HOMOMORPHIC IMAGES OF FINITE RING AUTOMATA

Mr.Y.IMMANUEL NELSON [1] and Dr.K.MUTHUKUMARAN [2 *]

[1] *Research Scholar, P.G. and Research Department of Mathematics, Saraswathi Narayanan College, Madurai-22,*

[2] *Head and Associate Professor, P.G. and Research Department of Mathematics, Saraswathi Narayanan College,*

Abstract : *Finite Ring Automata Homomorphism has been defined. Let* $(Q_1, +_1, ._1, \Sigma_1, \delta_1', \delta_1'', p_0, F_1)$ *and* $(Q_2, +_2, ._2, \Sigma_2, \delta_2', \delta_2'', q_0, F_2)$ *be two Finite Ring Automata. Let* $f:\Sigma_1 \rightarrow \Sigma_2$ *be a bijection. Let* $\Psi : Q_1 \rightarrow Q_2$ *be a Finite Ring Automata Homomorphism. If x is a string accepted by the Finite Ring Automata* $(Q_1, +_1, ._1, \Sigma_1, \delta_1', \delta_1'', p_0, F_1)$*, then f(x) is a string accepted by Finite Ring Automata* $(Q_2, +_2, ._2, \Sigma_2, \delta_2', \delta_2'', q_0, F_2)$*. A Two Initial States Finite Ring Automaton has been defined. Any Two Initial States Finite Ring Automaton* $(Q, +, ., \Sigma, \delta_1, \delta_2, p_0, q_0, F)$ *induces two Finite Binary Automata. Any Two Initial States Finite Ring Automaton* $(Q, +, ., \Sigma, \delta_1, \delta_2, p_0, q_0, F)$ *induces two Finite Group Automata if* $(Q, +, .)$ *is a field. If* $(Q, +, ., \Sigma, \delta_1, \delta_2, p_0, F)$ *is a Finite Ring Automaton, then there exists a Two Initial States Finite Ring Automaton corresponding to the given Finite Ring Automaton* $(Q, +, ., \Sigma, \delta_1, \delta_2, p_0, F)$*. If* $L_{TR}(M)$ *is a Language accepted by a Two Initial States Finite Ring Automaton* $(Q, +, ., \Sigma, \delta_1, \delta_2, p_0, q_0, F)$*, then any language accepted by the Induced Finite Group Automata* $(Q, +, \Sigma, \delta_1, p_0, F)$ *contains* $L_{TR}(M)$*. If* $L_{TR}(M)$ *is a Language accepted by a Two Initial States Finite Ring Automaton* $(Q, +, ., \Sigma, \delta_1, \delta_2, p_0, q_0, F)$*, then any language accepted by the induced Finite Group Automata* $(Q, ., \Sigma, \delta_2, q_0, F)$ *contains* $L_{TR}(M)$*. Let* $(Q_1, +_1, ._1, \Sigma_1, \delta_1', \delta_1'', p_0', p_0'', F_1)$ *and* $(Q_2, +_2, ._2, \Sigma_2, \delta_2', \delta_2'', q_0', q_0'', F_2)$ *be two TIS Finite Ring Automata. Let* $f:\Sigma_1 \rightarrow \Sigma_2$ *be a bijection. Let* $\Psi : Q_1 \rightarrow Q_2$ *be a TIS Finite Ring Automata Homomorphism. If x is a string accepted by the TIS Finite Ring Automata* $(Q_1, +_1, ._1, \Sigma_1, \delta_1', \delta_1'', p_0', p_0'', F_1)$*, then f(x) is a string accepted by TIS Finite Ring Automata* $(Q_2, +_2, ._2, \Sigma_2, \delta_2', \delta_2'', q_0', q_0'', F_2)$

Keywords : Finite Automaton, Finite Binary Automaton, Finite Group Automaton, Finite Ring Automaton, TIS Finite Ring Automata Finite Ring Automata Homomorphism

