ORIGINAL RESEARCH

Swine influenza inspired optimization algorithm and its application to multimodal function optimization and noise removal

Shyam S. Pattnaik[1], Devidas G. Jadhav[1], Swapna Devi[1], Radha Kanta Ratho[2]

1. National Institute of Technical Teachers' Training & Research (NITTTR), Chandigarh, India. 2. Postgraduate Institute of Medical Education & Research (PGIMER), Chandigarh, India.

Correspondence: Shyam S. Pattnaik. Address: National Institute of Technical Teachers' Training & Research (NITTTR), Sector-26, Chandigarh, 160019, India. Telephone: 91-987-287-9362. Fax: 91-172-279-3893. E-mail: shyampattnaik@yahoo.com

Received: March 31, 2012 **Accepted:** May 21, 2012 **Published:** September 1, 2012
DOI: 10.5430/air.v1n1p18 **URL:** http://dx.doi.org/10.5430/air.v1n1p18

Abstract

Swine Influenza Inspired Optimization (SIIO) is a search algorithm proposed for optimal solution. The authors followed the SIR (susceptible - infectious-recovered) virus spread model of Swine Influenza to develop the new evolutionary algorithm named as SIIO. SIR model is used to frame optimization algorithm following the spread and control phenomenon of the swine flu virus in the human population. The fitness based classes viz. susceptible (S), infectious (I) and recovered (R) of the individuals are made and treatment is used for the affected individuals by imitating the health information from the best fitness individual. The proposed algorithm shows improved performance on multi-dimensional unimodal and multimodal standard numerical benchmark functions than the compared optimization algorithms. The performance of the SIIO algorithm is better in terms of speed of convergence and quality of solutions. The SIIO is also applied for the Gaussian noise removal with Blind Source Separation (BSS) based on Independent Component Analysis (ICA).

Key words

Benchmark functions, Swine Influenza Inspired Optimization (SIIO), SIR-model, Independent Component Analysis (ICA), Blind Source Separation (BSS).

1 Introduction

Recent stochastic optimizers that have drawn the attentions include Genetic Algorithm, Differential Evolution, Evolutionary Programming etc. [1-4] where in a population of the solutions is utilized in the search process. These algorithms are capable of exploring and exploiting the promising regions in the search space but take relatively longer time [3]. This paper addresses two issues; one is the development of a new optimization technique Swine Influenza Inspired Optimization (SIIO) using the SIR model of influenza. Second is the application of this algorithm in Blind Source Separation (BSS) using Independent Component Analysis (ICA) for Gaussian noise removal from biomedical signal i.e. Electrocardiogram (ECG). The proposed algorithm is having capability to converge on multimodal functions having higher dimensions. The independent component analysis based blind source separation requires its demixing matrix optimization hence, the proposed algorithm is applied for the said optimization process for noise removal. The paper is organized as follows. In section 2, the proposed SIIO is presented. Section 3 gives the detail about the validation of the SIIO with standard numerical benchmark functions. Section 4 discusses about the ICA. Section 5 discusses the application of the proposed algorithm for ICA function maximization, while section 6 concludes the paper.

2 Swine influenza inspired optimization (SIIO)

Swine flu spreads just as the other seasonal flu [5-9]. It spreads from person to person via close contact and touch, or through respiratory droplets carrying the virus or from person to environmental surfaces by cough and sneeze. Wherever a person with swine flu touches the surfaces, infectious body fluid having effectiveness within specific duration remains on the surfaces and then gets transferred to the other persons. The swine flu is having the speed in spread of infection and instating capacity also.

Early in the 20th century Kermack and McKendrick [10] developed the first mathematical based model describing an influenza epidemic [6]. Since then various mathematical models have been proposed to limit the spreading of flu [5-9, 11, 12]. Virtual experiments have been carried out using simulation for worldwide outbreaks of influenza epidemic on a computer. The basic model is known as the Susceptible-Infectious-Recovered (SIR) model [10] and is shown in Figure 1. In this, the initial population is divided into the classes: Susceptible (S), Infectious (I), and Recovered (R). A susceptible class of swine influenza-A (H1N1) virus infection is of person having acute febrile respiratory illness with onset within 7 days of close contact with a person who is a confirmed case of swine influenza A (H1N1). Case of swine influenza A (H1N1) virus infection is confirmed with medical laboratory confirmation test. Depending on the test of confirmation of infection, individuals proceed from class S i.e. Susceptible to class I i.e. Infectious [10]. After the proper treatment with recommended medicinal dose, the recovered individual moves to class R i.e. Recovered, at that stage the individual acquires immunity to future infection. The infectious individual from class I transmits influenza to each susceptible individual in the class S. The researchers extended SIR model to represent and/or predict the spatial dynamics of an influenza epidemic. Rvachev [11] devised the first spatio-temporal model of influenza in the 1960s. This model was devised to understand the temporal and spatial synchrony of influenza epidemics [11]. Later the researchers modified the basic SIR model by including seasonality, vaccination, treatment etc. [6]. The SIR model extended with incorporation of vaccination and treatment [6] is shown in Figure 2. The individuals from class S can be vaccinated and infectious individuals from class I are treated with antiviral drugs. During the treatment phase, they are transferred into the class T [6]. Individuals from class S who are vaccinated are taken into class V and they are considered immune. After the treatment, individuals from class T go to class R due to recovery [6].

