AUTOMATA SYSTEMS COMMUNICATING BY TRANSITIONS

Patrik Petřík
Master Degree Programme (1), FIT BUT
E-mail: xpetri11@stud.fit.vutbr.cz

Supervised by: Alexander Meduna
E-mail: meduna@fit.vutbr.cz

ABSTRACT

This paper introduces automata systems communicating by transitions, whose components are finite state automata. Communication is based on transition rules, when one compoment can change his state according to state of other component. In conclusion are discussed their properties compared with grammar systems or grammars especially with parallel grammars and so we show the power of these systems.

1. ÚVOD

V tomto článku definujeme automatové systémy. Nejprve jsou uvedeny definice těchto systémů a poté i příklad. V závěru jsou zmíněny vlastnosti těchto systémů a jejich srovnání s gramatickými systémy či gramatikami.

2. ROZBOR

Základní terminologii z oblasti formálních jazyků naleznete v [1]. Další nutné pojmy pro pochopení dalšího jako jsou gramatické systémy, paralení gramatiky a v neposlední řadě i automatové systémy je možné nalézt v [1, 2, 3].

2.1. DEFINICE NOVÝCH POJMŮ

Paralelně komunikující automatový systém komunikující přechody, jehož komponenty jsou konečné automaty (*PC AS FSA*) je (*n*+1)-tice: $AS = (\Sigma, A_1, ..., A_n)$, kde Σ je vstupní abeceda, $A_i = (Q_i, \Sigma_i, R_i, s_i, F_i)$, $1 \le i \le n$ jsou konečné automaty (komponenty systému), kde Q_i je konečná množina stavů dané komponenty, $\Sigma_i = \Sigma$ je vstupní abeceda, $R_i = \{pa \to q \mid p, q \in Q_i, a \in \Sigma_i \cup \{\varepsilon\}\} \cup \left\{ p \xrightarrow{c} q \mid p, q \in Q_i, c \in \bigcup_{j=1}^{n} Q_j \right\}$ je přechodová funkce obsahující „normální“ pravidla typu $pa \to q$ a „komunikační“ pravidla typu $p \xrightarrow{c} q$, $s_i \in Q_i$ je počáteční stav, $F_i \subseteq Q_i$ je množina koncových (akceptujících) stavů a platí $\bigcap_{j=1}^{n} Q_j = \phi$.

Nechť $AS = (\Sigma, A_1, ..., A_n)$ je *PC AS FSA*(*n*). **Konfigurace** *PC AS FSA*(*n*) *AS* je řetězec $\chi = q_1 v_1 q_2 v_2 ... q_n v_n$, kde $q_i \in Q_i, v_i \in \Sigma^*, i \in \{1, ..., n\}$.

Nechť $AS = (\Sigma, A_1, ..., A_n)$ je *PC AS FSA*(*n*). Množina **komunikačních** stavů *i*-té komponenty, tedy stavů, ze kterých je možno provést komunikační přechod, je množina $C_i = \left\{ q \mid q \in Q_i \wedge \exists (q \xrightarrow{c} r) \in R_i, c \in \bigcup_{j=1}^{n} Q_j, r \in Q_i \right\}$.

Nechť $AS = (\Sigma, A_1, ..., A_n)$ je *PC AS FSA*(*n*). Množina **dotazovaných** stavů *i*-té komponenty je množina $D_i = \left\{ c \mid c \in Q_i \wedge \exists j \mid (q \xrightarrow{c} r) \in R_j, q, r \in Q_j, j \in \{1, ..., n\} \right\}$. Jsou to takové stavy, na které se může některá komponenta teoreticky dotázat, tj. existuje u jiné komponenty komunikační přechod, při němž je dotazováno na tento stav.

Nechť $q_1 v_1 q_2 v_2 ... q_n v_n$ a $q_{1'} v_{1'} q_{2'} v_{2'} ... q_{n'} v_{n'}$ jsou dvě konfigurace *PC AS FSA*(*n*) *AS*, kde $q_i, q_{i'} \in Q_i, a_i \in \Sigma \cup \{\varepsilon\}, v_i = a_i v_{i'}, v_i, v_{i'} \in \Sigma^*$ pro všechna $i, 1 \le i \le n$. Potom *AS* může provést **přechod** z konfigurace $q_1 v_1 q_2 v_2 ... q_n v_n$ do $q_{1'} v_{1'} q_{2'} v_{2'} ... q_{n'} v_{n'}$ za použití přechodů $r_1, r_2, ..., r_n$, zapsáno $q_1 v_1 q_2 v_2 ... q_n v_n \Rightarrow q_{1'} v_{1'} q_{2'} v_{2'} ... q_{n'} v_{n'} [r_1, r_2, ..., r_n]$ nebo zjednodušeně $q_1 v_1 q_2 v_2 ... q_n v_n \Rightarrow q_{1'} v_{1'} q_{2'} v_{2'} ... q_{n'} v_{n'}$, pokud platí jeden z následujících případů:

