Propriedades de fecho de linguagens (C, D)-Fuzzy

Valdigleis da Silva Costa[1], Antônio Diego S. Farias[2], Benjamín Bedregal[3], Regivan H.N. Santiago[3]

[1]Colegiado de Ciência da Computação
Universidade Federal do Vale do São Francisco (UNIVASF)
Campus Salgueiro – Salgueiro, PE – Brasil

[2]Departamento de Ciências Exatas e Naturais
Universidade Federal Rural do Semi-Árido (UFERSA)
Campus Pau dos Ferros, Pau dos Ferros, RN – Brasil

[3]Departamento de Informática e Matemática Aplicada
Universidade Federal do Rio Grande do Norte (UFRN) – Natal, RN – Brasil

valdigleis.costa@univasf.edu.br, antonio.diego@ufersa.edu.br,

{bedregal,regivan}@dimap.ufrn.br

Abstract. *We generalize the notion of fuzzy language associated with fuzzy automata by considering a conjunctive aggregation function C instead of a t-norm and a disjunctive aggregation function D instead of a t-conorm and we call it (C, D)-fuzzy. Then we investigate sufficient and necessary conditions on C and D for the class of (C, D)-fuzzy languages to be closed under fuzzy versions of traditional operators over languages. In this extended summary, we limit ourselves to the reverse operator.*

Resumo. *Generalizamos a noção de linguagem fuzzy associada a autômatos fuzzy considerando uma função de agregação conjuntiva C em lugar de uma t-norma e uma função de agregação disjuntiva D em lugar de uma t-conorma e a chamamos de linguagem (C, D)-fuzzy. Depois investigamos condições suficientes e necessárias sobre C e D para a classe das linguagens (C, D)-fuzzy ser fechada sob versões fuzzy de operadores tradicionais sobre linguagens. Neste resumo estendido, nos limitamos ao operador reverso.*

1. Introdução

William Go Wee em [Wee 1967] introduziu a noção de autômatos fuzzy como um modelo de sistema de aprendizado para ser aplicado em problemas de classificação de padrões[1]. Lee e Zadeh introduziram em [Lee and Zadeh 1969] os conceitos de linguagens e gramáticas fuzzy como sendo extensões de suas respectivas contra-partes clássicas. Formalmente, uma linguagem fuzzy $\mathbb{L}$ sobre um alfabeto Σ é um conjunto fuzzy, ou seja, uma função $\mathbb{L} : \Sigma^* \to [0, 1]$ [Costa 2016].

No entanto, em [Santos 1968] foi provado que, para cada $\alpha \in [0, 1)$, o α-corte de qualquer linguagem fuzzy aceita por um autômato fuzzy é regular. Uma vez que o

[1]Uma lista de algumas outras aplicações de autômatos fuzzy, com as respectivas referências, pode ser encontrada em [Butoianu and Todinca 2015, Li 2008, Singh et al. 2017].

inverso ocorre trivialmente, os autômatos fuzzy têm o mesmo poder de aceitação que os autômatos finitos [Mizumoto et al. 1969]. A definição de linguagem fuzzy aceita por um autômato fuzzy em [Santos 1968, Mizumoto et al. 1969, Mordeson and Malik 2002] usa a função de agregação mínima e máxima. Informações sobre diversas pesquisas na área até 2017 podem ser vistas em [Singh et al. 2017].

Analogamente às linguagens regulares, as linguagens fuzzy, no sentido de [Santos 1968], sao fechadas para as operações de união, interseção e fecho de Kleene, quando definidas como extensões das correspondentes noções em teria de linguagens formais [Lee and Zadeh 1969]. Em [Costa et al. 2021] foram estudadas as propriedades de fecho para linguagens lineares fuzzy valoradas em reticulados, e entre eles o caso da operação que estende de forma natural a reversa de linguagens formais para linguagens fuzzy valoradas em reticulados, no entanto, não encontramos qualquer estudo sobre a operação reversa de linguagens fuzzy aceitos por autômatos fuzzy.

