Electronic Journal of Mathematical Analysis and Applications
Vol. 10(2) Jul. 2022, pp. 105-114
ISSN: 2090-729X(online)
http://math-frac.org/Journals/EJMAA/

GRAPHS IN AUTOMATA

M. DUTTA, S. KALITA, H. K. SAIKIA

Abstract. A graph consists of some points and lines called vertices and edges. It is a mathematical representation of a network in which edges represent the existence of a particular relation among the vertices. Automata is a five tuple consisting of a set of states, inputs, outputs, one transition function, and one output function between these three sets. In this paper, an attempt is made to study automata with the help of the properties of graphs. For two given automation, with the help of the operations, viz. addition, union, intersection, complement, ring sum, product, and composition of graphs, another automaton is derived. The objective is to identify and analyze the automaton obtained. In the study, some connections are observed between various automata produced as well as between the adjacency matrix of the graph, and outputs of the automaton.

1. Introduction

An automaton is a system that spontaneously gives an output from an input. The input may be energy, information, materials, etc. The system works without the intervention of man. Simply, an automaton (plural: automata or automatons) is a self-operating machine. Its synonym is ROBOT. The term "automation" was invented by an engineer named D.S. Harder, in the automobile industry, in about 1946 to describe the increased use of automatic devices and controls in mechanized production lines. The term is used widely in the context of manufacturing. It is also used in which there is a significant substitution of mechanical, electrical, or computerized action for human effort and intelligence. Finite state automata are significant in many different areas, including Electrical Engineering, Linguistics, Computer Science, Philosophy, Biology, Mathematics, and Logic.[6,7] In computer science, finite state machines are widely used in modelling application behavior, designing hardware digital systems, software engineering, compilers, network protocols, and the study of computation and language. The number of possible states of the automaton, and hence the amount of information it implicitly stores, is finite. Therefore, the automaton is a finite-state automaton. In this paper, the objective is to relate graphs and automata and to study some uses of graphical properties on automata.

2010 *Mathematics Subject Classification.* 05C30, 05C76, 03D05.
Key words and phrases. Automata; Graph; Input; Output; Graph Operations.
Submitted Apr. 26, 2021.

2. Preliminaries

A *finite state automaton* consists of a finite set of states and a set of transitions from state to state that occur on input symbols from a set of alphabets. An *alphabet* is a finite, non-empty set of symbols denoted by A, e.g. $A = \{0, 1\}$, the set of binary alphabets. A *string* (or word) is a finite sequence of symbols chosen from the set A, e.g. $01101,\ 01,\ 1,\ 0$ are some strings over an alphabet $A = \{0, 1\}$. An *automata* is a quintuples of the type $\Sigma = (Q, A, B, F, G)$, where Q is a finite set of states, A is a finite set of inputs, B is a finite set of outputs, $F : Q \times A \to Q$ and $G : Q \times Q \to B$ are functions usually known as state *transition function* and *output function* respectively.[2,5]

A *graph* $G(V, E)$ consists of a non-empty set of objects V called set of *vertices/nodes* and a set E called set of —it edges/arcs whose elements belong to the set $V \times V$. If $(u, v) \in E$, then we say that u and v are adjacent in G. If each edge of a graph G has a direction then the graph is called *directed* graph. If each edge of the graph G has no direction, the graph is said to be —it undirected graph. A *loop* is an edge (v_i, v_i). A graph without loops and multiple edges is called a *simple* graph. A graph with a finite number of vertices as well as a finite number of edges is called a *finite* graph; otherwise, it is called an *infinite* graph.[1]

A graph that contains only an isolated node is called a *null* graph. The number of edges that are incident on the vertex is called the *degree* of the vertex. In a graph, if all vertices have the same degree, then it is called a *regular* graph. A simple graph, G is said to be *complete* if every vertex in G is connected with every other vertex. A *walk* is defined as a finite alternative sequence of vertices and edges. An *open walk* in which no vertex appears more than once is called a simple *path*. The cycle C_n, $n \leq 3$, is a closed path of n vertices and n edges. A matrix $A = [a_{ij}]$ of a labeled graph G with p points is the $n \times n$ matrix is called *adjacency matrix* in which $a_{ij} = 1$ if v_i is adjacent with v_j and $a_{ij} = 0$ otherwise.[1]