1. Pro každé *i*, $1 \le i \le n$ platí, že $q_i \notin C_i$ a $\exists r_i = (q_i a_i \to q_{i'}) \in R_i$.
2. Pro všechna j, $1 \le j \le n$, pro které $q_j \notin C_j$ existuje $r_j = (q_j a_j \to q_{j'}) \in R_j$ a pro každé i, $1 \le i \le n$,pro které platí $q_i \in C_i$ platí buď:
   1. $r_i = \left(q_i \xrightarrow{c} q_{i'} \right) \in R_i$ a existuje k, $1 \le k \le n$, pro které platí $c = q_k$ a $q_k \notin C_k$ a $r_k = (q_k a_k \to q_{k'}) \in R_k$ potom $a_i = \varepsilon$ nebo
   2. $r_i = \left(q_i \xrightarrow{c} q_{i'} \right) \in R_i$ a existuje k, $1 \le k \le n$, pro které platí $c = q_{k'}$ a $r_k = \left(q_k \xrightarrow{q_m} q_{k'} \right) \in R_k$, pro nějaké $m \in \{1, ..., n\}$ potom $a_i = \varepsilon$ nebo
   3. $r_i = \left(q_i \xrightarrow{c} q_{i'} \right) \in R_i$ a existuje k, $1 \le k \le n$, pro které platí $c = q_k$ a $r_k = (q_k a_k \to q_{k'}) \in R_k$ potom $a_i = \varepsilon$.
   4. Pokud není splněná žádná z uvedených možností, musí platit, že $r_i = (q_i a_i \to q_{i'}) \in R_i$.

V prvním případě žádná z komponent není schopna provést komunikační přechod.

Druhý případ je složitější neb existují komponenty, které jsou ve stavech, v nichž jsou schopné komunikovat, tj. dotazovat se na stavy jiných komponent. Z definice plyne, že nejprve se upřednostní komunikace a teprve po komunikaci provedou přechody i zbylé komponenty. Přitom je rozlišeno několik případů, které mohou při komunikaci nastat.

Nechť $AS = (\Sigma, A_1, ..., A_n)$ je *PC AS FSA*(n). **Jazyk přijímaný paralelně komunikujícím automatovým systémem** *AS*, L(*AS*), je definován: $L(AS) = \{w | s_1 v_1 s_2 v_2 ... s_n v_n \Rightarrow^* f_1 q_2 ... q_n, w \in \Sigma^*, v_1 v_2 ... v_n = w, f_1 \in F_1, q_i \in Q_i, i \in \{2,...,n\}\}$.

2.2. PŘÍKLAD

Nechť $AS = (\{a,b,c\}, A_1, A_2, A_3)$ je *PC AS FSA*, kde $A_1 = (\{Q_{11}, Q_{12}, Q_{13}, Q_{f1}\}, \{a,b,c\}, R_1, Q_{11}, \{Q_{f1}\})$, $R_1 = \{Q_{11}a \rightarrow Q_{11}, Q_{11}\varepsilon \rightarrow Q_{12}, Q_{12} \xrightarrow{Q_B} Q_{13}, Q_{13} \xrightarrow{Q_C} Q_{f1}\}$, $A_2 = (\{Q_{21}, Q_B, Q_{22}, Q_{23}\}, \{a,b,c\}, R_2, Q_{21}, \phi)$, $R_2 = \{Q_{21}b \rightarrow Q_{21}, Q_{21}\varepsilon \rightarrow Q_B, Q_B\varepsilon \rightarrow Q_{22}, Q_{22}\varepsilon \rightarrow Q_{23}\}$,
$A_3 = (\{Q_{31}, Q_{32}, Q_C, Q_{33}\}, \{a,b,c\}, R_3, Q_{31}, \phi)$, $R_3 = \{Q_{31}c \rightarrow Q_{31}, Q_{31}\varepsilon \rightarrow Q_{32}, Q_{32}\varepsilon \rightarrow Q_C, Q_C\varepsilon \rightarrow Q_{33}\}$.

AS má tedy tři komponenty, je centralizovaný, první komponenta je tzv. master a jazyk generovaný tímto systémem je $L(AS) = \{a^n b^n c^n \mid n \geq 0\}$.Tento systém příjme slovo *aaabbbccc* následovně: $Q_{11}aaaQ_{21}bbbQ_{31}ccc \Rightarrow Q_{11}aaQ_{21}bbQ_{31}cc \Rightarrow Q_{11}aQ_{21}bQ_{31}c \Rightarrow \Rightarrow Q_{11}Q_{21}Q_{31} \Rightarrow Q_{12}Q_BQ_{32} \Rightarrow Q_{13}Q_{22}Q_C \Rightarrow Q_{f1}Q_{23}Q_{33}$.

3. ZÁVĚR

Jak vidíme z příkladu mají tyto systémy větší vyjadřovací sílu než regulární gramatiky. Konkrétně se jejich síla rovná vyjadřovací síle tzv. paralelních gramatik uvedených v [2]. Stejně jako tyto gramatiky tvoří hierarchickou strukturu, tak i námi navržené systémy tvoří hierarchickou strukturu. Tedy přidáním další komponenty zvýšíme sílu systému, z čehož také plyne, že tak jako u některých gramatických systému lze minimalizovat počet komponent se zachováním generovaného jazyka, tak u těchto automatových systému toto neplatí. Více o těchto systémech lze nalézt v práci [4], která tyto systémy studuje podrobněji.

PODĚKOVÁNÍ

Na tomto místě bych rád poděkoval prof. RNDr. Alexandru Medunovi, CSc.

LITERATURA

[1] Meduna, A.: *Automata and Languages, Theory and Applications*, Springer, London, 2000

[2] Wood, D.: *m-parallel n-right linear simple matrix languages*, in: Utilitas Mathematica, Vol. 8, 1975, s. 3-28

[3] Martín-Vide, C., Mateescu, A., Mitrana, V.: *Parallel finite automata systems communicating by states*, in: International Journal of Foundations of Computer Science, Vol. 13, No. 5, 2002, s. 733-749

[4] Patrik Petřík: Automatové systémy [bakalářská práce], Brno, FIT VUT v Brně, 2007