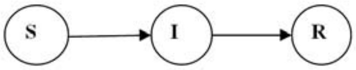


Figure 1. Basic Susceptible-Infectious-Recovered model of disease transmission

The SIIO is a new family of the optimization techniques, which is based on the extended SIR model. It shows optimization through vaccination and treatment based on probability. As mechanism of spreading the infection among the population is probabilistic, the mechanism of the control of the epidemic is also probabilistic. This concept is utilized to frame an optimization algorithm with spread and control of the swine flu virus in the population because every transition in the classes is probabilistic. These transitions are considered as optimization operations or happen in optimization steps. In these transitions or state change of individuals, the best individual can be used as reference. This is the fitness value that is taken as reference to decide the tag of the class for individuals and individual having this value is used for spreading health among the individuals. Other individuals try to enhance their health by imitating the health information like preventive care practiced, medicinal information etc. of the best individual. The individuals having best health are preserved during the process of optimization. In this mechanism ill individual tries to drag the population towards the viral spread i.e. poor fitness whereas healthier ones towards better health or optimum value during optimization process. Based on this logic, the SIIO is devised as follows.

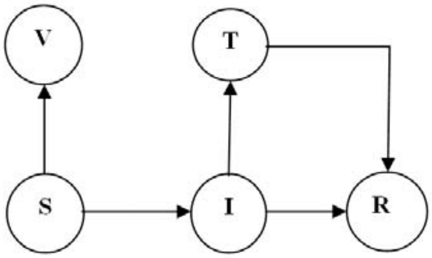


Figure 2. Extended Susceptible-Infectious-Recovered Model

2.1 Key terms and definitions

Population: Group of Individuals

Individual: Member of the population

Susceptible Individual (S): Member of population assumed as Susceptible and is in class S based on the fitness value obtained by test.

Infectious Individual (I): Individual infectious by swine flu virus and is in class I based on the fitness value obtained by test.

Recovered Individual(R): The individual having best health among the population based on the fitness value obtained by test. The immune property is realized by preserving individual during the process of optimization.

Days: It indicates state of each individual during various days. This counts to be equal to number of variables in the optimization. So the state of each individual is the dimension in the process of optimization.

When all the states of an individual are best, having optimum or near optimum fitness value then the solution is called the optimized or converged.

Pandemic-health (PH): Best fitness value among the entire individuals found up to current generation.

Swine Flu Test: It is test to check the health of the individuals which gives the values of fitness. These fitness values are used to classify the individuals.

Dose: Antiviral drugs given to individuals based on the current-health, pandemic-health and history of the dose.

2.2 Model of SIIO

The optimization model SIIO is presented in Figure 3. It does the optimization through treatment processes based on probability. Initially all individual except those having best fitness values due to vaccination are assumed as susceptible because one infectious individual in population may lead the total population towards suspected state. Then all the individuals undergo the swine flu test, which is nothing but declaring the current health (fitness) of individuals.

The individuals are now tagged with the status class R for recovered or healthy, class S for suspectible and class I for infectious with the help of appropriate optimization conditions based on fitness. The infectious individuals make all other susceptible to illness by viral spread. In this case, the preventive mechanisms like treatment, quarantine etc. are used. The recovered or healthy individual is taken as the reference individual for the others in the process of optimization and treatment process is not operated on such individual. The suspectible and infectious individuals are quarantined as per equation (1). Quarantine is based on probability and threshold given in equation (2).

$$P(t+1) = P(t) \times (1.5 \times rand - 0.25) \tag{1}$$

$$DT = CF(2) - 0.05 \times (CF(2) - CF(1)) \tag{2}$$

$P(t+1)$ is new individual value calculated by changing the current value $P(t)$. DT is dynamic threshold and CF is current fitness or current health of individual. CF(1) is current best individual health and CF(2) is next better fitness. The suspected individuals are treated by changing the current value based on probability. The treatment is given to individuals in population with varying amount of antiviral drugs. The dose of antiviral drugs is dependent on current health, pandemic health and changes adaptively based on the difference in the current individual value and best individual value. As the health of individual varies, dose quantity also changes as given in equation (3) and individual treatment is done by equation (4).

$$Dose = c \times Dose + rand \times (P(t) - P(1)) \tag{3}$$

$$P(t+1) = P(t) + \text{Dose} \tag{4}$$

$P(t)$ is current individual value and $P(1)$ is best individual value. $P(t+1)$ is new individual value calculated by changing the current value $P(t)$.

The SIIO performs optimization in four steps as given below. These steps are continued for all generations.

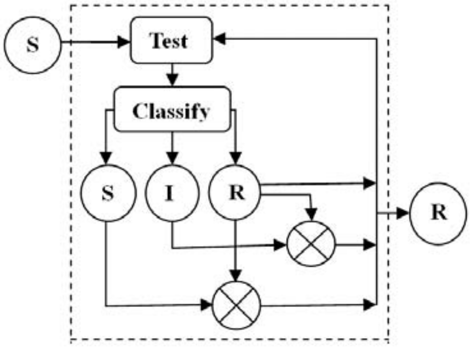


Figure 3. Model Of SIIO

2.2.1 Steps in SIIO

Step 1- Evaluate health of all individuals: The health of individual is calculated by fitness function.

Step 2- Classify individuals into S, I and R: Individuals are tagged with class S, class I and class R. The best current health individuals are considered as recovered and are in class R. When individuals are arranged according to health, better half of the individuals are considered as suspected individuals and are in class S and other half as infectious individuals and in class I.

Step 3- Quarantine of individual: The Individuals in class S and class I are isolated and they are not a part of the population during outbreak period. This is realized by changing the state of the individual randomly with the small value of probability of quarantine.

