SEECC: A Secure and Efficient Elliptic Curve Cryptosystem for E-health Applications

Golnaz Sahebi
Department of Information Technology
University of Turku, Finland
golnaz.sahebi@utu.fi

Amin Majd
Department of Information Technology
University of Turku, Finland
amin.majd@utu.fi

Masoumeh Ebrahimi
KTH Royal Institute of Technology, Sweden;
University of Turku, Finland
masebr@kth.se

Juha Plosila
Department of Information Technology
University of Turku, Finland
juplos@utu.fi

Jaber Karimpour
Department of Computer Science
University of Tabriz, Iran
karimpour@Tabrizu.ac.ir

Hannu Tenhunen
KTH Royal Institute of Technology, Sweden;
University of Turku, Finland
hannu@kth.se

***Abstract*—Security is an essential factor in wireless sensor networks especially for E-health applications. One of the common mechanisms to satisfy the security requirements is cryptography. Among the cryptographic methods, elliptic curve cryptography is well-known, as by having a small key length it provides the same security level in comparison with the other public key cryptosystems. The small key sizes make ECC very interesting for devices with limited processing power or memory such as wearable devices for E-health applications. It is vitally important that elliptic curves are protected against all kinds of attacks concerning the security of elliptic curve cryptography. Selection of a secure elliptic curve is a mathematically difficult problem. In this paper, an efficient elliptic curve selection framework, called SEECC, is proposed to select a secure and efficient curve from all the available elliptic curves. This method enhances the security and efficiency of elliptic curve cryptosystems by using a parallel genetic algorithm.**

Keywords— elliptic curve cryptography, secure elliptic curve, evolutionary computing, genetic algorithms, parallel genetic algorithms, multi-population parallel genetic algorithms, E-health

I. INTRODUCTION

Advances in information technology and wireless sensor networks allow the usage of sensors, computation, and communication devices on human body, so called Wireless Body Area Networks (WBANs) [1]. This development has enabled a large variety of new applications in several domains such as wellbeing and health care. One of the important advantages of WBANs is the capability of monitoring the human body at real-time. Figure 1 illustrates a primary design of a health care system, comprising of three main parts: WBAN, personal server (PS), and medical servers. WBAN includes different sensors to monitor the human body and measure medical data such as Electrocardiograms (ECG), Electroencephalogram (EEG), and temperature. The measurements are sent to the personal server via radio interface and wireless communication [1]. Finally, the collected data in the personal server are transferred to an external medical server on the condition that clinical services are required.

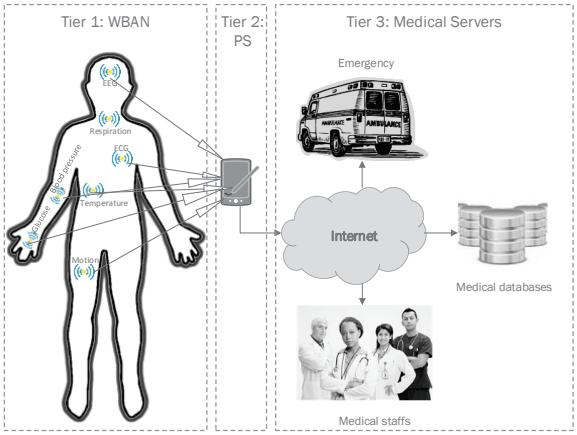


Figure 1. A primary schema of a health care system

Notwithstanding the enormous potential of wearable systems in improving the quality of lives, they face some obstacles to become a reality. First, the processors and architectures require a large amount of energy and sizable batteries. This brings a number of challenges toward miniaturization of the wearable devices. As one of the most important challenges, security becomes a critical factor as medical information must be safe against unauthorized use for dishonest acts that might threat life of the users [3]. In the wearable system domain, new sorts of security challenges arise due to its unique nature of sensor mobility and proximity to potential attackers [2], [4]. In the dynamic environment of using wearable systems, the system should be protected against different types of attacks to assure the privacy and integrity of medical data. Recently, several security solutions have been presented for protecting the bio-medical data. Among these solutions, cryptography has received much attention. Cryptography is divided into two main groups: cryptographic methods based on symmetric keys and those based on asymmetric keys. Among these two