I INTRODUCTION

We have different types of machine in the world. An Automaton is a typical machine. It is a mathematical model of a system with inputs and outputs. A machine may have an infinite variety of possible histories. But it will need an infinite capacity for storing them. We shall concentrate on those machines whose past histories can affect their future behavior in only a finite number of ways. We consider only finite automata and finite ring automata. There are some machines which works in different directions. Therefore we take Two Initial States Finite Ring Automaton. These type of machines may work in two different directions. Some times to know about a machine one can study homomorphic images of another machine. The theory of Automata plays an important role in many fields. It has become a part of computer science. It is very useful in electrical engineering. It provides useful techniques in a wide variety of applications. Therefore the theory of Finite Ring Automata, Two Initial States Finite Ring Automata and the homomorphic images of Finite Ring Automata will be play an important role in many fields.

II PRELIMINARIES

2.1 Definition : Alphabet: An alphabet is a finite set of symbols.

2.2 Example: $\sum = \{ a,b,c \}$ is an alphabet.

2.3 Example: $\sum = \{ 0,1 \}$ is an alphabet.

2.4 Definition : Strings (or Word): A string (or word) is a finite sequence of symbols juxtaposed.

2.5 Example : Letters and digits are examples of frequently used symbols.

2.6 Definition : They length of a string w, denoted $|w|$, is the number of symbols composing the string.

2.7 Definition : Concatenation : The concatenation of two strings is the string formed by writing the first followed by the second, with no intervening space.

2.8 Definition : Language : A language is a set of strings of symbols from some alphabet. The set of all strings over a fixed alphabet Σ is denoted by Σ^*.

2.9 Definition : Finite Automaton : A finite automaton is a 5–tuple $(Q, \Sigma, \delta, q_0, F)$, where Q is a finite set of states, Σ is a finite input alphabet, q_0 in Q is the initial state, $F \subseteq Q$ is the set of final states, and δ is the transition function mapping Q x Σ to Q.

2.10 Definition : A string x is said to be accepted by a finite automaton $(Q, \Sigma, \delta, q_0, F)$ if $\delta(q_0, x) = p$ for some p in F. The language accepted by the finite automaton is the set $\{x \mid \delta(q_0, x) \text{ is in } F\}$.

2.11 Definition : A language is a regular set (or just regular) if it is the set accepted by some finite automaton.

2.12 Definition : A nondeterministic finite automaton is a quintuple $(Q, \Sigma, \delta, q_0, F)$ with all components as in the deterministic finite automaton, but δ, the transition function, maps Q x Σ to 2^Q.

2.14 Theorem . Let L be a set accepted by a nondeterministic finite automaton. Then there exists a deterministic finite automaton that accepts L.

2.15 Definition : Finite Binary Automaton: A Finite Binary Automaton is a 6-tuple $(Q, *, \Sigma, \delta, q_0, F)$, where Q is a finite set of elements called states, * is a mapping from $Q \times Q$ to Q, Σ is a finite set of integers, q_0 in Q is a state called the initial state and $F \subseteq Q$ and F is the set of states called final states and δ is the transition function mapping from $Q \times \Sigma$ to Q defined by the operation in the group (Q, *), for example $\delta(q,n) = q^n$.

If Σ^* is the set of strings of inputs, then the transition function δ is extended as follows : For $m \in \Sigma^*$ and $n \in \Sigma$, $\delta': Q \times \Sigma^* \rightarrow Q$ is defined by $\delta'(q,mn) = \delta(\delta'(q,m),n)$. If no confusion arises δ' can be replaced by δ.

2.16 Definition : Finite Group Automaton : A Finite Group Automaton is a 6-tuple $(Q, *, \Sigma, \delta, q_0, F)$, where Q is a finite set of elements and the elements are called states and (Q, *) is a group, Σ is a subset of integers and the integers are called inputs, $q_0 \in Q$ and the state q_0 is called the initial state, $F \subseteq Q$ and the states (elements) of F are called final states, δ is the transition function mapping Q x Σ to Q defined by the operation in the group (Q, *).