Step 4- Treatment of Individual by dose: The amount of antiviral drug depends on current health, pandemic health and history of the dose. The dose is to change the health or fitness of individual as per probability of treatment.

2.2.2 Pseudo Code of SIIO

```
Initialize parameters: G, IN, D, Pq, Pt, Dose, P (D x IN)
for i=1:G
    for j = 1 : IN
        CF (j) =Evaluate Fitness Function
    end
    Update P and PH
    Sort individual in order of ascending health
    $DT = CF(2) - 0.05 \times (CF(2) - CF(1))$
    for k=1: IN
        if k==1
            tag individual as best health (R)
        elseif  k<IN/2
            tag individual as sucpected (S)
        else
            tag individual as infectious (I)
        end
        if (CF(k)-DT)>0 & rand<Pq& P(k) $\subset$ S,I
            for l=1:D
                P(l,k)=P(1:k) x (1.5 x rand-0.25)
            end
        end
        if rand <Pt& P(k) $\subset$ S
            Dose=c x Dose + rand x (P(k)-P(1))
            P(k)=P(k) + Dose
        end
    end
end
```

where G is number of generations, IN is size of population, D is dimension of population, Pq is probability of quarantine, Pt is probability of treatment, Dose is amount of dose for treatment, P (D x IN) is population of size D by IN and CF is current fitness.

2.3 Optimization by SIIO

The population is real coded for the algorithm, and it is obtained by generation process. The generation process generates the uniformly distributed random population in the specified range for functions shown in Table 1. Table 1 gives the details of the standard numerical benchmark functions, where D is the dimension of the function, S is the range of the function and fmin is the minimum value of the function [4]. The fitness of the population is determined and the current best individual is found out for the best health (elite) preservation.

In the SIIO algorithm, fitness is calculated and the state of each individual is decided as class S, class I, or class R and they are tagged accordingly. Then, according to the threshold and state of the individual, the individuals undergo quarantine and treatment processes as per quarantine probability and treatment probability respectively. In quarantine as well as the treatment processes, the current best individual in class R among the population is taken as the reference. The dose used for treatment is calculated depending on three parameters viz. current best individual, individual for which dose is being calculated and previous dose for the same individual. Then the dose is applied to the individual for recovery. The best individual is found out in all iterations during the optimization process. The SIIO is probability based and population-based algorithm hence, the possibility of avoiding local minima is higher. It strengthens the algorithm for convergence of both unimodal and multimodal functions, which are given in the Table 1. The performance of the algorithm is checked with 13 standard benchmark functions as given in Table 1. The proposed algorithm is proved on unimodal and multimodal standard benchmark functions and the performance is noted by determining accuracy, convergence speed and success rate.

Table 1. Standard Numerical Benchmark functions used for experimental study [4]

Function (F)	NFFE	S	f_{min}
$f_1 = \sum_{i=1}^{d} x_i^2$	150000	$[-100,100]^D$	0
$f_2 = \sum_{i=1}^{d} \lvert x_i \rvert + \prod_{i=1}^{d} \lvert x_i \rvert$	200000	$[-10,10]^D$	0
$f_3 = \sum_{i=1}^{d} (\sum_{j=1}^{i} x_j)^2$	500000	$[-100,100]^D$	0
$f_4 = \max_i \{\lvert x_i \rvert, 1 \le i \le d\}$	500000	$[-100,100]^D$	0
$f_5 = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	500000	$[-30,30]^D$	0
$f_6 = \sum_{i=1}^{d} [\lvert (x_i + 0.5) \rvert]^2$	150000	$[-100,100]^D$	0
$f_7 = \sum_{i=1}^{d} i x_i^4 + random\,[0,1)$	300000	$[-1.28,1.28]^D$	0
$f_8 = 418.9829 \times D - \sum_{i=1}^{d} (x_i \sin(\sqrt{\lvert x_i \rvert}))$	300000	$[-500,500]^D$	0
$f_9 = \sum_{i=1}^{d} (x_i^2 - 10 \cos(2\pi x_i) + 10)$	300000	$[-5.12,5.12]^D$	0
$f_{10} = -20 \exp\left(-0.2 \sqrt{\frac{1}{d} \sum_{i=1}^{d} x_i^2} \right) - \exp\left(\frac{1}{d} \sum_{i=1}^{d} \cos(2\pi x_i) \right) + 20 + \exp(1)$	150000	$[-32,32]^D$	0
$f_{11} = \frac{1}{4000} \sum_{i=1}^{d} x_i^2 - \prod_{i=1}^{d} \cos(\frac{x_i}{\sqrt{i}}) + 1$	300000	$[-600,600]^D$	0
$f_{12} = \frac{\pi}{d} \left\{ 10 \sin^2(\pi y_1) + \sum_{i=1}^{d-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\}$ $+ \sum_{i=1}^{d} u(x_i, 10, 100, 4) \quad where \ y_i = 1 + \frac{1}{4}(x_i + 1). \ u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i \succ a \\ 0, & -a \le x_i \le a \\ k(-x_i - a)^m, & x_i \prec a \end{cases}$	150000	$[-50,50]^D$	0
$f_{13} = 0.1 \left\{ \sin^2(\pi 3 x_1) + \sum_{i=1}^{d-1} (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + (x_n - 1)^2 [1 + \sin^2(2\pi x_d)] \right\}$ $+ \sum_{i=1}^{d} u(x_i, 5, 100, 4) \ where \quad u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i \succ a \\ 0, & -a \le x_i \le a \\ k(-x_i - a)^m, & x_i \prec -a \end{cases}$	150000	$[-50,50]^D$	0

3 Validation of SIIO on standard benchmark functions

3.1 Parameter setting and experimentation

For the proposed SIIO, a population of 100 individuals of real-valued representation is used. In the SIIO algorithm, the quarantine and treatment probabilities are taken as 0.3 and 0.8 respectively. Best individual value and its health are preserved as current_best_individual and current_best_health respectively. Number of fitness function evaluations

(NFFE) as shown in Table 1 is considered as one of the stopping criterion. Once the error in fitness value falls below 10^{-8}, the algorithm is considered to be reached its global optimum successfully and is used as its early terminating criterion.

