Applied Mathematics and Computational Intelligence
Volume 8, No.1, Dec 2019 [9-16]

Transformation Semigroups and State Machines

Nacer Ghadbane*

Laboratory of Pure and Applied Mathematics , Department of Mathematics, University of M'sila, BP 166 Ichebilia, 28000, M'sila, Algeria.

ABSTRACT

A transformation semigroup is a pair (Q; S) consisting of a finite set Q, a finite semigroup S and a semigroup action λ : Q X S ⟩ Q, (q, s ⎱ s (q) which means : i) □q ϵ Q, □s, t ϵ S : st (q) = s (t (q)) , and (ii) □s, t ϵ S,□q ϵ Q, s (q) = t (q) ⁄ s = t. A state machine or a semiautomation is an ordered triple M = (Q, ∑, F), where Q and are finite sets and F : Q X ∑ ⟩ Q is a partial function. This paper provides the construction of state machines associate a direct product, the cascade product, and wreath product of transformations semigroups.

Keywords: Semigroup, Semigroup Action, Morphism Semigroup, Transformation Semi-Group, State Machine.

1. INTRODUCTION

Actions of semigroups are important both in mathematics and computer science. The theory of machines developed so far have largely influenced the development of computer science and associated languages. The theory of machines that have been developed in the last twenty years, has had a considerable influence, not only on the computer systems, but also biology, biochemistry, and etc.

The remainder of this paper is organized as follows. Section 2 provides some elementary material concerning of transformation semigroups and state machines. Section 3 presents the construction of state machines associate a direct product of transformations semigroups. Section 4 includes the construction of state machines associated with the cascade product of transformations semigroups. In Section 5, the construction of state machines associated with the wreath product of transformations semigroups is provided. Finally, conclusions are drawn in Section 6.

2. PRELIMINARIES

A semigroup is an ordered pair (S, ·), where S is a nonempty set and the dot is an associative binary operation, i.e., a function $(s_1 , s_2) \longrightarrow s_1 \cdot s_2$ from S X S into S such that for all s_1, s_2, s_3, $(s_1 \cdot s_2) \cdot s_2 = s_1 \cdot (s_2\ s_3)$. (S, ·) will usually be abbreviated to S and $s_1 \cdot s_2$ to s_1s_2. A semigroup S is commutative or abelian if $s_1s_2 = s_2s_1$

A finite transformation semigroup is a pair X = (Q, S), where Q is a finite set, and S is a set of functions from Q into itself that forms a semigroup under composition. If S contains the identity mapping on Q, then we say that X is a transformation monoid. If X = (Q, S), then X denotes the transformation semigroup that results by adjoining to S all the constant mappings on Q.

---

*E-mail: *nacer.ghadbane@yahoo.com*

If S is a monoid, i.e., if S has an identify 1, then the semigroup action λ: Q **X** S ⎞ Q is further assumed to satisfy 1 (q) = q, for each $q \in Q$.

We formally define an alphabet as a non-empty finite set. A word over an alphabet is a finite sequence of symbols of $\sum$. Although one writes a sequence as $(\sigma_1, \sigma_2, \ldots, \sigma_n)$, in the present context, we prefer to write it as $\sigma_1 \sigma_2 \ldots \sigma_n$. The set of all words on the alphabet $\sum$ is denoted by $\sum^*$ and is equipped with the associative operation defined by the concatenation of two sequences. The concatenation of two sequences $\alpha_1 \alpha_2 \ldots \alpha_n$ and $\beta_1 \beta_2 \ldots \beta_m$ is the sequence $\alpha_1 \alpha_2 \ldots \alpha_n \beta_1 \beta_2 \ldots \beta_m$.

