

Fuzzy automaty

Martin Jašek

26. října 2014 — ??

Obsah

1	Základní pojmy	2
1.1	Množiny, relace, funkce	2
1.2	Uspořádání, uspořádané množiny, svazy	3
1.3	Symboly, řetězce a jazyky	3
2	Bivalentní automaty	4
2.1	Výpočetní modely, stavové stroje	4
2.2	Konečný bivaletní stroj	5
2.3	Konečný bivalentní automat	6
2.4	Konečný „bivalentní“ deterministický automat	6
3	Fuzzy automaty	6
3.1	Potřebné pojmy z teorie fuzzy množin	6
3.2	Nedeterministický fuzzy automat	7
3.3	Deterministický fuzzy automat	8
3.4	Nedeterministický fuzzy automat s ϵ -přechody	8
4	A tak dále ...	9
4.1	Zajímavé aplikace?	9
4.2	Další zdroje:	9

1 Základní pojmy

V této kapitole budou zopakovány základní pojmy (a především jejich značení) potřebné pro studium problematiky automatů. Pokud není řečeno jinak, definice a značení jsou převzaty z [1].

1.1 Množiny, relace, funkce

Množiny

Jako množinu označujeme kolekci objektů a značíme je velkými písmeny (M, A, B, \dots). Pokud objekt x náleží do množiny M , píšeme $x \in M$. V opačném případě píšeme $x \notin M$ a říkáme, že objekt x nenáleží do množiny M . Velikost (mohutnost, počet prvků) množiny M značíme $|M|$. Prázdnou množinu značíme \emptyset . Rovnost množin M a N značíme $M = N$. Podmnožinou M množiny N značíme $M \subseteq N$. Pro množiny M a N dále zavádíme $M \cap N$ (průnik) $M \cup N$ (sjednocení) $M \setminus N$ (rozdíl) a \overline{M} (doplnek).

Potenční množinu množiny M (systém množin, který je tvořen všemi podmnožinami množiny M) značíme 2^M . Symbol \mathbb{N} ($\mathbb{N}_0, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$) značí množinu všech přirozených (přirozených včetně nuly, celých, racionálních, reálných) čísel.

Relace

Kartézský součin množin M_1, \dots, M_n značíme $M_1 \times \dots \times M_n$ a jeho libovolnou podmnožinu nazýváme (n -ární) relace a značíme ji velkými písmeny (R, S, T). Pro $n = 2$ nazýváme relaci binární, pro $R \subseteq M \times M$ pak binární relací R na množině M . Binární relace R na množině M je:

- reflexivní, pokud pro každé $x \in M$ platí: $\langle x, x \rangle \in R$
- irreflexivní, pokud neexistuje žádné $x \in M$ takové, že: $\langle x, x \rangle \in R$
- symetrická, pokud pro každé $x, y \in M$ platí: jestliže $\langle x, y \rangle \in R$ pak $\langle y, x \rangle \in R$
- antisymetrická, pokud pro každé $x \in M$ platí: jestliže $\langle x, y \rangle \in R$ a $\langle y, x \rangle \in R$ pak $x = y$
- asymetrická, pokud neexistuje žádné $x, y \in M$ takové, že: jestliže $\langle x, y \rangle \in R$ pak $\langle y, x \rangle \in R$
- tranzitivní, pokud pro každé $x, y, z \in M$ platí: $\langle x, y \rangle \in R$ a $\langle y, z \rangle \in R$, pak $\langle x, z \rangle \in R$
- úplná, pokud pro každé $x, y \in M$ platí: $\langle x, y \rangle \in R$ nebo $\langle y, x \rangle \in R$

?? Uzávěry? Budu je používat?

Složení relací R a S značíme $R \circ S$ a inverzní relaci R^{-1} k relaci R .

Funkce

Relaci f nazýváme zobrazení (funkce) z množiny X (množin X_1, \dots, X_n) do množiny Y pokud pro každé $x \in X$ ($\langle x_1, \dots, x_n \rangle \in X_1 \times \dots \times X_n$) existuje právě jedno $y \in Y$ takové, že $\langle x, y \rangle \in f$ ($\langle x_1, \dots, x_n, y \rangle \in f$). Funkci f zkráceně zapisujeme $f : X \rightarrow Y$ ($f : X_1 \times \dots \times X_n \rightarrow Y$) případně $y = f(x)$ ($y = f(x_1, \dots, x_n)$).

