ISSN 2313-1527 (PRINT)
ISSN 2313-1535 (ONLINE)

IJEIE

International Journal of Electronics and Information Engineering

Vol. 8, No. 2 (June 2018)

INTERNATIONAL JOURNAL OF ELECTRONICS & INFORMATION ENGINEERING

PUBLISHING OFFICE

Min-Shiang Hwang
Department of Computer Science & Information Engineering, Asia University, Taichung 41354, Taiwan, R.O.C.
Email: mshwang@asia.edu.tw

International Journal of Electronics and Information Engineering is published both in traditional paper form (ISSN 2313-1527) and in Internet (ISSN 2313-1535) at http://ijeie.jalaxy.com.tw

PUBLISHER: Candy C. H. Lin

23-75, P.O. Box, Taichung, Taiwan 40199, R.O.C.

International Journal of Electronics and Information Engineering

Vol. 8, No. 2 (June 1, 2018)

Products of Monoids and Its Applications on the Monoids of State Machines

Nacer Ghadbane and Douadi Mihoubi

(Corresponding author: Nacer Ghadbane)

Laboratory of Pure and Applied Mathematics, Department of Mathematics, University of M'sila, Algeria

(Email: Nacer.ghadbane@yahoo.com)

(Received Dec. 12, 2017; revised and accepted Feb. 20, 2018)

Abstract

Let M be a monoid and X a non-empty set. M will be called a transformation monoid on X if there is a mapping $\phi : M \times X \longrightarrow X$, for which we write $\phi(m,x) = m \cdot x$ and which satisfies the conditions:

1) $(m_1 m_2) \cdot x = m_1 \cdot (m_2 \cdot x)$, for each $x \in X$ and for each $m_1, m_2 \in M$.
2) $1_M \cdot x = x$, for each $x \in X$.

Let M and N be two monoids. Let N^M be the set of all functions defined on M with values in N. In this paper, we prove that the set N^M forms a monoid shch that for any $\varphi, \psi \in N^M$, let $\varphi\psi \in N^M$ in N^M be defined for all $m \in M$ by: $(\varphi\psi)(m) = \varphi(m)\psi(m)$, the monoid M is a transformation monoid on N^M in the following was:
if $m \in M, \varphi \in N^M$, then $(m \cdot \varphi)(x) = \varphi^m(x) = \varphi(xm)$ for $x \in M$, and the set of all pairs (m, φ) where $m \in M, \varphi \in N^M$, with multiplications operation given by: $(m,\varphi)(m',\psi) = (mm', \varphi\psi^m)$ where $m, m' \in M$ and $\varphi, \psi \in N^M$ is a monoid. On the other hand, we present the direct product, the cascade product and wreath product of state machines, also we calculate the monoids of state machines.

Keywords: Free Monoid; Monoid of State Machine; Morphism of Monoids; State Machine; Transformation Monoid; Wreath Product of Monoids

1 Introduction

The theory of machines that has developed in last twenty years, has had a considerable influence, not only on the computer systems, but also biology, biochemestry, *etc.*

A semigroup is an ordered pair $(S, \cdot)$, where S is non -empty set and the dot " $\cdot$" is a binary operation on S, i.e., a mapping $(a,b) \longmapsto a \cdot b$ from $S \times S$ to S such that for all $a, b, c \in S, (a \cdot b) \cdot c = a \cdot (b \cdot c)$ (associative law).

A semigroup $(S, \cdot)$ with the identity element is called a monoid [10].

A state machine is a 3-tuple (Q, Σ, δ), where

1) Q is a finite nonempty set (the set of states);
2) Σ is a finite alphabet (the set of inputs);
3) δ is a function of $Q \times \Sigma$ into Q (the transition function) [6, 7].

The remainder of this paper is organized as follows. In Section 2, we begin with some elementary material concerning of monoids and state machines. In Section 3, we present direct product, semidirect product and wreath product of monoids. In Section 4, we introduce the direct product of state machines, the cascade product and wreath product, also we calculate the monoids of state machines. Finally, we draw our conclusions in Section 5.

2 Preliminaries

A monoid $(M, \cdot)$ consists of a set M together with a binary operation " $\cdot$ " on M such that

1) $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ for all $a, b, c \in M$;

2) There existes an identity $1_M \in M$ such that

$$a \cdot 1_M = 1_M \cdot a = a \text{ for all } a \in M.$$

A monoid $(M, \cdot)$ is called commutative if the operation " $\cdot$ " is commutative. Hence a semigroup $(S, \cdot)$ is just a set S together with an associative binary operation.

Let X be any set and let $X^X = \{f : X \longrightarrow X\}$ be the set of all function from X to itself. Then $\left(X^X, \circ\right)$ is a monoid, called the transformation monoide of X. In fact, the analogue of Cayley's theorem holds for monoids, and it can be shown that every monoid can be represented as a transformation monoid.

Suppose that $\mathcal{R}$ is an equivalence relation on a monoid $(M, \cdot)$. Then $\mathcal{R}$ is called a congruence relation on $(M, \cdot)$ if $a\mathcal{R}b$ implies $ac\mathcal{R}bc$ and $ca\mathcal{R}cb$ for all $a, b, c \in M$. The congruence class containing the element $m \in M$ is the set $[m] = \{x \in M : x\mathcal{R}m\}$.

If $\mathcal{R}$ is a congruence relation on the monoid $(M, \cdot)$, the quotient set $M/\mathcal{R} = \{[m] : m \in M\}$ is a monoid under the operation defined by $[m]\,[n] = [mn]$. This monoid is called the quotient monoid of M by $\mathcal{R}$.

If $(M, \cdot)$ and (N, Δ) are two monoids, with identities 1_M and 1_N, respectively, then the function $f : M \longrightarrow N$ is a monoid morphism from $(M, \cdot)$ to (N, Δ) if

1) $f(x \cdot y) = f(x)\,\Delta f(y)$ for all $x, y \in M$,

2) $f(1_M) = 1_N$.

A monoid isomorphism is simply a bijective monoid morphism.

Let M be a monoid and X a non-empty set. M will be called a transformation monoid on X if there is a mapping $\phi : M \times X \longrightarrow X$, for which we write $\phi(m, x) = m \cdot x$ and which satisfies the conditions:

1) $(m_1 m_2) \cdot x = m_1 \cdot (m_2 \cdot x)$, for each $x \in X$ and for each $m_1, m_2 \in M$.

2) $1_M \cdot x = x$, for each $x \in X$.

For every transformation monoid M on X, there is a homomorphism $\psi : M \longrightarrow E(X)$, the monoid of all mappings $f : X \longrightarrow X$, given by $\psi(m) = f$, where $f(x) = m \cdot x$ for all $x \in X$.

We formally define an alphabet as a non-empty finite set. A word over an alphabet Σ is a finite sequence of symbols of Σ. Although one writes a sequence as $(\sigma_1, \sigma_2, \cdots, \sigma_n)$, in the present context, we prefer to write it as $\sigma_1\sigma_2 \cdots \sigma_n$. The set of all words on the alphabet Σ is denoted by Σ^* and is equipped with the associative operation defined by the concatenation of two sequences [1, 5]:

$$\alpha_1\alpha_2 \cdots \alpha_n\beta_1\beta_2 \cdots \beta_m = \alpha_1\alpha_2 \cdots \alpha_n\beta_1\beta_2 \cdots \beta_m.$$

This operation is associative. This allows us to write $w = \sigma_1\sigma_2\cdots\sigma_n$. The string consisting of zero letters is called the empty word, written ϵ. Thus, $\epsilon, \alpha, \beta, \alpha\alpha\beta\alpha, \alpha\alpha\alpha\beta\alpha$ are words over the alphabet $\{\alpha, \beta\}$. Thus the set Σ^* of words is equipped with the structure of a monoid. the monoid Σ^* is called the free monoid on Σ. The length of a word w, in symbols $|w|$, 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 $|\alpha\beta\alpha| = 3$ and $|\alpha\beta\alpha\beta\alpha\alpha| = 6$. Let w be a word over an alphabet Σ. For $\sigma \in \Sigma$, the number of occurrences of σ in w shall be denoted by $|w|_\sigma$. For example $|\alpha\beta\alpha|_\beta = 1$ and $|\alpha\beta\alpha\beta\alpha\alpha|_\alpha = 4$ [3, 4].

Let $(\Sigma^*, \cdot)$ be the free monoid generated by Σ and let $i : \Sigma \longrightarrow \Sigma^*$ be the function that maps each element σ of Σ into the corresponding word of length 1, so that $i(\sigma) = \sigma$. Then if $f : \Sigma \longrightarrow M$ is any function into the underlying set of any monoid $(M, \cdot)$, there is a unique monoid morphism $h : (\Sigma^*, \cdot) \longrightarrow (M, \cdot)$ such that $h \circ i = f$.

A state machine is a triple $S = (Q, \Sigma, \delta)$ where Q is a finite set of states, Σ is a finite set of symbols called the input alphabet and $\delta : Q \times \Sigma \longrightarrow Q$ is a partial function called the transition function. The transition function can be extended naturally to sequences of input symbols, by letting $\delta(q, w\sigma) = \delta(\delta(q, w), \sigma)$ and $\delta(q, \epsilon) = q$, for all $w \in \Sigma^*, \sigma \in \Sigma$ and $q \in Q$.

A state machine $S = (Q, \Sigma, \delta)$ is called complete if the partial function $\delta : Q \times \Sigma \longrightarrow Q$ is in fact a function. In this situation we can specify what the resultant $\delta(q, \sigma)$ is for all possible combinations of $q \in Q$ and $\sigma \in \Sigma$.

Let $\sigma \in \Sigma$, define $\delta_\sigma : Q \longrightarrow Q$ by $\delta_\sigma(q) = \delta(q, \sigma)$ for each $q \in Q$. Let $w \in \Sigma^+$ be a word of length at least 1 with symbols from Σ. Suppose that $w = \sigma_1\sigma_2\cdots\sigma_n$ then we define $\delta_w : Q \longrightarrow Q$ by $\delta_w(q) = \delta_{\sigma_n}\delta_{\sigma_{n-1}}\cdots\delta_{\sigma_1}(q)$.

Let $S = (Q, \Sigma, \delta)$ be a state machine, define the monoid morphism $h : (\Sigma^*, \cdot) \longrightarrow (Q^Q, \circ)$ by $h(w) = \delta_w$. Define the relation $\mathcal{R}$ on Σ^* by $w\mathcal{R}w'$ if and only if $h(w) = h(w')$. This is easily verified to be an equivalence relation. Furthermore, it is a congruence relation. The qoutient monoid $(\Sigma^*/\mathcal{R}, \cdot)$ is called the monoid of the state machine (Q, Σ, δ).

3 Direct Product, Semidirect Product and Wreath Product of Monoids

In this section, we present direct product, semidirect product and wreath product of monoids.

Proposition 1. *Let M and N be monoids, consider the set $M \times N$ the cartesian product of M and N, and define a multiplication "$\cdot$" on $M \times N$ as follows:*

$$(m_1, n_1) \cdot (m_2, n_2) = (m_1m_2, n_1n_2).$$

This result is a monoid $(M \times N, \cdot)$ with is called the direct product of M and N.

Proof. It is easy to show that this product is associative and the identity element in $M \times N$ is $(1_M, 1_N)$. □

Example 1. *Let $M = (\mathbb{N}, +)$ and $N = (\mathbb{N}, \times)$, then in the direct product $M \times N$ we have $(m, n) \cdot (r, s) = (m + r, n \times s)$.*

Proposition 2. *Given three monoids M, N, L we can form the direct product $(M \times N) \times L$ similarly $M \times (N \times L)$ and the relationship between these two monoids is the isomorphism [6]:*

$$(M \times N) \times L \cong M \times (N \times L).$$

Proof. The isomorphism is $h : (M \times N) \times L \longrightarrow M \times (N \times L)$ defined by

$$h((m,n),l) = (m,(n,l)),$$

where $m \in M, n \in N, l \in L$. □

Proposition 3. *Given any monoids M and N, suppose that $\theta : N \longrightarrow End(M)$ is a monoid morphism. Then $(M \times N, \cdot)$ is a monoid under the operation " $\cdot$ " defined by $(m,n) \cdot (m',n') = (m\theta(n)(m'), nn')$ where $m, m' \in M, n, n' \in N$. The monoid $(M \times N, \cdot)$ is called the semidirect product of M and N with respect to θ and it is denoted by $M \times_\theta N$.* [1, 8, 9]

Proof. From [6]:

1) We will prove that " $\cdot$ " is associative on $M \times N$: let $m, m', m" \in M, n, n', n'' \in N$, we have

$$\begin{aligned} & ((m,n)\cdot(m',n'))\cdot(m'',n'') \\ = \; & (m\theta(n)(m'), nn')\cdot(m'',n'') \\ = \; & (m\theta(n)(m')\theta(nn')(m''), (nn')n'') \\ = \; & (m\theta(n)(m')\theta(n)(\theta(n')(m'')), (nn')n'') \\ = \; & (m\theta(n)(m'\theta(n')(m'')), (nn')n''). \end{aligned}$$

Also we have

$$\begin{aligned} & (m,n)\cdot((m',n')\cdot(m'',n'')) \\ = \; & (m,n)(m'\theta(n')(m''), n'n'') \\ = \; & (m\theta(n)(m'\theta(n')(m'')), n(n'n'')). \end{aligned}$$

Then " $\cdot$ " is associative on $M \times N$.

2) For $(m,n) \in M \times N$, $(m,n)\cdot(1_M, 1_N) = (m\theta(n)(1_M), n1_N) = (m1_M, n) = (m,n)$. Also $(1_M, 1_N)\cdot(m,n) = (1_M\theta(1_N)(m), 1_N n) = (1_M Id_M(m), n) = (m,n)$. Hence $(M \times N, \cdot)$ is a monoid.

□

Proposition 4. *Let M and N be two monoids. Let N^M be the set of all functions defined on M with values in N.*

1) The set N^M forms a monoid shch that for any $\varphi, \psi \in N^M$, let $\varphi\psi \in N^M$ in N^M be defined for all $m \in M$ by: $(\varphi\psi)(m) = \varphi(m)\psi(m)$.

2) The monoid M is a transformation monoid on N^M in the following was:

- *If $m \in M, \varphi \in N^M$, then $(m \cdot \varphi)(x) = \varphi^m(x) = \varphi(xm)$ for $x \in M$.*

3) The set of all pairs (m, φ) where $m \in M, \varphi \in N^M$, with multiplications operation given by: $(m,\varphi)(m',\psi) = (mm', \varphi\psi^m)$ where $m, m' \in M$ and $\varphi, \psi \in N^M$ is a monoid [1, 8, 9].

Proof.

1) First we will prove that the set N^M froms a monoid shch that for any $\varphi, \psi \in N^M$, let $\varphi\psi$ in N^M be defined for all $m \in M$ by: $(\varphi\psi)(m) = \varphi(m)\psi(m)$.

a. N^M is non-empty and is closed with respect to multiplication. If $\varphi, \psi \in N^M$, then $\varphi(m), \psi(m) \in N$, for all $m \in M$. Hence $\varphi(m)\psi(m) \in N$. This implies that $(\varphi\psi)(m) \in N$ and so $\varphi\psi \in N^M$.

b. Since multuplication in N is associative, so also is the multuplication in N^M.

c. The identity element in N^M is the map $e : M \longrightarrow N$ given by $e(m) = 1_N$, for all $m \in M$, where 1_N is the identity element of N.

2) Second, we will prove that M is a transformation monoid on N^M in the following was: if $m \in M, \varphi \in N^M$, then $(m \cdot \varphi)(x) = \varphi^m(x) = \varphi(xm)$ for $x \in M$.

Take $\varphi, \psi \in N^M$ and $m, m' \in M$, then

$$\begin{aligned} ((mm') \cdot f)(x) &= f(xmm') \\ (m \cdot (m' \cdot f))(x) &= \left(m \cdot f^{m'}\right)(x) \\ &= \left(f^{m'}\right)^m (x) \\ &= f^{m'}(xm) = f(xmm'). \\ \varphi^{1_M}(x) &= \varphi(x1_M) \\ &= \varphi(x). \\ (\varphi\psi)^m(x) &= \varphi\psi(xm) \\ &= \varphi(xm)\psi(xm) \\ &= \varphi^m(x)\psi^m(x). \\ (\varphi^m)^{m'}(x) &= \varphi^m(xm') \\ &= \varphi(xm'm) \\ &= \varphi^{m'm}(x). \end{aligned}$$

Then $(\varphi^m)^{m'} = \varphi^{m'm}$.

3) We will prove that $M \times N^M$ is a monoid with multiplication:

$$(m, \varphi)(m', \psi) = (mm', \varphi\psi^m)$$

where $m, m' \in M$ and $\varphi, \psi \in N^M$:

a. Closure property follows from the definition of multiplication.

b. Take $\varphi, \psi, \eta \in N^M$ and $m, m', m" \in G$, then

$$\begin{aligned} ((m, \varphi)(m', \psi))(m'', \eta) &= (mm', \varphi\psi^m)(m'', \eta) \\ &= \left((mm')m'', \varphi\psi^m\eta^{mm'}\right). \end{aligned}$$

Also we have

$$\begin{aligned} (m, \varphi)((m', \psi)(m'', \eta)) &= (m, \varphi)\left(m'm'', \psi\eta^{m'}\right) \\ &= \left(m(m'm''), \varphi\left(\psi\eta^{m'}\right)^m\right) \\ &= \left(m(m'm''), \varphi\psi^m\eta^{mm'}\right). \end{aligned}$$

c. We know that for every $\varphi \in N^M, \varphi^{1_M} = \varphi$, now for every $m \in M$, the map $\varphi \longrightarrow \varphi^m$ is an automorphism of N^M. Also if e is the identity element in N^M, then $e^m = e$. We have $(m,\varphi)(1_M, e) = (m1_M, \varphi e^m) = (m, \varphi e) = (m,\varphi)$. Also $(1_M, e)(m,\varphi) = \left(1_M m, e\varphi^{1_M}\right) = (m, e\varphi) = (m,\varphi)$. Thus identity element exists.

Hence $M \times N^M$ is a monoid with respect to the multiplication defined above. □

Remark 1. *If the monoid M is commutative, then $M \times N^M$ is a monoid with multiplication $(m,\varphi)(m',\psi) = \left(mm', \varphi^{m'}\psi\right)$ where $m, m' \in M$ and $\varphi, \psi \in N^M$.*

Proposition 5. *Let M and N be two monoids, and let M^N denote the set of all functions from the monoid N to the monoid M, then the set $M^N \times N$ is a monoid under the multiplication $(\varphi, n_1)(\psi, n_2) = (\varphi\psi, n_1 n_2)$ where $\varphi\psi \in M^N$ is defined by*

$$\varphi\psi(x) = \varphi(x)\psi(xn_1)$$

for $x, n_1, n_2 \in N$ and $\varphi, \psi \in M^N$.

We call the monoid $M^N \times N$ the wreath product of M and N [1, 8, 9].

Proof. From [6]:

1) We will prove that the multiplication is associative on $M^N \times N$. Let $\varphi, \psi, \eta \in M^N$ and $n_1, n_2, n_3 \in N$ then

$$\begin{aligned}((\varphi, n_1)(\psi, n_2))(\eta, n_3) &= (\varphi\psi, n_1 n_2)(\eta, n_3) \\ &= ((\varphi\psi)\eta, n_1 n_2 n_3).\end{aligned}$$

And

$$\begin{aligned}(\varphi, n_1)((\psi, n_2)(\eta, n_3)) &= (\varphi, n_1)(n_2 n_3, \psi\eta) \\ &= (\varphi(\psi\eta), n_1 n_2 n_3).\end{aligned}$$

The, we will prove that

$$(\varphi\psi)\eta = \varphi(\psi\eta).$$

Let $x \in N$, then

$$\begin{aligned}((\varphi\psi)\eta)(x) &= (\varphi\psi)(x)\eta(xn_1 n_2) \\ &= \varphi(x)\psi(xn_1)\eta(xn_1 n_2).\end{aligned}$$

And

$$\begin{aligned}(\varphi(\psi\eta))(x) &= \varphi(x)\psi\eta(xn_1) \\ &= \varphi(x)\psi(xn_1)\eta(xn_1 n_2).\end{aligned}$$

2) Let the map $e : N \longrightarrow M$ given by $e(n) = 1_M$ for all $n \in N$. We have

$$\begin{aligned}(\varphi, n)(e, 1_N) &= (\varphi e, n1_N) \\ &= (\varphi e, n),\end{aligned}$$

where

$$\begin{aligned}(\varphi e)(x) &= \varphi(x)\, e(xn) \\ &= \varphi(x)\, 1_M = \varphi(x)\end{aligned}$$

for $n, x \in N$ and $\varphi \in M^N$, then

$$(\varphi, n)(e, 1_N) = (\varphi, n).$$

Also

$$\begin{aligned}(e, 1_N)(\varphi, n) &= (e\varphi, 1_N n) \\ &= (e\varphi, n),\end{aligned}$$

where

$$\begin{aligned}(e\varphi)(x) &= e(x)\,\varphi(x 1_N) \\ &= 1_M \varphi(x) \\ &= \varphi(x).\end{aligned}$$

Then $(e, 1_N)(\varphi, n) = (\varphi, n)$. The identity element in $M^N \times N$ is $(e, 1_N)$.

□

4 Applications of Products of Monoids on The Monoids of State Machines

In this section, we present the direct product of state machines, the cascade product and wreath product.

Definition 1. *Let $S_1 = (Q_1, \Sigma_1, \delta_1)$ and $S_2 = (Q_2, \Sigma_2, \delta_2)$ be state machines. Suppose that S_1 and S_2 are state machines with the same unput Σ. Connecting them up in this way, will produce a new state machine $S_1 \times S_2 = (Q_1 \times Q_2, \Sigma, \delta_1 \times \delta_2)$ where $(\delta_1 \times \delta_2)((q_1, q_2), \sigma) = (\delta_1(q_1, \sigma), \delta_2(q_2, \sigma))$ for $\sigma \in \Sigma, (q_1, q_2) \in Q_1 \times Q_2$.*
We call this state machine the restricted direct product of S_1 and S_2 [5, 6].

Example 2. *Let $S_1 = (Q_1, \Sigma_1, \delta_1)$ be state machine where $Q_1 = \{0, 1\}, \Sigma_1 = \{\sigma\}$ and $\delta_1 : Q_1 \times \Sigma_1 \longrightarrow Q_1$ given by:*

δ_1	σ
0	1
1	0

and $S_2 = (Q_2, \Sigma_2, \delta_2)$ given by

$$\begin{aligned}Q_2 &= \{0, 1\}, \\ \Sigma_2 &= \{\sigma\} \\ \delta_2 : Q_2 \times \Sigma_2 &\longrightarrow Q_2\end{aligned}$$

δ_2	σ
0	1
1	1

Then $S_1 \times S_2 = (Q_1 \times Q_2, \Sigma, \delta_1 \times \delta_2)$ where $Q_1 \times Q_2 = \{(0,0), (0,1), (1,0), (1,1)\}$, $\delta_1 \times \delta_2 : (Q_1 \times Q_2) \times \Sigma \longrightarrow (Q_1 \times Q_2)$ given by:

$\delta_1 \times \delta_2$	σ
$(0,0)$	$(1,1)$
$(0,1)$	$(1,1)$
$(1,0)$	$(0,1)$
$(1,1)$	$(0,1)$

Define $\delta_\sigma : Q_1 \longrightarrow Q_1$ *by* $\delta_\sigma(q) = \delta(q,\sigma)$ *for each* $q \in Q_1$. *We have* $\delta_\sigma = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ *and* $\delta_\sigma \circ \delta_\sigma = Id_{Q_1}$, *Let* $w \in \{\sigma\}^+$ *be a word of length at least* 1 *with symbols from* $\{\sigma\}$. *Suppose that* $w = \sigma^n, n \in \mathbb{N}^*$ *then we define* $(\delta_1)_w : Q_1 \longrightarrow Q_1$ *by* $(\delta_1)_w(q) = \delta_\sigma\delta_\sigma \cdots \delta_\sigma(q)$. *We have*

$$\delta_w = \begin{cases} Id_{Q_1} & \text{if } n = 2k, k \in \mathbb{N} \\ \delta_\sigma & \text{if } n = 2k+1, k \in \mathbb{N} \end{cases}$$

$$h_1 : (\{\sigma\}^*, \cdot) \longrightarrow \left(Q_1^{Q_1}, \circ\right)$$

by $h_1(w) = \delta_w$. *Define the relation* $\mathcal{R}_1$ *on* $\{\sigma\}^*$ *by* $w\mathcal{R}_1w'$ *if and only if* $h_1(w) = h_1(w')$. *This is easily verified to be an equivalence relation. Furthermore, it is a congruence relation. The qoutient* $\{\sigma\}^*/\mathcal{R}_1 = \{[\epsilon], [\sigma]\}$.