The concatenation is an associative operation. The string consisting of zero letters is called the empty word, written ϵ. Thus, ϵ, α, β, ααβα, αααβα are words over the alphabet {α, β}. Thus the set $\sum^*$ of words is equipped with the structure of a monoid. The monoid $\sum^*$ is called the free monoid on $\sum$. The length of a word w, denoted $|w|$, is the number of letters in w when each letter is counted as many times as it occurs. Again by definition, $|\epsilon|= 0$. For example $|\alpha\alpha\beta\alpha|= 4$ and $|\alpha\alpha\alpha\beta\alpha| = 5$. Let w be a word over an alphabet $\sum$. For $\sigma \in \sum$, the number of occurrences of in w shall be denoted by $|w|_\sigma$. For example $|\alpha\alpha\beta\alpha|_\beta = 1$ and $|\alpha\alpha\alpha\beta\alpha|_\alpha = 4$.

A state machine or a semi-automation is an ordered triple M = (Q, $\sum$, F), where Q and $\sum$ are finite sets and $F: Q \times \Sigma \longrightarrow Q$ is a partial function. A state machine M = (Q, $\sum$, F) is said to complete if $F: Q \times \Sigma \longrightarrow Q$ is a function.

Corresponding to a transformation semigroup X = (Q, S), there is a state machine M = (Q, S, F), where $F: Q \times S \longrightarrow Q$ is defined by F(q, s) = s(q), for each $q \in Q$ and for each $s \in S$. M is called the state machine of X, denoted by M(X).

For every transformation semigroup X = (Q, S), there is a morphism $\upsilon|: S \longrightarrow E(Q)$, the semigroup of all mappings f : Q ⟶ Q, given by υ|(s) = f, where f (q) = s (q). E(Q) is usually called the full transformation semigroup on Q

Let M = (Q, $\sum$, F) and M′ = (Q', $\sum$', F') be state machines. A pair of mappings (δ, θ) : M ⟶ M' is said to be a state machine homomorphism if δ : Q ⎞ Q' and θ : $\sum$' is a pair of mappings such that $\delta o F_\sigma = F'_{\theta(\sigma)} o\delta$, $\forall \sigma \in \sum$, where F_σ : Q ⟶ Q is defined by F_σ (q) = F (q, σ), $\forall q \in Q$. A state machine homomorphism (δ, θ) : M⟶M' is said to be:

i. a monomorphism if δ and θ are both injective;
ii. an epimorphism if δ and θ are both surjective;
iii. an isomorphism if δ and θ is both a monomorphism and an epimorphism (written M ≅ M')

Let $M_1 = (Q_1, \sum_1, F_1)$ and $M_2 = (Q_2, \sum_2, F_2)$ be state machines. Suppose that M_1 and M_2 are state machines with the same input $\sum$. Connecting them up in this way, will produce a new state machine $M_1 \mathbf{X} M_2 = (Q_1 \mathbf{X} Q_2, \sum, F_1 \mathbf{X} F_2)$ where $(F_1 \mathbf{X} F_2) ((q_1, q_2), \sigma) = (F_1 (q_1, \sigma), F_2 (q_2, \sigma))$ for $\sigma \in \sum$ $(q_1, q_2) \in Q_1 \mathbf{X} Q_2$. We call this state machine the restricted direct product of M_1 and M_2.

Let $M_1 = (Q_1, \sum_1, F_1)$ and $M_2 = (Q_2, \sum_2, F_2)$ be state machines. We define $M_1 \mathbf{X} M_2 = (Q_1 \mathbf{X} Q_2, \sum_1 \mathbf{X} \sum_2, F_1 \mathbf{X} F_2)$ where $(F_1 \mathbf{X} F_2) ((q_1, q_2), (\sigma_1, \sigma_2)) = (F_1 (q_1, \sigma_1), F_2 (q_2, \sigma_2))$ for $(\sigma_1, \sigma_2) \in \sum_1 \mathbf{X} \sum_2$, $(q_1, q_2) \in Q_1 \mathbf{X} Q_2$.

We call this state machine the full direct product of M_1 and M_2.

