

AGENTE TOMADOR DE DECISÕES BASEADAS EM LÓGICA FUZZY NO JOGO DE PÔQUER

Heitor Magno Rodrigues Junior, heitormrjunior@hotmail.com

Programa de Pós-Graduação em Engenharia Elétrica - Associação ampla UFSJ/CEFET-MG
Pça. Frei Orlando, 170, Centro - 36307-352 - São João del-Rei, MG, Brasil

Leonardo Bonato Felix, leobonato@ufv.br

Departamento de Engenharia Elétrica - Universidade Federal de Viçosa
Av. Peter Henry Rolfs, s/n, Campus Universitário - 36570-900 - Viçosa, MG, Brasil

Erirelton Geraldo Nepomuceno, nepomuceno@ufs.edu.br

Departamento de Engenharia Elétrica - Universidade Federal de São João del-Rei
Pça. Frei Orlando, 170, Centro - 36307-352 - São João del-Rei, MG, Brasil

André Luiz Carvalho Ottoni, andreottoni@ymail.com

Programa de Pós-Graduação em Engenharia Elétrica - Associação ampla UFSJ/CEFET-MG
Pça. Frei Orlando, 170, Centro - 36307-352 - São João del-Rei, MG, Brasil

Resumo. *Este trabalho tem como objetivo o desenvolvimento de um agente tomador de decisões para o jogo de cartas pôquer. Para isso, foi desenvolvido um sistema baseado em lógica fuzzy, técnica discutida dentro do campo de conhecimentos da Inteligência Artificial. A complexidade envolvida no pôquer devido ao grande número de combinações de jogadas e cartas, aliada à falta de experiência, faz com que jogadores iniciantes tenham grande dificuldade nas tomadas de decisões e, principalmente, no aprendizado do jogo. Esse fato, portanto, foi a motivação para o desenvolvimento de um sistema fuzzy que auxilie a tomada de decisão do jogador, facilitando o entendimento e aprendizado do pôquer. O agente foi desenvolvido em Matlab e considera uma série de fatores para cada estado do jogo, como posição do jogador, quantidade de fichas restantes, quantidade de fichas já apostadas, ação do oponente, força da mão e estágio do jogo. O software usado para fazer o teste e a comparação do algoritmo foi o PokerTH, um simulador gratuito da modalidade Texas Hold'em, disponível em sistemas operacionais Unix-like e também no Microsoft Windows. A eficácia do sistema inteligente desenvolvido foi analisada pela comparação dos resultados obtidos em partidas desse agente e outros três tipos de jogadores virtuais contra um mesmo adversário. O jogador que toma decisões baseadas no algoritmo desenvolvido neste trabalho obteve o melhor resultado entre os agentes analisados. O sistema inteligente desenvolvido foi o único a vencer mais do que ser derrotado pelo jogador adversário.*

Palavras-chave: *Lógica Fuzzy, Inteligência Artificial em Jogos, Pôquer*

1. INTRODUÇÃO

A lógica *fuzzy* é a lógica baseada na teoria dos conjuntos *fuzzy*, desenvolvida por Zadeh (1965). Ela difere dos sistemas lógicos tradicionais em suas características e seus detalhes. Nesta lógica, o raciocínio exato corresponde a um caso limite do raciocínio aproximado, sendo interpretado como um processo de composição de relações nebulosas. Portanto, a lógica *fuzzy* constitui a base do campo de Inteligência Artificial (IA) para o desenvolvimento de métodos e algoritmos de modelagem e controle de processos, permitindo a redução da complexidade de projeto e implementação, tornando-se a solução para problemas intratáveis por técnicas clássicas (Gomide e Gudwin, 1994).

Dessa forma, informações difíceis de se quantificar e que fogem do conceito da lógica binária, podem ser tratadas pela

lógica *fuzzy*, que possui aplicação nas mais diversas áreas, como análise de dados (Moraes, 2008), reconhecimento de padrões (Cunha *et al.*, 2007), sistemas especialistas (Bilobrovec *et al.*, 2004), testes de software (Rabelo e Moita, 2014), modelagem computacional (Braga e Moita, 2014), diagnósticos de falhas (Cruz *et al.*, 2015), análise de incertezas (Pinto *et al.*, 2015), entre outras.