Table 2. Comparison of the experimental results of algorithms used

F	GA-BLX-α	SIIO vs GA-BLX- α	GA-SBX	SIIO vs GA-SBX	PSO	SIIO vs PSO	BFO	SIIO vs BFO	SIIO
	Mean (StdDev)	*p*-value	Mean (StdDev)	*p*-value	Mean (StdDev)	*p*-value	Mean (StdDev)	*p*-value	Mean (StdDev)
F1	3.14E+2(1.48E+2)	1.60E-10	9.35E+2(2.40E+2)	3.36E-16	8.72E-7(3.33E-6)	0.20	4.47E+3(1.21E+3)	1.19E-15	**1.82E-9(2.14E-9)**
F2	7.14E+0(2.56E+0)	5.29E-13	7.95E+1(4.86E+1)	2.17E-08	2.77E-3(9.17E-3)	0.14	9.11E-1(2.13E-1)	4.28E-17	**2.09E-9(1.38E-9)**
F3	7.30E+2(3.82E+2)	1.23E-09	8.09E+3(3.05E+3)	1.52E-12	9.11E-9(1.27E-9)	4.97E-15	3.05E+2(3.62E+2)	3.03E-4	**1.89E-9(2.41E-9)**
F4	4.74E+0(1.18E+0)	1.65E-16	4.12E+0(1.45E+0)	3.82E-13	9.56E-9(5.11E-10)	1.72E-16	1.01E+1(5.21E+0)	8.98E-10	**2.65E-9(2.03E-9)**
F5	5.04E+3(6.08E+3)	3.68E-4	6.90E+5(4.43E+5)	4.94E-08	1.26E+2(6.02E+2)	0.33	5.90E+1(1.25E+2)	0.056	**8.79E+0(2.78E-1)**
F6	2.95E+2(1.46E+2)	4.20E-10	9.09E+2(3.63E+2)	5.35E-12	**0.00E+0(0.00E+0)**	NA	6.56E+3(1.29E+3)	7.33E-19	**0.00E+0(0.00E+0)**
F7	9.26E-1(4.27E-1)	9.91E-11	9.06E-1(3.20E-1)	3.83E-13	3.02E-4(1.82E-4)	1.72E-08	5.22E-1(1.76E-1)	1.50E-13	**2.33E-9(2.50E-9)**
F8	1.13E+2(1.22E+2)	1.08E-4	3.09E+2(1.14E+2)	9.82E-13	7.11E+2(2.36E+2)	9.98E-14	1.47E+3(2.50E+2)	2.32E-20	**2.61E-2(9.85E-2)**
F9	1.64E+1(4.41E+0)	9.08E-16	1.70E+1(3.76E+0)	1.07E-17	7.28E+0(6.54E+0)	1.01E-05	4.24E+1(9.88E+0)	3.51E-17	**1.88E-9(2.49E-9)**
F10	4.20E+0(6.15E-1)	7.20E-22	3.53E+0(5.10E-1)	5.38E-22	3.63E-1(6.92E-1)	0.014	1.68E+1(9.90E-1)	2.58E-31	**2.31E-9(1.87E-9)**
F11	2.78E+0(6.54E-1)	4.41E-17	5.94E+0(1.75E+0)	8.01E-15	1.15E-1(8.22E-2)	2.88E-07	5.60E+1(9.75E+0)	4.30E-20	**1.89E-9(2.66E-9)**
F12	3.60E+1(1.10E+2)	0.120	1.56E+6(1.61E+6)	6.36E-05	**8.29E-3(2.87E-2)**	5.18E-08	1.13E+4(5.38E+4)	0.30	3.28E-1(2.20E-1)
F13	1.72E+4(4.52E+3)	0.0685	5.40E+6(4.05E+6)	6.80E-07	**2.19E-3(4.48E-3)**	3.97E-24	1.94E+5(2.91E+5)	2.76E-3	7.63E-1(9.26E-2)