Let $M_1 = (Q_1, \sum_1, F_1)$ and $M_2 = (Q_2, \sum_2, F_2)$ be state machines. We define the cascade product of M1 and M2 with respect to $w : Q_2 \mathbf{X} \sum_2 \longrightarrow \sum_1$ by $M_1 w M_2 = (Q_1 \mathbf{X} Q_2, \sum_2, F^w)$ where $F^w ((q_1, q_2), \sigma_2) = (F_1(q_1, w (q_2, \sigma_2)), F_2(q_2, \sigma_2))$ for $\sigma_2 \in \sum_2$, $(q_1, q_2) \in Q_1 \mathbf{X} Q_2$.

Let $M_1 = (Q_1, \sum_1, F_1)$ and $M_2 = (Q_2, \sum_2, F_2)$ be state machines. We define the wreath product of M_1 and M_2 by $M_1 w\, M_2 = (Q_1 \text{ X } Q_2,\ \text{X} \sum_2, F^w)$ where $F^w((q_1, q_2), (f, \sigma_2)) = (F_1(q_1, f(q_2)), F_2(q_2, \sigma_2))$ for $\sigma_2 \epsilon \sum_2$, $f \epsilon$, $(q_1, q_2) \epsilon Q_1 \mathbf{X} Q_2$.

3. DIRECT PRODUCT OF TRANSFORMATION SEMIGROUP AND STATE

In this section, we give the construction of state machines associate a direct product of transformations semigroups.

Preposition 1

Let X = (Q, S) and Y = (P, T) be transformation semigroups.

1. The direct product of X and Y, written X **X** Y, is defined as a transformation semigroup X **X** Y = (Q **X** P , S **X** T).
2. If X = (Q, S), Y = (P, T) and Z = (R, V) are transformation semigroup then (S **X** T) **X** V ≅ S **X** (T **X** V).
3. M (X **X** Y) = M (X) **X** M (Y).

Proof

1. The element of S **X** T being the ordered pairs (s, t), $s \epsilon S$, $t \epsilon T$ with (s, t) (q, p) = (s (q), t (p)), for each $q \epsilon Q$, $p \epsilon P$.
2. The mapping ((s, t), v) ⟼(s, (t, v)) is an isomorphism of (S **X** T) **X** V and S **X** (T **X** V).
3. We have M (X) = (Q, S, F_X), where F_X : Q **X** S⟶ Q is defined by F_X (q, s) = s (q), for all $s \epsilon S$, $q \epsilon Q$. Also we have M (Y) = (P, T, F_Y), where F_Y: P **X** T ⟶ P is defined by F_Y (p, t) = t (p), for all $t \epsilon T$, $p \epsilon P$. A similar argument, we have M (X **X** Y) = (Q **X** P, S **X** T, $F_{X \mathbf{X} Y}$), where $F_{X \mathbf{X} Y}$: (Q **X** P) (S **X** T)⟶ Q **X** P is defined by $F_{X \mathbf{X} Y}$((q, p) , (s, t)) = (s (q) , t (p)), for all (s, t) ϵ S **X** T, (q, p) ϵ Q **X** P. Consequently M (X **X** Y) = M (X) **X** M (Y).

Example 1

Consider the transformation semigroup X = (Q, S), where Q = {0, 1}, S = <s>, s defined by s (0) = 1, s (1) = 0, we have S = {s, s^2} with s^2 (0) = 0, s^2 (1) = 1 and the identity s^3 = s. M (X) = (Q, S, F_X) the state machine of X is given by the following table:

.

F_X	s	s^2
0	1	0
1	0	1

Let Y = (P, T) be transformation semigroup, where P = {0, 1}, T = <t>, t defined by t (0) = 1, t (1) = 1. We have T = {t}, and M(Y) = (P, T, F_Y) the state machine of Y is given by the following table:

F_Y	T
0	1
1	1

Consequently M (X **X** Y) = (Q **X** P, S **X** T, $F_{X \mathbf{X} Y}$) the state machine of X **X** Y is given by the following table:

$F_{X \times Y}$	**(s, t)**	**(s^2, t)**
(0,0)	(1,1)	(0,1)
(0,1)	(1,1)	(0,1)
(1,0)	(0,1)	(1,1)
(1,1)	(0,1)	(1,1)

4. CASCADE PRODUCT OF TRANSFORMATION SEMIGROUP AND STATE MACHINE

In this section, we give the construction of state machines associate a cascade product of transformations semigroups.