The monoid of the state machine $(Q_1, \{\sigma\}, \delta_1)$ *is given by:*

$\cdot$	$[\epsilon]$	$[\sigma]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$
$[\sigma]$	$[\sigma]$	$[\epsilon]$

Define $\delta_2 : Q_2 \longrightarrow Q_2$ *by* $\delta_\sigma(q) = \delta(q,\sigma)$ *for each* $q \in Q_2$. *We have* $\delta_\sigma = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$, *Let* $w \in \{\sigma\}^+$ *be a word of length at least* 1 *with symbols from* $\{\sigma\}$. *Suppose that* $w = \sigma^n, n \in \mathbb{N}^*$ *then we define* $(\delta_2)_w : Q_2 \longrightarrow Q_2$ *by* $(\delta_2)_w(q) = \delta_\sigma\delta_\sigma \cdots \delta_\sigma(q)$. *We have*

$$\delta_w = \begin{cases} Id_{Q_1} & \text{if } n = 0 \\ \delta_\sigma & \text{if } n \in \mathbb{N}^* \end{cases}$$

$$h_2 : (\{\sigma\}^*, \cdot) \longrightarrow \left(Q_2^{Q_2}, \circ\right)$$

by $h_2(w) = \delta_w$. *Define the relation* $\mathcal{R}_2$ *on* $\{\sigma\}^*$ *by* $w\mathcal{R}_2w'$ *if and only if* $h_2(w) = h_2(w')$. *This is easily verified to be an equivalence relation. Furthermore, it is a congruence relation. The qoutient* $\{\sigma\}^*/\mathcal{R}_2 = \{[\epsilon], [\sigma]\}$.

The monoid of the state machine $(Q_2, \{\sigma\}, \delta_2)$ *is given by:*

$\cdot$	$[\epsilon]$	$[\sigma]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$
$[\sigma]$	$[\sigma]$	$[\sigma]$

Define $(\delta_1 \times \delta_2)_\sigma : Q_1 \times Q_2 \longrightarrow Q_1 \times Q_2$ *by* $(\delta_1 \times \delta_2)_\sigma((q_1,q_2)) = (\delta_1(q_1,\sigma), \delta_2(q_2,\sigma))$ *for each* $(q_1,q_2) \in Q_1 \times Q_2$. *We have*

$$\begin{aligned} (\delta_1 \times \delta_2)_\sigma &= \varphi = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,1) & (1,1) & (0,1) & (0,1) \end{pmatrix}, \\ \varphi^2 &= \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,1) & (0,1) & (1,1) & (1,1) \end{pmatrix}, \\ \varphi^3 &= \varphi. \end{aligned}$$

Let $w \in \{\sigma\}^+$ *be a word of length at least* 1 *with symbols from* $\{\sigma\}$. *Suppose that* $w = \sigma^n, n \in \mathbb{N}^*$ *then we define*

$$(\delta_1 \times \delta_2)_w : Q_1 \times Q_2 \longrightarrow Q_1 \times Q_2$$

by

$$(\delta_1 \times \delta_2)_w((q_1,q_2)) = \delta_\sigma\delta_\sigma \cdots \delta_\sigma((q_1,q_2)).$$

We have

$$(\delta_1 \times \delta_2)_w = \begin{cases} Id_{Q_1} & \text{if } n = 0 \\ \varphi & \text{if } n = 2k+1, k \in \mathbb{N} \\ \varphi^2 & \text{if } n = 2k, k \in \mathbb{N}^* \end{cases}$$

$$\psi : (\{\sigma\}^*, \cdot) \longrightarrow \left((Q_1 \times Q_2)^{Q_1 \times Q_2}, \circ\right)$$

by $\psi(w) = (\delta_1 \times \delta_2)_w$.

Define the relation $\mathcal{R}$ *on* $\{\sigma\}^*$ *by* $w\mathcal{R}w'$ *if and only if* $\psi(w) = \psi(w')$. *This is easily verified to be an equivalence relation. Furthermore, it is a congruence relation. The qoutient* $\{\sigma\}^* / \mathcal{R} = \left\{[\epsilon], [\sigma], [\sigma^2]\right\}$. *The monoid of the state machine* $(Q_1 \times Q_2, \{\sigma\}, \delta_1 \times \delta_2)$ *is given by:*

$\cdot$	$[\epsilon]$	$[\sigma]$	$[\sigma^2]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$	$[\sigma^2]$
$[\sigma]$	$[\sigma]$	$[\sigma^2]$	$[\sigma]$
$[\sigma^2]$	$[\sigma^2]$	$[\sigma]$	$[\sigma^2]$

Definition 2. *Let* $S_1 = (Q_1, \Sigma_1, \delta_1)$ *and* $S_2 = (Q_2, \Sigma_2, \delta_2)$ *be state machines. We define*

$$S_1 \times S_2 = (Q_1 \times Q_2, \Sigma_1 \times \Sigma_2, \delta_1 \times \delta_2)$$

where

$$(\delta_1 \times \delta_2)((q_1, q_2), (\sigma_1, \sigma_2)) = (\delta_1(q_1, \sigma_1), \delta_2(q_2, \sigma_2))$$

for

$$(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2, (q_1, q_2) \in Q_1 \times Q_2.$$

We call this state machine the full direct product of S_1 *and* S_2 *[5, 6].*

Example 3. *Let* $S_1 = (Q_1, \Sigma_1, \delta_1)$ *be state machine where* $Q_1 = \{0, 1\}$, $\Sigma_1 = \{\sigma\}$ *and* $\delta_1 : Q_1 \times \Sigma_1 \longrightarrow Q_1$ *given by:*

δ_1	σ
0	1
1	0

and $S_2 = (Q_2, \Sigma_2, \delta_2)$ *given by*

$$Q_2 = \{0, 1\},$$
$$\Sigma_2 = \{\sigma, \tau\}$$
$$\delta_2 : Q_2 \times \Sigma_2 \longrightarrow Q_2$$

δ_2	σ	τ
0	1	0
1	1	0

Then $S_1 \times S_2 = (Q_1 \times Q_2, \Sigma_1 \times \Sigma_2, \delta_1 \times \delta_2)$ *where* $Q_1 \times Q_2 = \{(0,0), (0,1), (1,0), (1,1)\}$, $\delta_1 \times \delta_2 : (Q_1 \times Q_2) \times \Sigma_1 \times \Sigma_2 \longrightarrow (Q_1 \times Q_2)$ *given by:*

$\delta_1 \times \delta_2$	(σ, σ)	(σ, τ)
$(0,0)$	$(1,1)$	$(1,0)$
$(0,1)$	$(1,1)$	$(1,0)$
$(1,0)$	$(0,1)$	$(0,0)$
$(1,1)$	$(0,1)$	$(0,0)$

Define $\delta_\sigma : Q_1 \longrightarrow Q_1$ *by* $\delta_\sigma(q) = \delta(q, \sigma)$ *for each* $q \in Q_1$*. We have*

$$\delta_\sigma = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

and $\delta_\sigma \circ \delta_\sigma = Id_{Q_1}$*, Let* $w \in \{\sigma\}^+$ *be a word of length at least 1 with symbols from* $\{\sigma\}$*. Suppose that* $w = \sigma^n, n \in \mathbb{N}^*$ *then we define* $(\delta_1)_w : Q_1 \longrightarrow Q_1$ *by* $(\delta_1)_w(q) = \delta_\sigma\delta_\sigma \cdots \delta_\sigma(q)$*. We have*

$$\delta_w = \begin{cases} Id_{Q_1} & \text{if } n = 2k, k \in \mathbb{N} \\ \delta_\sigma & \text{if } n = 2k+1, k \in \mathbb{N} \end{cases}$$

$$h_1 : (\{\sigma\}^*, \cdot) \longrightarrow \left(Q_1^{Q_1}, \circ\right)$$

by $h_1(w) = \delta_w$*. Define the relation* $\mathcal{R}_1$ *on* $\{\sigma\}^*$ *by* $w\mathcal{R}_1w'$ *if and only if* $h_1(w) = h_1(w')$*. This is easily verified to be an equivalence relation. Furthermore, it is a congruence relation. The qoutient* $\{\sigma\}^* / \mathcal{R}_1 = \{[\epsilon], [\sigma]\}$*.*

The monoid of the state machine $(Q_1, \{\sigma\}, \delta_1)$ *is given by:*

$\cdot$	$[\epsilon]$	$[\sigma]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$
$[\sigma]$	$[\sigma]$	$[\epsilon]$

Define $\delta_\sigma : Q_2 \longrightarrow Q_2$ *by* $\delta_\sigma(q) = \delta_2(q, \sigma)$ *for each* $q \in Q_2$*,* $\delta_\tau : Q_2 \longrightarrow Q_2$ *by* $\delta_\tau(q) = \delta_2(q, \tau)$ *for each* $q \in Q_2$*. We have*

$$\begin{aligned} \delta_\sigma &= \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}, \\ \delta_\tau &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \\ \delta_\sigma \circ \delta_\tau &= \delta_\sigma \circ \delta_\sigma \\ &= \delta_\sigma, \delta_\tau \circ \delta_\sigma \\ &= \delta_\tau \circ \delta_\tau \\ &= \delta_\tau, \end{aligned}$$

Let $w \in \{\sigma, \tau\}^+$ *be a word of length at least 1 with symbols from* $\{\sigma, \tau\}$*. We have*

$$\delta_w = \begin{cases} Id_{Q_1} & \text{if } w = \epsilon \\ \delta_\sigma & \text{if } w \in \{\sigma, \tau\}^* \sigma \\ \delta_\tau & \text{if } w \in \{\sigma, \tau\}^* \tau \end{cases}$$

$$h_2 : (\{\sigma\}^*, \cdot) \longrightarrow \left(Q_2^{Q_2}, \circ\right)$$

by $h_2(w) = \delta_w$*. Define the relation* $\mathcal{R}_2$ *on* $\{\sigma, \tau\}^*$ *by* $w\mathcal{R}_2w'$ *if and only if* $h_2(w) = h_2(w')$*. This is easily verified to be an equivalence relation. Furthermore, it is a congruence relation. The qoutient* $\{\sigma, \tau\}^* / \mathcal{R}_2 = \{[\epsilon], [\sigma], [\tau]\}$*.*

The monoid of the state machine $(Q_2, \{\sigma, \tau\}, \delta_2)$ *is given by:*

$\cdot$	$[\epsilon]$	$[\sigma]$	$[\tau]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$	$[\tau]$
$[\sigma]$	$[\sigma]$	$[\sigma]$	$[\tau]$
$[\tau]$	$[\tau]$	$[\sigma]$	$[\tau]$

.

Define $(\delta_1 \times \delta_2)_{(\sigma,\sigma)} : Q_1 \times Q_2 \longrightarrow Q_1 \times Q_2$ *by* $(\delta_1 \times \delta_2)_{(\sigma,\sigma)}((q_1, q_2)) = (\delta_1(q_1, \sigma), \delta_2(q_2, \sigma))$ *for each* $(q_1, q_2) \in Q_1 \times Q_2$. *We have*

$$\begin{aligned} (\delta_1 \times \delta_2)_{(\sigma,\sigma)} = \eta &= \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,1) & (1,1) & (0,1) & (0,1) \end{pmatrix}, \\ \eta^2 &= \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,1) & (0,1) & (1,1) & (1,1) \end{pmatrix}, \\ \eta^3 &= \eta. \end{aligned}$$

And $(\delta_1 \times \delta_2)_{(\sigma,\tau)} : Q_1 \times Q_2 \longrightarrow Q_1 \times Q_2$ *by* $(\delta_1 \times \delta_2)_{(\sigma,\tau)}((q_1, q_2)) = (\delta_1(q_1, \sigma), \delta_2(q_2, \tau))$ *for each* $(q_1, q_2) \in Q_1 \times Q_2$. *We have*

$$\begin{aligned} (\delta_1 \times \delta_2)_{(\sigma,\tau)} = \mu &= \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,0) & (1,0) & (0,0) & (0,0) \end{pmatrix}, \\ \mu^2 &= \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,0) & (0,0) & (1,0) & (1,0) \end{pmatrix}, \\ \mu^3 &= \mu. \end{aligned}$$

The monoid of the state machine $S_1 \times S_2 = (Q_1 \times Q_2, \Sigma_1 \times \Sigma_2, \delta_1 \times \delta_2)$ *is given by:*

$\cdot$	$[\epsilon]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\tau)]$	$[(\sigma,\tau)^2]$
$[\epsilon]$	$[\epsilon]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\tau)]$	$[(\sigma,\tau)^2]$
$[(\sigma,\sigma)]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\sigma)]$	$[(\sigma,\tau)^2]$	$[(\sigma,\tau)]$
$[(\sigma,\sigma)^2]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\tau)]$	$[(\sigma,\tau)^2]$
$[(\sigma,\tau)]$	$[(\sigma,\tau)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\sigma)]$	$[(\sigma,\tau)^2]$	$[(\sigma,\tau)]$
$[(\sigma,\tau)^2]$	$[(\sigma,\tau)^2]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\tau)]$	$[(\sigma,\tau)^2]$

Definition 3. *Let* $S_1 = (Q_1, \Sigma_1, \delta_1)$ *and* $S_2 = (Q_2, \Sigma_2, \delta_2)$ *be state machines. We define the cascade product of* S_1 *and* S_2 *with respect to* $\omega : Q_2 \times \Sigma_2 \longrightarrow \Sigma_1$ *by*

$$S_1 \omega S_2 = (Q_1 \times Q_2, \Sigma_2, \delta^\omega)$$

where

$$\delta^\omega((q_1, q_2), \sigma_2) = (\delta_1(q_1, \omega(q_2, \sigma_2)), \delta_2(q_2, \sigma_2))$$

for $\sigma_2 \in \Sigma_2, (q_1, q_2) \in Q_1 \times Q_2$ *[5, 6].*

Example 4. *Let* $S_1 = (Q_1, \Sigma_1, \delta_1)$ *be state machine where* $Q_1 = \{0, 1\}, \Sigma_1 = \{\sigma, \tau\}$ *and* $\delta_1 : Q_1 \times \Sigma_1 \longrightarrow Q_1$ *given by:*

δ_1	σ	τ
0	1	0
1	1	0

and $S_2 = (Q_2, \Sigma_2, \delta_2)$ *given by* $Q_2 = \{0, 1\}$, $\Sigma_2 = \{\sigma\}$ *and* $\delta_2 : Q_2 \times \Sigma_2 \longrightarrow Q_2$

δ_2	σ
0	1
1	0

. *Define a mapping* $\omega : Q_2 \times \Sigma_2 \longrightarrow \Sigma_1$ *by* $\omega(0, \sigma) = \sigma, \omega(1, \sigma) = \tau$.

The cascade product $S_1 \omega S_2 = (Q_1 \times Q_2, \Sigma_2, \delta^\omega)$ *where given by*

δ^ω	σ
$(0,0)$	$(1,1)$
$(0,1)$	$(0,0)$
$(1,0)$	$(1,1)$
$(1,1)$	$(0,0)$

Define $(\delta^\omega)_\sigma : Q_1 \times Q_2 \longrightarrow Q_1 \times Q_2$ *by* $(\delta^\omega)_\sigma ((q_1, q_2)) = (\delta_1 (q_1, \omega (q_2, \sigma)), \delta_2 (q_2, \sigma))$ *for each* $(q_1, q_2) \in Q_1 \times Q_2$*. We have*

$$\begin{aligned} (\delta^\omega)_\sigma = \phi &= \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,1) & (0,0) & (1,1) & (0,0) \end{pmatrix}, \\ (\delta^\omega)_{\sigma\sigma} = \phi^2 &= \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,0) & (1,1) & (0,0) & (1,1) \end{pmatrix}. \end{aligned}$$

The monoid of the state machine $S_1 \omega S_2$ *is given by*

$\cdot$	$[\epsilon]$	$[\sigma]$	$[\sigma^2]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$	$[\sigma^2]$
$[\sigma]$	$[\sigma]$	$[\sigma^2]$	$[\sigma]$
$[\sigma^2]$	$[\sigma^2]$	$[\sigma]$	$[\sigma^2]$

Definition 4. *Let* $S_1 = (Q_1, \Sigma_1, \delta_1)$ *and* $S_2 = (Q_2, \Sigma_2, \delta_2)$ *be state machines. We define the wreath product of* S_1 *and* S_2 *by* $S_1 W S_2 = \left(Q_1 \times Q_2, \Sigma_1^{Q_2} \times \Sigma_2, \delta^W\right)$ *where* $\delta^W ((q_1, q_2), (f, \sigma_2)) = (\delta_1 (q_1, f(q_2)), \delta_2 (q_2, \sigma_2))$ *for* $\sigma_2 \in \Sigma_2, f \in \Sigma_1^{Q_2}, (q_1, q_2) \in Q_1 \times Q_2$ *[5, 6, 7].*

Example 5. *Let* $S_1 = (Q_1, \Sigma_1, \delta_1)$ *be state machine where* $Q_1 = \{0,1\}, \Sigma_1 = \{\sigma, \tau\}$ *and* $\delta_1 : Q_1 \times \Sigma_1 \longrightarrow Q_1$ *given by:*

δ_1	σ	τ
0	1	0
1	1	0

and $S_2 = (Q_2, \Sigma_2, \delta_2)$ *given by* $Q_2 = \{0,1\}$, $\Sigma_2 = \{\sigma\}$ *and* $\delta_2 : Q_2 \times \Sigma_2 \longrightarrow Q_2$

δ_2	σ
0	1
1	0

. Denote the four elements of $\Sigma_1^{Q_2}$ *by* f_1, f_2, f_3, f_4 *where*

$$\begin{aligned} f_1(0) &= f_1(0) = \sigma. \\ f_2(0) &= \sigma, f_2(1) = \tau. \\ f_3(0) &= \tau, f_3(1) = \sigma. \\ f_4(0) &= f_4(1) = \tau. \end{aligned}$$

Then the state machine $S_1 W S_2$ *has the table*

δ^W	$(0,0)$	$(0,1)$	$(1,0)$	$(1,1)$
(f_1, σ)	$(1,1)$	$(1,0)$	$(1,1)$	$(1,0)$
(f_2, σ)	$(1,1)$	$(0,0)$	$(1,1)$	$(0,0)$
(f_2, σ)	$(0,1)$	$(1,0)$	$(0,1)$	$(1,0)$
(f_4, σ)	$(0,1)$	$(0,0)$	$(0,1)$	$(0,0)$

Define $\left(\delta^W\right)_{(f_1,\sigma)} : Q_1 \times Q_2 \longrightarrow Q_1 \times Q_2$ *by* $\left(\delta^W\right)_{(f_1,\sigma)}((q_1,q_2)) = (\delta_1(q_1, f_1(q_2)), \delta_2(q_2,\sigma))$ *for each* $(q_1,q_2) \in Q_1 \times Q_2$. *We have*

$$\begin{aligned}
\left(\delta^W\right)_{(f_1,\sigma)} &= \alpha = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,1) & (1,0) & (1,1) & (1,0) \end{pmatrix}, \\
\left(\delta^W\right)_{(f_2,\sigma)} &= \beta = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,1) & (0,0) & (1,1) & (0,0) \end{pmatrix}, \\
\left(\delta^W\right)_{(f_3,\sigma)} &= \gamma = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,1) & (1,0) & (0,1) & (1,0) \end{pmatrix} \\
\left(\delta^W\right)_{(f_4,\sigma)} &= \lambda = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,1) & (0,0) & (0,1) & (0,0) \end{pmatrix}.
\end{aligned}$$

5 Conclusion

In this paper, we give a specific transformation monoid, after that, we give the monoids of state machines associate with the direct product, the cascade product and wreath product of state machines.

References

[1] A. A. Ibrahim and M. S. Audu, "On Wreath Product Of Permutation Groups," *Proyecciones Journal of Mathematics*, vol. 26, no 1, pp. 73–90, May 2007.

[2] B. Baumslag and B. Chandler, *Theory and Problems of Group Theory*, New York University, 1968.

[3] J. Berstel and D. Perrin, *Theory of Codes*, Academic Press, 1984.

[4] N. Ghadbane and D. Mihoubi, "A construction of some group codes", *International Journal of Electronics and Information Engineering*, vol. 4, 2016.

[5] A. Ginzburg, *Algebraic Theory of Automata*, Academic Press, New York, 1968.

[6] W. M. L. Holcombe, *Algebraic Automata Theory*, Cambridge University Press, 1982.

[7] J. M. Howie, *Fundamentals of Semigroup Theory*, Oxford Science Publications, 1995.

[8] S. Majumdar, K. Kumar Dey and M. A. Hossain, "Direct product and wreath product of transformation semigroups", *Ganit Journal of Bangladesh Mathematics*, vol. 31, pp. 1–7, 2011.

[9] A. E. Nagy and C. L. Nehaniv, *Cascade Product of Permutation Groups*, Centre for Computer Science and Informatics, U. K and Centre for Research in Mathematics, Australia, 2013.

[10] H. J. Shyr, *Free Monoids and Languages*, Soochow University Taipei, Taiwan, R.O.C, 1979.

Biography

Nacer Ghadbane is a research scholar in the Laboratory of Pure and Applied Mathematics, Department of Mathematics, University of M'sila, Algeria.

Douadi Mihoubi is a professor in the Laboratory of Pure and Applied Mathematics, Department of Mathematics, University of M'sila, Algeria.

Applying MVC to Shopping Management Application Using Ruby on Rails with Bangla Interface

Mahbub Alam[1], Al Nahian[2], and Shamimul Islam[3]
(Corresponding author: Shamimul Islam)

Institute of Electronics, Bangladesh Atomic Energy Commission, Bangladesh[1]
Bangladesh Telecommunications Company Limited, Bangladesh[2]
Institute of Computer Science, Bangladesh Atomic Energy Commission, Bangladesh[3]
E-12/A, Paramanu Bhaban, Shahid Shahabuddin Shorok, Agargaon, Dhaka
(Email: Nacer.ghadbane@yahoo.com)
(Received Jan. 7, 2018; revised and accepted Mar. 3, 2018)

Abstract

This work presents a combination of the Model View-Controller pattern with ruby on rails framework for web service development. Ruby on rails supports Agile Programming Methodologies prioritize to minimize the risk by developing software in shorter time boxes called iterations. They emphasize real-time communication, and give preference for working Software which satisfies all stake holders. They give less prominence to documentation, as compared to other methods.

Keywords: Active Record; Agile Development Methodologies; Bengali Interface; Rails Framework; Rural Development

1 Introduction

Web services are considered as self-contained, self describing, modular applications that can be published, located and invoked across the world. Nowadays, an increasing amount of companies and organizations only implement their core business and outsource other application services by desktop and web based application. Thus, the ability to efficiently and effectively select and integrate inter-organizational and heterogeneous services on the web at runtime is an important step towards the development of the applications. In particular, if no single application service can satisfy the functionality required by the user, there should be a possibility to combine existing services together in order to fulfill the request. This paper offers an answer, to combine the MVC (Model View-Controller) pattern with typical application development environment [2, 7, 10, 11, 13, 15].

2 Previous Work

Research indicates that many traditional IS development methodologies are based on outmoded concepts. These methodologies are being utilized to develop web sites and, not surprisingly, they are limited since they were never intended to be used for this localization purpose.

Before moving on to put forward a methodology for web development, it is worth considering traditional IS methodologies and their applicability to this process. The next part of this paper summarizes some of these methodologies, providing a brief explanation of their strengths and weaknesses in relation to the Web development process.

Waterfall: The waterfall methodology involves a series of cascading steps that cover the development process with a small level of iteration between each stage. The major problem with using the waterfall methodology for the development of Web-sites (and also IS) is the rigidity of its structure and lack of iteration between any stage other than adjacent stages. As has been described, the Web is fast moving environment and new technologies are becoming available almost daily. Any methodology used for the development of Web-sites must be flexible enough to cope with change.

Structures Systems Analysis and Design Method (SSADM): This methodology was designed for the development of traditional, fairly large IS projects. It does not cover the whole lifecycle of a development project, but emphasizes the analysis and design stages, in the hope of reducing costly errors and omissions that might arise later. Some elements of SSADM could be 'tailored' into a Web development methodology; this could hopefully reduce the likelihood of expensive code changes given the emphasis on analysis and design.

Prototyping: Prototyping helps people assess a version of a working system, rather than assess a description of an imagined future system. It is most useful for the development of "interactive applications" where the user is more concerned with the screen layout than the actual processes of the system. It may appear at first glance that prototyping would be ideal for the development of a Web site. However, there are potential problems if prototyping was to be used for the development of Web-sites. Firstly, prototyping has a tendency towards project "creep", where the users add components on to the prototype that are outside the initial system requirements. This leads to the problem of never knowing when the project is finished, as the Web-site would be in a constant state of prototype. Secondly, the target audience of a Web-site is much more diverse than that of a traditional IS. Therefore the interface must be designed for a broad group of users, which gives rise to a problem when defining a set of users to evaluate the prototype.

Rapid Action Development (RAD): RAD is a form of prototyping that involves building several small "throwaway" prototypes for the system and then discarding them once they have been analyzed; this means the prototype is never developed into the finished system unlike conventional prototyping methods. Powell discusses how IS development methodolo- gies are failing the Web development process and cites the RAD methodology as being the worse culprit: "RAD involves building the wrong site multiple times until the right site falls out of the process."