The *complement* G of $\overline{G}$ is defined as a simple graph with the same vertex set as G and where two vertices u and v adjacent only when they are not adjacent in G. Let $G_1 = G(V_1, E_1)$ and $G_2 = G(V_2, E_2)$ be two graphs. Union of graphs G_1 and G_2 denoted by $G_1 \cup G_2$ is a graph $G = G(V, E)$ such that $V = V_1 \cup V_2$ and $E = E_1 \cup E_2$. Intersection of graphs G_1 and G_2 with at least one vertex in common, denoted by $G_1 \cap G_2$ is a graph $G = G(V, E)$ such that $V = V_1 \cap V_2$ and $E = E_1 \cap E_2$. Sum of graphs G_1 and G_2, denoted by $G_1 + G_2$ is a graph $G = G(V, E)$ such that $V = V_1 \cup V_2$ and $E = E_1 \cup E_2 \cup \{(u, v) : u \in V_1,\ v \in V_2\}$. The ring sum of G_1 and G_2, denoted by $G_1 \oplus G_2$ is the graph $G = G(V, E)$ such that $V = V_1 \cup V_2$ and $E = E_1 \cup E_2 - E_1 \cap E_2$. The product $G_1 \times G_2$ of graphs G_1 and G_2 is a graph $G = G(V, E)$, where any two points $u = (u_1, u_2)$ and $v = (v_1, v_2)$ in $V = V_1 \times V_2$ are adjacent in $G_1 \times G_2$ whenever $[u_1 = v_1 \ and \ u_2 adj.v_2]$ or $[u_2 = v_2 \ and \ u_1 adj.v_1]$. The composition $G_1 [G_2]$ also has $V = V_1 \times V_2$ and $u = (u_1, u_2)$ and $v = (v_1, v_2)$ in $V = V_1 \times V_2$ are adjacent whenever $u_1 adj.v_1$ or $(u_1 = v_1 \ and \ u_2 adj.v_2)$.[1]

3. Main Work (Graph operations on Automaton)

Let $G_1 = G(V_1, E_1)$ and $G_2 = G(V_2, E_2)$ be two graphs. Applying automata theory to these two graphs, G_1 and G_2 and using graph operations viz. addition, union, intersection, complement, ring sum, product and composition of graphs on two automata, another automaton is derived.

Illustration: In this section, the impact of some graph operations on some automatons are studied. Two automata are considered for two cycles C_3 and C_4 and an effort is made to define a new automata using some graphical operations and the effect of these operations are studied on the derivation of the new automata.[3,8]

Automaton on C_4:

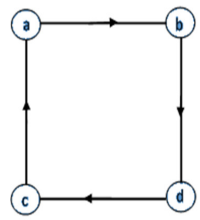

Fig. 1: Cycle C_4

Let us consider the finite state machine [2,5] for the above cycle C_4 is given by $\Sigma_1 = (Q_1, A_1, B_1, F_1, G_1, a)$ where $Q_1 = \{a, b, c, d\}$, $A_1 = \{1, 2, 3, 4\}$, $B_1 = \{0, 1\}$, a is the initial state,
the transition map $F_1 : Q_1 \times A_1 \to Q_1$ is defined by

$F_1(a, 1) = b,\ F_1(a, 2) = c,\ F_1(a, 3) = -,\ F_1(a, 4) = -,\ F_1(b, 1) = a,\ F_1(b, 2) = -,$
$F_1(b, 3) = -,\ F_1(b, 4) = d,\ F_1(c, 1) = -,\ F_1(c, 2) = a,\ F_1(c, 3) = d,\ F_1(c, 4) = -,$
$F_1(d, 1) = -,\ F_1(d, 2) = -,\ F_1(d, 3) = c\ F_1(d, 4) = b$
and the output function $G_1 : Q_1 \times Q_1 \to B_1$ defined by
$G_1(a, a) = 0,\ G_1(a, b) = 1,\ G_1(a, c) = 1,\ G_1(a, d) = 0,\ G_1(b, a) = 1, G_1(b, b) = 0,$
$G_1(b, c) = 0,\ G_1(b, d) = 1\ G_1(c, a) = 1,\ G_1(c, b) = 0,\ G_1(c, c) = 0,\ G_1(c, d) = 1,$
$G_1(d, a) = 0,\ G_1(d, b) = 1,\ G_1(d, c) = 1,\ G_1(d, d) = 0$
where 1 indicates that there is a path between the states and 0 indicates that there is no path between the states.