Dentre as diversas aplicações e problemas que envolvem dados incertos, está o problema de tomada de decisões em jogos de cartas, como no jogo de pôquer. Alguns trabalhos já trataram esse problema com o desenvolvimento de agentes com estratégias baseadas no comportamento dos adversários para definir quais ações deveriam ser tomadas (Benjamim *et al.*, 2012; Ziółko *et al.*, 2012). Nesse trabalho, procura-se desenvolver um sistema *fuzzy* com estratégia baseada em diversos outros fatores presentes no jogo, que auxilie na tomada de decisão de jogadores iniciantes.

O principal motivo de se desenvolver esse sistema, baseia-se no fato de que o pôquer aparenta ser trivial devido à simplicidade de suas regras, em que qualquer pessoa pode aprendê-las em poucos minutos. Porém, é um jogo que não exige apenas o conhecimento das regras, mas também a experiência dos jogadores para que as melhores decisões sejam tomadas (Sklansky, 1999).

Este artigo está organizado em cinco sessões: a próxima sessão descreve os conceitos que servirão como base para melhor entendimento do trabalho; a sessão 3 apresenta a aplicação do sistema *fuzzy* no jogo de pôquer; os resultados obtidos podem ser vistos na seção 4; finalmente, a sessão 5 apresenta as conclusões e propostas futuras de trabalho.

2. CONCEITOS BÁSICOS

2.1 Sistemas Fuzzy

Na teoria de conjuntos clássica, um elemento ou pertence a um conjunto ou não. Dado um universo U e um elemento particular $x \in U$, o grau de pertinência $\mu_A(x)$ com respeito a um conjunto $A \subset U$ é dado por:

$$\mu_A(x) = \begin{cases} 1, & \text{se } x \in A \\ 0, & \text{se } x \notin A \end{cases} \quad (1)$$

Zadeh (1965) propôs uma caracterização mais ampla, sugerindo que alguns elementos são mais membros de um conjunto do que outros. Segundo essa caracterização, o fator de pertinência pode assumir qualquer valor entre 0 e 1, sendo que o valor 0 indica uma completa exclusão e um valor 1 representa completa pertinência. Essa generalização aumenta o poder de expressão da função característica (Gomide e Gudwin, 1994). Formalmente, tem-se que:

$$\mu_A : U \rightarrow [0, 1] \quad (2)$$

Para expressar conceitos, é muito comum o uso de elementos qualitativos ao invés de valores quantitativos. Elementos típicos incluem "mais ou menos", "muito", "pouco", "médio", etc. Essas ideias são capturadas pela definição de variável linguística, que ao invés de assumir instâncias numéricas, assume valores dentro de um conjunto de termos linguísticos, ou seja, palavras ou frases. Por exemplo, uma variável linguística *Temperatura* pode assumir como valor um dos membros do conjunto $\{baixa, média, alta\}$. Para se atribuir um significado aos termos linguísticos, associa-se cada um destes a um conjunto *fuzzy* definido sobre um universo de discurso comum, como mostrado na Fig. 1 (Gomide e Gudwin, 1994).

A forma mais comum de expressar o conhecimento é por meio de regras. As regras *fuzzy* descrevem situações específicas que podem ser submetidas a análise de um painel de especialistas, e cuja inferência conduz a algum resultado desejado. Cada regra *fuzzy* é composta por uma parte antecedente e uma parte consequente, resultando em uma estrutura do tipo "se-então", onde a parte antecedente descreve uma condição e define uma região *fuzzy* no espaço das variáveis de entrada do sistema, enquanto a parte consequente descreve uma conclusão ou ação a ser tomada, além de descrever uma região no espaço das variáveis de saída do sistema. A elaboração das regras constitui um aspecto fundamental no desempenho do sistema de inferência *fuzzy* (Klir e Yuan, 1995).