2. Preliminares

Definição 2.1 *Um mapeamento $F : [0,1]^2 \to [0,1]$ é uma função de agregação bivariada*[2] *se $F(0,0) = 0$, $F(1,1) = 1$ e $F(x_1, x_2) \leq F(y_1, y_2)$ sempre que $x_1 \leq y_1$ e $x_2 \leq y_2$. Além disso, dizemos que F é*

Conjuntiva *se $F(x,y) \leq \min(x,y)$ para cada $x, y \in [0,1]$;*
Disjuntiva *se $F(x,y) \geq \max(x,y)$ para cada $x, y \in [0,1]$;*
Comutativa *se $F(x,y) = F(y,x)$ para cada $x, y \in [0,1]$;*
Associativa *se $F(x, F(y,z)) = F(F(x,y), z)$ para cada $x, y, z \in [0,1]$;*

e dizemos que F tem ***elemento neutro*** *se existe $r \in [0,1]$ tal que $F(x,r) = F(r,x) = x$ para cada $x \in [0,1]$.*

Uma função de agregação bivariada é uma **uninorma** se for comutativa, associativa e tiver elemento neutro. Uma uninorma é uma **t-norma** se o elementro neutro é 1 e se 0 for um elemento neutro então é uma **t-conorma**. Cada t-norma é conjuntiva e cada t-conorma é disjuntiva. Além disso, funções de agregação bivariadas, conjuntivas, comutativas e associativas são denominadas de **t-subnormas**. Para um estudo mais aprofundado sobre esses tópicos, recomendamos [Beliakov et al. 2007, Klement et al. 2000].

3. Autômatos fuzzy e linguagens (C, D)-fuzzy

Diversas formalizações e interpretações para a noção de autômatos fuzzy têm sido propostas [Costa 2016]. Aqui consideraremos a seguinte:

Definição 3.1 *[Komejwar 2012, Mordeson and Malik 2002] Um autômato fuzzy é uma quíntupla $M = \langle Q, \Sigma, \varphi, \mathbb{I}, \mathbb{F} \rangle$ onde*

- *Q é um conjunto finito e não vazio de estados;*
- *Σ é um alfabeto;*
- *$\varphi : Q \times \Sigma \times Q \to [0,1]$ é um conjunto de transições (ou instruções) fuzzy;*
- *$\mathbb{I} : Q \to [0,1]$ é um conjunto fuzzy de estados iniciais e*

[2]Funções de agregação podem ser definidas para qualquer aridade e inclusive para uma aridade variável, e são denominadas de funções de agregação estendida [Beliakov et al. 2007].

- $\mathbb{F} : Q \to [0,1]$ *é um conjunto fuzzy de estados finais.*

Os autômatos fuzzy funcionam de maneira semelhante aos autômatos finitos não determinísticos. Por conta desta natureza fuzzy de suas transições, fica implícito que autômatos fuzzy podem modelar erros de transição entre os estados.

Autômatos fuzzy têm a capacidade de determinar ou aceitar linguagens fuzzy [Lee and Zadeh 1969]. O grau de pertinência de cada cadeia à linguagem fuzzy é obtida de diversas formas. Por exemplo usando a composição Max-Min, como em [Mordeson and Malik 2002, Santos 1968, Wee 1967], ou usando a composição Max-T, onde T é t-norma, como por exemplo em [Li 2008] e ainda mais geral, baseado na composicão S-T, onde S é uma t-conorma e T uma t-norma, como em [Maciel 2006].

Mas, neste artigo, introduziremos uma nova forma, que considera funções de agregação bivariadas conjuntivas e disjuntivas em vez de t-normas e t-conormas, respectivamente.