For example $\delta(q,n) = q * q * q *.....* q$ (n times).
If Σ^* is the set of strings of inputs, then the transition function δ is extended as follows :

For $m \in \Sigma^*$ and $n \in \Sigma$, $\delta': Q \times \Sigma^* \rightarrow Q$ is defined by $\delta'(q,mn) = \delta(\delta'(q,m),n)$.

2.17 Definition : Nondeterministic Finite Group Automaton (NDFGA) : A Nondeterministic Finite Group Automaton is a 6-tuple (Q, *, Σ, δ , q_0, F), where Q,*, Σ, q_0 and F are the same as in Finite Group Automaton, but $\delta : Q \times \Sigma \rightarrow 2^Q$.

2.18 Theorem : Let L be a language accepted by a Nondeterministic Finite Group Automaton. Then there is a Deterministic Finite Group Automaton that accepts L.

2.19 Definition : Finite Ring Automaton : A Finite Ring Automaton is an 8-tuple (Q, +, . , Σ, δ_1, δ_2 ,q_0, F), where Q is a finite set of elements called states and (Q, +, .) is a ring, Σ is a finite set of integers, q_0 in Q is a state called the initial state and F⊆Q and F is the set of states called final states and δ_1, δ_2 are transition functions from Q×Σ to Q defined by $\delta_1(q,n) = nq$ and $\delta_2(q,n) = q^n$

If Σ* is the set of strings of inputs, then the transition functions δ_1, δ_2 can be extended as follows :

For m ϵ Σ* and n ϵ Σ, δ_1': Q×Σ* → Q is defined by $\delta_1'(q,mn) = \delta_1(\delta_1'(q,m),n)$. To reduce the number of notations, δ_1' can be replaced by δ_1.

For m ϵ Σ* and n ϵ Σ, δ_2': Q×Σ* → Q is defined by $\delta_2'(q,mn) = \delta_2(\delta_2'(q,m),n)$. To reduce the number of notations, δ_2' can be replaced by δ_2.

A Finite Ring Automaton can be simply expressed by **FRA**

2.20 Definition : A string x is said to be accepted by Finite Ring Automaton (Q, +, . , Σ, δ_1, δ_2 ,q_0, F) if $\delta_1(q_0,x)$ ϵ F and $\delta_2(q_0,x)$ϵ F. The language accepted by the given finite ring automaton is the set {x | $\delta_1(q_0,x)$ ϵ F and $\delta_2(q_0,x)$ϵ F }. We shall denote the language accepted by the given finite ring automaton by $L_R(M)$, where as the language accepted by the given finite group automaton can be denoted by $L_G(M)$.

2.21 Definition : Let (Q, +, . , Σ, δ_1, δ_2 ,q_0, F) be a Finite Ring Automaton. Then the transition ranges of δ_1 and δ_2 of the subset $\{a_1,a_2, a_3,\ldots\ldots\ldots,a_k\}$ of Q are defined to be

$\delta_1(\{a_1,a_2,a_3,\ldots,a_k\} , n)\} = \{ \delta_1(a_1 , n)\} \cup \{ \delta_1(a_2 , n)\} \cup \ldots\ldots \cup \{ \delta_1(a_k , n)\}$

$\delta_2(\{a_1,a_2,a_3,\ldots,a_k\} , n)\} = \{ \delta_2(a_1 , n)\} \cup \{ \delta_2(a_2 , n)\} \cup \ldots\ldots \cup \{ \delta_2(a_k , n)\}$

2.22 Definition : Nondeterministic Finite Ring Automaton (NDFRA) : A Nondeterministic Finite Ring Automaton is an 8-tuple (Q, +, . , Σ, δ_1, δ_2 ,q_0, F), where Q, +, . , Σ, q_0, F have the same meaning as in Deterministic Finite Ring Automaton but the transition functions δ_1, δ_2 are from Q×Σ to 2^Q defined by $\delta_1(q,n) = \{nq\}$ and $\delta_2(q,n) = \{q^n \}$

2.23 Theorem : Let $L_R(M)$ be a language accepted by a Nondeterministic Finite Ring Automaton. Then there is a Deterministic Finite Ring Automaton that accepts $L_R(M)$.