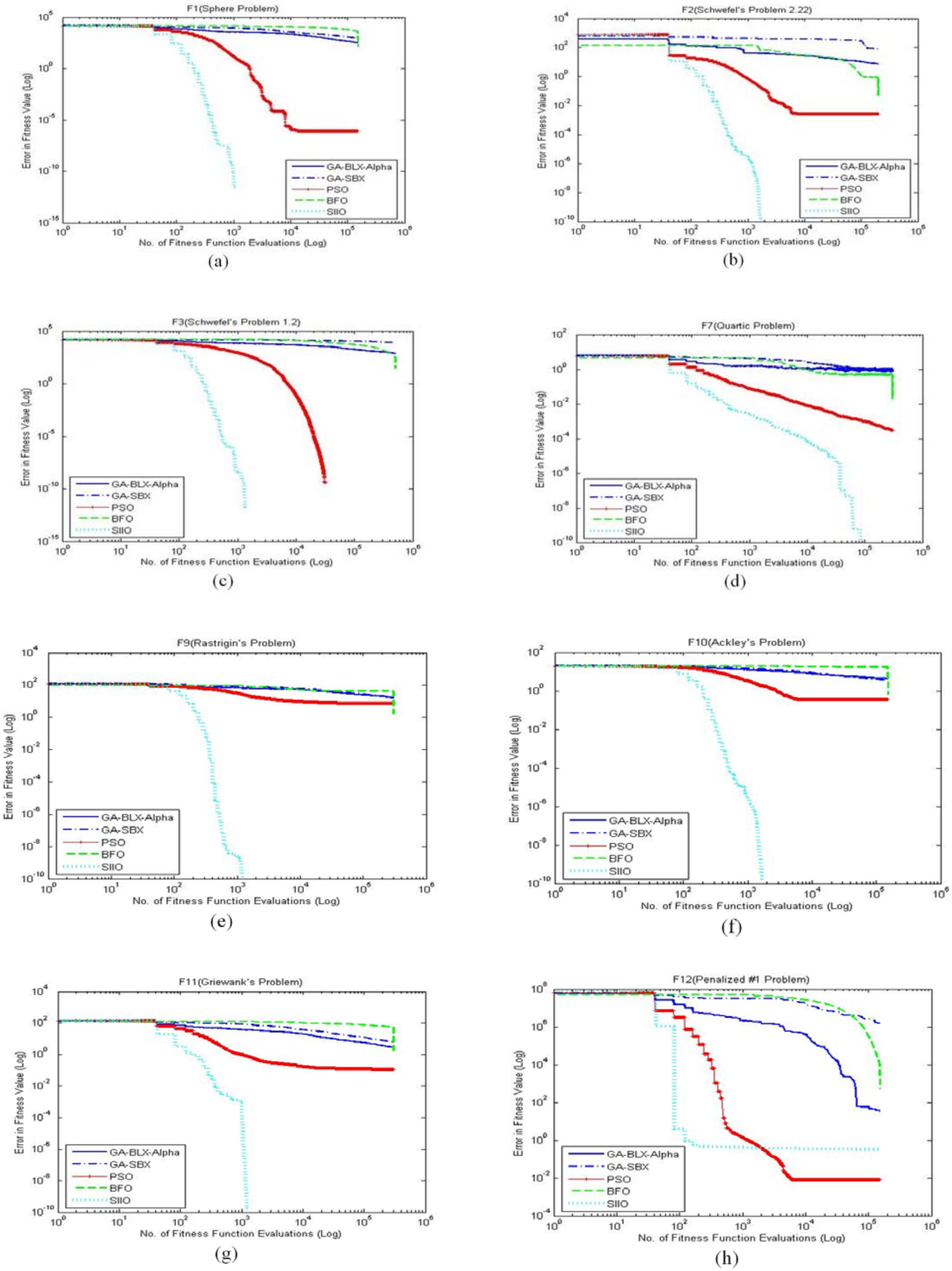
All functions have 10 dimensional inputs. The 25 independent trails are taken for testing the consistency of the convergence of the algorithms. The algorithms used for comparison with the SIIO are Genetic Algorithms having BLX-*α* cross over (GA-BLX-*α*) and SBX crossover (GA-SBX), Particle Swarm Optimization (PSO) and Bacterial Foraging Optimization (BFO). For comparison GA-BLX-*α*, GA-SBX, PSO, BFO and SIIO are run with same parameters and their results are shown in Table 2. Where ''Mean" is the mean best error values, ''STD Dev" is the standard deviation, shown in parenthesis. Student's t-test is performed for statistical testing of the obtained results and accordingly the P-values are reported. Here N.A. denotes 'not applicable'. From Table 2 it is clearly seen that the proposed algorithm SIIO generates better results compared to all the competitor algorithms. This is true in a statistically significant way for functions tested in Table 1. The SIIO demonstrates the superiority in solving the standard benchmark test problems. Table 3 shows error values achieved by SIIO in 25 runs on 13 standard numerical benchmark functions used, depicting the behavior of the SIIO in detail.

Table 3. Error Values Achieved by SIIO on Standard Benchmark Functions F1-F13

F	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13
1st (Best)	3.90E-11	5.76E-10	1.66E-11	4.29E-10	8.10 E+0	0.00E+0	2.08E-15	-2.23E-4	3.49E-11	1.71E-10	3.30E-11	8.93E-2	6.02E-1
7th	1.78E-10	1.04E-09	1.80E-10	1.14E-09	8.77 E+0	0.00E+0	4.74E-10	-1.32E-2	3.87E-10	7.86E-10	3.82E-10	1.76E-1	7.01E-1
13th (Median)	1.51E-09	1.75E-09	8.26E-10	1.91E-09	8.95 E+0	0.00E+0	1.19E-09	4.71E-2	7.28E-10	1.88E-09	6.23E-10	2.43E-1	7.22E-1
19th	2.25E-09	2.69E-09	2.22E-09	3.44E-09	8.97 E+0	0.00E+0	3.79E-09	7.66E-2	2.39E-09	3.32E-09	1.69E-09	5.96E-1	8.00E-1
25th (Worst)	8.00E-09	6.34E-09	8.81E-09	7.66E-09	8.98 E+0	0.00E+0	7.80E-09	2.77E-1	8.34E-09	8.52E-09	9.57E-09	7.39E-1	8.99E-1
Mean	1.82E-09	2.09E-09	1.89E-09	2.65E-09	8.79E+0	0.00E+0	2.33E-09	2.62E-2	1.88E-09	2.31E-09	1.89E-09	3.29E-1	7.63E-1
Std.Dev.	2.14E-09	1.38E-09	2.41E-09	2.03E-09	2.78E-1	0.00E+0	2.50E-09	9.86E-2	2.49E-09	1.87E-09	2.66E-09	2.20E-1	9.26E-2