Incremental Prototyping: Incremental prototyping allows large systems to be developed in phases, which avoids delays between specification and delivery. The most important features of the system are developed to completion first, and then less important features are added to the system later. This speeds up the implementation time of the project. The incremental approach to development would be useful in the dynamic world of the Web. Web sites grow both in size and functionality, so a methodology that utilized this incremental approach could not only speed up the development process and help build in new features as the enabling technologies emerge, but also help manage the problem of Web-site sprawl.

3 Methodology

Rails is a framework using Ruby language. It is designed to make programming Web applications easier, a feature every developer needs to get started. Ruby on Rails is a framework that gives support to develop, deploy, and maintain Web 2.0 applications. Rails applications gives use the Model-View Controller (MVC) architecture. Rails applications are written in Ruby which is an object-oriented scripting language. Rails is based on three philosophies: code is short and readable, DRY and convention over configuration. Using DRY (Don't Repeat Yourself) philosophy: Every piece of knowledge in a system should be expressed in just one place.

Convention Over Configuration - means that Rails makes assumptions about what we want to do and how we are going to do it, rather than requiring us to specify every little thing through endless configuration files. Rails supports Active Record in the object relational mapping (ORM) layer. Applying Active Record, it is possible to manage table relationships and in the process create, read, update, and delete operations (commonly referred to in the industry as CRUD methods) are covered [9]. Action Pack acts as core of Rails applications and it contains three Ruby modules: Action Dispatch, Action Controller, and Action View.

Action Dispatch routes requests to controllers and Action Controller converts requests into responses. The Action Controller makes use of Action View for the purpose of formatting the responses. Rails decodes information in the request URL and uses a scheme called Action Dispatch to determine what should be done with that request. At the end Rails determines the name of the controller that handles the particular request, along with a list of any other request parameters. In this process either of these parameters or the HTTP method itself is used to identify the action that has to be invoked in the target controller. Rails supports features such as Ajax and Restful interfaces that help the programmers in code integration. Figure 1 shows the MVC architecture.

Agile approach in Rails: Agile Manifesto as a set of three preferences as given below:

1) Individuals and interactions over processes and tools: Rails is all about individuals and interactions. Rails contains simple tool sets, no complex configurations, and no elaborate processes. There are small groups of developers, their favorite editors, and chunks of Ruby code which leads to transparency and what the developers do is reflected immediately in what the customer sees. It is basically interactive process.

2) Working software over comprehensive documentation: Rails does not support documentation and specifications. Rails delivers working software early in the development cycle.

3) Customer collaboration over contract negotiation: Rails project can quickly respond to changes required by the customers, who are thereby convinced that the team can deliver what is required, not just what has been requested.

Designing software to connect an object-oriented business system with a relational database is a tedious task. Object orientation and the relational paradigm differ quite a bit. An application that maps between the two paradigms needs to be designed with respect to performance, maintainability and cost to name just a few requirements. There are numerous patterns of object/relational access layers, but looking at the body of pattern literature we will find that some patterns are still to be mined, while there is no generative "one stop" pattern language for the problem domain. Object-relational mapping (ORM) is a programming technique for converting data between incompatible type systems in object-oriented programming languages.

ORM libraries map database tables to classes. In Rails if a class is defined as Order then the database has a table called orders. Rows in orders table are mapped to objects of the class-a specific order is represented as an object of class Order. The individual columns are set making use of the

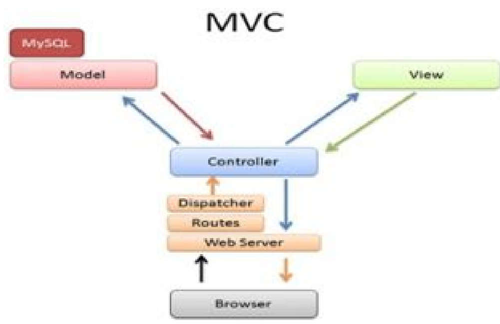


Figure 1: The MVC architecture

object attributes. ORM layer represents tables as classes, rows as objects, and columns as attributes of those objects. Class methods are used to perform table level operations, and methods can perform operations on the individual rows of a table. Active Record is the simplest of database design. Active Record has been released with the Rails framework to the public and is also available as a part of the core bundle with its own Ruby Gem. Active Record differs from other ORM Libraries because it makes a number of configuration assumptions, without requiring any outside configuration files or mapping details. With Rails, the model section is generally Active Record classes and other data-descriptive or data communication code. The view section is mainly for the user interface which involves elaborate HTML code in Rails applications. Figure 2 shows the architecture of active record.

Figure 2: Architecture of active record

We implement the existing system into ORM using Active Record. The knowledge of how to interact with the database is directly embedded into the class performing action, by making use of Active

Record. It is the ORM layer supplied with Rails. We will build on the mapping data rows and columns using Active Record to manage table relationships and in the implementation of CRUD Methods. Instances of Active Record classes correspond to rows in database table. The Instances have attributes corresponding to the columns in the table. Active Record determines the columns of the corresponding class dynamically. The Active Record object attributes in general correspond to the data in the row of the corresponding table. Figure 3 shows one of the sample screens of the project. Active Record is database-neutral, so it does not care which database software we use, and it supports nearly every database. Since it is a high-level abstraction, the code we write remains the same no matter which database we are using. For the present project we use PostgreSQL. Figure 4 shows the sample screen for a new product entry.

4 Results

The core architecture of the web service is a modified MVC pattern that is connected to the Requirement Analysis - through the user feedback, which should be an independent database system.

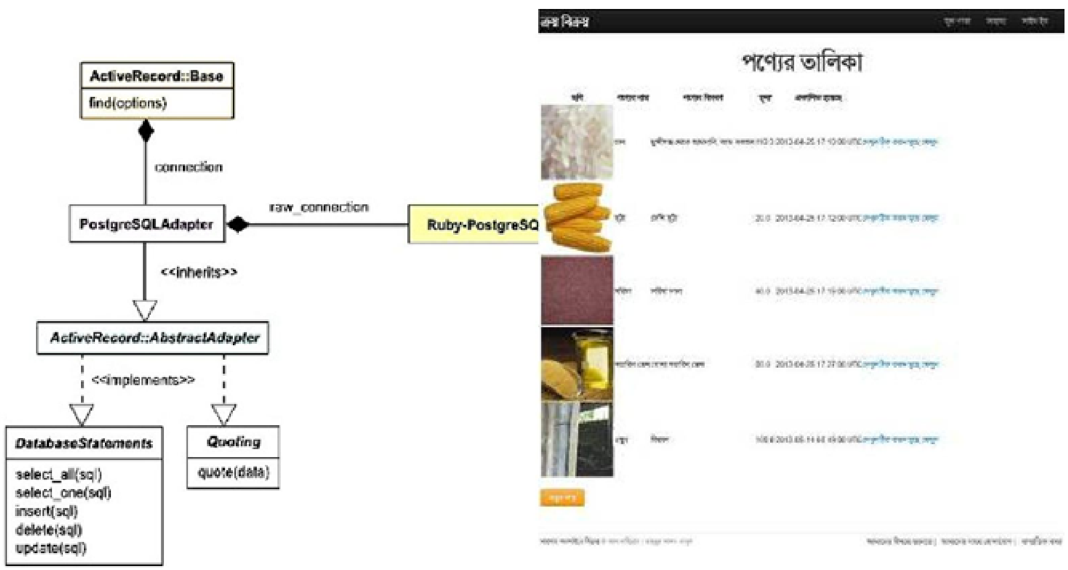


Figure 3: Sample homepage for Admin

Connection between web services' core and Requirement Analysis occurs in the following way:

1) Admin identifies a new product;

2) Admin access the view of relevant module, for instance, by pressing a specific button available in the user interface;

3) View queries Controller state;

4) Controller notifies View of change to state;

5) View displays the state to Admin;

6) Admin reports through View;

7) View transmits Admin's report to Controller;

8) Controller accesses feedback database system to report;

9) Refinements and update requirements are defined using feedback, technological opportunities and evolving threats considerations;

10) A decision to start a new action may be taken.

Figure 4: Sample screen for a new product entry

It is worth noticing that a new action may imply in a change in the system's architecture or a new block release.

5 Conclusions

Methodologies, whether used for traditional systems development or web development, have their uses and also their limitations. On the positive side they provide a useful crux for the novice developer [14], they act "as a comfort factor to reassure participants that 'proper' practices are being followed" [5] and the project management facility provides an audit trail, that helps ensure management viability of the development progress [12]. On the more negative side, they are often far too prescriptive and can actually constrain the developer while attempting to successfully complete a project in what is often a highly stressful and complex environment [4].

Not surprisingly, research shows that these methodologies are rarely applied as intended [3, 6] as developers creatively tailor them in order to meet the needs of the particular organizational context. Thus, it would be naive to assume that the methodology introduced here could provide some kind of universal panacea. Indeed, it is merely intended to act as a useful framework to aid the web development process. As each web site will have different goals and objectives and a unique set of problems, thus any methodology will require adaptation to the contingencies of each situation. It is hoped that the one proposed here will serve as a useful tool to complement the skills and creativity of the developer and ease the process of web development. Empirical research is currently taking place to explore its strengths and weaknesses in a real-world situation.

References

[1] Z. Cai, Z. He, X. Guan, Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks", *IEEE Transactions on Dependable and Secure Computing*, DOI: 10.1109/TDSC.2016.2613521, 2016.

[2] P. Chandrasekar, T. Sangeetha, "Smart shopping cart with automatic billing system through RFID and ZigBee", in *Proceedings of IEEE International Conference on Information Communication Embedded Systems (ICICES'14)*, pp. 1-4, 2014.

[3] Z. Fang, *et al.*, "Localization in wireless sensor networks with known coordinate database", *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, no. 1, pp. 1-17, 2010.

[4] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, "Internet of Things (IoT): A vision architectural elements and future directions", *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645-1660, 2013.

[5] C. Hu, *et al.*, "Efficient privacy-preserving schemes for dot-product computation in mobile computing", in *Proceedings of 2nd ACM Workshop Privacy Aware Mobile Computing*, pp. 51-59, 2016.

[6] R. Khan, S. U. Khan, R. Zaheer, S. Khan, "Future Internet: The Internet of Things architecture possible applications and key challenges", in *Proceedings of IEEE 10th International Conference on Frontiers of Information Technology (FIT'12)*, pp. 257-260, 2012.

[7] R. Kumar, K. Gopalakrishna, K. Ramesha, "Intelligent shopping cart", *International Journal of Engineering Science Innovation Technology*, vol. 2, no. 4, pp. 499-507, 2013.

[8] T. Lucas-Nülle, *Product Information Manage-ment in Deutschland*, Marketstudie, 2005.

[9] S. Ruby., *et al. Agile Web Development with Rails*, 4th Edition, The Pragmatic Bookshelf, 2016.

[10] M. R. Sawant, K. Krishnan, S. Bhokre, P. Bhosale, "The RFID based smart shopping cart", *International Journal of Engineering Research and General Science*, vol. 3, no. 2, pp. 275-280, 2015.

[11] T. Shanmugapriyan, "Smart cart to recognize objects based on user intention", *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 5, pp. 2049-2053, 2013.

[12] S. Shekar, P. Nair and A. Helel, "iGrocer - A Ubiquitous and Pervasive Smart Grocery Shopping System," in *Proceedings of the ACM Symposium on Applied computing*, pp. 645-652, 2003.

[13] T. Song, R. Li, X. Xing, J. Yu, X. Cheng, "A privacy preserving communication protocol for IoT applications in smart homes", in *Proceedings of International Conference on Identification Information Knowledge Internet Things (IIKI'16)*, 2016.

[14] Y. Wang, *et al.*, "An incentive mechanism with privacy protection in mobile crowdsourcing systems", *Computer Networks*, vol. 102, pp. 157-171, June 2016.

[15] A. Yewatkar, F. Inamdar, R. Singh, A. Bandal, "Smart cart with automatic billing product information product recommendation using RFID & ZigBee with anti-theft", in *Procedia Computing Science*, vol. 79, pp. 793-800, Jan. 2016.

Biography

Mahbub Alam was born in a rural area called Dhamrai of Dhaka in 1991. He has completed his B.Sc & M.Sc from the Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh. He worked as a lecturer in Gono University at the Dept. of CSE from January 2015 to February 2016. He worked as an Assintant Engineer in WALTON Group at the Computer R&D section from February 2016 to November 2017. Currently he is working as a Scientific Officer at the Institute of Electronics in Bangladesh Atomic Energy Commission. His research interests include big data analysis, artificial intelligence, pattern recognition and expert system, computer vision, system that can provide distinct service through internet protocol and any system that can be beneficiary for common people. He is also interested in entrepreneurship.

Al Nahian was born at Dhaka, Bangladesh in 1990. He has completed his B.Sc & M.Sc from the Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh. He worked as a lecturer in Bangladesh University at the Dept. of CSE from June 2015 to June 2017. He also worked as Software Developer for several Software Companies. Currently he is working as an Assistant Manager at Bangladesh Telecommunications Company Limited - BTCL. His research interests include data science, artificial intelligence, web services, e-commerce, software engineering etc.

Shamimul Islam was born in Satkhira, Bangladesh, on 28th October, 1988. He received the B.Sc. and M.Sc Degree in Computer Science and Engineering from the Jahangiranagar University, Bangladesh in 2011, and in 2012 respectively. He worked at Samsung R&D Institute Ltd., Bangladesh from 1 Oct, 2012 to 31 Dec,2014 as a Software Engineer. From 27th july,2016 he is being a Scientific Officer of Computer Science Division in Bangladesh Atomic Energy Commission(BAEC). He is a member of Bangladesh Computer Society. His research interest includes Network and Cyber Security, Communication Engineering, IoT, Software Engineering and Human Computer Interaction.

Cloud Security Using Markov Chain and Genetic Algorithm

Mazin H. R. Al-Shaikhly, Hazem M El-Bakry, and Ahmed A. Saleh
(Corresponding author: Mazin H. R. Al-Shaikhly)

Department of Information Systems, Faculty of Computer and Information Sciences, Mansoura University
Mansoura, Egypt
(Email: firasjawad118@yahoo.com)

(Received Oct. 30, 2017; revised and accepted Feb. 1, 2018)

Abstract

The fundamental change happening in the field of Information Technology this change represent with cloud computing, cloud providing virtual resources via internet but great challenges in the field of data security and privacy protection. Currently main problem with cloud computing is how to improve security "cloud is critical security" In this paper attempt to solve cloud security by using intelligent system with genetic algorithm and add user attack probability with Markov algorithm all as wall to provide cloud data secure, all service that provide by cloud must detect whom receive and register it to create list of user (trusted or un-trusted depend on behavior) with system implementation have good result.

Keywords: Cloud Services; Genetic Algorithms; Markov Algorithm

1 Introduction

Cloud is one of the main facility that provide multi virtual resource for difference user at same time while traffic may attack occurring the main objective of paper to implement data security for cloud starting from "cloud is critical security" [1, 3].

The main objective in implementation is security, Section 2 cloud computing type and service, Section 3 cloud -Security categories. Section 4 Markov and genetic algorithm. Section 5 implementation and experimental result and finally Section 6 conclusion.

2 Cloud Computing Service, Deployment Type and Risk

Cloud computing means storing and accessing data or applications over the Internet.

2.1 Cloud Service

Based on a service that the cloud is offering [2–8]:

- IaaS (Infrastructure-as-a-Service): gives virtual machines and other disconnected equipment and working frameworks which might be controlled through a benefit Application Programming Interface (API).

- PaaS (Platform-as-a-Service): enables clients to grow new applications utilizing APIs, actualized and worked remotely. The stages ordered in cloud advancement devices, configuration administration and organization stages.
- SaaS (Software-as-a-Service): is programming ordered by an outsider supplier, accessible on request, more often than not through a Web program, working in a remote way. Illustrations incorporate online word preparing and spreadsheet tools, CRM administrations and Web content conveyance administrations.
- Or, Storage, Database, Information, Process, Application, Integration, Security, Management, Testing-as-a-service [9].

2.2 Cloud Type

Cloud Deployment Models in figure 1.The cloud services can be implemented in four deployment models [7–11]:

- Public Cloud: The infrastructure of cloud is influenced accessible to the overall population or vast industry to gathering and is claimed by an association offering cloud administrations.
- Community Cloud: The infrastructure of cloud is shared by a few associations and backings an uncommon group. It might be overseen by the associations or an outsider, and may exist on-premises or off-premises.
- Private Cloud: The infrastructure of cloud is worked totally for a solitary organization. It might be overseen by the association or an outsider, and may exist on-premises or off-premises.
- Hybrid Cloud: The infrastructure of cloud is an arrangement of at least two mists (private, group or open) that are bound together by institutionalized or restrictive innovation that empowers transportability of information and application.

2.3 Cloud Security Categories

There are two number of security issues related with distributed computing and these issues classifications: Security issues looked by cloud suppliers (associations giving Software, Platform) and security issues looked by their clients. The supplier must guarantee that their framework is secure and that their customers' information and applications are secured while the client must guarantee that the supplier has taken the best possible safety efforts to ensure their data [1].

2.3.1 Security Dimensions (Security Domains)

A space, with regards to systems administration, alludes to any gathering of clients, workstations, gadgets, printers, PCs and database servers that offer diverse sorts of information by means of system assets. There are additionally many sorts of sub-domains. A area has a space controller that administers all fundamental space works and oversees organize security. In this way, a space is utilized to deal with all client capacities, including username/secret key and shared framework asset validation and access. An area is likewise used to dole out particular asset benefits, for example, client accounts [4, 7].

2.3.2 Security Risks

A large number of the distributed computing related dangers are not new and can be found in the figuring situations. There are many organizations and associations that outsource huge parts of their business because of the globalization. It implies not just utilizing the administrations and innovation of the cloud supplier, yet many inquiries managing the way the supplier pursues his security policy playing out an investigation the best dangers to distributed computing can be abridged as takes after [5]:

1) Abuse and Un allowed Use of Cloud Computing;
2) Insecure Application Programming Interfaces;
3) Malicious Insiders;
4) Shared Technology Vulnerabilities;
5) Data Loss and Leakage;
6) Account, Service and Traffic Hijacking;
7) Unknown Risk Profile.

2.3.3 Security Threats

In this section the major threats for cloud computing are explored [6, 11]. These are:

1) Data threats including data breaches and data loss.
2) Network threats including account or service hijacking, and denial of service.
3) Cloud environment specific threats including insecure interfaces and APIs, malicious insiders, abuse of cloud services, insufficient due diligence, and shared technology vulnerabilities.

3 Markov Algorithm

A Markov algorithm in computer science is a string rewriting system that uses grammar-like rules to operate on strings of symbols. Markov algorithms have been shown to be Turing-complete, which means that they are suitable as a general model of computation and can represent any mathematical expression from its simple notation [8, 9].

The *Rules* are a sequence of pair of strings, usually presented in the form of $pattern \Rightarrow replacement$. Each rule may be either ordinary or terminating. Given an input string:

1) Check the Rules in order from top to bottom to see whether any of the patterns can be found in the input string.
2) If none is found, the algorithm stops.
3) If one (or more) is found, use the first of them to replace the leftmost occurrence of matched text in the input string with its replacement.
4) If the rule just applied was a terminating one, the algorithm stops.
5) Go to Step 1.

Note that after each rule application the search starts over from the first rule.

4 Security System With Markov and Genetic

With this section attempted to described main algorithm that using to implement system function.

4.1 Coding

In this section attempted to give view on basic coding for variable as show below Service coding as

Table 1: Typical states of SEIR model

Bit From 33-34	Code
IAAS	01
PaAS	10
SAAS	11

Bits from 35-41 depend on status and visit control;
Bit 42 Accept 0 Reject 1.

4.2 Representation Of Request For Markov And Genetic As String

In this part represent IP address as string with sequence of binary cod "0", "1", binary string using with genetic and Markov as same string set for same source(request service) element in all request IP address.

Representation for:

1) IP with 32 bits;
2) Service 2 bits (33-34);
3) Operate code 7 bit as three high 3 (bits 35-37) from left to right bit genetic and (4) (bits 36-41) last bit for Markov all bits (35-41);
4) Flag code 1 bit is (42).

Remark 1. *String size 42 bits for all request and test:*
Requests convert IP and operate code as record of 42 bits to start algorithm (See Algorithms 1 – 4).

Algorithm 1 Main process engine

```
Start
while online for all requests do analyzing new request do
   Pre-search with un-trusted IP list
   if un-trusted then
      IP block
   end if
   Operate Markov - genetic algorithm for request IP
   Create new visitor IP list
   Re-Comparing with un-trusted list and pointed as block if repeated
   Sort and storing
end while
End
```

Algorithm 2 Pre-search with un-trusted IP list

```
Start
CIP = current IP 2
for IP_List = start IP to last IP in list do
   Depend on service type bit number 33, 34
   un-trusted IP deal with bits (35–37) and (38–41)
   if CIP with un-trusted IP then
      Call re-analysis request(CIP, T) with all branch
   else
      Call analysisIP(CIP, T)
   end if
end for
if flag of CIP(T) = true then
   pass one
end if
End
```

Algorithm 3 Markov - genetic algorithm for request IP

```
Start
Check Markov list-1, list-2, ···, list-n
if pass then
   check with genetic list-1, list-2, ···, list-n
   if pass then
      trusted IP
      give service with register service type and time ((log list for monitoring)
   else
      block
   end if
end if
End
```

Algorithm 4 IP visitor algorithm

```
Start
Create locale IP list depend on service control ( time and through put)
Grouping with unique IP list
Save IP though put behavior
End
```

Remark 2. *Using genetic to avoid intrusion to network from any out-said attack as is shown in Algorithm 5.*

Remark 3. *Using Markov to avoid intrusion to network from any out-said attack as is shown in Algorithm 6.*

Algorithm 5 Fitness for genetic

```
Start
rem using IP length =32bit (binary convert)= chromosome-42bit
rem fitness 0-7 refers to max value (block IP)
Loop
Check I/O port per time
if IP[current] throughput > max_value then
   Fitness[current]++
end if
Until I/O port off
End
```

Algorithm 6 Markov

```
Start
rem using IP length =32bit (binary convert)=chromosome-42bit
rem trusted 4 bit refers to max value (block IP)
Loop
Check I/O port per time
if IP[current] throughput near to critical value of system then
   weight[current]++
end if
Until I/O port off
End
```

5 Experimental Results

Running of simulated system with genetic feature nad Markove (in traffic mode) saves the following value from system with fitness, following chart gives sample fragment view, where X axis = 1 .. 145 sequence of IP; $Y1$ axis $s2$ = 1 .. 7 fitness value (threat degree); $Y2$ axis $s1$ if fitness value [current IP] = 7 "(7-1)" (max threat) and = 15 "(15-1)" then block IP[current], where current = current I/O value (See Figure 1).

Tables 2 – 4 result detail (IP address, 32 bits value for IP with Markov value test and fitness).

6 Conclusions

In this section, we summary the following four items:

1) Results obtained give good indication (test with simulated environment of real world).

2) While traffic increase, attaching attack easier.

3) Any change in fitness gives a new feature to system (the current fitness may be better).

4) High security performance with data when change of Markov weight.

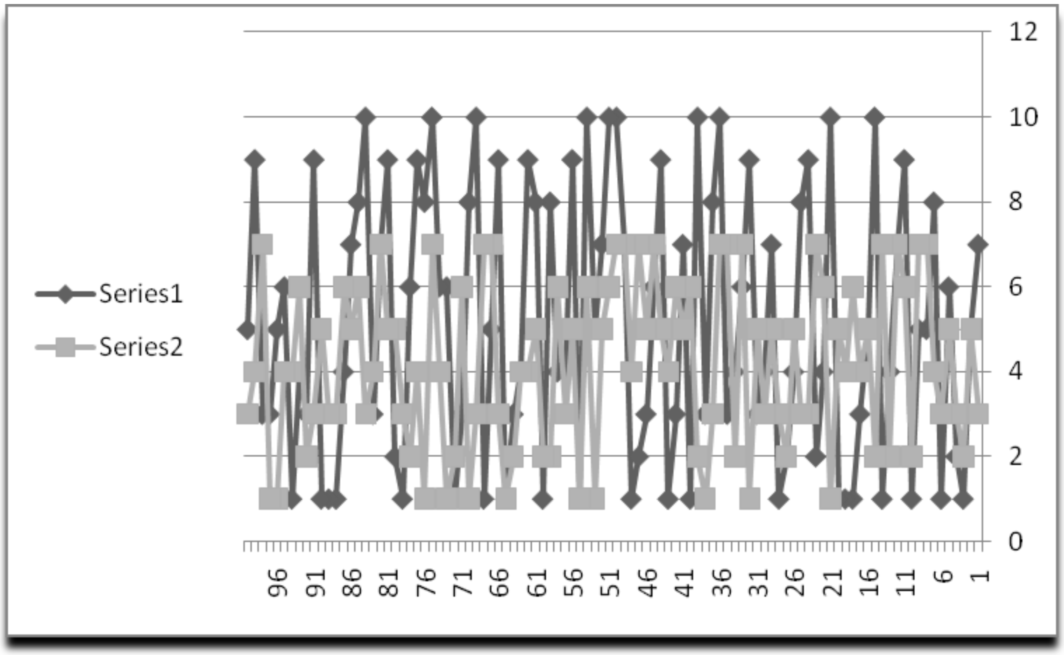


Figure 1: Chart fragment view for result of table 1 where s1=from Markov s2 from genetic

References

[1] M. Irfan, M. Usman, Y. Zhuang, and S. Fong, "A critical review of security threats in cloud computing," in *3rd International Symposium on Computational and Business Intelligence (ISCBI'15)*, pp. 105-111, 2016.