Definição 3.2 *Seja $M = \langle Q, \Sigma, \varphi, \mathbb{I}, \mathbb{F}\rangle$ um autômato fuzzy, $C : [0,1]^2 \to [0,1]$ uma função de agregação conjuntiva e $D : \bigcup_{n=1}^{\infty} [0,1]^n \to [0,1]$ uma função de agregação estendida comutativa e disjuntiva. O conjunto de transições fuzzy φ pode ser estendido para um conjunto fuzzy $\widehat{\varphi}_{C,D} : Q \times \Sigma^* \times Q \to [0,1]$ recursivamente como segue:*

$$\begin{aligned}
\widehat{\varphi}_{C,D}(q, \lambda, q') &= \begin{cases} 0, & \textit{se } q \neq q' \\ 1, & \textit{caso contrário} \end{cases} \\
\widehat{\varphi}_{C,D}(q, aw, q') &= D_{q'' \in Q}\Big\{ C(\varphi(q, a, q''), \widehat{\varphi}_{C,D}(q'', w, q')) \Big\}
\end{aligned} \tag{1}$$

para cada $a \in \Sigma, w \in \Sigma^*$ e $q, q' \in Q$.

Observe que quando D é o máximo, $\widehat{\varphi}_{C,D}(q, aw, q') = D_{q'' \in Q}\Big\{ C(\varphi(q, a, q''), \widehat{\varphi}_{C,D}(q'', w, q')) \Big\} = C(\varphi(q, a, q''), \widehat{\varphi}_{C,D}(q'', w, q'))$ para algum $q'' \in Q$ e generalizado na Proposição a seguir.

Proposição 3.1 *Seja $M = \langle Q, \Sigma, \varphi, \mathbb{I}, \mathbb{F}\rangle$ um autômato fuzzy, $C : [0,1]^2 \to [0,1]$ uma função de agregação conjuntiva e $D = \max$. Para todo $w = a_1 \ldots a_n \in \Sigma^*$ e $q, q' \in Q$, existe uma sequência $(q_1, \ldots, q_{n+1})$ de estados em Q tal que $q_1 = q$, $q_{n+1} = q'$ e*

$$\widehat{\varphi}_{C,D}(q, w, q') = C(\varphi(q_1, a_1, q_2), C(\varphi(q_2, a_2, q_3), C(\ldots, C(\varphi(q_n, a_n, q_{n+1}), 1)) \ldots)) \tag{2}$$

Definição 3.3 *Seja $M = \langle Q, \Sigma, \varphi, \mathbb{I}, \mathbb{F}\rangle$ um autômato fuzzy, $C : [0,1]^2 \to [0,1]$ uma função de agregação conjuntiva e $D : \bigcup_{n=1}^{\infty} [0,1]^n \to [0,1]$ uma função de agregação estendida comutativa e disjuntiva. A linguagem fuzzy aceita (ou computada) por M com respeito a (C, D), ou simplesmente a linguagem (C, D)-fuzzy, é o conjunto fuzzy $\mathbb{L}_{M,C,D} : \Sigma^* \to [0,1]$ definido por:*

$$\mathbb{L}_{M,C,D}(w) = D_{q,q' \in Q}\left(C(C(\mathbb{I}(q), \mathbb{F}(q')), \widehat{\varphi}_{C,D}(q, w, q')) \right).$$

Observe que quando $C = \min$ e $D = \max$, esta noção coincide com a de [Komejwar 2012] e quando C e D forem uma t-norma e t-conorma, respetivamente, esta noção coincide com [Maciel 2006].

4. A reversa de linguagens (C, D)-fuzzy

Nesta seção mostraremos uma condicao necessaria e suficiente que as aggregações C e D devem possuir para as linguagens (C, D)-fuzzy serem fechadas para a operação que estende de forma natural a operação reversa de linguagens formais.

Definição 4.1 *O reverso de uma linguagem fuzzy* $\mathbb{L} : \Sigma^* \to [0, 1]$ *é a linguagem fuzzy* $\mathbb{L}^{rev} : \Sigma^* \to [0, 1]$ *tal que* $\mathbb{L}^{rev}(w) = \mathbb{L}(rev(w))$ *onde* $rev(w)$ *é o inverso da cadeia* w*, ou seja,* $rev(\lambda) = \lambda$ *(com* λ *denotando a cadeia vazia) e* $rev(a_1a_2 \ldots a_n) = a_n \ldots a_2a_1$ *para cada* $a_1, \ldots, a_n \in \Sigma$.