3.2 Discussion

The algorithm is tested on standard benchmark functions having unimodal, multimodal properties with noise and discontinuity. F1-F5 are functions having unimodal property. Function F6 is step function, which has one minimum and is



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 4. Fitness Error versus number of function evaluation calls of test functions averaged over 25 runs (a) F1, (b) F2, (c) F3, (d) F7, (e) F9, (f) F10, (g) F11 and (h) F12.

a discontinuous function. Function F7 is a noisy quartic function comprising of random [0, 1) which is a uniformly distributed random variable in [0, 1). Functions F8 - F13 are multimodal functions in which the number of local minima increases exponentially with the problem dimension [4]. They are the most difficult class of problems for many optimization algorithms [4]. For unimodal functions, the convergence rates are more interesting than the final results of optimization. For unimodal functions, there are other methods like gradient based which are specifically designed to optimize these. In case of multimodal functions, the final results are important since they reflect an algorithm's ability of escaping from poor local optima and locating a good near-global optimum. The SIIO converged successfully in most of the functions viz. F1-F4 (unimodal), F6-F7 and F9-F11 (multimodal). In case of other functions also it has shown good performance. Convergence graphs shows the convergence rate as well as final results as seen in Figure 4a-h.

The graphs show the number of fitness function evaluations on x-axis and error in the optimum value on the y-axis. For comparison the convergence characteristics of GA-BLX-α, GA-SBX, PSO, BFO and SIIO are plotted. As per the graphs, the GA-BLX-α and GA-SBX performs similar. BFO converges similar to GAs but performs worst in many cases than GAs. PSO is better than GAs and BFO. Proposed SIIO converges better than all algorithms. When SIIO and PSO are compared in case of F1 and F2, SIIO has better convergence properties than PSO. In cases of F3 and F4, SIIO as well as PSO converged successfully but behaved differently. For F5 also, SIIO is better than PSO and BFO. In case of F6, SIIO as well as PSO shows successful and similar results. For F7-F11, SIIO performed better than all other algorithms. For functions F12 and F13, PSO achieves better average error among all the competing algorithms. For initial generations, SIIO shows fast convergence speed and at the end PSO show less average error in fitness. The speed of convergence is determined in terms of number of fitness function evaluations used by each of the algorithms for convergence or attaining the same error value by the algorithms. The graphs having number of fitness function evaluations on x-axis and error on y-axis, indicate the speed of the algorithms. Proposed SIIO algorithm wins for 10 functions; tie with PSO for 1 function and remains second after PSO for 2 functions. PSO also does better and achieves overall second position among all considered algorithms.

The configuration of computing machine is: Processor- Intel® Core™2Duo CPU, E8400 @ 3.00GHz 2.99GHz, Ram-1.94GB, Operating System- Microsoft Windows XP Professional Version 2002 Service Pack 2. The platform used for the simulation is MATLAB® 7.

4 Independent component analysis (ICA)

ICA is applied to many applications; one of these is blind source separation [13-21]. Blind source separation is an application that also can be solved using various theoretical approaches including ICA. ICA of a random vector x consists of finding a linear transformation y =Wx, so that the components y_i, i from 1 to m are as independent as possible. This can be achieved by maximizing some function $F(y_1,y_2,\ldots,y_m)$ that measures independence. W is demixing matrix. The maximization is done with deterministic methods for single signal [19] or for multiple signals [16, 17]. But there are multiple local maxima and many of these may not be the desired global maximum [21]. Therefore, the heuristic algorithms are suitable for the maximization problem and are applied for it [13, 18, 20]. In this paper, the SIIO is applied for noise removal.

For ICA, the assumptions are [14, 15]

1) The number of observed linear mixtures m must be at least as large as the number of independent components n, i.e., $m \geq n$.

2) The source signals must be statistically independent of each other or in practice as independent as possible (it includes uncorrelatedness). By definition, statistical independence means the joint probability density function (pdf) of the output sources can be factorized to the product of the marginal pdfs of each source.

3) Each source signal must be stationary with zero mean. Only one source is allowed to have Gaussian distribution because a linear combination of Gaussian signals is Gaussian again making it impossible to separate them.

4) The matrix A must be of full rank.

Two categories of ICA algorithms exists viz. optimization of suitable objective function and neural implementation of ICA [14]. In the first type, source separation is obtained by optimizing an objective function which is a scalar measure of some distributional property of the output y. The general measures are entropy, mutual independence, divergence between joint distribution of y and given mode and higher order decorrelation.