[2] M. H. R. Al-Shaikhly, H. M. El-Bakry, A. A. Saleh, "Intelligent firewall using genetic algorithm," *International Journal of Advanced Research in Computer Science & Technology*, vol. 5, no. 2, pp. 22–27, Apr. 2017.

[3] M. H. R. Al-Shaikhly, H. M. El bakry, A. A. Saleh, "An intelligent approach for improving stability and security of cloud computing," *International Journal of Artificial Intelligence and Mechatronics*, vol. 2, no. 3, pp. 51–56, Nov. 2017.

[4] S. E. El-Gazzar, A. A. Saleh, H. M. El-Bakry, "Overview of using private cloud model with GIS," *International Journal of Electronics and Information Engineering*, vol. 7, no. 2, pp. 68–78, Dec. 2017.

[5] N. S. Abu El-Ala, H. M. El-bakry, S. A. Abd El-Hafeez, "Personal cloud-based learning environment," *International Journal of Computer Science and Engineering*, vol. 8, no. 4, pp. 122–127, Apr. 2016.

[6] R. I. Louis, A. I. Saleh, M. F. Al Rahmawy, H. M. El-Bakry, S. M. Abd El-razek, "A best effort heuristic algorithm for scheduling timely constrained tasks in the cloud," *International Journal of Scientific & Engineering Research*, vol. 7, no. 2, pp. 390–400, Feb. 2016.

[7] R. Izzat, B. T. Shabana A. M. Riad and H. M. El Bakry, "Spatial query performance for GIS cloud," *International Journal of Information Engineering and Applications*, vol. 5, no. 8, pp. 56–65, Aug. 2015.

Table 2: Typical states of SEIR model

IP Addres	Binary Format	S1	S2
186:101:83:167	10111010011001010101001110100111	7	3
186:142:116:120	10111010100011100111010001111000	5	5
182:158:52:112	10110110100111100011010001110000	1	2
173:54:164:123	10101101001101101010010001111011	2	3
187:113:231:120	10111011011100011110011101111000	6	5
195:112:154:216	11000011011100001001101011011000	1	3
185:136:154:72	10111001100010001001101001001000	8	4
177:119:129:93	10110001011101111000000101011101	5	7
173:36:60:207	10101101001001000011110011001111	5	7
183:133:166:233	10110111100001011010011011101001	1	2
196:94:176:152	11000100010111101011000010011000	9	6
196:84:214:237	11000100010101001101011011101101	7	7
194:49:208:161	11000010001100011101000010100001	4	2
185:43:178:51	10111001001010111011001000110011	1	7
181:162:51:233	10110101101000100011001111101001	10	2
190:49:153:127	10111110001100011001100101111111	5	5
195:62:157:115	11000011001111101001110101110011	3	4
195:103:137:68	11000011011001111000100101000100	1	6
192:119:73:113	11000000011101110100100101110001	1	4
190:55:92:125	10111110001101110101110001111101	1	5
178:115:160:175	10110010011100111010000010101111	10	1
179:54:203:138	10110011001101101100101110001010	4	6
184:117:202:194	10111000011101011100101011000010	2	7
176:60:177:215	10110000001111001011000111010111	9	3
173:127:55:129	10101101011111110011011110000001	8	3
189:114:215:155	10111101011100101101011110011011	4	5
195:35:224:74	11000011001000111110000001001010	2	2
196:176:63:159	11000100101100000011111110011111	1	3
174:140:204:83	10101110100011001100110001010011	7	5
192:135:174:173	11000000100001111010111010101101	4	3
187:143:47:155	10111011100011110010111110011011	3	5
189:141:148:49	10111101100011011001010000110001	9	1
176:153:153:237	10110000100110011001100111101101	6	7
195:45:186:190	11000011001011011011101010111110	4	2
185:136:222:135	10111001100010001101111010000111	3	7
195:71:134:93	11000011010001111000011001011101	10	7
183:91:201:143	10110111010110111100100110001111	8	3
191:180:98:107	10111111101101000110001001101011	3	1
196:93:44:162	11000100010111010010110010100010	10	2
189:36:72:237	10111101001001000100100011101101	1	6
185:104:150:97	10111001011010001001011001100001	7	5
194:90:57:124	11000010010110100011100101111100	3	6

Table 3: Typical states of SEIR model (Cont.)

IP Addres	Binary Format	S1	S2
181:161:119:42	10110101101000010111011100101010	1	4
182:152:158:127	10110110100110001001111001111111	9	5
186:116:99:51	10111010011101000110001100110011	6	7
178:111:69:141	10110010011011110100010110001101	3	5
193:119:141:123	11000001011101111000110101111011	2	7
175:117:43:71	10101111011101010010101101000111	1	4
186:160:89:99	10111010101000000101100101100011	7	7
186:52:122:199	10111010001101000111101011000111	10	7
183:68:205:44	10110111010001001100110100101100	10	6
174:153:170:203	10101110100110011010101011001011	7	5
188:173:37:53	10111100101011010010010100110101	5	1
189:112:81:216	10111101011100000101000111011000	10	6
194:176:80:125	11000010101100000101000001111101	1	1
193:119:55:146	11000001011101110011011110010010	9	5
190:36:186:207	10111110001001001011101011001111	4	3
173:172:93:86	10101101101011000101110101010110	4	6
186:93:102:152	10111010010111010110011010011000	8	2
176:172:58:154	10110000101011000011101010011010	1	2
173:136:52:78	10101101100010000011010001001110	8	5
191:170:222:220	10111111101010101101111011011100	9	4
187:139:94:149	10111011100010110101111010010101	4	4
195:177:196:97	11000011101100011100010001100001	3	2
175:149:200:84	10101111100101011100100001010100	1	1
175:41:181:237	10101111001010011011010111101101	9	3
172:57:117:230	10101100001110010111010111100110	5	7
182:142:82:188	10110110100011100101001010111100	1	7
177:162:96:238	10110001101000100110000011101110	10	3
178:101:193:55	10110010011001011100000100110111	8	1
188:45:233:151	10111100001011011110100110010111	3	6
180:138:208:173	10110100100010101101000010101101	1	2
177:56:86:233	10110001001110000101011011101001	6	1
186:42:46:113	10111010001010100010111001110001	6	4
173:113:55:179	10101101011100010011011110110011	10	7
174:104:94:63	10101110011010000101111000111111	8	1
178:69:35:40	10110010010001010010001100101000	9	4
196:103:205:130	11000100011001111100110110000010	6	2
184:71:77:59	10111000010001110100110100111011	1	3
193:181:76:104	11000001101101010100110001101000	2	5
192:154:80:91	11000000100110100101000001011011	9	5
185:46:114:44	10111001001011100111001000101100	7	7
189:101:49:236	10111101011001010011000111101100	3	4
173:147:161:194	10101101100100111010000111000010	10	3
187:167:229:45	10111011101001111110010100101101	8	6
195:106:72:61	11000011011010100100100000111101	7	5

Table 4: Typical states of SEIR model (Cont.)

IP Addres	Binary Format	S1	S2
174:169:72:116	10101110101010010100100001110100	4	6
192:112:47:138	11000000011100000010111110001010	1	3
188:70:231:52	10111100010001101110011100110100	1	3
179:56:138:132	10110011001110001000101010000100	1	5
192:167:201:50	11000000101001111100100100110010	9	3
193:158:39:82	11000001100111100010011101010010	3	2
197:83:180:59	11000101010100111011010000111011	4	6
182:117:150:149	10110110011101011001011010010101	1	4
177:85:223:78	10110001010101011101111101001110	6	4
186:123:124:226	10111010011110110111110011100010	5	1
192:95:129:213	11000000010111111000000111010101	3	1
177:96:201:58	10110001011000001100100100111010	3	7
174:112:137:50	10101110011100001000100100110010	9	4
191:157:102:208	10111111100111010110011011010000	5	3

[8] S. Ziad, H. M. El Bakry, and I. M. Abdelhady, "A proposed framework for ranking and reservation of cloud services based on quality of service," *International Journal of Advanced Research in Computer Science & Technology*, vol. 3, no. 2, pp. 195–199, Apr. 2015.

[9] A. Mosa, H. M. El Bakry, and M. Abuelkhir, "Cloud computing in e-government: A survey," *International Journal of Advanced Research in Computer Science & Technology*, vol. 3, no. 2, pp. 132–139, Apr. 2015.

[10] M. Khaleel, H. M. El Bakry, and A. A. Saleh, "Developing e-learning services based on cache strategy and cloud computing," *International Journal of Information Science and Intelligent System*, vol. 3, no. 4, pp. 45–52, Oct. 2014.

[11] N. S. Abu El-Ala, W. A. Awad and H. M. El-Bakry, " Cloud computing for solving e-learning problems," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 12, pp. 135–137, Dec. 2012.

Biography

Mazin H. R. Al-Shaikhly graduated from Computer Science Dept. Baghdad University, Baghdad, IRAQ in 1998. He received his master degree in Computer Science from the Technical university, Baghdad, IRAQ in 2005. His main research interests are in the areas of Cloud Computing, Intelligent algorithms, and Information Systems.

Hazem M. El-Bakry (Mansoura, EGYPT 20-9-1970) received B.Sc. degree in Electronics Engineering, and M.Sc. in Electrical Communication Engineering from the Faculty of Engineering, Mansoura University - Egypt, in 1992 and 1995 respectively. Dr. El-Bakry received Ph. D degree from University of Aizu - Japan in 2007. Currently, he is a full professor at the Faculty of Computer Science and Information Systems - Mansoura University - Egypt. His research interests include neural networks, pattern recognition, image processing, biometrics, cooperative intelligent systems and electronic circuits. In these areas, he has published many papers in major international journals and refereed international conferences. According to academic measurements, now the total number of citations for his publications is 3148. The H-index of his publications is 28. Dr. El-Bakry has the United States Patent No. 20060098887, 2006. Furthermore, he is associate editor for journal of computer science and network security (IJCSNS) and journal of convergence in information technology (JCIT). In addition, is a referee for IEEE Transactions on Signal Processing, Journal of Applied Soft Computing, the International Journal of Machine Graphics & Vision, the International Journal of Computer Science and Network Security, Enformatika Journals, WSEAS Journals and many different international conferences organized by IEEE. Moreover, he has been awarded the Japanese Computer & Communication prize in April 2006 and the best paper prize in two conferences cited by ACM. He has also been awarded Mansoura university prize for science publication in 2010 and 2011. Dr. El-Bakry has been selected in who Asia 2006 and BIC 100 educators in Africa 2008. (Email: helbakry20@yahoo.com)

Ahmed A. Saleh is full professor at the Faculty of Computer Science and Information Systems - Mansoura University - Egypt. He is the vice dean for higher studies, research, and cultural affairs. His research interests include E-Business, GIS, and Information Systems. (Email: elfetouh@mans.edu.eg)

An Improved Cuckoo Algorithm Based on Adaptive Simulated Annealing Method Used for Traveling Salesman Problem

Jie Liu, Lin Teng, and Shoulin Yin
(Corresponding author: Lin Teng)

Software College, Shenyang Normal University
No. 253, HuangHe Bei Street, HuangGu District, Shenyang, China
(Email: 910675024@qq.com)

(Received Feb. 26, 2017; revised and accepted Apr. 9, 2018)

Abstract

Traditional Cuckoo algorithm has some disadvantages, such as inaccuracy results, low convergence speed and easily falling into local solution. Cuckoo search is a new intelligence meta-heuristic search algorithm which is based on the obligate brood parasitic behavior of some Cuckoo species in combination with the *Lévy* flight behavior of some birds and fruit flies. Therefore, an improved Cuckoo algorithm based on adaptive simulated annealing method is proposed in this paper. When it uses Cuckoo algorithm to search the optimal results. If the results fall into local optimal, then it utilizes adaptive simulated annealing method, has the strong ability of global search, to randomly select parts of bird's nests and conduct adaptive simulated annealing search. Finally, it can obtain a worse solution with a certain probability, which is conducive to jumping out of local solution for Cuckoo algorithm. By using this improved method, it can enhance the search ability of Cuckoo algorithm and accelerate the convergence time. At last, to inspect the performance of the proposed optimization method, experiments on the solution convergence, average execution time, and percentage deviations of both the best and average solutions to the best known solution are conducted. Similarly, in order to obtain unbiased and comprehensive comparisons, descriptive statistics such as mean, standard deviation, minimum, maximum and range were used to describe each of the algorithms. Empirical analysis results show that the quality of the final results as well as the convergence rate of the new algorithm in some cases produced even more superior solutions than the best known travelling salesman problem benchmarked results and it has the higher effectiveness than other optimization algorithms.

Keywords: Cuckoo Algorithm; Adaptive Simulated Annealing Method; Global Search; Convergence Time; Travelling Salesman Problem

1 Introduction

Currently, many problems in engineering technology fields need to be transformed into optimization problems, because it needs to search the global optimal value. Optimization problems can be divided into single objective and multi objective, which are generally difficult to solve. Many existed optimization algorithms are used for solving these problems, nevertheless, they are not the optimal choice in an appropriate convergence time. Due to the increasing of problem size, calculating amount is extremely large. Maybe travelling salesman problem (TSP) is the most famous example. TSP [5] is a typical optimization problem in the area of operational research. It belongs to Non-Deterministic Polynomial

(NP) problem and is used in many areas such as PCB drilling, goods delivery routes, control of robots and workshop scheduling. Although there are some precise algorithms which can be used to solve the problem, the principle of precise algorithms is complex. For small size TSP [22], Branch and bound method, greedy method and cutting plane method are often used to solve these problems. But for big size TSP, it is difficult to use above the methods.

So many researchers find inspiration from the nature to simulate creatures and solve global optimization problems. For example, genetic algorithms (GA), tabu search (TS), simulated annealing (SA), ant colonies optimization (ACO), particle swarm optimization (PSO), bee colonies optimization (BCO), monkey search algorithm (MS), harmony search algorithm (HS), firefly algorithm (FA), intelligent water drops (IWD), Cuckoo search, bat algorithm (BA), fireworks algorithm (FA), wolf search algorithm (WA), differential evolution and brainstorming algorithm etc., are presented in [2, 7–10, 19, 24, 26–31, 39, 40, 43] respectively. Most of these meta-heuristics are nature-inspired, mimicking successful feature in biological. All of these algorithms can solve combinatorial optimization problems depend on many factors, so there are no specific algorithms to solve all optimization problems. And many intelligence algorithms need to be improved, especially for N-P problems.

Every known, many these intelligence algorithms have been used in TSP. In [32], chaotic simulated annealing with a transiently chaotic neural network was applied to the travelling salesman problem. Reference [18] proposed an iterated tabu search algorithm for the well-known combinatorial optimization problem, the travelling salesman problem, which was based on so-called intensification and reconstruction paradigm. Also Discrete Particle Swarm Optimization [25], Genetic Simulated Annealing Ant Colony System with Particle Swarm Optimization Techniques (GSA-ACS-PSOT) [6] and Fuzzy Particle Swarm Optimization with Simulated Annealing and Neighborhood Information (PSO-NIC-SA) [1] are used for TSP.

The Cuckoo search (CS) algorithm is a new heuristic swarm intelligence algorithm proposed by Yang and Deb [38]. CS algorithm is derived from the behavior simulation of Cuckoo parasitic brood behavior and birds or drosophila melanogaster. CS algorithm has been used in engineering practice because of its relatively simple structure, less control parameters, the optimized search path and strong ability to jump out of local extremum value. The performance of CS algorithm is very close to differential evolution, so CS has become a hot topic in the field of intelligent algorithms. However, CS algorithm has some disadvantages, such as slow search speed in late of algorithm, low convergence precision and search energy shortage. Therefore, lots of researchers have proposed many improved CS algorithms to further improve the performance of the CS algorithm. In reference [33], the author adaptively modified parameter detection probability P_a and step factor α and applied it into the feed-forward neural network, which improved convergence speed and optimization accuracy of initial algorithm. Yin [17] proposed a new method based on discrete artificial bee colony algorithm for travelling salesman problem. Searching strategy and transforming mechanism of leading bees, following bees and scout bees were redefined according to discrete variables. The transition of swarm role was based on ratios factor of definition. Leading bees used 2-Opt operator and learning operator to accelerate the convergence speed and to search the neighborhood. Yang [15] considered multistage hybrid flow shop scheduling problems. Tehn NEH heuristic was incorporated with the initial solutions to obtain the optimal or near optimal solutions rapidly in the improved Cuckoo search algorithm. In [4, 11, 16, 37], Cuckoo search algorithm or improved Cuckoo search algorithm were used for flow shop scheduling problems. Wang [34] put forward a hybridizing harmony search algorithm with Cuckoo search for global numerical optimization. Li [14] used two new mutation rules based on the rand and best individuals among the entire population to improve Cuckoo search algorithm. What's more, CS was used for engineering design problems in [41]. Most of them mainly aim to optimize single objective problem.

Therefore, this paper proposes an improved Cuckoo search algorithm based on adaptive simulated annealing method to offset the disadvantage of traditional CS(easily falling into local solution, slow convergence speed). The new method utilizes the strong global search ability of adaptive simulated annealing. It can randomly select part of bird's nests to make simulated annealing search, which improves the convergence speed and accuracy of the algorithm. Finally, experiments are conducted by testing twelve standard functions and compared with other intelligence algorithms. The results show that new algorithm has the better optimization performance. The following is the structure of this paper. In Section 2 and Section 3, it briefly introduces cuckoo search algorithm and TSP respectively. Section 4 introduces the detailed new algorithm. Experiments are conducted to demonstrate the better performance in Section 5. There is a conclusion in the last section.

2 Cuckoo Search Algorithm

The most special habit of cuckoo is parasitic brood. At the time of reproduction, the cuckoo cannot hatch for themselves, rather than lay their eggs in other birds' nest, let them hatch. Hatched chicks instinctively can damage other eggs in the nest, and be able to emit a more loudly scream than chicks of the host. In order to reduce the probability that they are found, some cuckoos mimic their eggs as the selected birds' eggs. Many hosts judge the health degree of next generations through chirp size, and healthy offspring can get more food. So there is a higher survival rate. Under some certain conditions, when the host finds some strange eggs in the nest, host will abandon the nest and choose another place as a nest. Competing with host, the cuckoo eggs and chicks scream will have a development toward the direction of simulating host to against the increased ability of distinguishing next generation of host. The cuckoo search algorithm is derived from the reproduction behavior of cuckoo and *Lévy* flight mode. *Lévy* flight is a kind of random flight way combining long time and small scale local search with occasionally a larger scale exploring. In biological swarm intelligent optimization algorithms, this way can make a bigger search filed, more diversity of population. Also it is easy to jump out of local optimal solution. In the cuckoo search algorithm, in order to make a virtualization for laying eggs cuckoo, it needs to make three idealized assumptions as follows [42].

1) Every cuckoo lays only one egg, and then randomly finds a nest of other birds to hatch it.

2) In a group of randomly selected nest, the best bird's nest will be retained.

3) The number of available nest n is constant, and the probability of finding other eggs by the host is $pa \in [0, 1]$.

Based on the above assumptions, the mind of CS algorithm is that an egg in bird nest is a candidate solution, a cuckoo egg is a new solution, CS uses a new solution and a better solution in the next generation to replace the worst candidate solution in nest. Finally, it finds the best solution in the nest.

Assuming that D-dimension vector $X_i = [x_1, x_2, \cdots, x_D]$ denotes every egg or cuckoo. Levy method is used to produce offspring,

$$x_i^{t+1} = x_i^t + a \otimes levy(\lambda). \tag{1}$$

where x_i^{t+1} and x_i^t are the $i-th$ nest position in the $t+1$ iteration and t iteration respectively. $\otimes$ is point to point multiplication. a is step length control. $lévy(\lambda)$ denotes jumping path of random search, $lévy \sim u = t^{-\lambda}$. When position updating, random number $r = [0, 1]$ compares with pa. If $r > pa$, it makes a random transformation for x_i^t. Otherwise, it keeps unchanged.

Detailed processes of CS are as follows(summarized in **Algorithm 1**).

Step 1. Initializing the population. It randomly generates N bird's nests in D-dimension solution space and computes all the nests' fitness value, then it keeps the nest with the optimal fitness value.

Step 2. Assuming that the position of $i-th$ nest is $x_i = [x_{i1}, x_{i2}, \cdots, x_{iD}], 1 \le i \le N$. It calculates the fitness value of the whole nests in the $t-th$ iteration. The optimal nest's position will be kept in the next iteration. Updating nest's position by using the following two formulas.

$$x_i^{t+1} = x_i^t + stepsize \otimes N(0,1). \tag{2}$$

$$stepsize = \alpha \cdot step \otimes (x_i^t - x_{best}^t). \tag{3}$$

Where x_i^t is the $t-th$ iteration position of nest i. $N(0,1)$ denotes the D-dimension standard normal distribution. $\otimes$ denotes point to point multiplication. *step* is the random step length generated by *Lévy* distribution. x_{best}^t is the best nest in $t-th$ iteration.

Step 3. Nest host finds the external egg with probability P_a. And it randomly generates number $r \in [0,1]$. If $r > P_a$, the old nest will be replaced by new nest. Calculating all the nests' fitness value and keeping the nest with optimal fitness.

Step 4. Recording the optimal solution. Otherwise return to Step 2.

Algorithm 1 Cuckoo search algorithm

```
Begin
Assuming that objective function f(x), x = (x_1, ..., x_d)^T.
Generating initial population of n host nests x_i, i = 1, ..., n.
while t < MaxGeneration do
  Randomly obtain a cuckoo by Lévy flights.
  Evaluate its fitness F_i.
  Randomly select a nest from n(i.e.j).
  if F_i > F_j then
    Use new solution to replace j.
  end if
  A fraction p_a of worse nests are abandoned and new ones are built.
  Keep the best solutions (or nests with quality solutions);
  Rank the solutions and find the current best
end while
Postprocess results and visualization
End
```

3 Travelling Salesman Problem (TSP)

TSP is a hot issue, which was by the Irish mathematician W.R. Hamilton and the British mathematician Thomas Kirkman introduced in 1800s [20]. TSP can be described as a permutation problem with the objective of finding a shortest closed tour which visits all the cities in a given set; TSP can be modeled as completely connected graph in a D-dimensional Euclidean space (D is size of the problem), which cities are the graph's vertices, paths are the graph's edges, and a path's distance is the edge's length. Mathematically, it can be defined as given a set of n cities, named $(c_1, \cdots, c_n)$, and permutations $\pi_1, \cdots, \pi_n$, the goal is to find a number of permutation $\pi_i = c_1, c_2, \cdots, c_n$, such that minimizes $f(\pi)$ the sum of all Euclidean distances d between each city from the same path π and it is given as follows:

$$f(\pi) = \sum_{i=1}^{n-1} d_\pi(i)\pi(i+1) + d_\pi(n)\pi(1). \tag{4}$$

The Euclidean distance d, between any two cities with coordinate (x_1, y_1) and (x_2, y_2) is calculated by:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{5}$$

Broadly, the TSP is classified as STSP. In this symmetric TSP variation, all of the edge costs are symmetric. This means that, for all vertices in the graph, the cost incurred, when moving from vertex i to vertex j, is the same as the cost incurred when moving from vertex j to vertex i, i.e., $d_{ij} = d_{ji}$.

4 Cuckoo Search Algorithm Based on Adaptive Simulated Annealing

Basic CS adopts *Lévy* flight way to randomly generate position of new bird's nest. However, this method easily results in slow convergence speed and imprecise optimization results. Hence, this paper uses simulated annealing method to update part of nests, which avoids cuckoo algorithm falling into local solution. That can produce new optimal individual and execute next iteration that improve the search accuracy.

4.1 Adaptive Simulated Annealing Method

Simulated annealing (SA) method [3, 36] simulates cooling process of classic particle system in thermodynamics to solve the planning problem. As the temperature T of the isolated particle system falls at a slow speed, system approximates in the thermodynamic equilibrium state, which is equivalent to the global minimum point of energy function. SA adopts Metropolis acceptance rule for each value of control parameter T, continuously produces "new solution-judgment-accept or reject" iteration. SA utilizes a stochastic approach to direct the search. Even if the move causes the value of the objective function to become worse, SA allows the search to proceed to a neighboring state. SA guides the original local search method in the following way. If a move to a neighbor S' in the neighborhood $N_\lambda(S)$ decreases the objective function value, or leaves it unchanged then the move always accepted. More precisely, the solution S' is accepted as the new current solution if $\Delta < 0$, where $\Delta = C(S') - C(S)$. To allow the search to escape a local optimum, moves that increases the objective function value are accepted with a probability $e^{-\frac{\Delta}{T}}$, if $\Delta > 0$, where T is a parameter called the "temperature". The value of T varies from a relatively large value to a small value close to zero. These values are controlled by a cooling schedule which specifies the initial and temperature values at each stage of the algorithm. Detailed SA processes are as follows.