Table 1 : Transition table and Output table of C_4

F_1	1	2	3	4
a	b	c	-	-
b	a	-	-	d
c	-	a	d	-
d	-	-	c	b

G_1	a	b	c	d
a	0	1	1	0
b	1	0	0	1
c	1	0	0	1
d	0	1	1	0

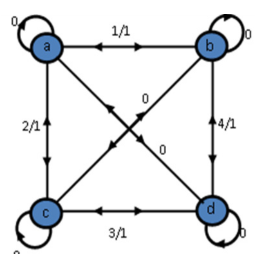

Fig. 2: State diagram of C_4

Automaton on C_3:

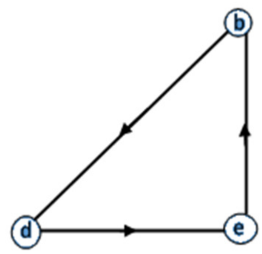

Fig. 3: State diagram of C_3

Let us consider the finite state machine for the above cycle C_3 is given by $\Sigma_2 = (Q_2, A_2, B_2, F_2, G_2, e)$ where $Q_2 = \{b, d, e\}$, $A_2 = \{4, 5, 6\}$, $B_2 = \{0, 1\}$, e is the initial state,
the transition map $F_2 : Q_2 \times A_2 \to Q_2$ is defined by
$F_2(b, 4) = d,\ F_2(d, 4) = b,\ F_2(e, 4) = -,\ F_2(b, 5) = -,\ F_2(d, 5) = e,\ F_2(e, 5) = d,\ F_2(b, 6) = e,\ F_2(d, 6) = -,\ F_2(e, 6) = b$
the output function $G_2 : Q_2 \times Q_2 \to B_2$ defined by
$G_2(b, b) = 0,\ G_2(d, b) = 1,\ G_2(e, b) = 1,\ G_2(b, d) = 1,\ G_2(d, d) = 0,\ G_2(e, d) = 1,\ G_2(b, e) = 1,\ G_2(d, e) = 1,\ G_2(e, e) = 0$
where 1 indicates that there is a path between the states and 0 indicates that there is no path.

Table 2 : Transition table and Output table of C_3

F_2	4	5	6
b	d	-	e
d	b	e	-
e	-	d	b

G_2	b	d	e
b	0	1	1
d	1	0	1
e	1	1	0

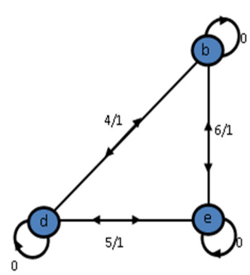

Fig. 4: State diagram of C_3

Automaton on $C_4 \cup C_3$:
The finite automata $C_4 \cup C_3$ can be obtained by using the union operation of graphs.

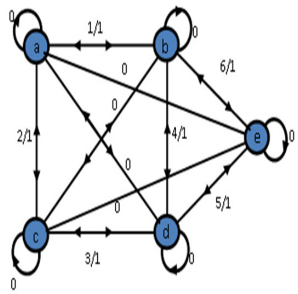

Fig. 5: State diagram of $C_4 \cup C_3$

Table 3 : Transition table and Output table of $C_4 \cup C_3$

F	1	2	3	4	5	6
a	b	c	-	-	-	-
b	a	-	-	d	-	e
c	-	a	d	-	-	-
d	-	-	c	b	e	
e	-	-	-	-	d	b

G	a	b	c	d	e
a	0	1	1	0	0
b	1	0	0	1	1
c	1	0	0	1	0
d	0	1	1	0	1
e	0	1	0	1	0

Automaton on $C_4 \cap C_3$:

The finite automata $C_4 \cap C_3$ can be obtained by using the intersection operation of graphs.

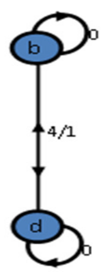

Fig. 6: State diagram of $C_4 \cap C_3$

Table 4 : Transition table and Output table of $C_4 \cap C_3$

F	1	2	3	4	5	6
a	-	-	-	-	-	-
b	-	-	-	d	-	-
c	-	-	-	-	-	-
d	-	-	-	b	-	-
e	-	-	-	-	-	-

G	a	b	c	d	e
a	0	0	0	0	0
b	0	0	0	1	0
c	0	0	0	0	0
d	0	1	0	0	0
e	0	0	0	0	0

Automaton on $\overline{C_3}$:

The finite automata $\overline{C_3}$ can be obtained by using the complement operation of graphs as follows