III FINITE RING AUTOMATA HOMOMORPHISM (FRA HOMOMORPHISM)

3.1 Definition : Let $(Q_1, +_1, ._1, \Sigma_1, \delta_1', \delta_1'', p_0, F_1)$ and $(Q_2, +_2, ._2, \Sigma_2, \delta_2', \delta_2'', q_0, F_2)$ be two Finite Ring Automata. Suppose there is a one to one correspondence between Σ_1 and Σ_2.
Let $f:\Sigma_1 \rightarrow \Sigma_2$ be a bijection. Then a mapping $\Psi : Q_1 \rightarrow Q_2$ is said to be a **Finite Ring Automata Homomorphism** or simply **FRA Homomorphism** if

1. $\Psi(a+_1b) = \Psi(a) +_2 \Psi(b)$, for all $a,b \epsilon Q_1$
2. $\Psi(a ._1 b) = \Psi(a) ._1 \Psi(b)$, for all $a,b \epsilon Q_1$
3. $\Psi(\delta_1'(a,n)) = \delta_2' (\Psi(a), f(n))$, for all $a \epsilon Q_1$ and $n \epsilon \sum_1$
4. $\Psi(\delta_1''(a,n)) = \delta_2''(\Psi(a), f(n))$, for all $a \epsilon Q_1$ and $n \epsilon \sum_1$
5. $\Psi(p_0) = q_0$
6. $a \epsilon F_1$ if and only if $\Psi(a) \epsilon F_2$.

3.2 Theorem : Let $(Q_1, +_1, ._1, \Sigma_1, \delta_1', \delta_1'', p_0, F_1)$ and $(Q_2, +_2, ._2, \Sigma_2, \delta_2', \delta_2'', q_0, F_2)$ be two Finite Ring Automata. Let $f:\Sigma_1 \rightarrow \Sigma_2$ be a bijection. Let $\Psi : Q_1 \rightarrow Q_2$ be a Finite Ring Automata Homomorphism. If x is a string accepted by the Finite Ring Automata $(Q_1, +_1, ._1, \Sigma_1, \delta_1', \delta_1'', p_0, F_1)$, then f(x) is a string accepted by Finite Ring Automata $(Q_2, +_2, ._2, \Sigma_2, \delta_2', \delta_2'', q_0, F_2)$

Proof : Let $(Q_1, +_1, ._1, \Sigma_1, \delta_1', \delta_1'', p_0, F_1)$ and $(Q_2, +_2, ._2, \Sigma_2, \delta_2', \delta_2'', q_0, F_2)$ be two Finite Ring Automata. Let $f:\Sigma_1 \rightarrow \Sigma_2$ be a bijection.
Let $\Psi : Q_1 \rightarrow Q_2$ be a Finite Ring Automata Homomorphism.
Suppose x is a string accepted by the Finite Ring Automata $(Q_1, +_1, ._1, \Sigma_1, \delta_1', \delta_1'', p_0, F_1)$.
Then $\delta_1'(p_0,x) \epsilon F_1$ and $\delta_1''(p_0,x) \epsilon F_1$.

It is enough to prove for any $n \epsilon \sum_1$

Since $\Psi : Q_1 \rightarrow Q_2$ is a Finite Ring Automata Homomorphism, $a \epsilon F_1$ if and only if $\Psi(a) \epsilon F_2$.

$\Psi (\delta_1'(p_0,x)) \epsilon F_2$ and $\Psi (\delta_1''(p_0,x)) \epsilon F_2$.

$\Psi(\delta_1'(p_0,n)) = \delta_2' (\Psi(p_0), f (n))$

$\Psi : Q_1 \rightarrow Q_2$ is a Finite Ring Automata Homomorphism => $\Psi(p_0)=q_0$

Therefore, $\delta_2' (\Psi(p_0), f (n)) = \delta_2' (q_0, f (n)) \in F_2$

$\Psi (\delta_1''(p_0,x)) = \delta_2''(\Psi(p_0), f (n)) = \delta_2''(q_0, f (n)) \in F_2$

Therefore, f (n) is accepted by the Finite Ring Automata (Q_2, $+_2$, $._2$, Σ_2 , δ_2', δ_2'', q_0, F_2)

Since x is a string formed by the states of Σ_1 , f(x) is a string formed by the states of Σ_2.