Step 1. Given iteration initial solution x_0, fitness value $f(x_0)$ and cooling schedule parameters (including cureent temperature T_i, attenuation coefficient A, final value f_1 and Markov chain length L). In this process, T_i is calculated by:

$$T_i = \frac{1}{2}(T_0 - T_N) \times (1 + cos(\frac{i\pi}{N_{max}})) + T_N. \tag{6}$$

Where T_0 is the initial temperature, T_N is the lower bound of temperature, N_{max} is the total iteration number.

Step 2. When $T = T_i$, it executes the following search.

- According to current solution's position x_k, it uses the following formula to generate next generation's position x_{k+1}.

$$x_{k+1} = x_k \oplus r. \tag{7}$$

Where $\oplus$ is point to point addition, r is a random number.

- Generate a random number β in $(0, 1)$. Use the following formula to calculate a corresponding transition probability P^* with the given current solution x_k and temperature T_i.

$$P = e^{\frac{f(x_k)-f(x_{k+1})}{T_i}}, f(x_{k+1}) \geq f(x_k). \quad (8)$$

  If $\beta < P$, then accept new solution $x_k = x_{k+1}$. Otherwise the current solution is unchanged.

- If search number is less than L, return to first step. Otherwise, return to Step 3.

Step 3. Judging whether it meets stop condition. If YES, then stop iteration. Otherwise return to step 2. And update temperature as T_{i+1}. Find the optimal solution in this temperature.

4.2 Judging Criteria for Adaptive Annealing Mechanism

Whether nest enters into simulated annealing process depending on whether algorithm falls into local solution. Quality evaluation is defined as change rate of optimal nest fitness value as formula(9).

$$\frac{|f(x^t_{best}) - f(x^{t-10}_{best})|}{|f(x^t_{best})|} < \varepsilon, t > 10. \quad (9)$$

Where $f(x^t_{best})$ is current optimal nest fitness value. $f(x^{t-10}_{best})$ is the optimal nest fitness value in $m-th$ iteration. $\varepsilon = 0.005$ is threshold value.

4.3 Nest's Update Mechanism and Parameter Setting in Adaptive SA

Although adaptive SA has the strong global search ability, its efficient is very low. In the proposed algorithm, it uses adaptive SA to avoid CS falling into local solution. Therefore, it sets nest's update mechanism and parameter in adaptive SA.

In basic adaptive SA, it uses Formula (6) to update current solution x_k. And Formula (10) is selected as update mechanism.

$$x_{k+1} = x_k + (x_k \otimes N(0, 1)). \quad (10)$$

Mainly the method selects initial control parameter $T_0 = 3$, attenuation coefficient $A = 0.9$, final value $T_{end} = 0.01$ and Markov chain length $L = 30$. After setting these parameters, it not only utilizes the strong global search ability, but avoids low efficiency of algorithm.

5 Detailed Process of Proposed Algorithm

Main idea of improved Cuckoo search algorithm based on adaptive simulated annealing method (ICASA) is that when algorithm falls into local optimal solution, it randomly selects part of nests to update and conducts adaptive SA search to generate new nests (Shown in **Algorithm 2**). The processes are as follows.

Step 1. Initializing parameters. Assuming that population number N, D-dimension search space, the biggest iteration number N_{iter}, step length factor α, detection probability P_a.

Step 2. Initializing population. Randomly produce N nests, calculate each nest's fitness value and keep the optimal nest.

Step 3. Conduct *Lévy* iteration and update nest's position by using (2)(3). Compute new nest's fitness value and compare to the last nest, keep the better nest.

Step 4. Use (9) to judge whether algorithm falls into global optimal solution. If YES, then randomly selects parts of nests to conduct adaptive SA, the rest nests conduct two-way crossover operation. Then it merges the results with the above ways. Compute all the nests' fitness value and keep the optimal nest. Return to step5. If it does not fall into global optimal solution, then directly returns to step5.

Step 5. Nest host finds outside egg with probability P_a and generates uniform distribution random number $r \in (0, 1)$. If $r > P_a$, it makes randomly mutation operation for the found nests and uses new nest to replace old nest. Compute all the nests' fitness value and keep the best nest.

Step 6. Record historical optimal solution. If it does not reach stop condition, then return to Step 2.

Algorithm 2 Improved Cuckoo search algorithm based on adaptive SA

Begin
Assuming that objective function $f(x)$, $x = (x_1, \cdots, x_d)^T$.
Generating initial population of n host nests $x_i, i = 1, \cdots, n$.
while $t < MaxGeneration$ **do**
  Randomly obtain a cuckoo position by $Lévy$ flights.
  Evaluate its fitness F_i.
  Randomly select a nest from $n(i.e.j)$.
  if $F_i > F_j$ **then**
    Conduct adaptive SA operation and use new solution to replace j.
  end if
  A fraction p_a of worse nests are abandoned and new ones are built.
  Keep the best solutions (or nests with quality solutions);
  Rank the solutions and find the current best
end while
Postprocess results and visualization
End

6 Experiments and Analysis

In order to verify the new algorithm, it makes experiments on TSP with the new method and take benchmarks data from TSPLIB library [23]. Each function runs 50 times independently. The experiments are conducted in MATLAB R2010a, with 2.2GB RAM, CPU 2.1GHz and Intel Core. Also comparisons are made to genetic algorithm (GA), particle swarm optimization algorithm (PSO), ant colony optimization (ACO), random-key cuckoo search (RKCS) algorithm [21], discrete monarch butterfly optimization (DMBO) [35], Kalman filter cuckoo search algorithm (KFCS) [13] and the new method (ICASA) using several standard functions in TSPLIB.

The experiment uses the same parameters: population number $N = 30$, maximum iteration number $N_{iter} = 500$, step length factor $\alpha = 0.01$, find probability $P_a = 0.25$.

Tables 1 and 2 are the function optimization results with GA, PSO, ACO, RKCS, DMBO, KFCS and ICASA. Table1 indicates that the optimal solution of the tested functions with the seven algorithms. Following is the explanation for every column. The first column indicates the instance name ended by the number of cities, the second column stands for the algorithms. the third column indicates the average value obtained by different algorithms, the fourth one represents the standard deviation, the fifth column indicates the best value, the sixth column denotes the obtained worst results, the seventh column represents average time.

The experiment uses the twelve functions to test the seven algorithms as Table 3 and evaluate them from average value, standard deviation, best value, worst value and average time respectively. Sphere and Rosenbraock is the high-dimensional unimodal function and only has global optimal value, which can test the mining ability. And Schaffer, Griewank, Rastrigin and Ackley is multimodal function and has several local extreme values with that the algorithm easily falls into local optimal value, which can test the exploring ability of algorithm. First, average value and best value can show convergence precision and optimization ability. So ICASA is superior to other six algorithms in table1 about average value and best value. For the Sphere function, optimization result approaches to theoretical optimum reaching at 10^{-16}. It illustrates that ICASA has a good mining ability due to SA operation. With the iteration number increasing, cuckoo search range gradually drops. Search range is more accuracy, which also increases convergence speed. For Schaffer, Griewank and Ackley function, theoretical optimum has been found, accuracy of Ackley is 10^{-21}, which demonstrates that ICASA has a good search ability. SA can help cuckoo communicate with each other and reduce the probability of falling into local value, improve the search ability of algorithm. Standard deviation and worst value presents the robustness of algorithm. From table1, it can be known that ICASA has the better robustness. Average time means complexity of the algorithm. GA, PSO, ACO is higher than ICASA. ICASA does not add running time, which shows that it is an effectiveness algorithm. Figure1-4 are average optimal value with the seven algorithms (X denotes iteration number, Y denotes average optimal value). (In the figure, deep red, red, orange, yellow, green, deep blue and purple denote the ICASA, KFCS, DMBO, RKCS, ACO, PSO and GA respectively.).

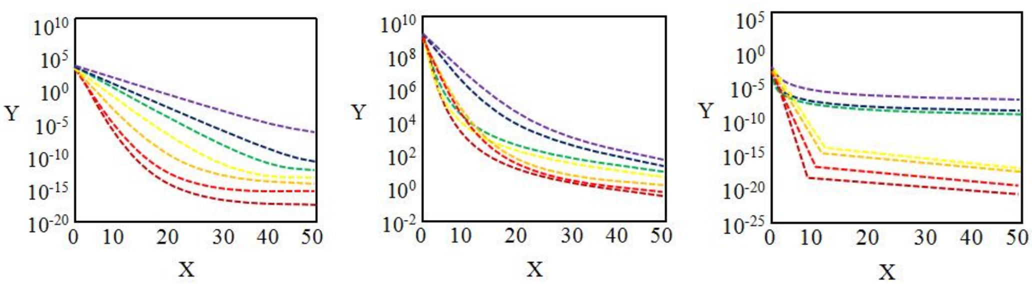


Figure 1: Sphere, Rosenbrock and Schaffer function

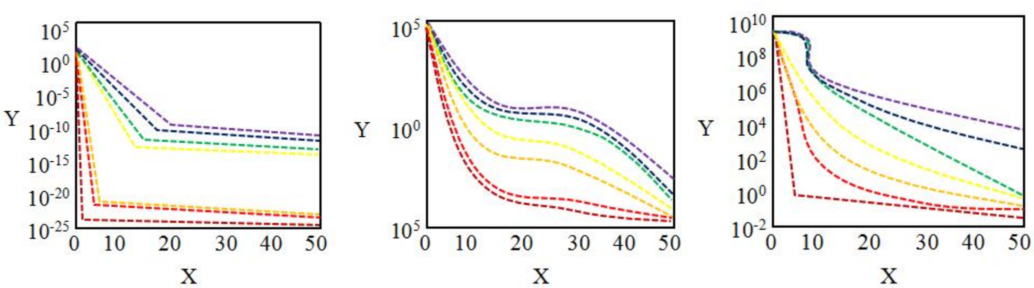


Figure 2: Griewank, Rastrigin and Ackley function

This paper also selects six typical TSP problems in TSPLIB: bays29, eil51, berlin52, gr96, pr144

Table 1: The function optimization results with GA, PSO, ACO, RKCS, DMBO, KFCS and ICASA

Function	Algorithm	Average value	Standard deviation	Best value	Worst value	Average time
Stepere	GA	16.843	14.574	20.584	12.649	16.87
	PSO	12.366	10854	16.549	8.172	11.58
	ACO	8.05	7.64	14.249	3.121	11.23
	RKCS	$5.424e^{-18}$	$8.563e^{-18}$	$1.507e^{-17}$	$1.532e^{-20}$	7.71
	DMBO	$6.785e^{-18}$	$6.527e^{-18}$	$6.3836e^{-17}$	$2.571e^{-20}$	7.65
	KFCS	$5.424e^{-18}$	$8.563e^{-18}$	$1.507e^{-17}$	$1.532e^{-20}$	7.56
	ICASA	0	0	0	0	6.52
Rosenbrock	GA	$3.217e^{-3}$	$2.229e^{-3}$	$4.672e^{-3}$	1.579^{-3}	10.34
	PSO	$3.996e^{-7}$	$2.387e^{-7}$	$6.024e^{-7}$	1.112^{-7}	6.18
	ACO	$6.649e^{-7}$	$5.448e^{-7}$	$9.187e^{-7}$	4.44^{-7}	5.64
	RKCS	$6.254e^{-9}$	$6.661e^{-9}$	$10.254e^{-9}$	2.288^{-9}	4.26
	DMBO	$6.579e^{-15}$	$5.331e^{-15}$	$8.314e^{-15}$	3.947^{-15}	3.78
	KFCS	$5.624e^{-17}$	$4.624e^{-17}$	$9.645e^{-17}$	$1.264e^{-17}$	3.87
	ICASA	0	0	0	0	1.06
Schaffer	GA	$8.942e^{-3}$	$7.778e^{-3}$	$15.649e^{-3}$	$1.674e^{-3}$	22.38
	PSO	$10.318e^{-5}$	$9.543e^{-5}$	$14.619e^{-5}$	$6.447e^{-5}$	20.54
	ACO	$2.784e^{-9}$	$5.174e^{-9}$	$4.316e^{-9}$	$1.294e^{-9}$	16.94
	RKCS	$6.549e^{-10}$	$5.943e^{-10}$	$9.544e^{-10}$	$3.856e^{-10}$	14.25
	DMBO	$6.748e^{-14}$	$5.174e^{-14}$	$8.354e^{-13}$	$4.587e^{-15}$	14.25
	KFCS	$3.472e^{-14}$	$1.455e^{-14}$	$4.218e^{-13}$	$2.642e^{-15}$	12.67
	ICASA	0	0	0	0	10.24
Griewank	GA	1.123	0.975	3.214	0.642	6.78
	PSO	0.484	0.3141	0.7081	0.2532	5.67
	ACO	$7.364e^{-11}$	$6.669e^{-11}$	$8.673e^{-11}$	$6.084e^{-11}$	4.55
	RKCS	$4.589e^{-13}$	$4.661e^{-13}$	$6.638e^{-13}$	$3.342e^{-13}$	3.38
	DMBO	$6.245e^{-14}$	$5.414e^{-14}$	$2.225e^{-13}$	$9.286e^{-17}$	3.38
	KFCS	$10.661e^{-20}$	$9.876e^{-20}$	$15.436e^{-20}$	$5.647e^{-20}$	2.21
	ICASA	0	0	0	0	2.08
Rastrigin	GA	$6.477e^{-3}$	$5.123e^{-3}$	$9.582e^{-3}$	$3.189e^{-3}$	7.85
	PSO	$5.123e^{-7}$	$4.875e^{-7}$	$7.316e^{-7}$	$3.487e^{-7}$	7.85
	ACO	$6.149e^{-7}$	$5.318e^{-7}$	$7.649e^{-7}$	$3.335e^{-7}$	7.85
	RKCS	$6.245e^{-13}$	$6.314e^{-13}$	$9.057e^{-13}$	$7.945e^{-13}$	6.66
	DMBO	$1.614e^{-13}$	$3.58e^{-13}$	$7.778e^{-12}$	$5.331e^{-15}$	6.66
	KFCS	$4.656e^{-20}$	$3.986e^{-20}$	$6.478e^{-20}$	$2.345e^{-20}$	6.66
	ICASA	0	0	0	0	5.61
Ackley	GA	$3.365e^{-9}$	$3.549e^{-9}$	$4.112e^{-9}$	$2.645e^{-9}$	16.08
	PSO	$3.798e^{-15}$	$3.146e^{-15}$	$6.487e^{-15}$	$1.669e^{-15}$	15.91
	ACO	$5.219e^{-15}$	$5.113e^{-15}$	$7.336e^{-15}$	$3.674e^{-15}$	14.64
	RKCS	$6.798e^{-19}$	$6.491e^{-19}$	$8.557e^{-19}$	$4.679e^{-19}$	12.59
	DMBO	$6.875e^{-19}$	$5.347e^{-19}$	$8.457e^{-19}$	$5.664e^{-19}$	12.59
	KFCS	$3.461e^{-19}$	$4.102e^{-19}$	$4.397e^{-19}$	$2.672e^{-19}$	12.58
	ICASA	$7.328e^{-21}$	$5.943e^{-21}$	$9.478e^{-21}$	$5.943e^{-21}$	11.67

Table 2: The function optimization results with GA, PSO, ACO, RKCS, DMBO, KFCS and ICASA (Cont.)

Function	Algorithm	Average value	Standard deviation	Best value	Worst value	Average time
U574	GA	$6.748e^{-13}$	$6.541e^{-13}$	$8.388e^{-13}$	$5.847e^{-13}$	15.51
	PSO	$7.846e^{-15}$	$7.564e^{-15}$	$8.346e^{-15}$	$7.285e^{-15}$	13.24
	ACO	$7.842e^{-15}$	$7.321e^{-15}$	$9.714e^{-15}$	$6.123e^{-15}$	12.64
	RKCS	$6.486e^{-19}$	$6.664e^{-19}$	$7.358e^{-19}$	$5.674e^{-19}$	8.88
	DMBO	$3.942e^{-19}$	$3.673e^{-19}$	$6.488e^{-19}$	$1.947e^{-19}$	8.67
	KFCS	$3.287e^{-20}$	$3.242e^{-20}$	$4.258e^{-20}$	$2.357e^{-20}$	7.86
	ICASA	$7.457e^{-23}$	$7.264e^{-23}$	$9.364e^{-23}$	$5.347e^{-23}$	6.78
D1291	GA	11.556	10.641	16.342	6.785	9.15
	PSO	13.809	12.851	22.192	4.789	9.12
	ACO	$4.618e^{-3}$	$5.346e^{-3}$	$6.742e^{-3}$	$2.557e^{-3}$	7.75
	RKCS	$6.849e^{-5}$	$5.249e^{-5}$	$9.127e^{-5}$	$4.467e^{-5}$	6.67
	DMBO	$3.849e^{-5}$	$3.338e^{-5}$	$6.669e^{-5}$	$1.349e^{-5}$	5.23
	KFCS	$5.627e^{-8}$	$4.794e^{-8}$	$8.644e^{-8}$	$3.243e^{-8}$	5.21
	ICASA	0	0	0	0	4.57
Pr2392	GA	5.874^{-7}	5.532^{-7}	6.477^{-7}	4.582^{-7}	21.34
	PSO	7.187^{-9}	7.054^{-9}	8.996^{-9}	5.347^{-9}	20.59
	ACO	5.874^{-9}	5.532^{-9}	6.477^{-9}	4.582^{-9}	18.64
	RKCS	7.121^{-12}	6.482^{-12}	10.674^{-12}	5.397^{-12}	12.38
	DMBO	4.668^{-12}	4.167^{-12}	7.542^{-12}	1.396^{-12}	12.38
	KFCS	5.645^{-12}	$5.484e^{-12}$	$6.584e^{-12}$	$4.127e^{-12}$	12.38
	ICASA	$5.978e^{-17}$	$4.567e^{-17}$	$8.917e^{-17}$	$2.526e^{-17}$	10.54
Usa13509	GA	6.749^{-9}	6.428^{-9}	8.333^{-9}	5.971^{-9}	20.87
	PSO	6.889^{-13}	6.478^{-13}	8.943^{-13}	4.879^{-13}	18.99
	ACO	4.387^{-13}	4.137^{-13}	5.677^{-13}	3.121^{-13}	17.54
	RKCS	3.228^{-13}	3.216^{-13}	3.471^{-13}	2.994^{-13}	15.44
	DMBO	$6.487e^{-16}$	$6.348e^{-16}$	$7.336e^{-16}$	$5.471e^{-16}$	15.34
	KFCS	$5.477e^{-19}$	$5.617e^{-19}$	$6.341e^{-19}$	$4.627e^{-19}$	10.54
	ICASA	0	0	0	0	8.69
KroD100	GA	3.334^{-9}	3.411^{-9}	4.189^{-9}	2.552^{-9}	20.08
	PSO	7.846^{-13}	7.431^{-13}	9.706^{-13}	6.663^{-13}	17.23
	ACO	3.843^{-13}	3.174^{-13}	5.313^{-13}	1.888^{-13}	16.11
	RKCS	$7.089e^{-15}$	$6.943e^{-15}$	$7.791e^{-15}$	$6.477e^{-15}$	15.78
	DMBO	$3.788e^{-17}$	$3.412e^{-17}$	$4.617e^{-17}$	$2.997e^{-17}$	13.98
	KFCS	$6.784e^{-18}$	$6.327e^{-18}$	$8.713e^{-18}$	$4.747e^{-18}$	11.27
	ICASA	0	0	0	0	7.45
rat783	GA	$4.428e^{-12}$	$5.314e^{-12}$	$3.592e^{-12}$	$5.638e^{-12}$	20.37
	PSO	$4.428e^{-12}$	$5.314e^{-12}$	$3.592e^{-12}$	$5.638e^{-12}$	16.73
	ACO	$6.493e^{-8}$	$6.354e^{-8}$	$9.647e^{-8}$	$3.189e^{-8}$	12.54
	RKCS	$7.549e^{-10}$	$4.512e^{-10}$	$12.579e^{-10}$	$2.546e^{-10}$	12.47
	DMBO	$6.582e^{-7}$	$5.473e^{-7}$	$8.547e^{-7}$	$8.426e^{-7}$	10.05
	KFCS	$7.724e^{-9}$	$6.127e^{-9}$	$4.743e^{-9}$	$9.547e^{-9}$	10.14
	ICASA	$4.428e^{-12}$	$5.314e^{-12}$	$3.592e^{-12}$	$5.638e^{-12}$	8.54

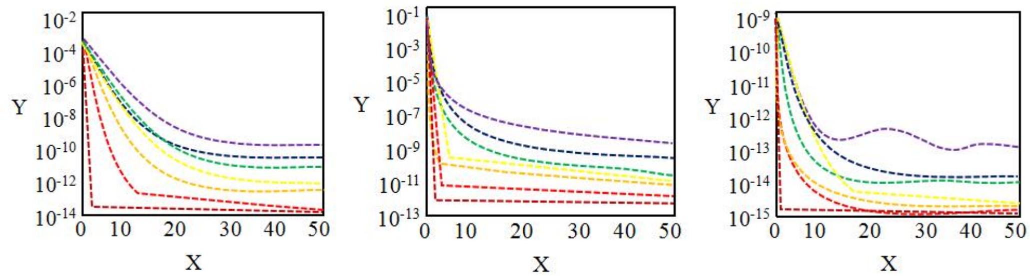


Figure 3: U574, D1291 and Pr2392 function

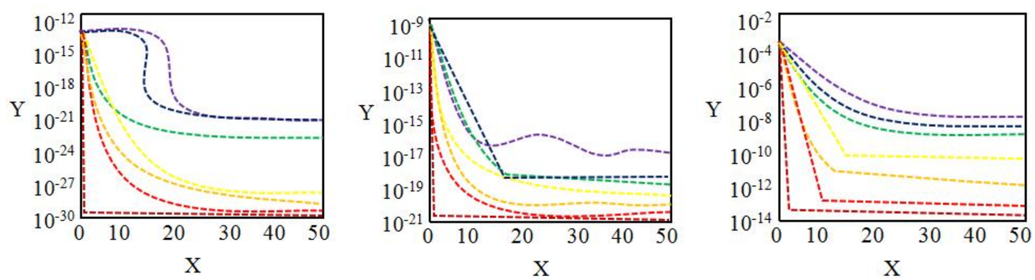


Figure 4: Usa13509, KroD100 and rat783 function

and gr202 to make much optimization calculation. Evaluation criterion: optimal value (OV), success rate (SR%), deviation (DE%), percentage deviation of a solution (PDs%) and average iteration times (AIT). Supposing S_0 is the shortest path value. If after calculating optimal path value $best/S_0 < 1.03$, so the current optimization calculation is successful [12]. DE can be calculated by :

$$DE = \frac{\sum_{i=1}^{T}(B_i - S_0)}{T_{S_0}} \times 100\%. \tag{11}$$

$$PDs = \frac{(s_1 - s_2)}{s_2} \times 100\%. \tag{12}$$

Where B_i is the shortest path value at $i - th$ iteration. s_1 is solution length and s_2 is optimal solution length. T=30 is the number of dependently calculation. Additionally, in order to evaluate the performance and to give more credibility of results, Table 3 (it selects six functions from above test functions) makes respectively a fair comparison to discrete particle swarm optimization (DPSO), original cuckoo search algorithm (CS) and improved CS (ICS) as presented in their original papers.

From Table 3 it can know that ICASA can obtain the optimal path solution for different TSP problems. And it can reach the optimal solutions of 0 from all tested instances of PDs, which indicates that all the solutions found over 1000 running times are the same as the optimal solutions known ICASA has the higher optimization accuracy. It furthest ensures that city vertexes constitute convex polygon, so it can better construct shortest loop. Adaptive SA operation avoids CS falling into local solution. That helps CS move toward the good solution, which further reduces the search range. So ICASA is very efficient for solving small-size as well as large-size instances of TSP problem in a reasonable time. Figure 5,6 are the optimal path for solving the six TSP with ICASA method.