The ICA method when formulated as optimization of a suitable objective function, termed as the contrast function [14, 15]. The problem in optimization of contrast function is that, it is batch computation using the estimated higher order statistics of data or leads to complicated adaptive separation [14, 15]. It is often sufficient to use simple higher order statistics such as kurtosis, which is a fourth order cummulant with zero time lags [14, 15]. The kurtosis for the ith source signal s(i) is given by equation (5).

$$Cum\left[s(i)^4\right] = E\left\{s(i)^4\right\} - 3\left[E\left\{s(i)^2\right\}\right]^2 \tag{5}$$

If s(i) is Gaussian, then its kurtosis is zero. Source signals that have negative kurtosis are called sub-Gaussian and have a probability distribution flatter than usual Gaussian distribution. Source signals having a positive kurtosis are called super-Gaussian and have a probability distribution with sharp peaks and longer tails than the standard Gaussian ones. A contrast function based on kurtosis is given by equation (6).

$$\mathrm{F}(y) = \sum_{i=1}^{n}\left|Cum\left[y(i)^4\right]\right| = \sum_{i=1}^{n}\left|E\left\{y(i)^4\right\} - 3\left[E\left\{y(i)^2\right\}\right]^2\right| \tag{6}$$

5 Application of SIIO for noise removal

For N source signals $s = \{s_1(t), \ldots, s_N(t)\}$, linearly mixed by multiplying a mixing matrix A, produce N mixture signals $x = \{x_1(t), \ldots, x_N(t)\} = A*s$. In present case one channel ECG is considered mixed with Gaussian noise, so it gives N=2. Given the signal mixtures x, to recover a version y = W*x identical to the original sources s, optimization is done by finding a square unmixing matrix W. The main assumption used in ICA for solving this problem is that the source signals are as statistically independent as possible during the time of recordings.

To evaluate the resulting independent components, backprojection is used to plot body surface projection maps (ECG) for each component. From the ICA algorithm, we obtain the unmixing matrix W. The signal mixing matrix is W^{-1}, i.e., $x = W^{-1}*y$. Let $W^{-1}(:, i)$ denotes the ith column of W^{-1}, and y(i, :) denotes the ith row of y, then the back-projection of component i, z_i is $z_i = W^{-1}(:, i) * y(i, :)$. The column vector $W^{-1}(:,i)$ represents the relative (signed) weight of the ith component in each body surface channel of ECG. The back-projection map of each component may be plotted for each subject, which is shown in the Figure 5.

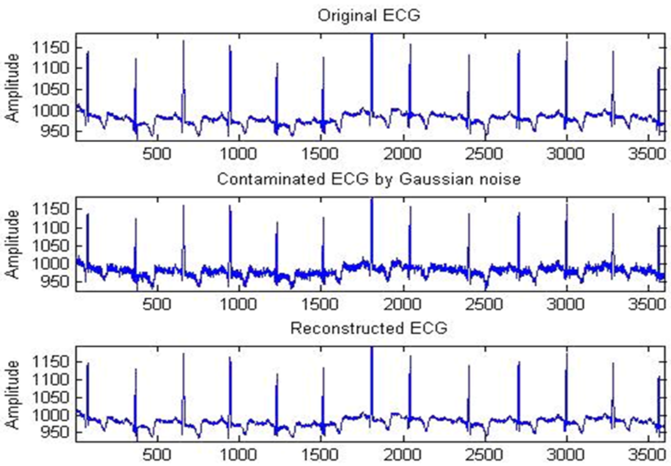


Figure 5. ECG - Original, Contaminated by Gaussian noise of mean=0, σ=1 and Reconstructed

The ECG which is contaminated by Gaussian noise with mean=0, standard deviation=1 and variance=1, is submitted for the processing of denoising with the help of SIIO with fitness function in equation (6). The inverse of optimized matrix W is used for backprojecting the signal from demixed version to remove noise from signal. Mean Square Error (MSE) is used to measure the difference between the original "clean" ECG and the reconstructed ECG. MSE is mainly due to the residual noise and also ECG distortion after filtering process. MSE is defined as in equation (7).

$$MSE = \frac{\sum_{t=0}^{L-1} \left(x(t) - \hat{x}(t)\right)^2}{L} \tag{7}$$

The results are presented in the Table 4 which indicates the MSE between the original signal and reconstructed signal, achieved with noise removal by SIIO. The MSE is also given for the comparison between original signal and contaminated signal. MSEo-r is MSE between original and reconstructed ECG signal; MSEo-c is between original and contaminated signal by Gaussian noise.

Table 4. Mean Square Error achieved by noise removal by SIIO

	100		**101**	
NFFE	1120	2008	5011	10011
MSEo-r	3.074	2.297	19.720	19.196
MSEo-c	25.231	25.231	25.231	25.231

The MSE between original and reconstructed ECG shows that the SIIO optimizes the demixing matrix for removal of noise as shown in Table 4. In the Figure 5 the original ECG, Contaminated ECG and Reconstructed ECG are shown for some interval. The ECG used is from MIT-BIH Arrhythmia Database [22, 23]. ECG is for 10 Sec, in Figure 5 it is shortened to 3500 samples for display. To convert from raw unit to the physical unit (mV), Base=0 is subtracted and then it is divided by Gain=143.062.

6 Conclusion

In this paper, a new optimizing algorithm named as Swine Influenza Inspired Optimization (SIIO) is developed. The mechanism of virus spreading (communicating), virus instating capacity and individual recovery by treatment deciding the fitness are used to build the algorithm. This information is utilized in the algorithm for individual learning by the method of recovery of the individual with the treatment mechanism. The proposed SIIO performed robustly as compared to Genetic Algorithms, Particle Swarm Optimization and Bacterial Foraging Optimization in terms of high accuracy and speed of convergence. The algorithm is having convergence accuracy up to the error of 10^{-8} for most of the functions and shows better speed, success and consistency of the convergence. The speed of convergence is determined in terms of number of fitness function evaluations used by each of the algorithms for convergence or attaining the same error value by these algorithms. The graphs indicate the speed of the algorithms. The SIIO optimizes the parameters for efficient Gaussian noise removal from the ECG signal by Blind Source Separation (BSS) using Independent Component Analysis (ICA).