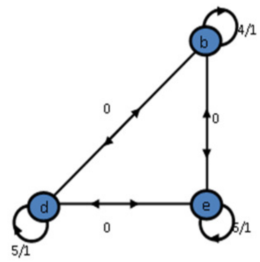

Fig. 7: State diagram of $\overline{C_3}$

Table 5 : Transition table and Output table of $\overline{C_3}$

F	4	5	6
b	b	-	-
d	-	d	-
e	-	-	e

G	b	d	e
b	1	0	0
d	0	1	0
e	0	0	1

Automaton on $C_3 \cup \overline{C_3}$:

The finite automata of $C_3 \cup \overline{C_3}$ is given by

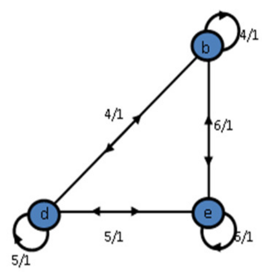

Fig. 8: State diagram of $C_3 \cup \overline{C_3}$

Table 6 : Transition table and Output table of $C_3 \cup \overline{C_3}$

F	4	5	6
b	d, b	-	e
d	b	e, d	-
e	-	d	b, e

G	b	d	e
b	1	1	1
d	1	1	1
e	1	1	1

Automaton on $C_4 \cup \overline{C_4}$:

The finite automata of $C_4 \cup \overline{C_4}$ can be obtained by using the union operation of graphs.

Automaton on $C_4 + C_3$:

The finite automata $C_4 + C_3$ can be obtained by using the addition operation of graphs.

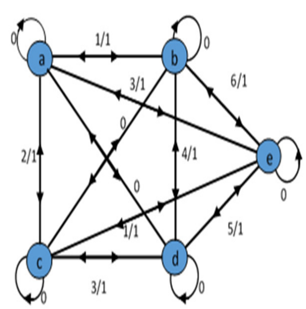

Fig. 9 : State diagram of $C_4 + C_3$

Table 7 : Transition table and Output table of $C_4 + C_3$

F	1	2	3	4	5	6
a	b	c	e	-	-	-
b	a	-	-	d	-	e
c	e	a	d	-	-	-
d	-	-	c	b	e	-
e	c	-	a	-	d	b

G	a	b	c	d	e
a	0	1	1	0	1
b	1	0	0	1	1
c	1	0	0	1	1
d	0	1	1	0	1
e	1	1	1	1	0

Automaton on $C_4 \oplus C_3$:

The finite automata $C_4 \oplus C_3$ can be obtained by using the ring sum operation of graphs.

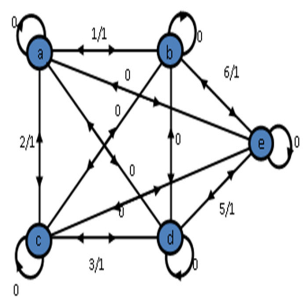

Fig. 10: State diagram of $C_4 \oplus C_3$

Table 8 : Transition table and Output table of $C_4 \oplus C_3$

F	1	2	3	4	5	6
a	b	c	-	-	-	-
b	a	-	-	-	-	e
c	-	a	d	-	-	-
d	-	-	c	-	e	-
e	-	-	-	-	d	b

G	a	b	c	d	e
a	0	1	1	0	0
b	1	0	0	0	1
c	1	0	0	1	0
d	0	0	1	0	1
e	0	1	0	1	0

Automaton on $C_4 \times C_3$:

The automata $C_4 \times C_3$ can be obtained by using the product operation of graphs as follows