Proposition 2

Given any transformation semigroup X = (Q, S) and Y = (P, T), suppose that w: P **X** T⟶ S is a mapping. Then

1. The transformation semigroup X $\mathbf{X}_w$ Y = (Q **X** P, Tw) is called the cascade product of X and Y with respect to w, on Q **X** P with Tw = {t_w: Q **X** P ⟶ Q **X** P,}, t_w (q, p) = (w (p, t) (q), t (p)) for all t ϵ T, (q, p) ϵ Q **X** P .
2. The state machine M (X X_w Y) is the cascade product of M (X) and M (Y), i.e., M (X $\mathbf{X}_w$ Y) = M (X) $\mathbf{X}_w$ M (Y).

Proof

1. Since T_w is a set of functions from Q **X** P into itself, then X $\mathbf{X}_w$ Y = (Q **X** P, T_w) is transformation semigroup.
2. 2. We have M (X) = (Q, S, F_X), where F_X: Q **X** S ⟶ Q is defined by F_X (q, s) = s (q), for all s ϵ S, q ϵ Q. Also we have M (Y) = (P, T, F_Y), where F_Y: P **X** T ⟶ P is defined by F_Y (p, t) = t (p), for all t ϵ T, p ϵ P. A similar argument, we have M (X X_w Y) = (Q P; Tw; F_{X} $X_{w\,Y}$), where F_X $X_{w\,Y}$: (Q X P) Tw ⟶ Q X P is defined by F_X $X_{w\,Y}$ ((q, p) , t_w) = (w (p, t) (q) , t (p)), for all t_w ϵ T_w , (q, p) ϵ Q X P . Consequently M (X $\mathbf{X}_w$ Y) = M (X) $\mathbf{X}_w$ M (Y).

Example 2

Consider the transformation semigroup X = (Q, S), where Q = {0, 1}, S = **<s>**, s defined by s (0) = 1, s (1) = 0, we have S = {s, s^2 } with s^2 (0) = 0, s^2 (1) = 1 and the identity s^3 = s. M (X) = (Q, S, F_X) the state machine of X is given by the following table:

F_X	**s**	s^2
0	1	0
1	0	1

Let Y = (P, T) be transformation semigroup, where P = {0, 1}, T = <t> defined by t (0) = 1, t(1) =1. We have T = {t}, and M (Y) = (P, T, F_Y) the state machine of Y is given by the following table:

F_Y	**t**
0	1
1	1

Define the mapping w : P **X** T ⟶ S by w (o,t) = s, w (1, t) = s^2. We have X X_w Y = (Q **X** P, Tw), where t_w : Q **X** P ⟶ Q **X** P, t_w (q, p) = (w (p, t)(q), t(p)) for all t ϵ T and (q, p) ϵ Q **X** P. Since T = {t}, then Tw = {t_w} where,

t_w (0, 0) = (w (0, t) (0), t(0)) = (s(0), 1) = (1,1);
t_w (0, 1) = (w (1, t) (0), t(1)) = (s^2(0), 1) = (0,1);
t_w (1, 0) = (w (0, t) (0), t(0)) = (s(1), 1) = (0,1);
t_w (1, 1) = (w (1, t) (1), t(1)) = (s^2(1), 1) = (1,1).