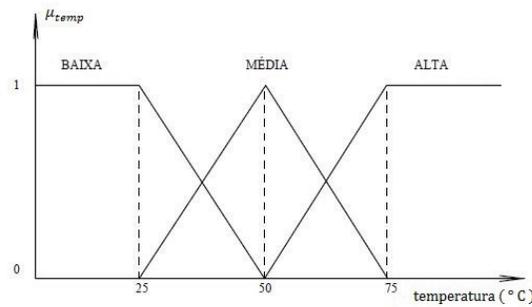


Figura 1. Variável linguística *temperatura*. Fonte: (Gomide e Gudwin, 1994).

Além da base de regras, um sistema de inferência é composto pelo módulo de *fuzzificação*, que é o que modela matematicamente a informação das variáveis de entrada por meio de conjuntos *fuzzy*. O sistema também é composto pelo módulo de inferência, que é onde se definem quais são os conceitos lógicos usados para estabelecer a relação *fuzzy* que modela a base de regras. Finalmente, no módulo de *defuzzificação*, os resultados difusos da inferência são transformados em valores de saída, onde uma interpretação dessas informações é realizada (Pedrycz e Gomide, 1998).

Portanto, por serem sistemas capazes de processar eficientemente informações imprecisas e qualitativas de forma geral, os modelos de inferência *fuzzy* são especialmente adequados em processos que exigem tomadas de decisão baseadas em conhecimento e experiência sobre um determinado estado do processo ou da situação (Rezende, 2003).

2.2 Lógica Fuzzy no Matlab

O software *Matlab* apresenta o *Fuzzy Logic Toolbox*, ferramenta usada para analisar, projetar e simular sistemas baseados em lógica *fuzzy*. Essa ferramenta permite criar e editar sistemas *fuzzy* de inferência no *Matlab* ou Simulink (MathWorks e Wang, 1998).

O tipo de método de análise *fuzzy* utilizado no trabalho foi o *Mandani*, padrão do *Matlab*, que também apresenta como opção o método *Sugeno*. Para melhor entendimento destes e de outros métodos de inferência e *defuzzificação*, recomenda-se a leitura de (Pedrycz e Gomide, 1998).

2.3 Texas Hold'em Poker

O *Texas Hold'em Poker* é uma das variáveis do pôquer e atualmente é a modalidade mais popular desse jogo. Basicamente, os jogos de pôquer consistem em maximizar os ganhos e minimizar as perdas de fichas, que são apostadas em cada estágio da rodada. Ao final de cada rodada, ganha o jogador que ainda estiver no jogo e possuir a melhor combinação possível com cinco cartas de um baralho completo. O jogador que ganha a rodada, leva o total das apostas feitas, chamado de pote. As definições abaixo foram retiradas de Bello (2008).

Cada rodada do *Texas Hold'em Poker* é composta por quatro estágios, onde, em cada estágio, há uma rodada de apostas. Essa modalidade geralmente é disputada entre até nove jogadores. A distribuição das cartas e a ordem das apostas é sempre realizada no sentido horário. Em cada rodada, um dos jogadores terá o botão do *dealer* a sua frente, indicando que as ações começam com o jogador a sua esquerda, de modo que o jogador com o botão seja o último a agir.

Os *blinds* são apostas obrigatórias que devem ser feitas pelos jogadores nas duas posições imediatamente a esquerda do *dealer*, antes mesmo de receber suas cartas. O primeiro jogador a esquerda, chamado de *small blind* deposita metade do valor do segundo jogador, que aposta o *big blind*.

Feito isso, são distribuídas duas cartas fechadas a cada um dos jogadores da mesa. O primeiro jogador a esquerda do *big blind* começa a primeira rodada de apostas, onde os jogadores possuem 3 opções de ação:

- *Fold*: desistir da mão;

- *Call*: pagar a aposta anterior e ir para o próximo estágio da rodada;
- *Raise*: aumentar a aposta feita anteriormente.
- *Check*: caso nenhum jogador tenha aumentado e o jogador que está no *big blind* não deseje aumentar, ele passa a jogada.