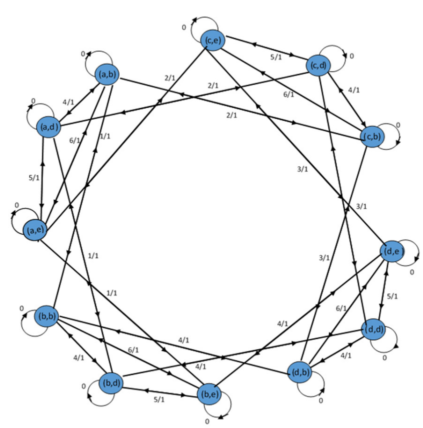

Fig. 11: State diagram of $C_4 \times C_3$

Table 9 : Transition table of $C_4 \times C_3$

F	1	2	3	4	5	6
(a,b)	(b,b)	(c,b)	-	(a,d)	-	(a,e)
(a,d)	(b,d)	(c,d)	-	(a,b)	(a,e)	-
(a,e)	(b,e)	(c,e)	-	-	(a,d)	(a,b)
(b,b)	(a,b)	-	-	$(b,d),(d,b)$	-	(b,e)
(b,d)	(a,d)	-	-	$(b,b),(d,d)$	(b,e)	-
(b,e)	(a,e)	-	-	(d,e)	(b,d)	(b,b)
(c,b)	-	(a,b)	(d,b)	(c,d)	-	(c,e)
(c,d)	-	(a,d)	(d,d)	(c,b)	(c,e)	-
(c,e)	-	(a,e)	(d,e)	-	(c,d)	(c,b)
(d,b)	-	-	(c,b)	$(b,b),(d,d)$	-	(d,e)
(d,d)	-	-	(c,d)	$(d,b),(b,d)$	(d,e)	-
(d,e)	-	-	(c,e)	(b,e)	(d,d)	(d,b)

Table 10 : Output table of $C_4 \times C_3$

G	(a,b)	(a,d)	(a,e)	(b,b)	(b,d)	(b,e)	(c,b)	(c,d)	(c,e)	(d,b)	(d,d)	(d,e)
(a,b)	0	1	1	1	0	0	1	0	0	0	0	0
(a,d)	1	0	1	0	1	0	0	1	0	0	0	0
(a,e)	1	1	0	0	0	1	0	0	1	0	0	0
(b,b)	1	0	0	0	1	1	0	0	0	1	0	0
(b,d)	0	1	0	1	0	1	0	0	0	0	1	0
(b,e)	0	0	1	1	1	0	0	0	0	0	0	1
(c,b)	1	0	0	0	0	0	0	1	1	1	0	0
(c,d)	0	1	0	0	0	0	1	0	1	0	1	0
(c,e)	0	0	1	0	0	0	1	1	0	0	0	1
(d,b)	0	0	0	1	0	0	1	0	0	0	1	1
(d,d)	0	0	0	0	1	0	0	1	0	1	0	1
(d,e)	0	0	0	0	0	1	0	0	1	1	1	0

Automaton on $C_4\,[C_3]$:

The finite automata $C_4\,[C_3]$ can be obtained by using the composition operation of graphs.

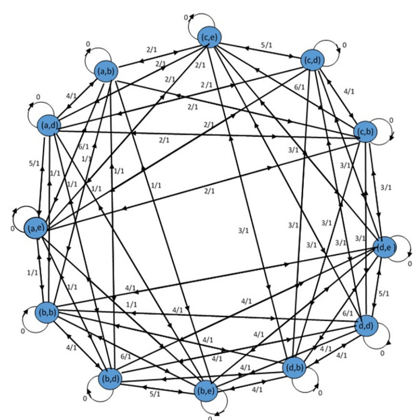

Fig. 12: State diagram of $C_4\,[C_3]$

Table 11 : Transition table of $C_4\,[C_3]$

F	1	2	3	4	5	6
(a,b)	$(b,b),(b,d),(b,e)$	(c,b),(c,d), (c,e)	-	(a,d)	-	(a,e)
(a,d)	$(b,b),(b,d),(b,e)$	$(c,b),(c,d),(c,e)$	-	(a,b)	(a,e)	-
(a,e)	$(b,b),(b,d),(b,e)$	$(c,b),(c,d),(c,e)$	-	-	(a,d)	(a,b)
(b,b)	$(a,b),(a,d),(a,e)$	-	-	$(b,d),(d,b),(d,d),(d,e)$	-	(b,e)
(b,d)	$(a,b),(a,d),(a,e)$	-	-	$(b,b),(d,b),(d,d)(d,e)$	(b,e)	-
(b,e)	$(a,b),(a,d),(a,e)$	-	-	$(b,b),(d,b),(d,d)(d,e)$	(b,d)	(b,b)
(c,b)	-	$(a,b),(a,d),(a,e)$	$(d,b),(d,d),(d,e)$	(c,d)	-	(c,e)
(c,d)	-	$(a,b),(a,d),(a,e)$	$(d,b),(d,d),(d,e)$	(c,b)	(c,e)	-
(c,e)	-	$(a,b),(a,d),(a,e)$	$(d,b),(d,d),(d,e)$	-	(c,d)	(c,b)
(d,b)	-	-	$(c,b),(c,d),(c,e)$	$(b,b),(b,d),(b,e),(d,d)$	-	(d,e)
(d,d)	-	-	$(c,b),(c,d),(c,e)$	$(b,b),(b,d),(b,e),(d,b)$	(d,e)	-
(d,e)	-	-	$(c,b),(c,d),(c,e)$	$(b,b),(b,d),(b,e)$	(d,d)	(d,b)