M(X $\mathbf{X}_w$ Y) = (Q **X** P, Tw, F_X $\mathbf{X}_{w\ Y}$) the state machine of X $\mathbf{X}_w$ Y is given by the following table:

$\mathbf{F_X\ X_{w\ Y}}$	$\mathbf{t_w}$
(0, 0)	(1, 1)
(0, 1)	(0, 1)
(1, 0)	(0, 1)
(1, 1)	(1, 1)

5. WREATH PRODUCT OF TRANSFORMATION SEMIGROUP AND STATE MACHINE

In this section, we give the construction of state machines associated with the cascade product of transformations semigroups.

Proposition 3

Let X = (Q, S) and Y = (P, T) be transformation semigroups. Let S^P the set of all mappings f : P ⟶ S. Then

1. The transformation semigroup XwY = (Q **X** P, S^P **X** T) is called the wreath product of X and Y, on Q **X** P with S^P **X** T = {(f, t} : Q **X** P ⟶ Q **X** P, f ϵ S^P, t ϵ T}, (f, t) (q, p) = (f (p) (q), t (p)) for all (q, p) ϵ Q **X** P.
2. The state machine M (Xw Y) is the wreath product of M (X), and M (Y), i.e., M (XwY) M = M (X) wM(Y).

Proof

1. We show that S^P **X** T is closed under composition. Let (f_1, t_1), (f_2, t_2) ϵ S^P **X** T, i.e., for all (q, p) ϵ Q **X** P, (f_1, t_1) (q, p) = (f_1 (p) (q), t_1 (p)) and (f_2, t_2) (q, p) = (f_2 (p) (q), t_2 (p)). We have (f_1, t_1) o (f_2, t_2) (q, p) = (f_1, t_1) ((f_2 (p) (q), t_2 (p))) = (f_1 (t_2 (p)(f_2 (p) (q))), t_1 (t_2 (p))).
2. We have M (X) = (Q, S, F_X), where F_X : Q **X** S ⟶ Q is defined by F_X (q,s) = s (q), for all s ϵ S, q ϵ Q. Also we have M (Y) = (P, T, F_Y), where F_Y : P **X** T ⟶ P is defined by F_Y (p, t) = t (p), for all t ϵ T, p ϵ P. A similar argument, we have M (XwY) = (Q **X** P, S^P **X** T, F_{XwY}), where F_{XwY} : (Q **X** P) **X** S^P **X** T ⟶ Q X P is defined by F_{XwY} ((q, p), (f, t)) = (f (p) (q), t (p)), for all (f, t) ϵ S^P **X** T, (q, p) ϵ Q **X** P. Consequently M (XwY) = M (X) wM (Y).

Example 3

Consider the transformation semigroup X = (Q, S), where Q = {0, 1}, S = <s>, s defined by s (0) = 1, s (1) =0, we have S = {s, s^2} with s^2 (0) = 0, s^2 (1) = 1 and the identity s^3 = s. M (X) = (Q, S, F_X) the state machine of X is given by the following table:

F_X	s	s^2
0	1	0
1	0	1

Let Y = (P, T) be transformation semigroup, where P = {0, 1}, T = <t>, t defined by t (0) = 1, t (1).
We have T = {t}, and M (Y) = (P, T, F_Y), the state machine of Y is given by the following table:

F_Y	T
0	1
1	1

We have P = {0, 1} and S = {s, s^2}, then $S^P = \{f_i : \{0, 1\} \longrightarrow \{s, s^2\}, 1 \leq i \leq 4\}$, where f_1 (0) = s, f_1 (1) = s; f_2 (0) = s, f_2 (1) = s^2; f_3 (0) =s^2, f_3 (1) = s; f_4 (0) = s^2, f_4 (1) = s^2. We have T = {t}, then S^P **X** T = {(f_1, t), (f_2, t), (f_3, t), (f_4, t)}.