Figure 7 is the comparison of average computation time with the seven algorithms. From this figure it can be seen that GA needs long time to finish iteration and has a long convergence time. Compared

Table 3: Performance results for different algorithms

Function	Algorithm	OV	PDs	SR	DE	AIT
bays29	CS	$9.529e^3$	0.68	95.47	-5.51	55
	ICS	$9.488e^3$	0.56	96.13	-4.18	45
	DPSO	$9.475e^3$	0.48	98.88	-3.47	25
	ICASA	$8.652e^3$	0	100	-0.21	20
eil51	CS	452.213	0.72	94.36	4.18	213
	ICS	430.3457	0.48	95.37	3.64	129
	DPSO	425.9948	0.13	99.28	2.89	72
	ICASA	425.6417	0	100	2.32	68
berlin52	CS	$9.6417e^3$	0.66	95.57	1.54	151
	ICS	$7.6492e^3$	0.64	96.87	1.35	94
	DPSO	$4.5517e^3$	0.22	97.32	1.24	66
	ICASA	$3.146e^3$	0	100	1.05	48
gr96	CS	532.663	0.48	95.87	2.74	213
	ICS	530.641	0.32	97.66	2.64	129
	DPSO	521.647	0.14	98.64	1.54	72
	ICASA	512.674	0	100	1.05	68
pr144	CS	$6.847e^4$	0.47	96.77	1.58	368
	ICS	$4.318e^4$	0.35	98.67	1.32	249
	DPSO	$3.341e^4$	0.24	99.65	0.25	189
	ICASA	$1.947e^4$	0	100	0.08	157
gr202	CS	532.781	0.49	96.37	1.58	148
	ICS	512.322	0.41	97.98	0.96	134
	DPSO	507.621	0.23	99.66	0.84	71
	ICASA	496.327	0	100	0	70

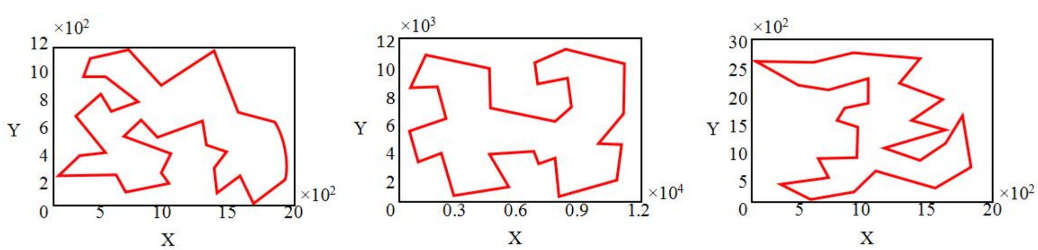


Figure 5: bays29, eil51 and berlin52

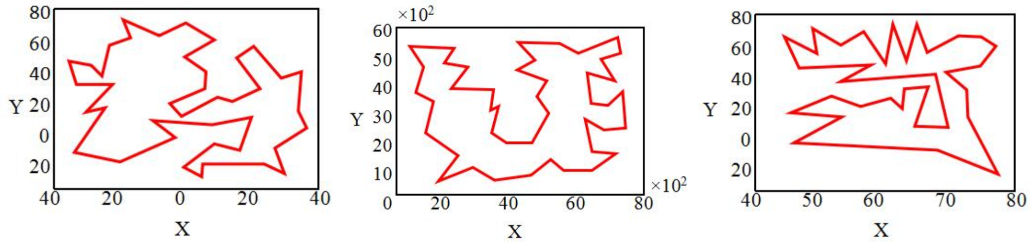


Figure 6: gr96, pr144 and gr202

to GA, PSO would run nearly 510 times when reaching to convergence time approximately 40ms. However, computation time of BKCS will be 30ms less than ACO and exceed that of DMBO (about 10ms). Finally, ICASA consumes closely 4ms to get convergence with only 20 times. Therefore, new method ICASA obtains the best performance in computing TSP time.

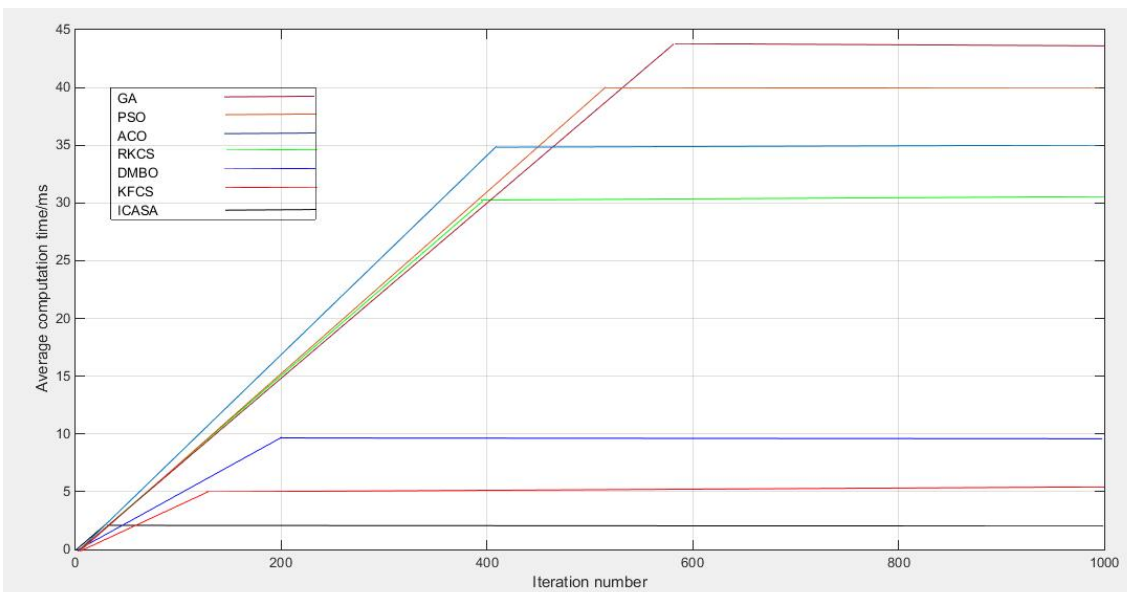


Figure 7: Comparison of computation time

Figure 8 shows evidently that the curve associated with ICASA is better in terms of solutions quality compared to KFCS(the latest method).

7 Conclusion

In this paper, an improved cuckoo search algorithm based on adaptive SA is proposed to optimize the traditional CS with disadvantage of slow convergence speed and inaccuracy results. To verify the new method's performance, twelve testing functions in TSPLIB are conducted. Experiments show that ICASA has the best optimization effect on function. In addition, new method is applied into solving

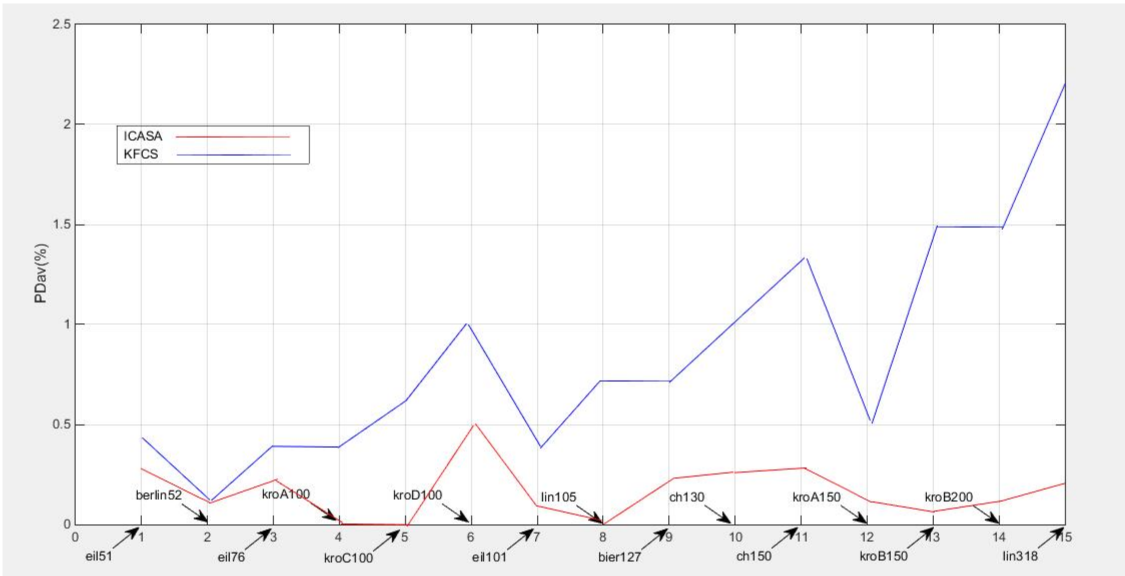


Figure 8: PDav(%)(over 30 running) for 15 instances from TSPLIB

TSP. By introducing path crossover elimination mechanism and testing standard TSP problems, new method has the better performance than other algorithms. Finally, it may also be interesting to hybrid the ICASA with other heuristics or meta-heuristics and to analyze the performance of each algorithm in solving TSP problem. In the future, more intelligence artificial algorithm will be proposed. We also study other advanced methods to apply them into actual applications.

Acknowledgments

Authors thank Software College in Shenyang Normal University for supporting the experiment.

References

[1] R. F. Abdel-Kader, "Fuzzy particle swarm optimization with simulated annealing and neighborhood information communication for solving TSP," *International Journal of Advanced Computer Science and Applications*, vol. 2, 2011.

[2] V. Bhargava, S. E. K. Fateen, A. Bonilla-Petriciolet, "Cuckoo search: A new nature-inspired optimization method for phase equilibrium calculations," *Fluid Phase Equilibria*, vol. 337, pp. 191–200, 2013.

[3] P. Borges, T. Eid, E. Bergseng, "Applying simulated annealing using different methods for the neighborhood search in forest planning problems," *European Journal of Operational Research*, vol. 233, no. 3, pp. 700–710, 2014.

[4] S. R. Boussalia, A. Chaoui, M. Ouederni, P. Queinnec, "Multi-objective quantum inspired Cuckoo search algorithm and multi-objective bat inspired algorithm for the web service composition problem," *International Journal of Intelligent Systems Technologies & Applications*, vol. 15, no. 2, pp. 95–126, 2016.

[5] F. Carrabs, J. F. Cordeau, G. Laporte, "Variable neighborhood search for the pickup and delivery travelling salesman problem with LIFO loading," *Informs Journal on Computing*,vol. 19, no. 4, 2015.

[6] S. M. Chen, C. Y. Chien, "Solving the travelling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques," *Expert Systems with Applications*, 2011.

[7] M. Dorigo, M. Birattari, T. Stutzle, "Ant colony optimization," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28–39, 2006.

[8] Z. W. Geem, J. H. Kim, G. V. Loganathan, "A new heuristic optimization algorithm: harmony search," *Simulation*, vol. 76, no. 2, pp. 60–68, 2001.

[9] F. Glover, M. Laguna, *Tabu Search*, Springer New York, 2013.

[10] D. E. Goldberg, *Genetic Algorithms*, Pearson Education India, 2006.

[11] W. Hui, J. Wen, H. Wang, Sun Z. H. Cui, S. Rahnamayan, S. Zeng, "A new Cuckoo search algorithm with hybrid strategies for flow shop scheduling problems," *Soft Computing*, vol. 21, no. 15, pp. 4297–4307, 2017.

[12] I. H. Kuo, S. J. Horng, T. W. Kao, T. L. Lin, C. L. Lee, Y. H. Chen, Y. I. Pan and T. Terano, "A hybrid swarm intelligence algorithm for the travelling salesman problem," *Expert Systems*, vol. 27, pp. 166–179, 2010.

[13] J. Li, T. Zhang, "The Kalman filter Cuckoo search algorithm to solve the TSP problem," *Applied Mechanics and Materials*, vol. 733, pp. 918–921, 2015.

[14] X. Li, M. Yin, "Modified Cuckoo search algorithm with self adaptive parameter method," *Information Sciences*, vol. 298, pp. 80–97, 2015.

[15] M. K. Marichelvam, T. Prabaharan, X. S. Yang, "Improved Cuckoo search algorithm for hybrid flow shop scheduling problems to minimize makespan," *Applied Soft Computing*, vol. 19, no. 1, pp. 93–101, 2014.

[16] M. K. Marichelvam, T. Prabaharan, X. S. Yang, "Improved Cuckoo search algorithm for hybrid flow shop scheduling problems to minimize makespan," *Applied Soft Computing*, vol. 19, no. 1, pp. 93–101, 2014.

[17] L. Meng, S. Yin, X. Hu, "A new method used for travelling salesman problem based on discrete Artificial Bee Colony algorithm," *Telecommunication Computing Electronics and Control*, vol. 14, no. 1, pp. 342–348, 2016.

[18] A. Misevičius, "Using iterated tabu search for the travelling salesman problem," *Information Technology and Control*, vol. 32, no. 3, 2015.

[19] S. N. Omkar, J. Senthilnath, R. Khandelwal, G. N. Naik, S. Gopalakrishnan, "Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures," *Applied Soft Computing*, vol. 11, no. 1, pp. 489–499, 2011.

[20] A. Ouaarab, B. Ahiod, X. S. Yang, "Discrete Cuckoo search algorithm for the travelling salesman problem," *Neural Computing and Applications*, vol. 24, no. 7, pp. 1659–1669, 2014.

[21] A. Ouaarab, B. Ahiod, X. S. Yang, "Random-key Cuckoo search for the travelling salesman problem," *Soft Computing*, vol. 19, no. 4, pp. 1099–1106, 2015.

[22] A. Purbasari, I. S. Suwardi, O. S. Santoso, R. Mandala, "Data Partition and Communication On Parallel Heuristik Model Based on Clonal Selection Algorithm," *Telkomnika*, vol. 13, no. 1, 2015.

[23] G. Reinelt, "A travelling salesman problem library," *Journal of the Operational Research Society*, vol. 11, no. 1, pp. 19–21, 1992.

[24] H. Shah-Hosseini, "Intelligent water drops algorithm: A new optimization method for solving the multiple knapsack problem," *International Journal of Intelligent Computing and Cybernetics*, vol. 1, no. 2, pp. 193–212, 2008.

[25] X. H. Shi, Y. C. Liang, H. P. Lee, C. Lu, Q. X. Wang, "Particle swarm optimizationbased algorithms for TSP and generalized TSP," *Information Processing Letters*, vol. 103, pp. 169–176, 2007.

[26] Y. Shi, "Particle swarm optimization," *IEEE Connections*, vol. 2, no. 1, pp. 8–13, 2004.
[27] Y. Shi, "Brain storm optimization algorithm," in *International Conference in Swarm Intelligence*, pp. 303–309, 2011.
[28] R. Storn, K. Price, "Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
[29] H. Szu, R. Hartley, "Fast simulated annealing," *Physics Letters A*, vol. 122, no. 3-4, pp. 157–162, 1987.
[30] R. Tang, S. Fong, X. S. Yang, S. Deb, "Wolf search algorithm with ephemeral memory," in *IEEE Seventh International Conference on Digital Information Management (ICDIM'12)*, pp. 165–172, 2012.
[31] Y. Tan, Y. Zhu, "Fireworks algorithm for optimization," in *International Conference in Swarm Intelligence*, pp. 355–364, 2010.
[32] J. Tunnell, S. Abdul-Wahid, *Solving the TSP Using Chaotic Simulated Annealing*, 2014.
[33] E. Valian, S. Mohanna, S. Tavakoli, "Improved cuckoo search algorithm for feedforward neural network training," *International Journal of Artificial Intelligence & Applications*, vol. 2, no. 3, 2011.
[34] G. G. Wang, A. H. Gandomi, X. Zhao, H. C. E. Chu, "Hybridizing harmony search algorithm with Cuckoo search for global numerical optimization," *Soft Computing*, vol. 20, no. 1, pp. 273–285, 2016.
[35] G. G. Wang, G. S. Hao, S. Cheng, Q. Qin, "A discrete monarch butterfly optimization for Chinese TSP problem," in *International Conference in Swarm Intelligence*, pp. 165–173, 2016.
[36] S. Xavierdesouza, J. A. Suykens, J. Vandewalle, D. Bolle, "Coupled simulated annealing," *IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics*, vol. 40, no. 2, pp. 320–335, 2010.
[37] H. H. Yang, P. M. Xie, X. F. Zhang, W. Ma, Z. B. Liu, "Improved Cuckoo search algorithm for multi-objective optimization problems," *Journal of Zhejiang University (Engineering Science Edition)*, vol. 49, no. 8, pp. 1600–1608, 2015.
[38] X. Yang and S. Deb, "Cuckoo search via Levy flights," in *World Congress on Nature & Biologically Inspired Computing*, pp. 210–214, 2010.
[39] X. S. Yang, "Firefly algorithm," in *Engineering Optimization*, pp. 221–230, 2010.
[40] X. S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization (NICSO'10)*, pp. 65–74, 2010.
[41] X. S. Yang, S. Deb, "Multiobjective Cuckoo search for design optimization," *Computers & Operations Research*, vol. 40, no. 6, pp. 1616–1624, 2013.
[42] X. S. Yang, S. Deb, "Cuckoo search via *lévy* flights," *IEEE World Congress on Nature & Biologically Inspired Computing (NaBIC'09)*, pp. 210–214, 2009.
[43] R. Zhao, W. Tang, "Monkey algorithm for global numerical optimization," *Journal of Uncertain Systems*, vol. 2, no. 3, pp. 165–176, 2008.

Jie Liu obtained his master degree in School of Electronic and Information Engineering from Harbin Institute of Technology. Jie Liu is a full professor of the Kexin software college at Shenyang Normal University. He is also the Dean of Kexin software college. From 1990 to 1991, he was a visiting scholar at the school of computer engineering, university of Genoa, Italy. From 1999 to 2000, he received an MBA degree from Roosevelt university in the United States. He has research interests in wireless networks, mobile computing, cloud computing, social networks and network security. He had published more than 20 international journal and conference papers on the above research fields. Email: nan127@sohu.com.

Lin Teng received the B.Eng. degree from Shenyang Normal University, Shenyang , Liaoning province, China in 2016 . Now, she is working as a Master Degree Candidate in Shenyang Normal University. Her research interests include Multimedia Security, Network Security, Filter Algorithm and Data Mining.

Email:910675024@qq.com.

Shoulin Yin received the B.Eng. And M.Eng. degree from Shenyang Normal University, Shenyang, Liaoning province, China in 2013 and 2015 respectively. His research interests include Multimedia Security, Network Security, Filter Algorithm and Data Mining. He received School Class Scholarship in 2015. Email: 352720214@qq.com.

Flower Classification and Its Physiological and Psychological Relaxing Effects of Viewing Flowers for Sportive

Diaa Salama Abdul Minaam[1] and Nermin Rafiq Mohammed[2]
(Corresponding author: Diaa Salama Abdul Monem)

Faculty of Computers and Information, Benha University, Egypt[1]
Benha city , Qalubiya Governorate, Arab Republic of Egypt
(Email: diaa.salama@fci.bu.edu.eg)
Faculty of Physical Education, El Sadat University, Egypt[2]
(Received Jan. 29, 2018; revised and accepted Apr. 11, 2018)

Abstract

Flowers have played a key role all throughout human history. Some types of flowers were used as food and continue to be used as food until the present day; others have healing properties and are used as medicine. For Psychological sportive also used flower for relaxation by creating positive energy for sportive. Early man also recognized the aesthetic function of flowers; in fact, some ancient cultures believed that they could appease the spirits of their ancestors by decorating their tombs with a particular type of flower. The most important function that flowers have, however, is their power to convey deep human emotions especially for sportive. the physiological relaxing effect brought by nature is becoming clear; however, many workers find it difficult to be exposed to nature in their working environment. So, we work to classify flower dataset. the problem concerns the identi?cation of IRIS plant species on the basis of plant attribute measurements. Classification of IRIS data set would be discovering patterns from examining petal and sepal size of the IRIS plant and how the prediction was made from analyzing the pattern to form the class of IRIS plant and comparing between accuracy results and ratio of errors. By using this pattern and classification to simplify the problem of classification naive Bayes, decision tree, support vector machine and PSO are being introduced. This paper focuses on IRIS plant classification using Nave Bayes, decision tree and SVM (kernel=radial, polynomial, sigmoid) and compare between accuracy and error ratio of this algorithms.

Keywords: Decision Tree; Flower Classification; Naive Bayes; PSO; Psychological Relaxing; SVM

1 Introduction

In modern society, many sportive spend the majority of their time in intensely stressful states and have no time for contact with nature outside of their immediate surroundings. Ever increasing development and job pressures have resulted in an overly stressed society far removed from possible calming effects of nature.

Previous studies have evaluated the psychological stress for sportive [5], and they have shown that stressors for players not only cause psychological symptoms but also increase risk of heart disease [4] and conditions such as sleeplessness [8].

In our stressful modern societies, the relaxing effect of natural stimuli is considered advantageous compared with other stimuli. Many people are thus attracted to the physiological and psychological relaxing effect of exposure to nature [7, 9–11, 13, 14]. The flower that holds the most obvious meaning is the rose, the eternal symbol of romantic love and desire. The most important function that flowers have, however, is their power to convey profound human emotions and thoughts the way no other object can. Whenever your feelings are too intense for words - be it love, grief, happiness, or gratitude - try expressing them through the gift of flowers. You will find that, it will express your feelings perfectly. Besides conveying messages, flowers can also affect our moods and influence our emotions. the effect of flowers on sportive was conducted as follows.

1) Flowers have an immediate impact on happiness. Research participants expressed "true" or "excited" smiles upon receiving flowers, demonstrating extraordinary delight and gratitude.

2) Flowers have a lasting positive effect on moods. Those who participated in the study reported feeling less depressed, anxious or agitated after receiving flowers. Additionally, they exhibited a greater sense of enjoyment and life satisfaction.

3) Flowers tended to make intimate connections between people. The presence of flowers in the surroundings of individuals led to increased contact with family and friends.

4) Flowers evoke feelings of compassion. Study participants who had fresh-cut flowers in their home environment felt an increase in feelings of compassion and kindness for others. This true even when the lowers were in the home for a relatively short time.

5) Flowers relieve "the blues". Simply put, study participants felt less negative after being around lowers at home. Flowers placed in rooms frequented in the morning after arising can help to dissipate the "morning blahs", they concluded.

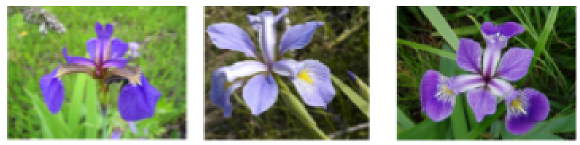
6) Flowers give people a physical and psychological "lift". Increases in energy, happiness and enthusiasm at work were reported by those who had flowers in their home environment.

Perhaps the positive effect flowers have on sportive because Flowers create happiness, it Boost creativity, and it reduce stress and anxiety so we work to make classification on IRIS flower dataset.

Classification consists of assigning a class label to a set of unclassi?ed cases there are two types of classification are supervised which is The set of possible classes is known in advance and unsupervised which is Set of possible classes is not known. After classification we can try to assign a name to that class. Unsupervised classification is called clustering. Machine learning is the science of getting computers to act without being explicitly programmed. Machine learning is so pervasive today that you probably use it dozens of times a day without knowing it. Neural network simulations appear to be a recent development. However, this field was established before the advent of computers, and has survived at least one major setback and several eras. Deep learning (also known as deep structured learning or hierarchical learning) is part of a broader family of machine learning methods based on learning data representations, as opposed to task-specific algorithms. Learning can be supervised, semi- supervised unsupervised [2, 3].

2 Iris Flower Dataset

The iris flower dataset consist of 50 samples from each of three species of Iris (Iris setosa, Iris virginica and Iris ersicolor) (See Figure 1). Four features were measured from each sample: The length and the width of the sepals and petals, in centimeters. We try to classify flowers by using Nave Bayes, decision tree and SVM (Kernel = Radial, Polynomial, Sigmoid) to obtain more exact results and compare between this algorithms.



(a) (b) (c)

Figure 1: Types of flowers: (a) Iris setosa; (b) Iris virginica; (c) Iris ersicolor

2.1 Nave Bayes Classifier

In machine learning, naive Bayes classi?ers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features.

For each target value of $c_i(c_i = c_1, \cdots, c_l)$ $P(c_i) \longleftarrow$ estimate $P(c_i)$ with examples in S.

For every feature X_{ik} of each feature $X_j(j = 1, \cdots, F;\ k = 1, \cdots, N_j) P(X_j = X_{ik}|C_i) \longleftarrow$ estimate $P(X_{jk}|C_i)$ with examples in S.

In test data set $[P(a_1|C)\cdots P(a_n|C]P(c) > [P(a_1|c_i)\cdots P(a_n|C_i)]p(c_i)$, $c_i = c$, $c_i = c_1 \cdots c_l$.

2.2 Decision Tree Classification

Decision tree builds classification or regression models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed the final result is a tree with decision nodes and leaf nodes.

A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values (homogenous). ID3 algorithm uses entropy to calculate the homogeneity of a sample. If the sample is completely homogeneous the entropy is zero and if the sample is an equally divided it has entropy of one (See Figure 2).

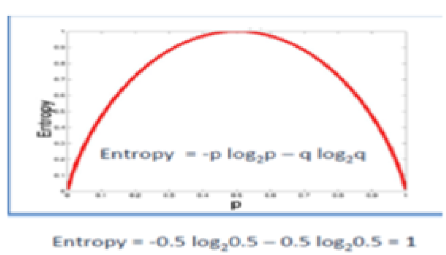


Figure 2: Entropy of decision tree classification

2.3 Support Vector Machine

A support vector machine (SVM) constructs a hyper plane or set of hyper planes in a high dimensional space, which can Be used for classification, regression, or other tasks like outliers detection intuitively, a good separation is achieved by the hyper plane that has the largest distance to the nearest training data point of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier [1, 6, 12].