Hence f(x) is a string accepted by Finite Ring Automata (Q_2, $+_2$, $._2$, Σ_2 , δ_2', δ_2'', q_0, F_2)

IV TWO INITIAL STATES FINITE RING AUTOMATON

4.1 Definition : Two Initial States Finite Ring Automaton : A Two Initial States Finite Ring Automaton is a 9-tuple (Q, +, . , Σ, δ_1, δ_2 ,p_0, q_0, F), where Q is a finite set of elements called states and (Q, +, .) is a ring, Σ is a finite set of integers and p_0 , q_0 are two initial states in Q and F⊆Q and F is the set of states called final states and δ_1, δ_2 are transition functions from Q×Σ to Q defined by $\delta_1(q,n) = nq$ and $\delta_2(q,n) = q^n$.

Note : Finite Ring Automaton and Two Initial States Finite Ring Automaton differ not only by the number of initial states but they differ by the acceptance of strings.

If Σ* is the set of strings of inputs, then the transition functions δ_1, δ_2 can be extended as follows :

For m ϵ Σ* and n ϵ Σ, δ_1': Q×Σ* → Q is defined by $\delta_1'(q,mn) = \delta_1 (\delta_1'(q,m),n)$. To reduce the number of notations, δ_1' can be replaced by δ_1.

For m ϵ Σ* and n ϵ Σ, δ_2': Q×Σ* → Q is defined by $\delta_2'(q,mn) = \delta_2 (\delta_2'(q,m),n)$. To reduce the number of notations, δ_2' can be replaced by δ_2.

A Two Initial States Finite Ring Automaton can be simply expressed by **TIS FINITE RING AUTOMATON** or **TISFRA**

4.2 Definition : A string x is said to be accepted by a Two Initial States Finite Ring Automaton (Q, +, . , Σ, δ_1, δ_2 , p_0, q_0, F) if $\delta_1(p_0,x) \in F$ and $\delta_2(q_0,x) \in F$. The language accepted by the given Two Initial States finite ring automaton is the set {x l $\delta_1(p_0,x) \in F$ and $\delta_2(q_0,x) \in F$ }. We shall denote the language accepted by the given Two Initial States finite ring automaton by $L_{TR}(M)$.

4.3 Theorem : Any Two Initial States Finite Ring Automaton (Q, +, . , Σ, δ_1, δ_2 ,p_0, q_0, F) induces two Finite Binary Automata.

Proof : They are (Q, +, Σ, δ_1, p_0, F) and (Q, . , Σ, δ_2, q_0, F).

4.4 Theorem : Any Two Initial States Finite Ring Automaton (Q, +, . , Σ, δ_1, δ_2 ,p_0, q_0, F) induces two Finite Group Automata if (Q, +, .) is a field.

Proof : They are (Q, +, Σ, δ_1, p_0, F) and (Q\{0}, . , Σ, δ_2, q_0, F).

4.5 Theorem : If (Q, +, . , Σ, δ_1, δ_2 ,p_0, F) is a Finite Ring Automaton, then there exists a Two Initial Finite Ring Automaton corresponding to the given Finite Ring Automaton (Q, +, . , Σ, δ_1, δ_2 ,p_0, F).

Proof : It is (Q, +, . , Σ, δ_1, δ_2 ,p_0, p_0, F).

4.6 Theorem : If $L_{TR}(M)$ is a Language accepted by a Two Initial States Finite Ring Automaton (Q, +, . , Σ, δ_1, δ_2 ,p_0, q_0, F), then any language accepted by the Induced Finite Group Automata (Q, +, Σ, δ_1, p_0, F) contains $L_{TR}(M)$.