Ekvivalence a rozklady

Symetrická, reflexivní a tranzitivní relace \equiv se nazývá ekvivalence. Rozklad Π množiny M je systém neprázdných disjunktních podmnožin množiny M . Třída rozkladu Π prvku $x \in M$ se značí $[x]_\Pi$.

1.2 Uspořádání, uspořádané množiny, svazy

Reflexivní, antisymetrická a tranzitivní relace $\leq \subset M \times M$ se nazývá (částečné) uspořádání. Místo $\langle x, y \rangle \in \leq$ píšeme $x \leq y$. Dvojice $\langle M, \leq \rangle$ se nazývá uspořádaná množina. Pokud pro každé $x, y \in M$ platí buď $x \leq y$ nebo $y \leq x$, pak $\langle M, \leq \rangle$ nazýváme úplné uspořádání (řetěz).

Mějme množinu $N \subset M$. Prvek $x \in N$ se nazývá minimální (maximální), pokud pro každé $y \in N$ takové, že $x \leq y$ ($y \leq x$), platí $x = y$. Prvek $x \in N$ se nazývá nejmenší (největší), pokud neexistuje žádné $y \in N$ takové, že $y \leq x$ ($x \leq y$). Pokud existuje nejmenší (největší) prvek, budeme jej značit 0 (1).

Dolní kužel $\mathcal{L}(A)$ (horní kužel $\mathcal{U}(A)$) je množina všech $x \in M$ takových, pro které platí $x \leq y$ ($y \leq x$) pro každé $y \in A$. Pokud má $\mathcal{L}(A)$ ($\mathcal{U}(A)$) nejmenší (největší) prvek, nazývá se A infimum (supremum) a značí se $\inf A$ nebo $\bigwedge A$ ($\sup A$ nebo $\bigvee A$).

Pokud má $\langle M, \leq \rangle$ infimum i supremum pro každou neprázdnou $A \subseteq M$, nazývá se svaz. Pokud svaz má infimum i supremum pro každou $A \subseteq M$ nazývá se úplný.

(zde bude doplněno: hustě uspořádaný svaz??)

1.3 Symboly, řetězce a jazyky

V této kapitole budou zdefinovány a rozebrány základní pojmy z teorie automatů. Značení bylo přejato z [4].

Abeceda

Základním pojmem při studiu automatů je abeceda. Abeceda je neprázdná množina symbolů a značí se velkými řeckými písmeny (nejčastěji Σ). Abecedou může být například „všechna velká písmena latinky“, množina $\{\neg, \rightarrow, (,), \varphi_1, \dots, \varphi_n\}$ (abeceda formulí výrokové logiky) nebo např. číslice 0 – 9 a symboly plus, mínus a tečka.

Řetězec

Posloupnost $u = a_1 a_2 \dots a_n$ kde $a_1, a_2, \dots, a_n \in \Sigma$ se nazývá řetězec u nad abecedou Σ . Číslo n je pak délka řetězce u , která se také značí $|u|$. Řetězec, který

má nulovou délku, značíme ε . Řetězcem nad abecedou všech velkých písmen latinky pak může být například *KMI* nebo *INFORMATIKA*.

Řetězec $u \circ v = a_1 \dots a_n b_1 \dots b_m$ (častěji však uv) se nazývá zřetězení (konkatenace) řetězců $u = a_1 \dots a_n$ a $v = b_1 \dots b_m$ a má délku $|uv| = n + m$. Jako n -tá mocnina u^n řetězce u se označuje řetězec:

$$u^n = \begin{cases} \varepsilon & \text{pokud } n = 0 \\ uu^{n-1} & \text{jinak} \end{cases}$$

Symbolem Σ^* se značí množina všech řetězců nad abecedou Σ (včetně ε). Symbol Σ^+ pak značí všechny řetězce nad abecedou Σ vyjma ε .

Jazyk

Pojmem (formální) jazyk se označuje určitá vybraná množina L řetězců nad abecedou Σ . Jazyk L nad abecedou Σ je tedy podmnožina Σ^* .