Observe que claramente se $\mathbb{L} : \Sigma^* \to [0, 1]$ é uma linguagem fuzzy *crisp*, ou seja $\mathbb{L}(w) \in \{0, 1\}$ para todo $w \in \Sigma^*$, então $\mathbb{L}^{rev}$ é uma linguagem fuzzy *crisp* que coincide com a reversa de $\mathbb{L}$, no sentido que $\mathbb{L}^{rev}(w) = \mathbb{L}(rev(w))$ para todo $w \in \Sigma^*$.

Definição 4.2 *Seja* $M = \langle Q, \Sigma, \varphi, \mathbb{I}, \mathbb{F}\rangle$ *um autômato fuzzy.* ***O autômato fuzzy reverso*** *de* M *é* $M^{rev} = \langle Q, \Sigma, \varphi^{rev}, \mathbb{F}, \mathbb{I}\rangle$ *onde* $\varphi^{rev} : Q \times \Sigma \times Q \to [0, 1]$ *é definido para todo* $q, q' \in Q$ *e* $a \in \Sigma$ *como* $\varphi^{rev}(q, a, q') = \varphi(q', a, q)$.

Teorema 4.1 *Seja* $M = \langle Q, \Sigma, \varphi, \mathbb{I}, \mathbb{F}\rangle$ *um autômato fuzzy,* $C : [0, 1]^2 \to [0, 1]$ *uma função de agregação conjuntiva e* $D : \bigcup_{n=1}^{\infty} [0, 1]^n \to [0, 1]$ *uma função de agregação estendida comutativa e disjuntiva. Então,* $\mathbb{L}_{M^{rev},C,D} = \mathbb{L}^{rev}_{M,C,D}$ *se, e somente se,* C *é uma t-subnorma.*

Mas isto não significa que funções de agregação conjuntivas que não forem t-subnormas devam ser descartadas, pois em alguns cenários pode ser que a reversa de uma linguagem fuzzy não seja relevante, como por exemplo nas aplicações de autômatos fuzzy mencionadas em [Mordeson and Malik 2002, Rajasekar and Thilagavathi 2022, Singh et al. 2017].

5. Consideraçõe finais

Na literatura há diversas propostas de generalizações dos autômatos finitos que adicionam pesos nos estados e arestas, petencentes ao conjunto base de algum tipo de álgebra, como por exemplo em [Eilenberg 1974, Gomes et al. 2020, Madhuri and Amudhambigai 2019, Mordeson and Malik 2002]. As linguagem associadas a esses autômatos usam as operações da álgebra para determinar o valor ou o grau com que uma cadeia pertence à linguagem. No entanto, $\langle[0, 1], C, D\rangle$, onde C é uma função de agregação conjuntiva e D é uma função de agregação disjuntiva estendida, não é em geral um caso especial dessas álgebras, tome por exemplo $C(x, y) = x^2 \cdot y^3$ e $D(x, y) = \min(1, \sqrt[3]{x} + \sqrt{y})$. Também existem na literatura, diversas propostas de autômatos fuzzy que usam algum tipo de função de agregação, como por exemplo t-normas e t-conormas em [Li 2008, Maciel 2006], mas nenhuma delas usa funções de agregação conjuntivas e disjuntivas, como proposto aqui.

Neste trabalho, também introduzimos uma nova forma de definir linguagens fuzzy a partir de um autômato fuzzy, via uma função de agregação conjuntiva C e uma função de agregação disjuntiva estendida D. Além disso, fornecemos um teorema de caracterização dessas funções que levam à classe das linguagens fuzzy serem fechadas sobre a operação reversa. Seguindo esta mesma linha podemos fornecer teoremas de caracterização de outras operações, como união, interseção, complemento, composição, fecho estrela, etc.

Referências

Beliakov, G., Pradera, A., and Calvo, T. (2007). *Aggregation Functions: A Guide for Practitioners*, volume 221 of *Studies in Fuzziness and Soft Computing*. Springer, Berlin, Heidelberg.