groups, asymmetric key methods such as Rivest-Shamir-Adleman (RSA) and Elliptic Curve Cryptography (ECC) provide a better data confidentiality [5]. Meanwhile, when the target is the security of compact devices, ECC can better satisfy the requirement of limited memory and processing capacity. The main feature of ECC, making it a popular choice in mobile healthcare systems, is a high level of security with a minor key bit size [3], [4], [5], [6]. For these reasons, ECC is commonly applied in mobile healthcare systems to both increase the security and decrease the computational costs [5].

Unfortunately, all kinds of cryptographic systems have vulnerabilities and are challenged by various attacks. Regarding ECC, attackers can infiltrate into an ECC protected system by solving the Elliptic Curve Discrete Logarithm Problem (ECDLP). The ECDLP is the main factor of security protection in ECC. Existence of a secure elliptic curve is fundamental for guaranteeing the security of ECC and its resistance against the ECDLP attacks. This resistance is directly dependent on the number of elliptic curve points, also called the order of the curve. ECC can be made more secure by increasing the number of the points. The main problem is then to find and select a secure elliptic curve which is suitable for various cryptosystems.

The selection of a secure elliptic curve, which is performed in the first phase of the ECC algorithm, is a non-linear problem [16] with large search spaces. There are some approaches for solving this problem such as metaheuristic methods (e.g. Evolutionary Algorithms (EAs)) and deterministic methods. Since it is a non-linear problem with a large number of initial points, EA methods will lead to more powerful solutions. Evolutionary algorithms refer to different methods such as Genetic Algorithms (GAs), Imperialist Competitive Algorithm (ICA), and Ant Colony Optimization (ACO). Previously, a simple GA has been utilized to select a secure elliptic curve [16].

In this paper, a new framework for selection of secure elliptic curves based on mathematical selection principles is presented. The selection is performed by using a parallel multi-population genetic algorithm which is implemented on a shared memory architecture. The major goal of this work is improvement of security and resource efficiency of elliptic curve cryptography.

The rest of the paper is organized as follows: In Section II, some definitions and facts about elliptic curves, mathematical principles of secure curve selection, and parallel genetic algorithm are given. Section III discusses the related work and motivation of this work. In Section IV, the proposed selection algorithm for ECC (SEECC) is presented. Section V demonstrates the efficiency of the proposed algorithm and experimental results. Finally, Section VI concludes the paper.

II. Background on Elliptic Curve Cryptosystem

In this section, we review the main concepts of Elliptic Curve Cryptography (ECC) and Parallel Genetic Algorithms (PGA). With advances in computing abilities, the need for efficient cryptographic algorithms, which can provide high speed, high security and low memory usage, is a necessity. According to the latest studies, ECC is a viable candidate to satisfy the mentioned properties. ECC provides an equal level of security with a far smaller key size than for example the RSA method, significantly reducing the processing overhead [5], [9], [28], [29]. TABLE I compares the well-known cryptographic methods based on the security levels and key sizes [31][31]. This table indicates that the size of ECC keys is much smaller for the same level of security when compared to RSA. Asymmetric methods have efficiently solved the main challenge of symmetric methods, which is the key distribution; therefore, their usage has recently been prevalent.

TABLE I. A key length and security level comparison of cryptosystems

Cryptosystems	Symmetric	Asymmetric	
Security Levels	AES (key bit size)	RSA (key bit size)	ECC (key bit size)
1	128	3072	256
2	192	7680	384
3	256	15360	512

Neal Koblitz and Victor S. Miller suggested the usage of elliptic curves in cryptography in 1985 [15]. Since then, ECC has been officially accepted and been followed by many standards such as NIST, ISO, and IEEE [10].