Nad jazyky L , L_1 a L_2 nad abecedami Σ , Σ_1 a Σ_2 se zavádí:

$$L_1 L_2 = \{uv \mid u \in L_1, v \in L_2\} \quad \text{zřetězení (produkt)}$$

$$L^n = \begin{cases} \{\varepsilon\} & \text{pokud } n = 0 \\ LL^{n-1} & \text{jinak} \end{cases} \quad n\text{-tá mocnina}$$

$$L^* = \bigcup_{i=0}^{\infty} L^i \quad \text{Kleeneho uzávěr}$$

$$L^+ = \bigcup_{i=1}^{\infty} L^i \quad \text{pozitivní uzávěr}$$

(zde bude doplněno: ještě něco?)

2 Bivalentní automaty

V této kapitole je intuitivně zaveden pojem „automat“. Poté následuje formální definice klasického, bivalentního konečného automatu, a to jak deterministického, tak neterministického. Pro úplnost bude také uveden konečný bivalentní stroj. Definice a pojmy související s konečnými bivalentními automaty byly převzaty z [4], terminologie související s konečnými bivalentními stroji byla přejata z [6].

Poznámka: Jak bude v následujících kapitolách ukázáno, mezi pojmy „konečný bivalentní automat“ a „konečný bivalentní stroj“ je určitý rozdíl. Pro zjednodušení však budou souhrnně nazývány „bivalentní automaty“.

Důkladné nastudování těchto pojmů je důležité, protože jak fuzzy automaty, tak pravděpodobnostní automaty z bivalentních automatů vycházejí a sdílejí z velké části jejich principy.

2.1 Výpočetní modely, stavové stroje

Před tím, než bude automat zadefinován formálně je vhodné si nejprve uvědomit jeho intuitivní význam. Automat je jeden z mnoha formálních nástrojů pro řešení problémů. Takovýmto nástrojům se obecně říká výpočetní modely.

Výpočetní model je obecný formální mechanismus, který řeší rozhodovací problém. Rozhodovací problém P je problém určení, zda-li řetězec u (vstupní instance problému P) náleží do množiny přípustných řešení pro problém P (zde bude doplněno: zdroj?). Kromě automatů se mezi výpočetní modely řadí například Turingův stroj a λ -kalkul.

Dalším označením, které automat nese je stavový stroj. Stavový stroj je systém (nemusí to být výpočetní model), u kterého lze uvažovat nějaký vnitřní stav. Mezi těmito stavy poté bývají definovány přechody, pomocí kterých systém přechází z jednoho stavu do jiného (nebo stávajícího). Mezi stavové stroje lze kromě automatů zařadit například Turingův stroj nebo buněčný automat [4].

2.2 Konečný bivaletní stroj

Jak bude ukázáno v následující kapitole, konečný bivaletní automat je speciálním případem konečného bivaletního stroje. Je tedy na místě začít nadefinováním konečného bivaletního stroje.

Konečný bivaletní stroj je výpočetní model, který disponuje vstupní a výstupní páskou. Na vstupní pásce jsou čteny symboly vstupní abecedy, podle kterých se rozhodují přechody mezi stavy a také, jaké symboly budou zapsány na výstupní pásku. Výstupem chodu konečného bivaletního stroje je tedy řetězec nad výstupní abecedou.

Definice 2.1 (Konečný bivaletní stroj[6]). *Konečný bivaletní stroj je šestice $\mathcal{M} = \langle Q, \Sigma, \Upsilon, \delta, \sigma, s \rangle$ kde*

*Q je konečná množina stavů
 Σ je konečná vstupní abeceda
 Υ je konečná výstupní abeceda
 $\delta : Q \times \Sigma \rightarrow Q$ je přechodová funkce
 $\sigma : Q \times \Sigma \rightarrow \Upsilon$ je výstupní funkce
 $s \in Q$ je počáteční stav*

Poznámka: Takto definovaný automat je ve skutečnosti Mealeho automat [5].

Přechodová funkce δ nám říká, že nachází-li se stroj \mathcal{M} ve stavu $q \in Q$ a na vstupu je $x \in \Sigma$, tak stroj přejde do stavu $q' \in Q$ takového, že $\delta(q, x) = q'$. Obdobně - výstupní funkce σ na výstup zapíše symbol $y \in \Upsilon$ právě když se automat \mathcal{M} nachází ve stavu $q \in Q$, na vstupu je symbol $x \in \Sigma$ a $\sigma(q, x) = y$.