Proof : The proof comes from the definition of the language accepted by Finite Ring Automaton and Two Initial States Finite Ring Automaton

4.7 Theorem : If $L_{TR}(M)$ is a Language accepted by a Two Initial States Finite Ring Automaton (Q, +, . , Σ, δ_1, δ_2 ,p_0, q_0, F), then any language accepted by the induced Finite Group Automata (Q, . , Σ, δ_2, q_0, F) contains $L_{TR}(M)$.

Proof : The proof is similar to the proof of Theorem 4.6

V TWO INITIAL STATES FINITE RING AUTOMATON AND HOMOMORPHISM

5.1 Definition : Let (Q_1, $+_1$, $\cdot_1$, Σ_1, δ_1', δ_1", p_0', p_0",F_1) and (Q_2, $+_2$, $\cdot_2$, Σ_2 , δ_2', δ_2", q_0', q_0",F_2) be two Finite Ring Automata. Suppose there is a one to one correspondence between Σ_1 and Σ_2. Let $f:\Sigma_1 \rightarrow \Sigma_2$ be a bijection. Then a mapping $\Psi : Q_1 \rightarrow Q_2$ is said to be a **TIS Finite Ring Automata Homomorphism** or simply **TISFRA Homomorphism** if

1. $\Psi(a +_1 b) = \Psi(a) +_2 \Psi(b)$, for all $a,b \in Q_1$

2. $\Psi(a \cdot_1 b) = \Psi(a) \cdot_1 \Psi(b)$, for all $a,b \in Q_1$

3. $\Psi(\delta_1'(a,n)) = \delta_2'(\Psi(a), f(n))$, for all $a \in Q_1$ and $n \in \sum_1$

4. $\Psi(\delta_1''(a,n)) = \delta_2''(\Psi(a), f(n))$, for all $a \in Q_1$ and $n \in \sum_1$

5. $\Psi(p_0') = q_0'$ and $\Psi(p_0'') = q_0''$
6. $a \in F_1$ if and only if $\Psi(a) \in F_2$.

5.2 Theorem : Let $(Q_1, +_1, \cdot_1, \Sigma_1, \delta_1', \delta_1'', p_0', p_0'', F_1)$ and $(Q_2, +_2, \cdot_2, \Sigma_2, \delta_2', \delta_2'', q_0', q_0'', F_2)$ be two TIS Finite Ring Automata. Let $f:\Sigma_1 \to \Sigma_2$ be a bijection. Let $\Psi : Q_1 \to Q_2$ be a TIS Finite Ring Automata Homomorphism. If x is a string accepted by the TIS Finite Ring Automata $(Q_1, +_1, \cdot_1, \Sigma_1, \delta_1', \delta_1'', p_0', p_0'', F_1)$, then f(x) is a string accepted by TIS Finite Ring Automata $(Q_2, +_2, \cdot_2, \Sigma_2, \delta_2', \delta_2'', q_0', q_0'', F_2)$

Proof : Let $(Q_1, +_1, \cdot_1, \Sigma_1, \delta_1', \delta_1'', p_0', p_0'', F_1)$ and $(Q_2, +_2, \cdot_2, \Sigma_2, \delta_2', \delta_2'', q_0', q_0'', F_2)$ be two TIS Finite Ring Automata.
Let $f:\Sigma_1 \to \Sigma_2$ be a bijection. Let $\Psi : Q_1 \to Q_2$ be a TIS Finite Ring Automata Homomorphism.
Suppose x is a string accepted by the TIS Finite Ring Automata $(Q_1, +_1, \cdot_1, \Sigma_1, \delta_1', \delta_1'', p_0, F_1)$.
Then $\delta_1'(p_0', x) \in F_1$ and $\delta_1''(p_0'', x) \in F_1$.

It is enough to prove for any $n \in \sum_1$

Since $\Psi : Q_1 \to Q_2$ is a TIS Finite Ring Automata Homomorphism, $a \in F_1$ if and only if $\Psi(a) \in F_2$.