Elliptic Curve Cryptography is a public key cryptosystem based on algebraic structures of elliptic curves on finite fields. Elliptic curves only consist of a few equations and in order to be used in cryptography, they must be utilized in the other public key cryptosystems, like Elgamal. Figure 2 illustrates the Elgamal Elliptic Curve cryptosystem. In this algorithm, some parameters must be generated by a trusted center and published in the network. These parameters are the secure elliptic curve equation ($E(a, b)$), the number of the field (q), and a point on the curve (G).

In the IEEE P1363 standard [11], the parameters of ECC are defined as a 7-tuple T = (q, FR, a, b, G, r, h), where q is a prime number or a binary to represent the finite field F_q, FR implies the representation basis of the finite field, a and b, which describe the curve equation, are elements of the finite field, G is the base point, r is a large prime and is also the order of the base point G that can be computed by the baby step-giant step algorithm [17], and h (cofactor) is a small integer that is obtained by:

$$h = \frac{\#E(F_q)}{r} \tag{1}$$

where #E (F_q) is the number of F_q-points of an elliptic curve defined over a finite field F and is computed by Schoof and baby step-giant step algorithms [17].

In the set of real numbers, the curve equation is defined as:

$$y^2 + xy = x^3 + ax + b \tag{2}$$

However, to reach the goals of cryptography, the elliptic curve equation is defined as follows:

$$y^2 = x^3 + ax + b \tag{3}$$

This is known as the Weierstrass Equation. Since computations in real fields are slow and also inaccurate, and cryptographic methods need accurate and fast computations, elliptic curve groups on finite fields, such as prime (F_q) and

binary (F_{2^m}) fields, are utilized for these purposes. In this paper, elliptic curves on prime fields are employed.

An elliptic curve $E(F_q)$ on a finite field F_q, where q is a prime number and larger than 3, is a set of points (x, y) that satisfy the equation:

$$y^2 \bmod q = (x^3 + ax + b) \bmod q \quad (4)$$

1. Public parameters generation	
A trusted center selects the elliptic curve, i.e., *E (a, b)*, the number of field, *i.e., q*, and a base point on E, i.e., *G,* then publishes them in the network	
2. Keys generation	
Alice	Bob
Selects a random integer n_A, and keeps it secret as her private key. Calculates the point Q_A in this way: $Q_A = n_A G$ (5) Then publishes it as her public key.	Does not do anything.
3. Encryption	
Does not do anything.	Converts the plaintext, which should be encrypted to a point (P_m) or points on the curve. Then he performs these steps to send the converted message to Alice: 1. Selects a random positive integer *k*. 2. Performs the following computations to obtain two new points on the curve using the Alice's public key (Q_A). $C_1 = kG$ (6) $C_2 = P_m + kQ_A$ (7) 3. Sends the ordered pair of points (C_1, C_2) as the cipher text to Alice.
4. Decryption	
Calculate the $C_2 - C_1 n_A$ (8) coordinate on the curve to detect the plaintext.	Does not do anything.

Figure 2. Elgamal Elliptic Curve Cryptosystem

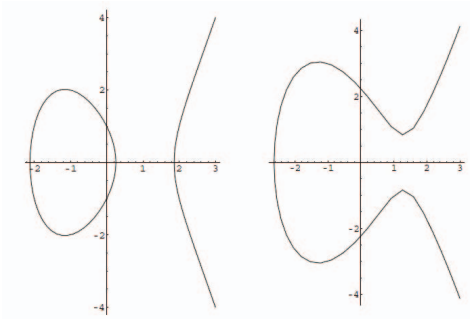


Figure 3. Graph of an instance EC (a) Δ<0 (b) Δ>0 [31]