Takto získáme přechod pro jeden vstupní symbol. Jak probíhá výpočet stroje pro celé slovo, tedy jak pro vstupní slovo stroj určí výstupní slovo určuje následující definice:

Definice 2.2 (Výstupní řetězec[6]). *Mějme konečný bivaletní stroj $\mathcal{M} = \langle Q, \Sigma, \Upsilon, \delta, \sigma, s \rangle$. Pak řetězec $y = y_1 \dots y_n \in \Upsilon^*$ se nazývá výstupní řetězec automatu \mathcal{M} pro vstupní řetězec $x = x_1 \dots x_n \in \Sigma^*$ pokud existují stavy $q_0, \dots, q_n \in Q$ takové, že:*

$$\begin{aligned} q_0 &= s \\ q_i &= \delta(q_{i-1}, x_i) \quad \text{pro } i = 1, \dots, n \\ y_i &= \sigma(q_{i-1}, x_i) \quad \text{pro } i = 1, \dots, n \end{aligned}$$

Stroj \mathcal{M} na svém vstupu čte řetězec x . Stroj postupně čte jednotlivé symboly řetězce x a mění svou konfiguraci na základě přechodové a výstupní funkce

následovně: Pro symbol vstupní abecedy x_i přejde ze stavu q_{i-1} do stavu q_i pokud $\delta(q_{i-1}, x) = q_i$ a na výstup zapíše y_i z výstupní abecedy, pokud $\sigma(q_{i-1}, x_i) = y_i$. Výsledkem je slovo y zapsané na výstupu.

(zde bude doplněno: Příklad?)

2.3 Konečný bivalentní automat

Uvažujme nyní stroj $\mathcal{M} = \langle Q, \Sigma, \Upsilon, \delta, \sigma, s \rangle$, jehož výstupní abeceda je dvouprvková a obsahuje hodnoty nula a jedna, tedy $\Upsilon = \{0, 1\}$. Dále předpokládejme, že všechny stavy $q_i \in Q$ splňují vlastnost, že při přechodu do stavu q_i je výstup zapsán 0 nebo 1 a tato hodnota je stejná pro všechny přechody do tohoto stavu. Podle toho, zda-li je tato hodnota 0 nebo 1 můžeme o stavu q_i říci, zda-li je koncový nebo nekonečný.

Dle [6] je takovýto stroj konečným bivalentním automatem. Pro účely této práce však bude použita definice z [4], která je však s definicí pomocí stroje na první pohled ekvivalentní.

Následuje nadefinování konečného deterministického automatu.

2.4 Konečný „bivalentní“ deterministický automat

Definice 2.3 (Konečný deterministický automat[?]). (zde bude doplněno: FJAA)

Dále se zavádí pojmy konfigurace a výpočet.

Definice 2.4 (Konfigurace[?]). Konfigurace c konečného stroje \mathcal{M} je libovolný prvek z množiny $\Sigma^* \times Q$.

(zde bude doplněno: značení, zdroj = FJAA)

Definice 2.5 (Výpočet[?]). Výpočet c konečného stroje \mathcal{M} je posloupnost konfigurací c_1, c_2, \dots, c_n .

3 Fuzzy automaty

3.1 Potřebné pojmy z teorie fuzzy množin

V této kapitole se nachází výpis základních definic z teorie fuzzy logiky a fuzzy množin nutný pro práci s fuzzy automaty. Definice byly přejaty z [10] a značení případně upraveno tak, aby korespondovalo se značením v kapitole 1.

Fuzzy množina je kolekce objektů zobecňující „klasickou“ množinu. Uvažuje se nad univerzem X (nebo jeho libovolnou neprázdnou?? podmnožinou) a u každého objektu z tohoto univerza se uvádí, jak moc do fuzzy množiny náleží. K tomu slouží tzv. charakteristická funkce. To je zobrazení f_A z univerza X do intervalu $[0, 1]$ ¹.

Pokud je pro $x \in X$ $f_A(x) = 0$, pak objekt do fuzzy množiny A jistě nenáleží, pokud je $f_A(x) = 1$, pak objekt do fuzzy množiny A jistě náleží (zde bude doplněno: jak je český překlad?). Pokud je $0 < f_A(x) < 1$, pak říkáme, že objekt x do fuzzy množiny A náleží ve stupni $f_A(x)$.

Pokud je $f_A(x) = 0$ pro každé $x \in X$, pak fuzzy množinu A nazýváme prázdnou fuzzy množinou. Dvě fuzzy množiny A a B se rovnají, pokud $f_A = f_B$

¹obecně do libovolného svazu??