Após os jogadores terem tomado suas decisões no primeiro estágio, são abertas três cartas comunitárias na mesa, o que é o chamado *flop*. Então uma nova rodada de apostas se segue. O próximo estágio é o *turn*, onde mais uma carta comunitária é aberta na mesa, seguido por mais uma rodada de apostas. Finalmente, o último estágio é o *river*, onde mais uma carta comunitária é aberta na mesa e mais uma rodada de apostas é feita. Em qualquer estágio, caso algum jogador faça uma aposta e todos os demais desistam, ele leva o pote e a rodada é finalizada.

A modalidade abordada no trabalho foi o *Texas Hold'em Heads-Up Poker*, que segue as mesmas regras descritas acima, porém a disputa é entre dois jogadores.

2.3.1 Classificação das Mãos

No pôquer, as combinações de cartas ou mãos, são classificadas melhores que outras baseando-se nos termos abaixo:

- O valor individual de cada carta: onde a carta de menor valor é o 2 e a de maior valor o Ás, apesar deste também poder assumir o valor de 1 no caso de sequências A-2-3-4-5;
- Naipes não tem valor: mãos idênticas com naipes diferentes são equivalentes;
- Uma mão sempre consiste da combinação de cinco cartas;
- Cartas são combinadas primeiramente por combinação, depois por valor individual: mesmo a menor mão de uma combinação possível vence todas as outras mãos de combinações menores.

3. METODOLOGIA

O agente tomador de decisões no jogo de pôquer *Texas Hold'em Heads-Up* apresenta um algoritmo desenvolvido no software *Matlab* que considera uma série de fatores para cada estágio do jogo, de acordo com a posição em que o jogador se encontra na mesa e a ação do jogador oponente. Sendo assim, são usadas várias regras de inferência de acordo com o estado em que o jogo se encontra. A Figura 2 mostra, através de um diagrama de blocos, como são feitas as tomadas de decisões no estágio *flop*.

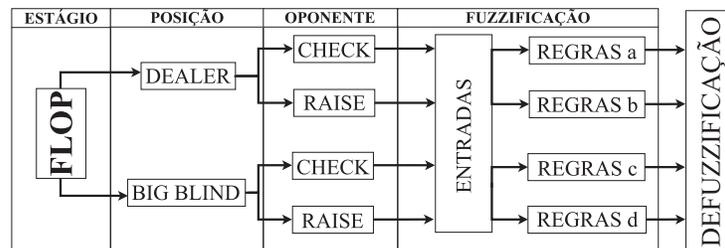


Figura 2. Esquema de tomada de decisão no *flop*.

3.1 Parâmetros e Variáveis Linguísticas

Além do estágio do jogo (*pré-flop*, *flop*, *turn* ou *river*), posição do jogador (*dealer* ou *big blind*) e ação do oponente (*fold*, *call*, *check* ou *raise*), os parâmetros utilizados como entrada para a tomada de decisão na jogada foram:

- Tamanho do pote: pequeno, médio ou grande;
- Tamanho da aposta do oponente: baixa, média ou alta;
- Fichas do jogador: poucas, médio ou muitas;
- Força da Mão: baixa, média ou alta.

Os pesos dados para cada parâmetro de entrada são diferentes e dependem do estágio da rodada. Em geral, parâmetros como posição e força da mão, possuem mais influência na tomada de decisão do que o tamanho do pote, por exemplo.

3.2 Regras de Inferência

Para cada estágio do jogo, existe uma respectiva regra de inferência. No total, o algoritmo é composto por 23 regras de inferência, cada uma considerando um tipo de estratégia. As regras foram definidas e adaptadas de literaturas relacionadas à modalidade (Moshman, 2010; Bello, 2009; Zadeh, 1977). A Figura 3 apresenta o conjunto das regras de inferência para o estágio *flop* e posição *dealer*, após uma aposta do oponente.

