, 39, 27-41.

Lin, C. e Chern, M. S. (1993), The fuzzy shortest path problem and its most vital arcs. *Fuzzy Sets and Systems*, 58, 343-353.

Nayeem, S. M. A. e Pal, M. (2005), Shortest path problem on a network with imprecise edge weight, *Fuzzy Optimization and Decision Making*, 4, 293-312.

Okada, S. e Soper, T. (2000), A shortest path problem on a network with fuzzy arc lengths, *Fuzzy Sets and Systems*, 109, 129-140.

Okada, S. (2004), Fuzzy shortest path problems incorporating interactivity among paths, *Fuzzy Sets and Systems*, 142, (3), 335-357.

Pedrycz, W. e Gomide, F., *An introduction to fuzzy sets: analysis and design,* London: MIT Press, 1998.

Sengupta, A. e Pal, T. K. (2000), On comparing interval numbers. *European Journal of Operational Research*, 127, 29-43.

Shaw, I. S. e Simões, M. G., *Controle e modelagem fuzzy*. Edgar-Blucher/FAPESP, 1998.

Shih, H-S. e Lee, E. S. (1999), Fuzzy multi-level minimum cost flow problems, *Fuzzy Sets and Systems*, 107, 159-176.

Takahashi, M. T., *Contribuições ao estudo de grafos fuzzy: teoria e algoritmos*, Tese de doutorado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 2004.

Takahashi, M. T. e Yamakami, A. (2004), The assignment problem with fuzzy weights. *Anais do North American Fuzzy Information Processing Society*, Alberta, Canadá.

Zadeh, L. (1965), Fuzzy sets, *Information and Control*, 8, 338-353.

Zadeh, L. (1968), Fuzzy algorithms*, Information and Control*, 12.

Zadeh, L. (1978), Fuzzy sets as a theory of possibility, *Journal of Fuzzy Sets and Systems*, 1.