(pro každé $x \in X$ platí $f_A(x) = f_B(x)$). Fuzzy množina A je podmnožinou fuzzy množiny B (píšeme $A \subseteq B$) pokud je $f_A \leq f_B$.

Mějme fuzzy množiny A a B s charakteristickými funkcemi f_A a f_B . Pak definujeme $A \cap B$ (průnik fuzzy množin), $A \cup B$ (sjednocení fuzzy množin) a \overline{A} (doplňěk fuzzy množiny) jako fuzzy množiny s charakteristickými funkcemi $f_{A \cap B}$ a $f_{A \cup B}$, $f_{\overline{A}}$ takovýmito:

$$\begin{aligned} f_{A \cap B}(x) &= \min(f_A(x), f_B(x)) \\ f_{A \cup B}(x) &= \max(f_A(x), f_B(x)) \\ f_{\overline{A}}(x) &= 1 - f_A(x) \end{aligned}$$

Množina všech \mathbb{L} -množin nad univerzem X se značí \mathbb{L}^X .

(zde bude doplněno: Pro průnik a sjednocení platí De Morganovy zákony a zákony distributivity.?? (zde bude doplněno: a tak dále))

(zde bude doplněno: Fuzzy relace, fuzzy zobrazení)

[3]

Definice 3.1 (Max-min kompozice[7], strana 2). (zde bude doplněno: nějak to pořešit, moc nerozumím zápisu té definice) Max-min kompozice \mathbb{L} -relací $R \in \mathbb{L}^X$ a $T \in \mathbb{L}^Z$ je relace $R \circ T \in \mathbb{L}^{X \times Z}$ taková, že

$$(R \circ T)(\langle x, z \rangle) = \bigvee_{y \in Y} (R(x, y) \wedge S(y, z))$$

pro libovolná $\langle x, z \rangle \in X \times Z$.

3.2 Nedeterministický fuzzy automat

Nyní přejdeme k definici fuzzy automatu. Vzhledem k tomu, že teorie fuzzy množin pracuje s určitou formou neurčitosti, bude proto vhodné definovat nejprve fuzzy automat nedeterministický.

Stejně tak, jako jsou fuzzy množiny zobecněním klasických množin, očekáváme, že i fuzzy automat bude zobecněním klasického automatu. Současně však stále vyžadujeme jeho konečnost. Jeho abeceda i množina stavů tedy musí být konečné „klasické“ množiny. Stupně pravdivosti se proto budou uvažovat pouze u počátečních a koncových stavů a u přechodové funkce.

Definice 3.2 (Fuzzy automat[1]). Mějme konečný?? reziduovaný svaz \mathbb{L} a konečnou abecedu Σ . Pak \mathbb{L} -automat \mathcal{M} nad abecedou Σ je pětice $\mathcal{M} = \langle Q, \Sigma, Q_I, Q_F, \delta \rangle$ kde

Q je konečná množina stavů

Q_I je \mathbb{L} -množina nad Q počátečních stavů

Q_F je \mathbb{L} -množina nad Q koncových stavů

δ je \mathbb{L} -relace $\mathbb{L}^{Q \times \Sigma \times Q}$?? (přechodová funkce)

Tato definice je v souladu s našimi požadavky. Každému stavu q z konečné množiny stavů Q je přiřazen stupeň $Q_I(q)$, v jakém je tento stav počáteční a stupeň $Q_F(q)$ jeho koncovosti. Přechodová funkce δ má význam: „Po přečtení symbolu a přejdi ze stavu q do stavu q' ve stupni pravdivosti $\delta(q, a, q')$ “.

Vzhledem k podobě definice nedeterministického fuzzy automatu a klasického automatu

Definice 3.3 (Rozšířená přechodová funkce[?]). Pro \mathbb{L} -automat \mathcal{M} s přechodovou funkcí δ definujeme rozšířenou přechodovou funkci jako \mathbb{L} -relaci $\delta^* \subseteq Q \times \Sigma^* \times Q$ definovanou:

$$\delta^*(q, \alpha, q') = \bigvee_{q_0=q, q_1, \dots, q_n=q' \in Q} \delta(q_0, s_1, q_1) \wedge \dots \wedge \delta(q_{n-1}, s_n, q_n)$$

kde $\alpha = s_1 s_2 \dots s_n$ a $s_1, \dots, s_n \in \Sigma$.