3 Results of Iris Flower Dataset

Naive Bayes Classifier:

Table 1 shows the result of naive Bayes algorithm by using Weka tool. In naive, we compute number of attributes and class label to classify points of flower as flower type (v iris color, vergencia, setosa) then compute accuracy to know error ratio of this algorithm by using confusion matrix in Table 2.

The way to compute the rate of error from confusion matrix is $\longrightarrow$ Assume classification = Absolute Error = 1 - sum(diagonal)/sum(table) $\longleftarrow$ 1 - (50 + 48 + 46)/ (50 + 48 + 46 + 2 + 4) = 1 - 0.96 = 0.04 (See Table 3).

Table 1: Naive Bayes classifier

Attribute	Class IRis-Setosa (0.33)	 IRis-Versicolor (0.33)	 IRis-Virginica (0.33)
Sepallength			
mean	4.9913	5.9379	5.5795
std. dev.	0.3550	0.50420	0.6353
weight sum	50	50	50
precision	0.1059	0.1059	0.1059
Sepalwidth			
mean	3.4015	2.7687	2.9629
std. dev.	0.3925	0.3038	0.3088
weight sum	50	50	50
precision	0.1091	0.1091	0.1091
petallength			
mean	1.4694	4.2452	5.5516
std. dev.	0.1782	0.4712	0.5529
weight sum	50	50	50
precision	0.1405	0.1405	0.1405
petalwidth			
mean	0.2743	1.3094	2.0343
std. dev.	0.1096	0.1915	0.2646
weight sum	50	50	50
precision	0.1143	0.1143	0.1143

Table 2: Detailed accuracy by class

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1	0	1	1	1	1	1	1	Setosa
	0.96	0.4	0.923	0.96	0.941	0.911	0.992	0.983	versicolor
	0.92	0.02	0.958	0.92	0.939	0.91	0.992	0.986	virginica
weight Avg	0.96	0.02	0.96	0.96	0.96	0.94	0.994	0.989	

Decision Tree:

In decision tree make classification by using entropy and information gain then split max information gain as decision node and repeated until built tree consist of leaf and decision node. Figure 3 shows the result of decision tree which describe dataset that contain 5 (sepal length, sepal width,

Table 3: Naive Bayes confusion matrix

Predicted	Actual		
	Setosa	Versicolor	Virginica
Setosa	50	0	0
versicolor	0	48	2
virginica	0	4	46

petal width, petal length) attribute and 150 instance then split petal width that least than 0.6 as iris versicolor and so on to produce rules of tree then compute accuracy by using confusion matrix.

Rules of Decision Tree:

R1: If (Petal Width) $\leq$ 0.6 THEN species = iris - setosa;

R2: If (Petal Width) $>$ 0.6 And (Petal Width) $\leq$ 1.7 And (Petal Length) $\leq$ 4.9 THEN species = iris - versicolor;

R3: If (Petal Width) $>$ 0.6 And (Petal Width) $\leq$ 1.7 And (Petal Length) $>$ 4.9 And (Petal Width) $\leq$ 1.5 THEN species = iris - virginica[

R4: If (Petal Width) $>$ 0.6 And (Petal Width) $\leq$ 1.7 And (Petal Length) $>$ 4.9 And (Petal Width) $>$ 1.5 THEN species = iris - versicolor;

R5: If (Petal Width) $>$ 0.6 And (Petal Width) $>$ 1.7 THEN species = iris - verginica.

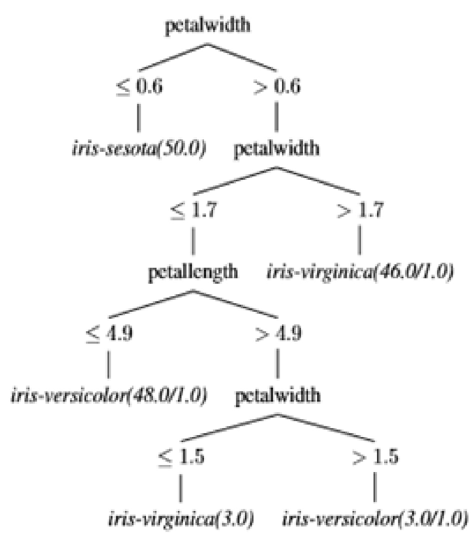


Figure 3: Rules of decision tree

Support Vector Machine:

By using R Studio draw points of dataset then colored each class with different color as shown in Figure 4 then use 3 kernel type of SVM to make classify and compute accuracy.

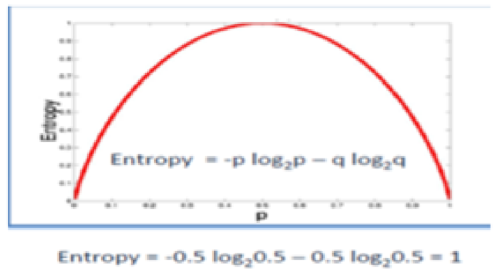


Figure 4: Ploting of Iris dataset

SVM (Kernel = Radial):

This is confusion matrix of support vector machine radial kernel algorithm on accuracy = 0.266 (See Figure 5).

	Actual		
predicted	setosa	versicolor	virginica
setosa	50	0	0
versicolor	0	48	2
virginica	0	2	48

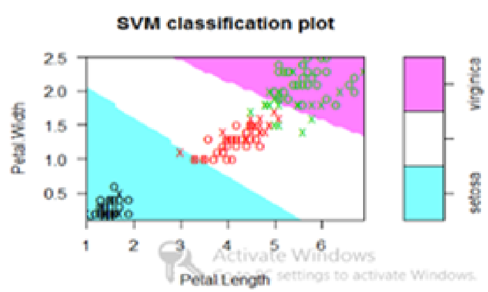


Figure 5: Confusion matrix of support vector machine radial kernel algorithm

SVM (Kernel = Polynomial):

This is confusion matrix of support vector machine polynomial kernel algorithm on accuracy = 0.466 (See Figure 6).

SVM (Sigmoid Kernel):

This is confusion matrix of support vector machine polynomial kernel algorithm on accuracy = 0.1 (See Figure 7).

The best classification is (See Table 4:

$$SVM(Radial\ kernel) \longrightarrow \text{Nave Bayes} \longrightarrow \text{decision tree} \longrightarrow SVM(polynomial) \longrightarrow SVM(Sigmoid). \quad (1)$$

	Actual		
predicted	setosa	versicolor	virginica
setosa	50	0	0
versicolor	0	48	2
virginica	0	2	48

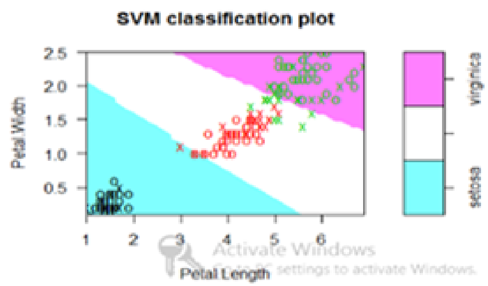


Figure 6: Confusion matrix of support vector machine polynomial kernel algorithm on accuracy = 0.466

	Actual		
predicted	setosa	versicolor	virginica
setosa	50	0	0
versicolor	0	48	2
virginica	0	2	48

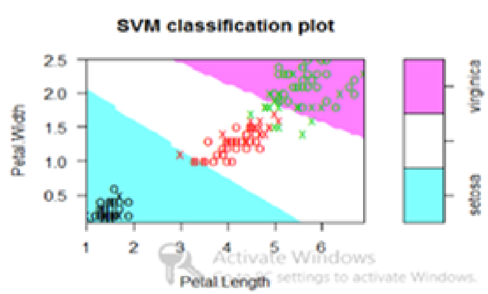


Figure 7: Confusion matrix of support vector machine polynomial kernel algorithm on accuracy = 0.1

Table 4: Classification results

Algorithm	Absolute Error	
Naive Bayes	0.0342	2
Decision Tree	0.035	3
SVM(radial)	0.02666	1
SVM(polynomial)	0.0466	4
SVM(sigmoid)	0.1	5

4 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a computational algorithm which trying to solve problem by improving solution by many regard given solutions. The algorithm solves the problem by having populations and divided it into particles and moving them to search space and performing many mathematical equations over the position. Velocity every particle of swarm movement is effect by 2 things (See Figure 8):

1) The local best position;

2) The best velocity and in every iteration the position and velocity are updated by the other particles. In the final, we reach the best solutions of the swarm.

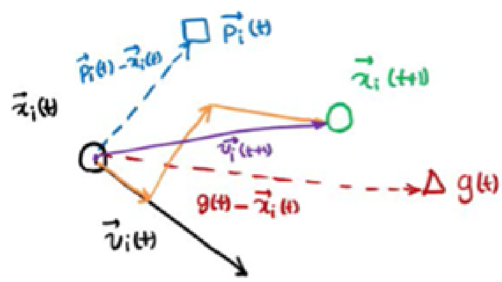


Figure 8: PSO procedures

The PSO Mechanism:

1) Algorithm:

$$V_{ij}(t+1) = WV_{ij}(t) + r_1c_1(P_{ih}(t) - X_{ij}(t)) + r_2c_2(g_j(t) - X_{ij}(t))$$
$$X_{ij}(t+1) = X_{ij}(t) + V_{ij}(t+1).$$

2) Results (See Figure 9):

$Trial >>$ Global best
Global best = struct with field :
$positions : [-0.7206, 5.0267, 2.2930, 1.2321, 5.8240]$
$Cost : 66.4824$
$Trial >>$ Global best.position
$ans : -0.7206, 5.0267, 2.2930, 1.2321, 5.8240$

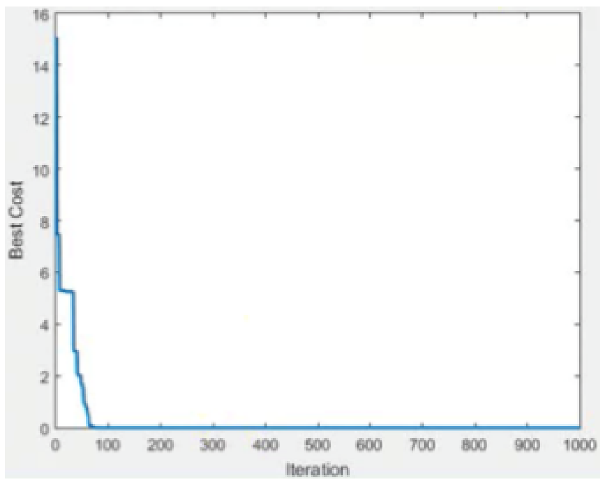


Figure 9: Results of PSO

3) Tools: The software that is used: RStudio, Weka, and Matlab.

5 Conclusion

This paper focuses on IRIS plant classification using Nave Bayes, decision tree and SVM (Kernel = Radial, Polynomial, Sigmoid) and compare between accuracy and error ratio of this algorithms. The problem concerns the identi?cation of IRIS plant species on the basis of plant attribute measurements. Classification of IRIS data set would be discovering patterns from examining petal and sepal size of the IRIS plant and how the prediction was made from analyzing the pattern to form the class of IRIS plant. By using this pattern and classification. then we made PSO algorithm and present Results.

References

[1] A. Chedad, D. Moshou, J. M. Aerts, *et al.*, "Recognition system for pig cough based on probabilistic neural networks," *Journal of Agricultural Engineering Research*, vol. 79, no. 4, pp. 449–457, 2001.
[2] M. Clerc, J. Kennedy, "The particle swarm: Explosion, stability, and convergence in a multi-dimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

[3] O. Hegazy, S. O. Soliman and M. A. Salam, "Comparative study between FPA, BA, MCS, ABC, and PSO algorithms in training and optimizing of LS-SVM for stock market prediction," *International Journal of Advanced Computer Research*, vol. 5, pp. 891–921, 2015.

[4] M. G. Kang, S. B. Koh, B. S. Cha, J. K. Park, S. K. Baik, S. J. Chang, "Job stress and cardiovascular risk factors in male workers," *Preventive Medicine*, vol. 40, no. 5, pp. 583–588, 2005.

[5] N. Kawakami, T. Haratani, "Epidemiology of job stress and health in Japan: Review of current evidence and future direction," *Ind Health*, vol. 37, no. 2, pp. 174-186, 1999.

[6] K. Kirchner, K. H. Tolle, J. Krieter, "The analysis of simulated sow herd datasets using decision tree technique," *Computers and Electronics in Agriculture*, vol. 42, no. 2, pp. 111–127, 2004.

[7] J. Lee, B. J. Park, Y. Tsunetsugu, T. Kagawa, Y. Miyazaki, "Restorative effects of viewing real forest landscapes: based on a comparison with urban landscapes," *Scandinavian Journal of Forest Research*, vol. 24, no. 3, pp. 227–234, 2009.

[8] A. Ota, T. Masue, N. Yasuda, A. Tsutsumi, Y. Mino, H. Ohara, "Association between psychosocial job characteristics and insomnia: An investigation using two relevant job stress models - the demand-control-support (DCS) model and the effort-reward imbalance (ERI) model" *Sleep Medicine*, vol. 6, no. 4, pp. 353–358, 2005.

[9] B. J. Park, Y. Tsunetsugu, H. Ishii, S. Furuhashi, H. Hirano, T. Kagawa, Y. Miyazaki, "Physiological effects of Shinrin-yoku (taking in the atmosphere of the forest) in a mixed forest in Shinano Town," *Scandinavian Journal of Forest Research*, vol. 23, no. 3, pp. 278–283, 2008.

[10] B. J. Park, Y. Tsunetsugu, T. Kasetani, T. Morikawa, T. Kagawa, Y. Miyazaki, "Physiological effects of forest recreation in a young conifer forest in Hinokage town," *Silva Fennica*, vol. 43, no. 2, pp. 291–301, 2009.

[11] B. J. Park, Y. Tsunetsugu, J. Lee, Y. Miyazaki, "Effect of the forest environment on physiological relaxation - the results of field tests at 35 sites throughout Japan," *Forest Medicine*, Editor: Q. Li, pp. 57–68, 2013.

[12] T. Simon, K. Daphne, "Support vector machine active learning with applications to text classification," *Journal of Machine Learning Research*, vol. 2, pp. 45–66, 2001.

[13] Y. Tsunetsugu, J. Lee, B. J. Park, L. Tyrvainen, T. Kagawa, Y. Miyazaki, "Physiological and psychological effects of viewing urban forest landscapes assessed by multiple measurement," *Landscape and Urban Planning*, vol. 113, pp. 90–93, 2013.

[14] Y. Tsunetsugu, B. J. Park, Y. Miyazaki, "Trends in research related to "Shinrin-yoku" (taking in the forest atmosphere or forest bathing) in Japan," *Environmental Health and Preventive Medicine*, vol. 15, pp. 27–37, 2010.

Diaa Salama Abdul Minaam was born on November 23, 1982 in KafrSakr, Sharkia, Egypt. He received the B.S from Faculty of Computers & Informatics, Zagazig University, Egypt in 2004 with grade very good with honor, and obtains master degree in information system from faculty of computers and information, menufia university, Egypt in 2009 specializing in Cryptography and network security. He obtained his Ph.D. degree in information system from faculty of computers and information, menufia university, Egypt in 2015 . He is currently a Lecturer in Information systems department, Faculty of Computers and Information, Benha University, Egypt since 2011. He has worked on a number of research topics .Diaa has contributed more than 30+ technical papers in the areas of wireless networks, wireless network security, Information security and Internet applications, Cloud Computing, Mobile Cloud Computing, Internet of Things, Machine learning in international journals, international conferences, local journals and local conferences. He majors in Cryptography, Network Security, IoT, Big Data, Cloud Computing (Mobile: +201019511000; E-mail: ds_desert@yahoo.com).

Nermin Rafiq Mohammed is a Professor, Department of psychological Sport, Faculty of Physical Education, El Sadat University, El Sadat City, Egypt. She obtain her B.Sc. and M.Sc from Faculty of Physical Education, Helwan University, Cairo, Egypt . she obtain her PhD Sports sciences, Department of sports psychology and management, Faculty of sports sciences, Menufyia University . she is currently a member of Women's Committee Decision, The African Wrestling Federation, The Egyptian Wrestling Federation from, the Technical Committee and follow-up , Egyptian Olympic Committee, and the Olympic Academy for the preparation of the athletes leaders. Prof Nermin has contributed more than 20+ technical papers in the areas of sports psychology, and psychology in international journals, international conferences, local journals and local conferences. He majors in sports psychology (Dr.nerm@gmail.com, nermin.rafiq@phed.usc.edu.eg).

Threat Minimization by Design and Deployment of Secured Networking Model

Shamimul Islam[1], Haidar Ali[1], Ahsan Habib[1], Nur Nobi[1], Mahbub Alam[2], Dulal Hossain[1]
(Corresponding author: Shamimul Islam)

Institute of Computer Science, Bangladesh Atomic Energy Commission, Bangladesh[1]
E-12/A, Paramanu Bhaban, Shahid Shahabuddin Shorok, Agargaon, Dhaka 1207, Bangladesh
(Email: shamimul321@yahoo.com)
Institute of Electronics, Bangladesh Atomic Energy Commission, Bangladesh[2]
(Received Mar. 18, 2018; revised and accepted Apr. 22, 2018)

Abstract

Security is an important issue for organizational network design and development. With an increasing technology in cloud computing sector, enterprise Sector and specific organizational network Infrastructure development, network security always has remained as a great challenge. Our considerable Organization like different scientific/institutional institution faces core security issues challenges in network architecture design and development. A Secured infrastructure of a network always considers or concerns about different security attacks. Network security will prevent a organization network infrastructure from different types of attacks and threats. This paper intends to give an idea to design and deployment of a simple but better network security model and cost effective approach using routers and firewall. This research aim is also that how a network will be protected against vulnerabilities, configuration and security policy weaknesses. Our proposed network infrastructure is adaptable with secure structure.

Keywords: Firewall; Threats and Attacks; VLAN; VPN

1 Introduction

Increasing and overgrowing of networking volumes and internet, the network and information related threats has been risen significantly. These threats are exercised attacks causing damage of a network infrastructure and committing different types of criminal network or cyber activities. Now a days Internet and networking plays a important role in personal, enterprise, organization and different government application. So using network based application and services, Network Security is a major concern [1]. When we concerns about network security terms it includes authorization, identification, authentication and surveillance to the protection of computer hardware or network physical equipment and all other virtual and sophisticated things related with network infrastructure. Threats may arise from mis-configuration of hardware or software equipment, poor network design and deployment, technology weaknesses, or carelessness of end-user [3]. There is no specific laid-down procedure for secure network design and deployment. So Network security must be considered for designing and fit the needs of an organization especially for scientific/institutional organization.

An important secured network deployment and design consideration for today's networks is creating the potential to enhancement for future expansion in a scalable, reliable, and secure way [4]. This paper focuses on the simple but hierarchical network design in which the proposed system will be scalable; better performance and security will be ensured and easy to maintain [7]. It also focuses on review of different types of attacks on routers, switch and its prevention and mitigation procedure. As Routers and firewall are crucial parts of network operations and network security, careful management of router and firewall operations and by avoiding of redundant installation of software and hardware equipments can reduce network downtime, prevent attacks, and proceed in helping in the analysis of security breaches [8].

2 Literature Review: Network Security, Attacks, Weakness and Threats

As the networks of today are more open and with the increased number of LAN, WLAN and personal computers, Wi-Fi and the Internet are beyond a huge numbers of security risks. Network security a important component in information security because it is responsible for securing all information passed through networked computers [2].

For the protection for hardware, software, and information within a network with a acceptable level administrative and management policy, access controls, hardware and software specific functions, features and operational procedures are required. Network security ensures three fundamental aspects: integrity, confidentiality, availability of information from top to bottom.

Real-world security includes prevention, detection, and response. As no prevention mechanism is perfect. Detection and response are more effective than prevention. We need to address protection different OSI layer protection for a secured networking.

Multilayer firewall is need for different level network security in network OSI layer protection [10]. When we considering or imagine a total secured network architecture then we need to consider the overall security in every layer of a network architecture. The protection in every layer provides a secured system where the end user are satisfied and secured within the network. So before designing a network we must be introduced with every layered function for ensuring a robust security firewall against total architecture (See Table 1).

Table 1: Multilayer activities

OSI Layer	Areas
Application Layer	- OS and Application level threats - Application-level gateway
Presentation	encryption
Session	Socks Proxy server
Transport	- Packet filtering -TCP/UDP Flooding.
Network	-NAT/PAT -IP Alternation and DHCP attacks
Data link & Physical	MAC Address alternation, physical port, Traffic Flow changing *etc.*

Firewall security provides centralization, Identify weak points in security system so it can be strengthened, identify intruders so they can be apprehended, provide for authentication, and contribute to a VPN. In Transport layer packet filtering firewalls scan network data packets looking for compliance with, or violation of, rules of firewall's database. Restrictions most commonly implemented in packet filtering firewalls are based on IP source and destination address, Direction (bound and unbound) and TCP or UDP source and destination port [6] network-level proxy; convert IP addresses of internal hosts to IP address assigned by firewall. NAT uses pool of valid external IP addresses, assigning one of these actual addresses to each internal computer requesting an outside connection. Application gateway Frequently installed on a dedicated computer and known as application-level firewall, proxy server, or application firewall. It also can control applications inside a network that access the outside world by setting up proxy services. This security techniques covers: 1) IP address mapping; 2) URL filtering; 3) Content filtering.

Security Weakness and Threats Issues in Network:

In network security every network and device consists of its weak points which are inherently described as its weakness. The weakness can be categorized into 3 ways:

1) Technological Weaknesses: Protocol related weaknesses like TCP/IP protocol, network equipment and Operating System weaknesses.
2) Configuration Weaknesses: it concerns with correctly configuration of computing and network devices. Some configuration weakness is: unsecured user accounts which deals with transmission of insecure user account information over network. System accounts with easily guessed passwords. Misconfigured internet devices such as turn on JavaScript on web browser, Misconfigured network equipment can cause a large security hole.
3) Security Policy Weaknesses: It can create unforeseen security threats. It may consists of points like lack of written security policy, logic access control not applied, software and hardware installation and changes which do not follow the proper policy and lack of disaster recovery plan existence [9].

Network Security:

It faces potential threats generally. The threats may include: 1) Passive attack; 2) Active attack; 3) Distributed attack; 4) Insider attack; 5) Close in attack; 6) Phishing attack; 7) Hijack attack; 8) Spoofing attack; 9) Buffer overflow; 10) Exploit attack; 11) Password attack; 12) Session hijacking attacks; 13) TCP SYN attack; 14) Smurf attacks; 15) Routing attacks; and 16) Masquerade attacks, *etc.* Explanations of some attacks are given here:

1) Denial of Services(DoS): DoS often attacks the specific target by traffic flooding. An attacker tries to prevent actual users from accessing information or services. Flood attacks occur by receiving too much traffic and it cause the server to buffer, causing them to slow down and stop. In this attack, the specified server does not know from which actual port the request are sending. Attacker overloads the server with requests. Buffer overflow attacks, ICMP flood, SYN flood are popular DoS attack. Again DDoS(Distributed Denial of Services) attack occurs when multiple systems target a synchronized DoS attack to a single target. Figure 1 shows a general server connected with specific address with different IP address.

   Figure 2 shows after DoS attack server for IP flooding.
2) ARP Spoofing Attacks (See Figure 3): This kind of attack occurs over a Local Area Network (LAN). Generally ARP Protocol translates IP addresses into MAC addresses. This attack occurs when malicious ARP packets are sent to a default gateway on a LAN to change the pairings in its IP to MAC address table. ARP Poisoning attacks are easy to carry out as the attacker has the control of a machine within the target LAN or is directly connected to it [5].