Table 12 : Output table of $C_4\,[C_3]$

G	(a,b)	(a,d)	(a,e)	(b,b)	(b,d)	(b,e)	(c,b)	(c,d)	(c,e)	(d,b)	(d,d)	(d,e)
(a,b)	0	1	1	1	1	1	1	1	1	0	0	0
(a,d)	1	0	1	1	1	1	1	1	1	0	0	0
(a,e)	1	1	0	1	1	1	1	1	1	0	0	0
(b,b)	1	1	1	0	1	1	0	0	0	1	1	1
(b,d)	1	1	1	1	0	1	0	0	0	1	1	1
(b,e)	1	1	1	1	1	0	0	0	0	1	1	1
(c,b)	1	1	1	0	0	0	0	1	1	1	1	1
(c,d)	1	1	1	0	0	0	1	0	1	1	1	1
(c,e)	1	1	1	0	0	0	1	1	0	1	1	1
(d,b)	0	0	0	1	1	1	1	1	1	0	1	1
(d,d)	0	0	0	1	1	1	1	1	1	1	0	1
(d,e)	0	0	0	1	1	1	1	1	1	1	1	0

4. Observations

From above, it is seen that the graph-theoretic properties are applicable to automata and are helpful in deriving new automatons from two known automata. It is also observed that the graphical representation of the resultant automaton contains the given automata as its subgraphs. From $C_4 \cup C_3 \subseteq C_4 \oplus C_3 \subseteq C_4 + C_3$, it is noted that the automaton obtained from the union is contained by the automaton obtained from the ring sum, and both of them are contained by the automaton obtained from the sum of the graphs. The output table of the resulting automaton can be obtained from the adjacency matrix of its graph.

5. Future Work and Discussion

From the above study, it is clear that there is a broad field of study on automata with other properties of the graph. With the help of the adjacency matrix and its eigenvalues, non-linear dynamical systems (automata) can be studied. A directed graph, on the other hand, will help to understand the direction of the transition in a finite automaton. So, there is a broad scope of study awaiting in this direction that can bring automatons and graph theory together.[3,4]

6. Conclusion

Automata and graph theory both have great applications in different branches of science, viz., computer science, networking, communication, transportation, robotics, etc. So if a connection can be established between them, then it can be very helpful in studying various topics with the help of each other in all those branches where there are many applications. It is seen from the study that there is definitely a connection between graph theory and automata, and further study will bring out more fruitful results.

References

[1] F. Harary, Graph Theory, Narosa Publishing House, Reprint, 2013.

[2] J.E. Hopcroft, R. Motwani, J.D. Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, 2nd edition, 2000.

[3] J. Julie, J.B. Babujee, Automata Resulting from Graph Operations, IEEE-fourth International Conference on Advanced Computing,(ICoAC), MIT, Anna University, Chennai, pp. 13 – 15, 2012.

[4] B. Mikolajczak, Algebraic and Structural Automata Theory, Annals of Discrete Mathematics, 44, Elsevier Science Publishing Company, 1991.

[5] K.L.P. Mishra, M. Chandrasekaran, Theory of Computer Science; Automata, Languages and Computation, Third edition, Publisher-Prentice-Hall of India, 2007.

[6] M. Nouri, S. Talatahari, A.S. Shamlou, Graph Products and its Applications in Mathematical Formulation of Structures, Journal of Applied Mathematics, Jan-1, Vol.2012, Article ID 510180, 16 pages, Publisher: Hindawi Publishing Corporation, 2012.
[7] P. Patel, C. Patel, Various Graphs and their Applications in Real World, International Journal of Engineering Research and Technology (IJERT), Vol. 2(12), pp. 208 - 226, Paper ID : IJERTV2IS120024, 2013.
[8] N. Subashini, K. Thiagarajan, Automaton for Connected Graph and its Complement Graph, International Journal of Pure and Applied Mathematics, Vol.119, No. 10, pp. 981-988, 2018.

M. Dutta
Department of Mathematics, Dudhnoi College, Dudhnoi-783124, Goalpara, Assam, India
Email address: mrldulduttamc@gmail.com

S. Kalita
Department of Mathematics, School of Fundamental and Applied Sciences, Assam Don Bosco University, Tapesia-782402, Kamrup, Assam, India
Email address: sanjoykalita1@gmail.com

H. K. Saikia
Department of Mathematics, Gauhati University, Guwahati-781014, Assam, India
Email address: hsaikia@yahoo.com