Journal of Engineering and Applied Sciences 3 (11): 808-810, 2008
ISSN: 1816-949X

Parallel Communicationing Parallel Access Automata Systems with External Memory

M. Ramakrishnan and S. Balasubramanian
Department of Computer Science and Engineering,
Anna University Coimbatore, Coimbatore-641013, India

Abstract: In this study, we introduced parallel communicating parallel access memory automaton with external memory. It works independently and communication between states based on their request. We proved that the whole contents can be modified during single transition and this modal is more powerful than parallel communicating finite automata systems.

Key words: Finite automata, parallel communicating automata, multi head automata, external memory, singal transition

INTRODUCTION

Multiprocessor automata system consist of several finite automata, called processors (Budo, 1977; Hartmanis, 1972), which are coordinated by a central processing unit and it decides, which processor is to become active or frozen at a given steps. The processors works independently from the other ones based on the internal transition function which depends on the internal state and current input symbols. The states achieved by the processors depend on their current input symbol and current state. Parallel communicating finite automata systems are finite collections of automata working independently but communicating their states to each other by request (Sakthibalan *et al*., 2003). In this study, we introduced parallel communicating parallel access memory automata with external memory for parallel commutation. In this modal, each automaton is permitted to the states of all automata and know the memory status. In this modal, we assumed X is a parallel memory which means that its whole contents can be modified during a single write operation and can determine state transition through a single read operation. This modal reduces space and time also. The communicating strategy is similar to the earlier systems (Vide *et al*., 2002; Sudborough, 1977).

PARALLEL COMMUNICATING PARALLEL ACCESS MEMORY AUTOMATION

In this study, we introduced a new model parallel communicating finite automata with external memory.

Parallel communicating parallel access memory automation system is a construct

$$A = (V_1, A_1, \ldots..An, K, X)$$

where,

V = The input alphabet

$$Ai = (Q_i, V_1, f_i, q_i, F_i, X_i), 1 \le i \le n$$

are finite automata with its set of states Q_i, q_i belongs to Q_i and F_i is a subset of Q_i and f_i is the transition function of the automation i and it is defined as

$$f_i: Q_i \times (V \cup \{\epsilon\}) \rightarrow 2^{Q_i} \times X$$

$K = \{K_1 \ldots\ldots K_n\}$ is a subset of union of Q_i the set of query states and $\{0, 1\}^N$ be the memory configuration. The automata $A_1, A_2 \ldots\ldots A_n$ are the elements of A. The system is said to be centralized if $k \le Qi$, $1 \le i \le n$.

Whenever a system is centralized, the first component is the master and automation system is said to be deterministic. If the following conditions are satisfied

$$|f_i(s, a, z_i)| \le 1 \text{ for all } s \in Q_i \text{ and } a \in V\{\epsilon\}, z_i \in X_i \quad \text{(i)}$$

$$\text{If } |f_i(s, a, z_i)| \neq 0 \text{ for some } s \in Q_i, \text{ then } |f_i(s, a, x_i)| = 0 \text{ (ii)}$$
$$\text{for all } a \in V$$

where, $i \in [1, n]$

Confirmation of parallel communicating external memory automation is $(s_1, x_1, z_1, s_2, x_2, z_2, \ldots\ldots s_n, x_n, z_n)$ is a 3n tuple.

Corresponding Author: M. Ramakrishnan, Department of Computer Science and Engineering, Anna University Coimbatore, Coimbatore-641013, India

where:

s_i = The current sate of the component T

X_i = The remaining part of the input word which has not been read yet, by the number i, i ϵ [1, n]

Now, we define the confirmation of the system A is $(s_1, x_1, z_1, s_2, x_2, z_2, \ldots\ldots s_n, x_n, z_n)$ $(p_1, y_1, m_1, p_2, y_2, m_2, \ldots\ldots p_n, y_n, m_n)$. One of the following two conditions hold

$$k \cap \{s_1, \ldots s_n\} = 0 \text{ and } x_i = a_i y_i m_i,\ a_i \in V \cup \{\epsilon\},\ p_i \in f_i(s_i, a_i, z_i),\ 1 \le i \le n \quad \text{(i)}$$

for all $1 \le i \le n$ such that $s_i = k_i, j_i$ and $s_{ji} \in K$ (ii)

put $p_i = s_{ji}$, $p_r = s_r$, $1 \le i \le n$ and $y_t = x_t$, $1 \le t \le n$

$$(s_1, x_1, z_1, s_2, x_2, z_2, \ldots\ldots s_n, x_n, z_n) \text{ to } (p_1, y_1, m_1, p_2, y_2, m_2, \ldots\ldots p_n, y_n, mn)$$