Butoianu, D.-E. and Todinca, D. (2015). The efficiency of minimum operator for fuzzy automata: A case study. In *2015 IEEE 10th Jubilee International Symposium on Applied Computational Intelligence and Informatics*, pages 203–208.

Costa, V. S. (2016). *Linguagens Lineares Fuzzy*. Master's thesis, Programa de Pós-graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, UFRN, Natal, RN.

Costa, V. S., Bedregal, B., and Santiago, R. H. N. (2021). On closure properties of $\mathcal{L}$-valued linear languages. *Fuzzy Sets and Systems*, 420:54–71.

Eilenberg, S. (1974). *Automata, Languages, and Machines*, volume 59A. Academic Press. Pure and Applied Mathematics.

Gomes, L., Madeira, A., and Barbosa, L. S. (2020). Introducing synchrony in fuzzy automata. *Electronic Notes in Theoretical Computer Science*, 348:43–60. 14th International Workshop on Logical and Semantic Frameworks, with Applications (LSFA 2019).

Klement, E. P., Mesiar, R., and Pap, E. (2000). *Triangular norms*, volume 8. Kluwer Academic, Dordrecht/Boston/London.

Komejwar, D. D. (2012). *A Study of Some Aspects of Fuzzy Automata, Fuzzy Grammars and Languages*. PhD thesis, Shivaji University, Kolhapur.

Lee, E. T. and Zadeh, L. A. (1969). Note on fuzzy languages. *Information Sciences*, 1:421–434.

Li, Y. (2008). Approximation and robustness of fuzzy finite automata. *International Journal of Approximate Reasoning*, 47(2):247–257.

Maciel, A. (2006). Aplicação de autômatos finitos nebulosos no reconhecimento aproximado de cadeias. Master's thesis, Universidade de São Paulo, São Paulo, SP.

Madhuri, V. and Amudhambigai, B. (2019). A new view on fuzzy automata normed linear structure spaces. *Iranian Journal of Fuzzy Systems*, 16(6):65–74.

Mizumoto, M., Toyoda, J., and Tanaka, K. (1969). Some considerations on fuzzy automata. *Journal of Computer and System Sciences*, 3(4):409–422.

Mordeson, J. N. and Malik, D. S. (2002). *Fuzzy automata and languages: theory and applications*. CRC Press, New York.

Rajasekar, M. and Thilagavathi, T. S. (2022). A new DNA implementation and pattern analysis using intuitionistic fuzzy finite automata. In *Proc. 2nd Int. Conf. Math. Techniques and Applications ICMTA2021, 24–26 March 2021, Kattankulathur, India*, volume 2516.

Santos, E. S. (1968). Maximin automata. *Information and Control*, 13:363–377.

Singh, R. K., Rani, A., and Sachan, M. K. (2017). Fuzzy automata: A quantitative review. *International Journal on Future Revolution in Computer Science & Communication Engineering*, 3(7):11–17.

Wee, W. G. (1967). *On Generalizations of Adaptive Algorithms and Application of the Fuzzy Sets Concept to Pattern Classification*. PhD thesis, Purdue University, Lafayette, Indiana.

Apêndice: Prova Teorema 4.1

Demonstração: ($\Rightarrow$) Se C não é comutativa então existem $r, s \in [0,1]$ tais que $C(r,s) \neq C(s,r)$. Seja $M = \langle Q, \Sigma, \varphi, \mathbb{I}, \mathbb{F} \rangle$ onde $Q = \{q,p\}$, $\Sigma = \{a,b\}$, $\varphi(q,a,p) = r$, $\varphi(p,b,q) = s$, $\varphi(q,a,q) = \varphi(q,b,q) = \varphi(q,b,p) = \varphi(p,a,q) = \varphi(p,a,p) = \varphi(p,b,p) = 0$, $\mathbb{I}(q) = \mathbb{F}(q) = 1$ e $\mathbb{I}(p) = \mathbb{F}(p) = 0$. Neste caso, $\mathbb{L}^{rev}_{M,C,D}(ba) = \mathbb{L}_{M,C,D}(rev(ba)) = \mathbb{L}_{M,C,D}(ab) = C(r,s) \neq C(s,r) = \mathbb{L}_{M^{rev},C,D}(ba)$, o que é uma contradição. Logo C é comutativa.