References

[1] Das S, Suganthan P N. Differential evolution – a survey of the state-of-the-art. IEEE Transactions on Evolutionary Computation. 2010; 15(1): 4-31. http://dx.doi.org/10.1109/TEVC.2010.2059031

[2] Michalewicz Z. Genetic Algorithms + Data Structures = Evolution Programs. (3rd ed.). London, UK: Springer-Verlag. 1996.

[3] Nguyen QH, Ong Y-S, Krasnogor N. A Study on the Design Issues of Memetic Algorithm. Proceedings of the IEEE Congress on Evolutionary Computation. 2007; 2390 - 2397. http://dx.doi.org/10.1109/CEC.2007.4424770

[4] Yao X, Liu Y, Lin G. Evolutionary Programming Made Faster. IEEE Transactions on Evolutionary Computation. 1999; 3(2): 82-102. http://dx.doi.org/10.1109/4235.771163

[5] Arino J, Brauer F, Driessche PVD, Watmough J, Wu J. A model for influenza with vaccination and antiviral treatment. Journal of Theoretical Biology. 2008; 253: 118-130. PMid:18402981 http://dx.doi.org/10.1016/j.jtbi.2008.02.026

[6] Coburn B J, Wagner B G, Blower S. Modeling influenza epidemics and pandemics: insights into the future of swine flu (H1N1). BMC Medicine. 2009. PMid:19545404 http://dx.doi.org/10.1186/1741-7015-7-30

[7] Flahault A, Deguen S, Valleron AJ. A mathematical model for the European spread of influenza, European Journal of Epidemiology. 1994; 10(4): 471-474. PMid:7843359 http://dx.doi.org/10.1007/BF01719679

[8] Khazeni N, Hutton DW, Garber AM, Hupert N, Owens DK. Effectiveness and Cost-Effectiveness of Vaccination Against Pandemic Influenza (H1N1). Annals of Internal Medicine. 2009; 151(12): 829-839. PMid:20008759

[9] Rvachev LA, Longini IMJr. A mathematical model for the global spread of influenza. Mathematical Biosciences. 1985; 75(1): 3-22. http://dx.doi.org/10.1016/0025-5564(85)90064-1

[10] Kermack WO, McKendrick AG.A contribution to the mathematical theory of epidemics.Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character. 1927; 15(772): 700-721. http://dx.doi.org/10.1098/rspa.1927.0118

[11] Rvachev LA. Modeling experiment of a large-scale epidemic by means of a computer. (In Russian.) Trans. USSR Acad. Sci. Ser. Mathematics and Physics. 1968; 180(2): 294-296.

[12] Patel M, Dennis A, Flutter C, Khan Z. Pandemic (H1N1) 2009 influenza. Oxford University Press on behalf of the British Journal of Anaesthesia. 2010; 104(2): 128-142. (doi:10.1093/bja/aep375.)

[13] Acharya DP, Panda G, Mishra S, Lakshmi YVS. Bacteria Foraging Based Independent Component Analysis. International Conference on Computational Intelligence and Multimedia Applications. 2007; 527-531.

[14] Das N, Routray A, Dash P K. ICA Methods for Blind Source Separation of Instantaneous Mixtures: A Case Study. Neural Information Processing - Letters and Reviews. 2007; 11(11): 225-246.

[15] Li H, Adali T. A Class of Complex ICA Algorithms Based on the Kurtosis Cost Function. IEEE Transactions on Neural Networks. 2008; 19(3): 408-420. PMid:18334361 http://dx.doi.org/10.1109/TNN.2007.908636

[16] Moreau, Macchi O. New self-adaptive algorithms for source separation based on contrast functions. Proceedings of the IEEE Signal Processing Workshop on Higher Order Statistics. 1993; 215-219. http://dx.doi.org/10.1109/HOST.1993.264564

[17] Papadias C. Globally convergent blind source separation based on a multiuser kurtosis optimization criterion. IEEE Transactions on Signal Processing. 2000; 48(12): 3508-3519. http://dx.doi.org/10.1109/78.887044

[18] Rojas F, Puntonet CG, Rodriguez-Alvarez M, Rojas I, Martin-Clemente R. Blind Source separation in postnonlinear mixtures using competitive learning, simulated annealing and a genetic algorithm, IEEE Transactions On Systems, Man and Cybernatics - Part C: Applications and Reviews. 2004; 34(4): 407-416.

[19] Shalvi O, Weinstein E. New criteria for blind deconvolution of nonminimum phase systems (channels). IEEE Transactions on Information Theory. 1990; 36(2): 312-321. http://dx.doi.org/10.1109/18.52478

[20] Tan Y, Wang J. Nonlinear Blind Source Separation Using Higher Order Statistics and a Genetic Algorithm. IEEE Transactions On Evolutionary Computation. 2001; 5(6): 600-611. http://dx.doi.org/10.1109/4235.974842

[21] Tugnait JK. Comments on "New Criteria for Blind Deconvolution of Nonminimum Phase Systems (Channels)". IEEE Transaction on Information Theory. 1992; 38(1): 210-213. http://dx.doi.org/10.1109/18.108276

[22] Moody GB, Mark RG. The impact of the MIT-BIH Arrhythmia Database. IEEE Engineering in Medicine and Biology. 2001; 20(3): 45-50. PMid:11446209 http://dx.doi.org/10.1109/51.932724

[23] Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PCh, Mark RG, et al. Components of a New Research Resource for Complex Physiologic Signals. Circulation. 2000; 101(23): e215-e220. Available from: http://circ.ahajournals.org/cgi/content/full/101/23/e215]. PMid:10851218 http://dx.doi.org/10.1161/01.CIR.101.23.e215