Se	MÃO	BAIXA					então	AÇÃO	FOLD
Se	MÃO	MÉDIA	FICHAS	POUCAS			então	AÇÃO	ALL IN
Se	MÃO	MÉDIA	FICHAS	MÉDIO	APOSTA	BAIXA	então	AÇÃO	CALL
Se	MÃO	MÉDIA	FICHAS	MÉDIO	APOSTA	MÉDIA	então	AÇÃO	CALL
Se	MÃO	MÉDIA	FICHAS	MÉDIO	APOSTA	ALTA	então	AÇÃO	FOLD
Se	MÃO	MÉDIA	FICHAS	MUITAS	APOSTA	BAIXA	então	AÇÃO	ALL IN
Se	MÃO	MÉDIA	FICHAS	MUITAS	APOSTA	MÉDIA	então	AÇÃO	CALL
Se	MÃO	MÉDIA	FICHAS	MUITAS	APOSTA	ALTA	então	AÇÃO	FOLD
Se	MÃO	ALTA	FICHAS	POUCAS			então	AÇÃO	ALL IN
Se	MÃO	ALTA	FICHAS	MUITAS			então	AÇÃO	ALL IN
Se	MÃO	ALTA	FICHAS	MÉDIO	APOSTA	BAIXA	então	AÇÃO	RAISE
Se	MÃO	ALTA	FICHAS	MÉDIO	APOSTA	MÉDIA	então	AÇÃO	CALL
Se	MÃO	ALTA	FICHAS	MÉDIO	APOSTA	ALTA	então	AÇÃO	ALL IN

Figura 3. Regras de inferência para o *flop*, na posição *dealer*, depois de uma aposta do oponente.

3.3 Saída Obtida

Para todos os casos, a saída obtida é um valor numérico entre 0 e 1, que representa a agressividade da ação a ser tomada pelo jogador. O algoritmo *defuzzifica*, ou seja, interpreta o valor obtido na saída e o relaciona a uma informação precisa para a tomada de decisão do jogador.

Desse modo, saídas próximas de 0 tendem resultar em ações passivas, como desistência de jogar a rodada (*fold*). Já saídas com valores próximos a 1 tendem a culminar em ações mais agressivas, como apostas (*raise*), apostas em cima de apostas (*re-raise*) ou até mesmo apostas de todas as fichas do jogador (*all in*).

3.4 Eficiência do Algoritmo

O método usado para verificar a eficácia do tomador de decisões desenvolvido, foi comparar os resultados obtidos de um jogador controlado pelo algoritmo e outros tipos de jogadores contra um mesmo adversário. Foram usados nessa comparação, além do tomador de decisões baseadas em lógica *fuzzy*, outros três tipos de jogadores:

- **Jogador Check/Call:** jogador que sempre paga (*call*) as apostas feitas pelo oponente, a não ser que este não tenha feito nenhuma aposta no estágio. Neste último caso, o jogador passa a jogada (*check*);
- **Jogador Raise:** jogador que sempre faz apostas (*raise*) em qualquer estágio do jogo. Além disso, quando o oponente faz alguma aposta ou aumenta uma aposta já feita, ele aumenta ainda mais;
- **Jogador Random:** jogador que toma decisões aleatórias a cada turno, ou seja, não tem um padrão de jogo e não define a ação baseando-se em algum fator.

3.5 Software de Pôquer Usado

O software usado para fazer o teste e a comparação do algoritmo do agente tomador de decisões no jogo de pôquer foi o *PokerTH*, um simulador gratuito da modalidade *Texas Hold'em* disponível em sistemas operacionais *Unix-like*, como *Linux*, *BSD*, *OS X*, *Android* e também no *Microsoft Windows*. O software permite que o jogador jogue contra oponentes controlados pela máquina ou outros usuários online.

A escolha deste software se deu por muitos fatores, principalmente por ser um software acessível, bastante conhecido, além apresentar facilidade na configuração dos jogos e qualidade dos programas que controlam os jogadores virtuais. Além disso, para jogadores iniciantes, o *PokerTH* é uma grande ferramenta para se adquirir experiência e aprender mais sobre o jogo.

Foram jogadas 50 partidas com cada tipo de jogador contra o virtual controlado pelo software, de modo que a análise feita a partir da comparação dos resultados fosse bastante ampla. É importante salientar que uma partida se completa quando um dos jogadores fica sem todas as fichas. Portanto, geralmente, são necessárias várias rodadas para que se chegue a uma partida completa.