Analogamente, se C não é associativa então existem $r, s, t \in [0,1]$ tais que $C(r, C(s,t)) \neq C(C(r,s),t)$. Seja $M = \langle Q, \Sigma, \varphi, \mathbb{I}, \mathbb{F} \rangle$ onde $Q = \{q,p\}$, $\Sigma = \{a,b\}$, $\varphi(q,a,p) = r$, $\varphi(p,b,p) = s$, $\varphi(p,b,q) = t$, $\varphi(q,a,q) = \varphi(q,b,q) = \varphi(q,b,p) = \varphi(p,a,q) = \varphi(p,a,p) = 0$, $\mathbb{I}(q) = \mathbb{F}(q) = 1$ e $\mathbb{I}(p) = \mathbb{F}(p) = 0$. Neste caso, como C é comutativa, $\mathbb{L}^{rev}_{M,C,D}(bba) = \mathbb{L}_{M,C,D}(rev(bba)) = \mathbb{L}_{M,C,D}(abb) = C(r, C(s,t))) \neq C(t, C(s,r)) = C(C(r,s),t) = \mathbb{L}_{M^{rev},C,D}(bba)$, o que é uma contradição. Logo C é associativa.

Portanto, C é uma função de agregação conjuntiva, comutativa e associativa, ou seja uma t-subnorma.

($\Leftarrow$) Provaremos por indução que, para todo $q, q' \in Q$ e $w \in \Sigma^*$, $\widehat{\varphi}_{C,D}(q,w,q') = \widehat{\varphi^{rev}}_{C,D}(q', rev(w), q)$. Trivialmente isto vale se $w = \lambda$ e $q \neq q'$. Se $w = \lambda$ e $q = q'$ então $\widehat{\varphi}_{C,D}(q,w,q') = 1 = \widehat{\varphi^{rev}}_{C,D}(q', \lambda, q) = \widehat{\varphi^{rev}}_{C,D}(q', rev(w), q)$. Se $w = a_1 \ldots a_n$ para $n \geq 1$ então seja a sequência de estados $(q_1, \ldots, q_{n+1})$ da Proposição 3.1. Logo,

$$\begin{aligned}
\widehat{\varphi}_{C,D}(q,w,q') &= C(\varphi(q_1,a_1,q_2), C(\varphi(q_2,a_2,q_3), C(\ldots, C(\varphi(q_n,a_n,q_{n+1}),1))\ldots) \\
&= C(\varphi(q_n,a_n,q_{n+1}), C(\varphi(q_{n-1},a_{n-1},q_n), C(\ldots, C(\varphi(q_1,a_1,q_2),1))\ldots) \\
&= C(\varphi^{rev}(q_{n+1},a_n,q_n), C(\varphi^{rev}(q_n,a_{n-1},q_{n-1}), C(\ldots, C(\varphi^{rev}(q_2,a_1,q_1),1))\ldots) \\
&\leq \widehat{\varphi^{rev}}_{C,D}(q', rev(w), q).
\end{aligned}$$

De forma análoga, obtemos que

$$\widehat{\varphi^{rev}}_{C,D}(q', rev(w), q) \leq \widehat{(\varphi^{rev})^{rev}}_{C,D}(q, rev(rev(w)), q') = \widehat{\varphi}_{C,D}(q,w,q').$$

Portanto, $\widehat{\varphi^{rev}}_{C,D}(q', rev(w), q) = \widehat{\varphi}_{C,D}(q,w,q')$ para todo $q, q' \in Q$ e $w \in \Sigma^*$ e consequentemente, $\mathbb{L}^{rev}_{M,C,D} = \mathbb{L}_{M^{rev},C,D}$. $\square$