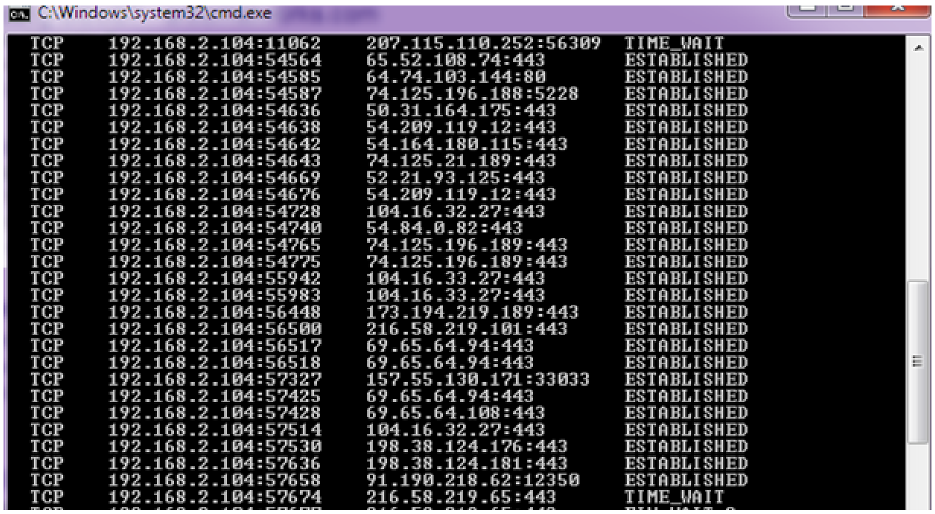


Figure 1: A general server connected with specific address with different IP address

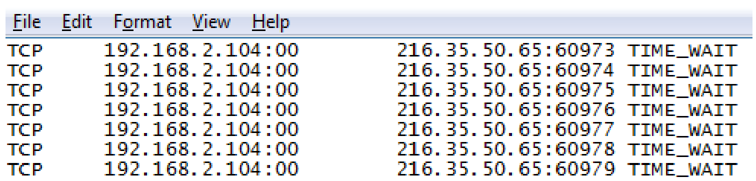


Figure 2: After attack of DoS for IP flooding

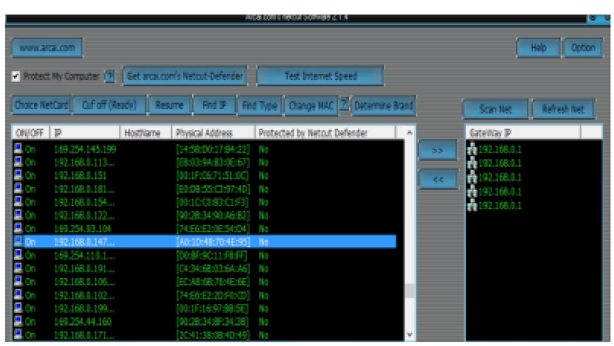


Figure 3: ARP spoofing

3) Session Hijacking Attacks: In this kind of attacks, attacker insert falsified IP packets after session establishment, sequence number alternation and prediction.

4) Man-in-the-Middle Attacks (MITM): This kind of attacks rely on ARP spoofing for intercepting and modify traffic.

3 Building a Secure Network by Decreasing Familiar Attacks

To design a Secure Network we need to define the security function against a network intrusion by decreasing the familiar attacks. When we consider secure but cost effective way for designing and deployment a network infrastructure we must have to analyses the feasibility both. To design a cost effective secured organizational network system here we proposed the steps generally: 1) Implementation of both Hardware and software firewall. 2) Establishment of Virtual Private Network (VPN) for one or more network connection. 3) Implementation of MAC-Bindings with IP addresses registration on server side for network broadcasting. 4) For Wi-Fi Router establishment within same server use the Authentication server for end user identification and tracking. 5) Synchronizing, Observing and tracing of internal gateway router and core router and core switch. 6) VLANs (Virtual LAN) creation.

1) Implementation of Both Hardware and Software Firewall: Firewall concept is adapted to centralize access control. It works as the gatekeeper between the untrusted Internet and the more trusted internal networks. A firewall may be software and hardware firewall. Firewalls provide several types of protection:

   a. It blocks unwanted traffic.

   b. It can direct incoming traffic to more trustworthy internal systems.

   c. It hides vulnerable systems which aren't easily be secured from the Internet.

   d. It can log traffic to and from the private network.

   e. It can hide information like system names, network device types, network topology and internal user ID's from the Internet.

   A hardware firewall is in Figure 4.

2) Establishment of Virtual Private Network (VPN) for One Or More Network Connection (See Figure 5): A Virtual Private Network (VPN) extends a private network across a public network, such as the Internet. By establishing a virtual point-to-point connection through the use of virtual tunnelling protocols, dedicated connections, or traffic encryption a VPN is created. For encrypting and secure user different VPN protocol are used such as IPsec, TLS (Transport layer Security), SSL (Secure Socket Layer), Open VPN or Point-to-Point Tunnelling Protocol, *etc.* Common uses of the organization VPN include access to file sharing/shared drives.

3) Implementation of MAC-Bindings with IP Addresses Registration on Server Side for Network Broadcasting (See Figure 6): Binding IP addresses to MAC addresses could avoid IP address changing with reconnection. Once specified device's MAC address and IP address are bound, the IP address will be reserved for the device and the device is easy to trace if any occurrence is happened. It is easy for the Administrator to manage critical devices and a great method to manage the LAN clients.

4) For Wi-Fi Router Establishment within Same Server Use the Authentication Server for End User Identification and Tracking: If we want to keep end user track for Wi-Fi connection where Wi-Fi

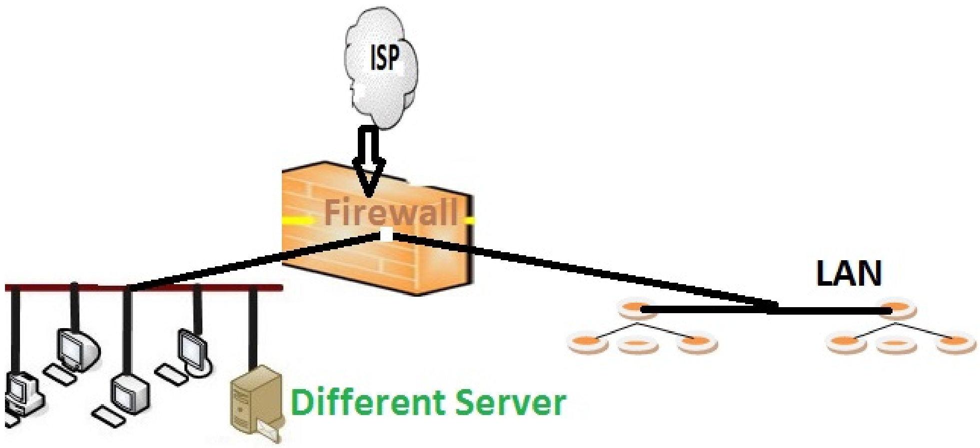


Figure 4: A simple firewall implementation

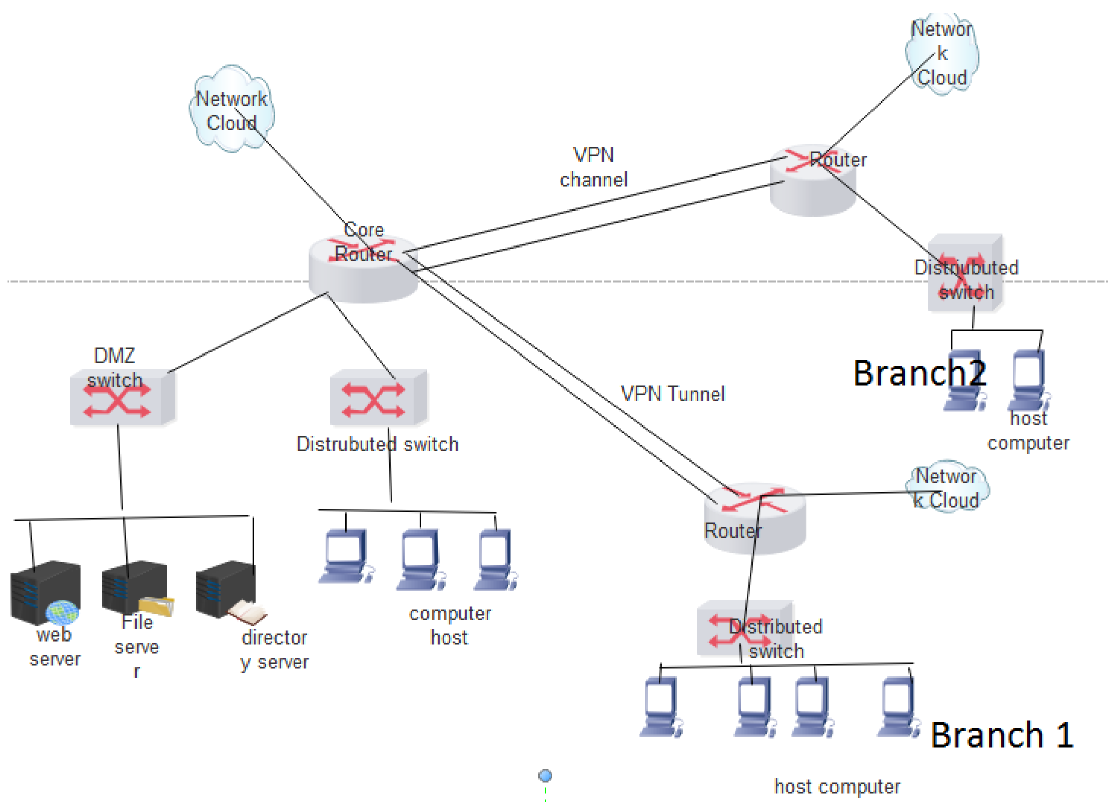


Figure 5: VPN connection for organizational branch

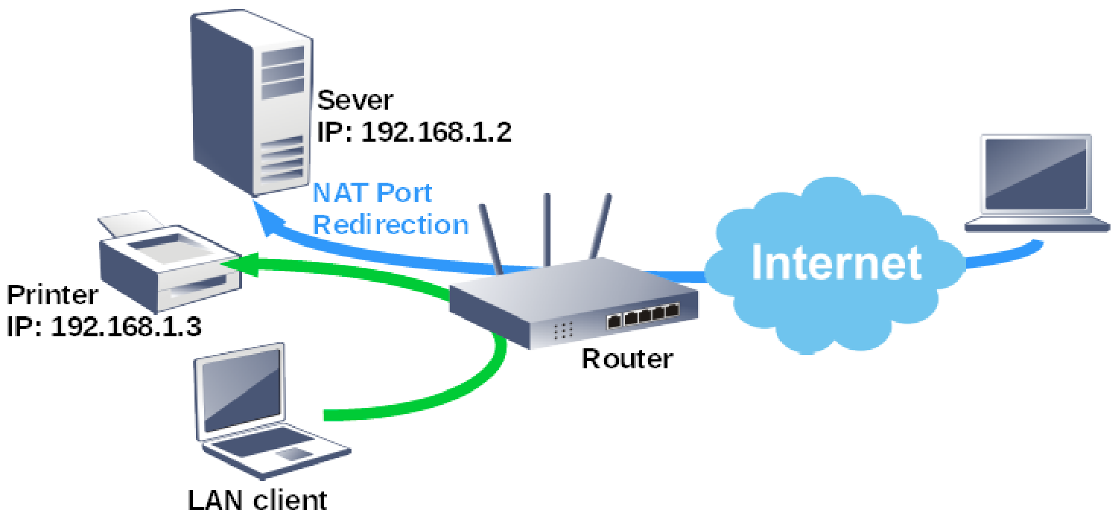


Figure 6: MAC Bindings with IP for secure networking

router and WLAN are configured in same proxy server then we must use an authentication server for user identification and tracking. Radius Server and AAA server are used for authentication, authorization, accountability and user logs filtering. Scalability, security and flexibility are ensured by using those server mechanisms. When the terms wireless network comes into front the Strong authentication, Strong data encryption, Protects broadcast and multicast traffic, User authentication, Secures access to the WLAN instead of just to the packets and Additional network devices required are most considerable things in wireless establishment.

5) Synchronizing, Observing and Tracing of Internal Gateway Router and Core Router and Core Switch. As we know that every internal private network of organization is created from a Real IP that is connected to the real world network like internet. In this sense a core router and switch is designed for creating network. For a secure network design and deployment we need to synchronize, observe and trace of internal gateway router and core router for every user. The other objective we need to consider that monitoring this it is easy to avoid external and internal threat. Flexibility and scalability can be assured through it.

6) VLANs (Virtual LAN) Creation. It works in data link layer. VLAN partitions and isolate a computer network. In practical terms, multiple VLANs are pretty much the same as having multiple separate physical networks within a single organization - without managing multiple switches and cable plants. Because VLANs segment a network, creating multiple broadcast domains, they effectively allow traffic from the broadcast domains to remain isolated. We have suggested some VLANs for better security of campus network and reducing broadcast [11]. Table 2 shows an example of the proposed VLAN creation.

By examine all above the designed structure of the suggested model are as in Figure 7.

Table 2: An example of the proposed VLAN creation

VLAN ID	VLAN Name
1	A
5	D
10	L
15	C
20	F
40	I

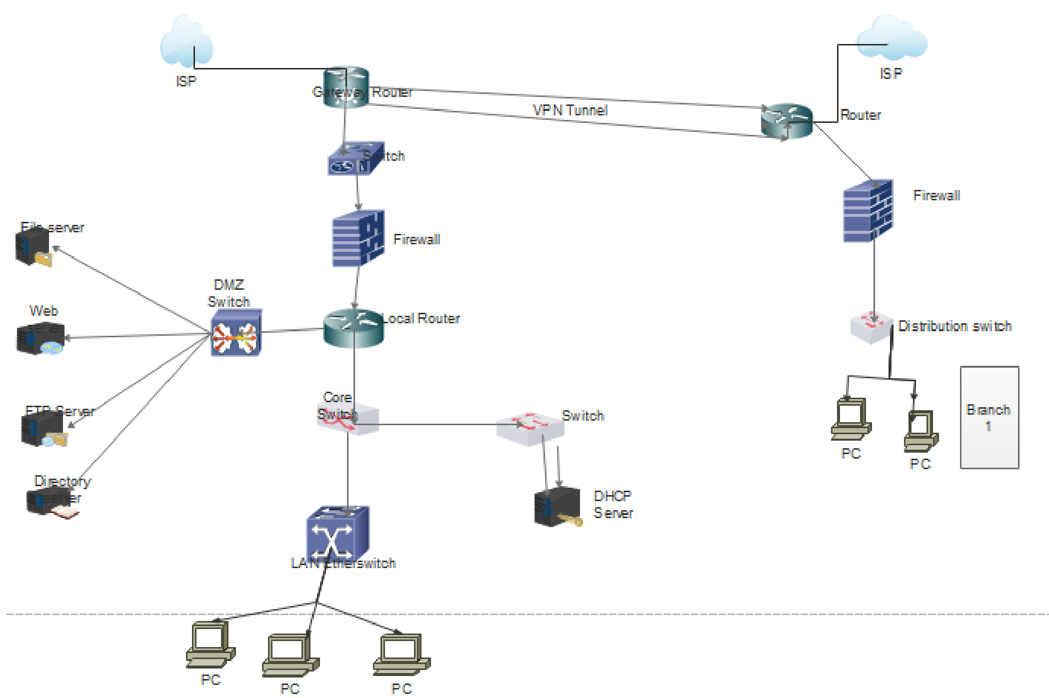


Figure 7: Structure of suggested secure network model

4 Conclusion

When we design the architecture and deployment of a network system, its security becomes an important issue and considerable matter for any organization. So by following the above network design and considering the above discussed security terms, one organization can design and deploy a scalable, better performer and secured network which is easy to maintain. In this discussion and work, we have proposed a cost effective secure network design based on the work environment and required security, scalability and other aspects. The terms we have introduced and considered in designing network architecture are helpful in mitigating the known attacks and reoccurrence attacks.

This paper presented the tips and recommendations to achieve a best security and to protect the network from threats, vulnerabilities and attacks by applying security configurations such on strong routing filtering using router and firewall which can assure a better network security.

References

[1] S. Alabady, "Design and implementation of a network security model using static VLAN and AAA server," in *3rd International Conference on Information and Communication Technologies: From Theory to Applications (ICTTA'08)*, 2008.

[2] S. Alabady, "Design and implementation of a network security model for cooperative network," *International Arab Journal of e-Technology*, vol. 1, no. 2, June 2009.

[3] M. N. B. Ali, M. L. Rahman, S. A. Hossain, "Network architecture and security issues in campus networks," in *Fourth International Conference on Computing, Communications and Networking technologies (ICCCNT'14)*, 2014.

[4] G. Gur, S. Bahtiyar1, F. Alagoz, "Chapter 30: Security analysis of computer networks: Key concepts and methodologies," in *Modeling and Simulation of Computer Networks and Systems Methodologies and Applications*, pp. 861–898, 2015.

[5] S. Keshav, *An Engineering Approach to Computer Networking*, Addison Wesley, 2010.

[6] J. F. Kurose and W. R. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, Pearson, 2000.

[7] S. Al Maskari, D. K. Saini, S. Y. Raut and L. AHadimani, "Security and vulnerability issues in university networks," in *Proceedings of the World Congress on Engineering (WCE'11)*, London, U.K., 2011.

[8] B. Mulyawan, "Campus network design and implementation using top down approach," in *Proceedings of the 1st International Conference on Information Systems for Business Competitiveness (ICISBC'11)*, 2011.

[9] L. L. Peterson and B. S. Davie, *Computer Networks: A System Approach*, Elsevier, 1996.

[10] M. T. Soleimani, M. Kahvand, " Defending packet dropping attacks based on dynamic trust model in wireless ad hoc networks," in *IEEE 17th Mediterranean Electrotechnical Conference (MELECON'14)*, Beirut, Lebanon, pp. 362-366, 2014.

[11] Z. Xiaozhe, W. Zaifang, L. Qiong, F. Guiming, *The Design and Implementation of the Integration between Switches and Routers*, Computer Engineering and Application, 2014.

Md.Shamimul Islam was born in Satkhira, Bangladesh, on 28th October, 1988. He received the B.Sc. and M.Sc Degree in Computer Science and Engineering from the Jahangiranagar University, Bangladesh in 2011, and in 2012 respectively. He worked at Samsung R&D Institute Ltd., Bangladesh from 1 Oct, 2012 to 31 Dec, 2014 as a Software Engineer. From 27th July, 2016 he is being as a Scientific Officer of Institute of Computer Science Division in Bangladesh Atomic Energy Commission (BAEC). He is a member of Bangladesh Computer Society. His research interest includes Network and Cyber Security,

Communication Engineering, IoT, Software Engineering and Human Computer Interaction.

Md. Haidar Ali was born in Kishorgonj, Bangladesh, on 25th January, 1985. He received the B.Sc. Degree in Computer Science and Engineering from the Daffodil International University, Bangladesh in 2010, and in 2012, he received the M.Sc. degree in Computer Science and Engineering from Daffodil International University. He worked at Padma Multipurpose Bridge Project, Bridge Division under Ministry of Road Transport & Bridges, Bangladesh from 14 October 2014 to 05 November, 2016 as a Assistant Programmer. From 15 November 2016 to still now, He is being a scientific officer of Computer Science Division in Bangladesh Atomic Energy Commission. He is a Life Time Member of Bangladesh Computer Society and Also Life Time Member of IEB. His research interest includes Network and Cyber security, communication Engineering.

Md. Ahsan Habib was born in Comilla, Bangladesh. On 05th December, 1984, he received the B.Sc degree in Electrical, Electronics and Telecommunication Engineering from Dhaka International University, Bangladesh in 2010 and in 2014, received the M.Sc degree in Computer Science from Jahangirnagar University, Bangladesh. He worked at Bangladesh Chemical Corporation, Bangladesh from 01 February, 2016 to 14 November as an Assistant Engineer. From 15 November, 2016 he is being a Scientific Officer of Institute of Computer Science in Bangladesh Atomic Energy Commission. He is a member of Bangladesh Computer Society.

Mohammad Nur Nobi was born in Bangladesh in 1987. He received the B.Sc. (Engg.) and M.Sc. (Engg.) degrees from Mawlana Bhashani Science and Technology University, Tangail, Bangladesh, in 2011 and 2015, respectively. He joined Samsung R&D Institute Bangladesh, in 2011. Since 2016, he has been with the Bangladesh Atomic Energy Commission, where he is a Scientific Officer. His main areas of research interest are algorithm, machine learning, artificial intelligence and cyber security. Mr. Nobi is a member of the Bangladesh Computer Society (BCS).

Md. Mahbub Alam was born in a rural area called Dhamrai of Dhaka in 1991. He has completed his B.Sc & M.Sc from the Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh. He worked as a lecturer in Gono University at the Dept. of CSE from January 2015 to February 2016. He worked as an Assintant Engineer in WALTON Group at the Computer R&D section from February 2016 to November 2017. Currently he is working as a Scientific Officer at the Institute of Electronics in Bangladesh Atomic Energy Commission. His research interests include big data analysis, artificial intelligence, pattern recognition and expert system, computer vision, system that can provide distinct service through internet protocol and any system that can be beneficiary for common people. He is also interested in entrepreneurship.

Dr. Hossain has been working at Bangladesh Atomic Energy Commission (BAEC) for more than 18 years. He actively involves in the R&D, and training activities in BAEC. He received his PhD in Innovation Technology & Management, MS in Electrical & IT Engineering from Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea, and BS in Electrical & Electronic Engineering from Chittagong University of Engineering & Technology (CUET), Bangladesh. Hossain's research interest includes; (i) Nuclear innovations/knowledge management (ii) Nuclear cyber security, (iii) Computer security in nuclear facilities. His articles have appeared in several international journals and conferences proceedings as well.

Reviewers (Volume 8, 2018)

Riyadh Abdulamir
Abayomi Agbeyangi
Ozgur B. Akan
Saud Althunibat
Jafar Ahmad Abed Alzubi
Hajar Aroubandi
Majid Bayat
Yu Bi
Zhengjun Cao
Arup Kumar Chattopadhyay
Jui-Chi Chen
Mei-Juan Chen
Chen-Yang Cheng
Yung-Chen Chou
Christos Chrysoulas
Sankhanil Dey
Christo Dichev
Xuedong Dong
Hazem M. El-Bakry
Hagar M. El-Hadad
Salah Elgharib Elgazzar
Hamsa Emad
Nacer Nacer Ghadbane
Abdalaah Hamroush
Wajid Hassan
Dariusz Jacek Jakobczak
Biswapati Jana
Thangakumar Jeyaprakash
Hatem M. Abdul Kader
Marimuthu Karuppiah
Santosh Ravva Kumar
N. Muthu Kumaran
Tzu-Liang Kung
Andrew Kusiak
Wei-Chao Lin
Yang-Bin Lin
Wei-Fu Lu
Diaa Salama Abdul Minaam
Diaa Salama Abdul Monem
Reem Essam Mohamed
Fadiyya Mohsin
Ahmed Mosa
Chinthagunta Mukundha
Inam Muhannad
Faisal Nabi
Asifa Nazir
Amar Pandey
Sachin Paranjape
Ravinder Paspula
S. R. Boselin Prabhu
B. R. Pushpa
Rasoul Ramezanian
Hemraj Saini
Michael Sheng
Jiunn-I Shieh
Yuriy S. Shmaliy
M. Shobana
Rajeev Singh
Thaer Taha
Lin Teng
P. Thanapal
Gulom Tuychiev
S. Umamaheswaran
Jing-Doo Wang
Chia-Chun Wu
Nan-I Wu
Wei-Chiang Wu
Chao-Chen Yang
Cheng-Ying Yang
Yinghui Zhang

Guide for Authors
International Journal of Electronics and Information Engineering

International Journal of Electronics and Information Engineering (IJEIE) will be committed to the timely publication of very high-quality, peer-reviewed, original papers that advance the state-of-the art and applications of Electronics and Information Engineering. Topics will include, but not be limited to, the following: Algorithms, Bioinformatics, Computation Theory, AI, Fuzzy Systems, Embedded Systems, VLSI/EDA, Computer Networks, Information Security, Database, Data Mining, and Information Retrieval, Digital Content, Digital Life, and Human Computer Interaction, Image Processing, Computer Graphics, and Multimedia Technologies, Information Literacy, e-Learning, and Social Media, Mobile Computing, Wireless Communications, and Vehicular Technologies, Parallel, Peer-to-peer, Distributed, and Cloud Computing, Semiconductor, Software Engineering and Programming Languages, Telecommunication, etc.

1. Submission Procedure

Authors are strongly encouraged to submit their papers electronically by using online manuscript submission at http://ijeie.jalaxy.com.tw/.

2. General

Articles must be written in good English. Submission of an article implies that the work described has not been published previously, that it is not under consideration for publication elsewhere. It will not be published elsewhere in the same form, in English or in any other language, without the written consent of the Publisher.

2.1 Length Limitation:

All papers should be concisely written and be no longer than 30 double-spaced pages (12-point font, approximately 26 lines/page) including figures.

2.2 Title page

Title page should contain the article title, author(s) names and affiliations, address, an abstract not exceeding 100 words, and a list of three to five keywords.

2.3 Corresponding author

Clearly indicate who is willing to handle correspondence at all stages of refereeing and publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address.

2.4 References

References should be listed alphabetically, in the same way as follows:

For a paper in a journal: M. S. Hwang, C. C. Chang, and K. F. Hwang, ``An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445--446, 2002.

For a book: Dorothy E. R. Denning, *Cryptography and Data Security*. Massachusetts: Addison-Wesley, 1982.

For a paper in a proceeding: M. S. Hwang, C. C. Lee, and Y. L. Tang, ``Two simple batch verifying multiple digital signatures," in *The Third International Conference on Information and Communication Security (ICICS2001)*, pp. 13--16, Xian, China, 2001.

In text, references should be indicated by [number].

2.5 Author benefits

No page charge is made.

Subscription Information

Individual subscriptions to IJEIE are available at the annual rate of US$ 200.00 or NT 6,000 (Taiwan). The rate is US$1000.00 or NT 30,000 (Taiwan) for institutional subscriptions. Price includes surface postage, packing and handling charges worldwide. Please make your payment payable to "Jalaxy Technique Co., LTD." For detailed information, please refer to http://ijeie.jalaxy.com.tw or Email to ijeieoffice@gmail.com.