4. RESULTADOS

A Tabela 1 apresenta os resultados de cada tipo de jogador contra o mesmo oponente controlado pelo software *PokerTH*. A tabela detalha o número médio de mãos jogadas a cada rodada e a porcentagem de vitórias e derrotas.

Tabela 1. Resultados de cada tipo de jogador contra o *PokerTH*.

Jogador	Média de Mãos	Vitórias	Derrotas
Random	16,0	10,0%	90,0%
Check/Call	9,68	20,0%	80,0%
Raise	5,58	32,0%	68,0%
Fuzzy	15,54	56,0%	44,0%

Pela análise da Tabela 1, pode ser observado que o jogador que apresentou o pior resultado foi o jogador *Random*, que define suas ações aleatoriamente. Esse jogador apresentou uma taxa de sucesso de 10,0%, vencendo apenas 5 mãos das 50 disputadas. O segundo pior caso foi o do jogador *Check/Call*, que sempre passa a jogada ou paga a aposta do oponente, apresentando uma taxa de fracasso de 80,0%, ou seja, o jogador perdeu 40 mãos das 50 jogadas. Já o jogador *Raise* obteve melhores resultados em relação aos anteriores, com uma taxa de sucesso de 32,0%, vencendo 16 mãos das 50 disputadas.

Finalmente, o jogador que toma decisões baseadas no algoritmo desenvolvido neste trabalho e leva em conta a lógica *fuzzy*, foi o que obteve não só o melhor resultado entre os jogadores analisados, mas o único a vencer mais do que ser derrotado pelo jogador virtual. Para esse caso, obteve-se uma taxa de sucesso de 56,0%, ou seja, das 50 partidas jogadas, o jogador *Fuzzy* venceu 28.

Outro fator descrito na Tabela 1 é o número médio de mãos jogadas a cada rodada por cada tipo de jogador enfrentando o mesmo oponente. Pode-se observar que esse parâmetro não tem influência significativa no resultado final, uma vez que tanto o pior quanto o melhor resultado apresentam números médios de mãos jogadas a cada rodada próximos. Como era de se esperar, o número médio de mãos jogadas dos jogadores com ações fixas (*Check/Call* e *Raise*) é menor devido à agressividade das ações.

5. CONCLUSÃO

Pela análise dos resultados, pode-se concluir que o objetivo do trabalho foi alcançado, uma vez que o resultado obtido pelo agente tomador de decisões baseadas em lógica *fuzzy* sugere que o algoritmo desenvolvido é melhor ou pelo menos tão bom quanto o usado para controlar o jogador virtual do software. Isso mostra que é uma ferramenta confiável e útil no aprendizado de jogadores iniciantes no jogo de pôquer.

Os casos dos jogadores *Random*, *Check/Call* e *Raise* lançam sugestões no sentido em que o jogo de pôquer não é definido apenas pelo fator sorte, mostrando que tomadas de decisões que não sejam baseadas em fatores como posição, número de fichas, força da mão, entre outras, apresentam grande chance de levar o jogador ao fracasso.

Além disso, os resultados para esses três tipos de jogadores sugerem que o fator agressividade tem influência nos resultados obtidos, uma vez que, para esses três casos, os melhores resultados foram obtidos com o tipo de jogador com ações mais agressivas.

Pretende-se no futuro associar este trabalho com Ziółko *et al.* (2012), de modo que o algoritmo seja capaz de não somente tomar decisões baseadas em lógica fuzzy, mas também usar as redes neurais para prever as ações do oponente. Além disso, pretende-se expandir o código para outras modalidades mais conhecidas como *Omaha* e *Five-Card Draw*, além do próprio *Texas Hold'em* para mais jogadores.

AGRADECIMENTOS

Agradecemos à CAPES, CNPq/INERGE, FAPEMIG e à Universidade Federal de São João del-Rei pelo apoio.