One of the following conditions are holds

$$k \cap \{s_1, s_2, \ldots., s_n\} = 0 \text{ and } x_i = a_i y_i m_i,\ a_i \in V \cup \{\epsilon\},\ p_i \epsilon f_i(s_i, a_i, z_i),\ i \epsilon [1, n] \quad \text{(i)}$$

for $1 \le i \le n$ such that $s_i = k_{ji}$ and $s_{j\,i} \epsilon\ k$ (ii)

put $p_i = s_{ji}$, $p_r = s_r$

$p_{ji} = q_{ji}$, $V_1 \le t \le r)$,

and $y_i = x_t$, $1 \le t \le n$

The language accepted if a PCFAS, A consists of all strings $x \epsilon V$ such that the system starts in an initial confirmation $(q_1, x, z_1, q_2, x, z_2, \ldots., q_n, x, z_n)$ and reuses the final configuration is $(s_1, \epsilon, \epsilon, s_2, \epsilon, \epsilon, \ldots\ldots, s_n, \epsilon, \epsilon)$ with $s_i \epsilon F_i$:

$$\text{Recem}(A) = \{x \in V \mid (q_1, x, z_1, q_2, x, z_2, \ldots., q_n, x, z_n) (s_1, \epsilon, \epsilon, s_2, \epsilon, \epsilon, \ldots..s_n, \epsilon, \epsilon)$$

where, $S_i \in F_i$, $1 \le i \le n$

$$\text{Recem}_r(A) = \{x \epsilon V \mid (q_1, x, z_1, q_2, x, z_2, \ldots.q_n, x, z_n) (s_1, \epsilon, \epsilon, s_2, \epsilon, \epsilon, \ldots..s_n, \epsilon, \epsilon)$$

where, $s_i \epsilon F_i$, $1 \le i \le n$

Rcpcem (N): A retrieving parallel communicating external memory finite automation of design.

Epcemta(n): A centralized parallel communicating external finite automation system of degree (n).

Pcemfa (n): A parallel communicating external memory finite automation system of degree n.

Rcpcemfa (n): Is the class of all languages accepted by rcpcemfa (n) automation systems

Xrctc: Of x (n) is a type of automation system, them Xm (n) is the classes of all languages accepted by automation systems of type X_{em}.

Then

$$\text{Recem}_r(A) = \text{Recem}(A)$$

Theorem 1: A language is accepted by a deterministic n and finite external memory automation if it always Pcemfa (n).

A language is accepted by a deterministic n head External memory finite automation if and only if it belongs to Dpcemfa (n).

Proof: Let $A = (n, Q_i, V, f_i, q_0, F, X)$. Be a n head external memory finite automation. We define

$$\text{Pcemfa (n) } A = (V, A_1, \ldots A_n, K, X)$$

where:

$$A_i = (Q_i, V, f_i, q_0, F, X)$$

$$Q_i = K \cup Q \cup (Q \times (V \cup \{\epsilon\})^{i-1})$$

$$\cup (Q \times (V\{\epsilon\}^i) \cup R_i \times T_i$$

where:

R_i = {o, i≤2

{{P:|Pϵ Q, 1 ≤ i ≤ i-2}, i>2}

T_i = {0, i = n

{{S_i|i+1 ≤ i ≤ n}, i<n

The transition function f_i is defined as follows

$$i = 1,\ f_1(p, a, z) = \{(p, a, z), a \in V \cup \{\epsilon\}\},\ p \epsilon q,\ z \epsilon X$$

$$f_1((p, a, z) \epsilon) = \{S_2\},\ a \epsilon V \cup \{\epsilon\},\ p \in Q,\ z \in X$$

$$f_1(s_j, \epsilon, z) = \{s_{j+1}\},\ 2 \le i \le n\text{-}1$$

$$f_1(S_n, \epsilon, \epsilon) = \{k_n\}$$

for $i = n$, $f_n(p, \epsilon, z) = \{p_1\}$

$$f_n(p_j, \epsilon, z) = \{p_{j+1}\}$$

$$f_n(p_j, w, z), \epsilon) = f(p, w)$$

where:

w = Sequence of input strings

This completes the proof of the above theorem.

CONCLUSION

The result shows in this study reveal, the importance of considering the behavior of more complicated devices than parallel communicating finite automata when reading their input are determining the acceptance by the parallel communicating parallel access memory automaton with external memory. We have proved that the whole contents can be modified during a single read operation and immediately determine the state transition during a single write operation. We conclude that what is crucial is the type of external memory is used in computations and reduce the computation time.

REFERENCES

Budo, A.O., 1977. Multiprocessor automata. Inform. Process. Lett., 25: 257-261.

Hartmanis, J., 1972. On nondeterminancy in simple computing devices. Acta Inform., 1: 336-344.

Sudborough, I.H., 1977. Some remarks on Multi head automata. RAIRO Inform. Theorique, 11: 181-195.

Sakthibalan, M., K. Krithivasan and M. Madhu, 2003. Some variants in communication of Parallel communicating pushdown automata. J. Automata, Lang. Combinator., 8 (3): 401-416.

Vide, C.V., A. Mateescu and V. Mitrana, 2002. Parallel finite automata systems communicating by states. IJFCSE., 13: 733-749.