X*w*Y = (Q **X** P, S^P **X** T), where

(f_1, t) (0, 0) = (f_1 (0) (0), t (0)) = (s (0), t (0)) = (1, 1)
(f_1, t) (0, 1) = (f_1 (1) (0), t (1)) = (s (0), t (1)) = (1, 1)
(f_1, t) (1, 0) = (f_1 (0) (1), t (0)) = (s (1), t (0)) = (0, 1)
(f_1, t) (1, 1) = (f_1 (1) (1), t (1)) = (s (1), t (1)) = (0, 1)
(f_2, t) (0, 0) = (f_2 (0) (0), t (0)) = (s (0), t (0)) = (1, 1)
(f_2, t) (0, 1) = (f_2 (1) (0), t (1)) = (s^2 (0), t (0)) = (0, 1)
(f_2, t) (1, 0) = (f_2 (0) (1), t (0)) = (s (1), t (0)) = (0, 1)
(f_2, t) (1, 1) = (f_2 (1) (1), t (1)) = (s^2 (1), t (1)) = (1, 1)
(f_3, t) (0, 0) = (f_3 (0) (0), t (0)) = (s^2 (0), t (0)) = (0, 1)
(f_3, t) (0, 1) = (f_3 (1) (0), t (1)) = (s (0), t (1)) = (1, 1)
(f_3, t) (1, 0) = (f_3 (0) (1), t (0)) = (s^2 (1), t (0)) = (1, 1)
(f_3, t) (1, 1) = (f_3 (1) (1), t (1)) = (s (1), t (1)) = (0, 1)
(f_4, t) (0, 0) = (f_4 (0) (0), t (0)) = (s^2 (0), t (0)) = (0, 1)
(f_4, t) (0, 1) = (f_4 (1) (0), t (1)) = (s^2 (0), t (1)) = (0, 1)
(f_4, t) (1, 0) = (f_4 (0) (1), t (0)) = (s^2 (1), t (0)) = (1, 1)
(f_4, t) (1, 1) = (f_4 (1) (1), t (1)) = (s^2 (1), t (1)) = (1, 1)

M (XwY) = (Q **X** P, S^P **X** T, F_{XwY}) the state machine of X*w*Y is given by the following table:

F_{XwY}	(f_1, t)	(f_2, t)	(f_3, t)	(f_4, t)
(0, 0)	(1, 1)	(1, 1)	(0, 1)	(0, 1)
(0, 1)	(1, 1)	(0, 1)	(1, 1)	(0, 1)
(1, 0)	(0, 1)	(0, 1)	(1, 1)	(1, 1)
(1, 1)	(0, 1)	(1, 1)	(0, 1)	(1, 1)

6. CONCLUSION

In this paper, we give the construction of state machines associated with the direct product of transformation semigroups, the cascade product of transformation semigroups and the wreath product of transformation semigroups, illustrated by some examples.

REFERENCES

[1] A. E. Nagy, C. L. Nehaniv, Cascade Product of Permutation Groups. Centre for computer science and informatics, U. K and Centre for Research in Mathematics, Australia, (2013).
[2] A. Ginzburg, Algebraic Theory of Automata, Academic Press, New York, (1968).
[3] H. J. Shyr. Free Monoids and Languages, Soochow University Taipei, Taiwan, (1979).
[4] H. Straubing, "Finite Automata, Formal Logic and Circuit Complexity", Springer Science+ Business Media, LLC, (1994).
[5] J. Berstel, D. Perrin, Theory of Codes. Academic Press, (1984).
[6] J. D. P. Meldrum, Wreath Products of Groups and semigroups, University of Edinburgh, Longman, (1995).
[7] J. M. Howie, Fundamentals of Semigroup Theory, Oxford Science Publications, (1995).
[8] M. A. Arbib, Algebraic Theory of Machines, Languages and Semigroups, University of California, Academic Press, (1968).
[9] M. R. Adhikari, A. Adhikari, Basic Modern Algebra with Applications, University of Calcutta, Springer, (2014).
[10] S. Majumdar, K. Kumar Dey, M. A. Hossain, Direct Product and Wreath Product of Transformation Semigroups, Ganit J., Bangladesh Math. Soc **31** (2011) 1-7.
[11] W. M. L. Holcombe, Algebraic Automata Theory, Cambridge University Press, (1982).