NOMENCLATURA

IA	Inteligência Artificial
U	Conjunto Universo
x	Elemento que pertencente ao Conjunto Universo
A	Subconjunto do Conjunto Universo
$\mu_A(x)$	Função de Pertinência com respeito ao conjunto A

REFERÊNCIAS

- Bello, L., 2008. *Aprendendo a Jogar Poker - Princípios, Técnica e Prática*. Nova Fronteira.
- Bello, L., 2009. *Dominando a Arte do Poker*. Nova Fronteira.
- Benjamim, A.G., Junior, J.J.P.F., Diniz, Y.S. e Gomes, I.R., 2012. “Abordagem híbrida fuzzy-bayesiana para um agente jogador de pôquer”. *XXXIV CNMAC, Águas de Lindóia - SP/Brasil*.
- Bilobrovec, M., Marçal, R.F.M. e Kovalski, J.L., 2004. “Implementação de um sistema de controle inteligente utilizando a lógica fuzzy”. *XI SIMPEP, Bauru/Brasil*.
- Braga, H.C. e Moita, G.F., 2014. “Modelagem computacional com lógica fuzzy da movimentação de pessoas: Determinação e relevância de aspectos ergonômicos cinemáticos”. *XI SIMMEC, Juiz de Fora - MG/Brasil*.
- Cruz, A.G.A., Lima Filho, A.C., Belo, F.A. e Silva, J.F., 2015. “Sistema híbrido para diagnósticos de falhas em motores de indução trifásicos com base no método vibracional, corrente de armadura e lógica fuzzy”. *XXXVI CILAMCE, Rio de Janeiro - MG/Brasil*.
- Cunha, F.L., Franca, J.E., Ortolan, R.L. e Junior Cliquet, A., 2007. “O uso de redes neurais artificiais para o reconhecimento de padrões em uma prótese mioelétrica de mão”. In: *VIII Congresso Brasileiro de Redes Neurais*.
- Gomide, F.A.C. e Gudwin, R.R., 1994. “Modelagem, controle, sistemas e lógica fuzzy”. *SBA controle & Automação*, Vol. 4, No. 3, pp. 97–115.
- Klir, G. e Yuan, B., 1995. *Fuzzy sets and fuzzy logic*, Vol. 4. Prentice Hall New Jersey.
- MathWorks, I. e Wang, W., 1998. *Fuzzy Logic Toolbox: for Use with MATLAB: User's Guide*. Mathworks, Incorporated.
- Moraes, O.B.d., 2008. *Método de análise de dados para avaliação de áreas urbanas recuperadas-uma abordagem utilizando a lógica fuzzy*. Tese (Doutorado), Universidade de São Paulo.
- Moshman, C., 2010. *Heads-Up: No-Limit Hold'em Poker*. Raise.
- Pedrycz, W. e Gomide, F., 1998. *An introduction to fuzzy sets: analysis and design*. Mit Press.

- Pinto, W.J.G.S., Silva, A.D.G., Cavalini Jr., A. e Steffen Jr., V., 2015. “Uncertainty analysis in a rotor system by using the fuzzy logic approach”. *XXXVI CILAMCE, Rio de Janeiro - MG/Brasil*.
- Rabelo, A.F. e Moita, G.F., 2014. “Aprimorar massa de testes com lógica fuzzy”. *XI SIMMEC, Juiz de Fora - MG/Brasil*.
- Rezende, S.O., 2003. *Sistemas inteligentes: fundamentos e aplicações*. Editora Manole Ltda.
- Sklansky, D., 1999. *The theory of poker*. Two Plus Two Publishing LLC.
- Zadeh, L.A., 1965. “Fuzzy sets”. *Information and control*, Vol. 8, No. 3, pp. 338–353.
- Zadeh, N., 1977. “Computation of optimal poker strategies”. *Operations Research*, Vol. 25, No. 4, pp. 541–562.
- Ziółko, B., Bochniak, D. e Jankowski, G., 2012. “Neural network application for automatic decisions in poker”. *Journal of Applied Computer Science*, Vol. 20, No. 1, pp. 119–127.

NOTA DE RESPONSABILIDADE

Os autores são os únicos responsáveis pelo material reproduzido nesse artigo.