$\Psi(\delta_1'(p_0', x)) \in F_2$ and $\Psi(\delta_1''(p_0'', x)) \in F_2$.

$\Psi(\delta_1'(p_0',n)) = \delta_2'(\Psi(p_0'), f(n))$

$\Psi : Q_1 \to Q_2$ is a TIS Finite Ring Automata Homomorphism

$\Rightarrow \Psi(p_0') = q_0'$ and $\Psi(p_0'') = q_0''$

Therefore, $\delta_2'(\Psi(p_0'), f(n)) = \delta_2'(q_0', f(n)) \in F_2$

$\Psi(\delta_1''(p_0'',x)) = \delta_2''(\Psi(p_0''), f(n)) = \delta_2''(q_0'', f(n)) \in F_2$

Therefore, f (n) is accepted by the TIS Finite Ring Automata $(Q_2, +_2, \cdot_2, \Sigma_2, \delta_2', \delta_2'', q_0', q_0'', F_2)$

Since x is a string formed by the states of Σ_1 , f(x) is a string formed by the states of Σ_2.
Hence f(x) is a string accepted by the TIS Finite Ring Automata $(Q_2, +_2, \cdot_2, \Sigma_2, \delta_2', \delta_2'', q_0', q_0'', F_2)$

VI **CONCLUSION :** There are some machines which works in different directions. Therefore we take Two Initial States Finite Ring Automata also. These type of machines may work in two different directions so that the number of machines may be reduced. Some times to know about a machine one can study homomorphic images of another machine. The study of Finite Ring Automata, Two Initial States Finite Ring Automata and their homomorphic images will lay a new milestone.

REFERENCES :

1. Y.Immanuel Nelson and Dr.K.Muthukumaran, "Finite Ring Automata" ***"GIS SCIENCE JOURNAL"****, ISSN:1869-9391, Volume 9, Issue 12, Pages 857-863*
2. Dr.K.Muthukumaran and S.Shanmugavadivoo, "Finite Abelian Automata" ***"IOSR Journal Of Mathematics"****, A Journal of "International Organization of Scientific Research" e-ISBN:2278-5728, p-ISBN:2319- 765X,Volume 14, Issue 2, Ver.II (March.-April.2018),PP 01-04.*
3. Dr.K.Muthukumaran and S.Shanmugavadivoo, "Finite Subgroup Automata" ***"IOSR Journal of Mathematics"****, A Journal of "International Organization of Scientific Research", e-ISBN:2278-5728, p-ISBN:2319-765X,Volume 15, Issue I, Ver.I (Jan.-Feb.2019),PP 50-56.*
4. S.Shanmugavadivoo and Dr. M.Kamaraj, "Finite Binary Automata" **"International Journal of Mathematical Archive"**, ISSN 2229 - 5046, 7(4),2016, Pages 217-223.
5. Danish Ather, Raghuraj and Vinodani Katiyar, "An efficient algorithm to design DFA that accepts strings over input symbol a, b having at most x number of a and y number of b" Journal of Nature Inspired Computing, Vol.I, No. 2, 2013, pages 30-33.
6. Danish Ather, Raghuraj and Vinodani Katiyar, "To develop an efficient algorithm that generalize the method of design of Finite Automata that accepts "N" Base Number such that when "N" is divided by M leaves remainder "X", IJCA ISBN, December 2012.
7. S.Shanmugavadivoo and Dr. M.Kamaraj, "An Efficient Algorithm To Design DFA That Accept Strings Over The Input Symbol A,B,C Having Atmost X Number of A, Y Number of B, & Z Number of C" **"Shanlax International Journal of Arts, Science And Humanities"** Volume 3, No. 1, July 2015, Pages 13-18
8. John E. Hopcroft, Jeffery D.Ullman, Introduction to Automata Theory, Languages, And Computation, Narosa Publising House.

9. J.P.Tremblay and R.Manohar, Discrete Mathematical Structures with Applications To Computer Science, Tata Mcgraw-Hill Publishing Company Limited, New Delhi, 1997.