Definice 3.4 (Jazyk rozpoznávaný automatem[?]). Pro \mathbb{L} -automat \mathcal{M} definujeme \mathbb{L} -jazyk $\mathcal{L}(\mathcal{M})$ rozpoznávaný automatem \mathcal{M} jako \mathbb{L} -množinu:

$$(\mathcal{L}(\mathcal{M}))(\alpha) = \bigvee_{q, q' \in Q} Q_I(q) \wedge \delta(q, \alpha, q') \wedge Q_F(q')$$

Poznámka 3.1. Místo \mathbb{L} -automat \mathcal{M} se bude nadále používat značení automat \mathcal{M} (za předpokladu, že není nutné uvádět \mathbb{L}).

(zde bude doplněno: bacha na značení jazyka (v textech o automatech).
!!!)

3.3 Deterministický fuzzy automat

Obdobně, jako u klasických automatů, lze i u fuzzy automatů zavést deterministický automat. Je zjevné, že k dosažení determinizace bude nutné omezit množinu počátečních stavů jen na právě jeden stav.

3.4 Nedeterministický fuzzy automat s ϵ -přechody

Viz [9].

4 A tak dále ...

4.1 Zajímavá aplikace?

L-Systémy (Lindenbayer?)(grafické znázornění stonků?). HP Modely (predikce modelů proteinových struktur?) [Learning Classifier Systems in Data Mining. Larry Bull, Ester Bernadó-Mansilla, John Holmes. 2008].

4.2 Další zdroje:

Původní zmínka [8], Definice Fuzzy množin NEpoužít z [?] (použít [?]?). BelKru [2]

Bělohávek - determinismus fuzzy matu: [7].

Další materiály na fuzzy množiny: [51, 108, 110, 114, 115, 266].

Pseudoautomaton (45) Automaton (46)

Max-min automata (48+++) Asynchronous max-product automaton (111+)

Max-product automaton (114+)

Fuzzy automaton (393) Fuzzy tree automaton (158), Fuzzy finite state machine (237) Finite fuzzy automaton (463, 404, 367) Finite quasi-fuzzy automaton (384) Subfuzzy finite state machine (305) Partial fuzzy automaton (325) \mathbb{L} -automaton (415)

Probabilistic automata (173 - 175) Probabilistic pushdown automaton (218+)

Probabilistic pseudoautomaton (223) Deterministic pseudoautomaton (233)

Weighted automaton (449) Subautomaton (503) automaton with tolerance (504)

https://www.researchgate.net/figure/4151164_fig7_Fig-7-State-Transition-Graph-of-the-Fuzzy-Automaton

<https://books.google.cz/books?id=8ONbAwAAQBAJ&pg=PA44&lpg=PA44&dq=max-min+fuzzy+automata&source=bl&ots=5Wat8DeC1l&sig=hA8wcbWeXOtUOen2YF5gD9kMbmM&hl=cs&btnBs-vJAhXJvxQKHVnCCokQ6AEIVTAH#v=onepage&q=automata&f=false>

Reference

- [1] Radim Bělohlávek. *Fuzzy Relational Systems: Foundations and Principles*. Kluwer, New York, 2002.
- [2] Radim Bělohlávek and Michal Krupka. On approximate minimalization of fuzzy automata. *Journal of Multiple-Valued Logic and Soft Computing*, 2009.
- [3] George J. Klir and Bo Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall P T R, 1995.
- [4] Dexter C. Kozen. *Automata and Computability*. Springer Science & Business Media, 1977.
- [5] G. H. Mealy. A method for synthesizing sequential circuits. 1955.
- [6] John N. Mordeson and Davender S. Malik. *Fuzzy Automata and Languages: Theory and Applications*. Chapman and Hall/CRC, 2002.
- [7] Bělohlávek Radim. Determinism and fuzzy automata. *Information Sciences*, 2002.
- [8] William G. Wee. Organization of adaptive algorithm and application of the fuzzy sets concept to pattern classification (disertační práce). *Prude University*, 1967.
- [9] Yoshinori Ezawa. Yongzhi Cao. Nondeterministic fuzzy automata. *Information Sciences*, 2012.
- [10] Lotfi A. Zadeh. Fuzzy sets *. *Published by Elsevier*, 1965.