A group of elliptic curves consists of the set of points that satisfy the equation (4), the point at infinity (identity element), and an addition operation. In the best conditions, the order of the curve ($\#E(F_q)$) is a prime number. In this case, the group forms a cyclic group, which can be utilized as a framework for cryptographic purposes [29]. In addition, the cryptographic use of elliptic curves requires that the curves are non-singular. To satisfy this condition, the discriminant Δ of the elliptic curve equation must be non-zero, that is:

$$\Delta = -16(4a^3 + 27b^2) \neq 0 \quad (5)$$

The discriminant delineates the real graph of a non-singular curve. The sign of the discriminant determines the shape of the curve, i.e., it has two cases corresponding to the negative discriminant and the positive discriminant. Figure 3 geometrically indicates these two cases. Among all parameters of ECC, the most affecting security parameter is $\#E(F_q)$, the number of points of an elliptic curve defined over a finite field.

The key point of public-key cryptography is the intractability of certain mathematical problems, and for ECC this problem is the Elliptic Curve Discrete Logarithm Problem (ECDLP).

ECC will be insecure if the ECDLP can be computed. There are some algorithms to solve the ordinary ECDLP such as Weil and Tate [18], Pollard-ρ [19] and Pohlig-Hellman [20]. In order to protect ECC, it is essential that the selected elliptic curves are secure against the aforementioned algorithms. However, there are only 15 curves that have been recommended by NIST for everyone to access [10], which is a potential vulnerability. The security of ECC is improved, if we can generate a new standard secure curve instead of using the 15 recommended ones. The following conditions must be satisfied for an elliptic curve over the field F_q to resist against all known attacks [16], [27]:

- $\#E(F_q)$, i.e., the number of points of an elliptic curve defined over a finite field, should be divisible by an adequately large prime number N (e.g. $N > 2^{160}$) to resist the Pollard-ρ attack [21].
- To resist the Weil and Tate pairing attacks, $q^k - 1$ should not be divisible by $\#E(F_q)$ ($1 \leq k \leq C$). Here C should be large enough ($C=20$ is enough in practice) to ensure that the selected curve will not become a super-singular elliptic curve [22], [23].
- The number of the points of an elliptic curve defined over a finite field should not be equal to q (i.e. $\#E(F_q) \neq q$) to resist against the Semaev-Smart-Satoh-Araki attack, ensuring that the selected curve will not become an anomalous elliptic curve [24].

The selection of a secure elliptic curve is a non-linear problem as well as a multi-objective optimization problem. This indicates that evolutionary computing methods provide an efficient approach to solve this problem [16].

III. Related Work

It has been efficient improvements on ECC in recent years that are divided into two major categories: speed and security.

The speed-up improvements accelerate the speed of computations (e.g. scalar multiplications) in the ECC algorithm by several methods such as parallelization, pipeline

architectures, and the use of improved Montgomery algorithms. TABLE II presents security solutions for different architectures, recently proposed for protecting bio-medical sensor networks [4]. Efficient implementations of ECC have recently been developed for wireless sensor networks. NanoECC [9], proposed by Szczechowiak et al., is comparatively faster than the other existing ECC implementations, but it requires a massive amount of ROM and RAM memory [4]. Uhsadel et al. [7] have proposed an efficient implementation of ECC. TinyECC [8] has been developed by Liu et al. It is an ECC library, which uses inline assembly code to speed up critical operations of ECC.

Improvements in security of ECC have rarely been carried out, but Wang et al. have used a simple genetic algorithm to select a secure curve in the finite fields [16] to increase the security of ECC.

On the other hand, computability is a significant challenge for researchers especially in non-linear problems such as selecting a secure elliptic curve. There are different methods for solving these problems, but evolutionary algorithms are the most popular ones [32], [33].

TABLE II. SECURITY SCHEMES USED IN HEALTH CARE ARCHITECTURES [4]

System architecture	Security Scheme
ALARM-NET	Hardware Encryption
Code Blue	ECC & TinySec
WBAN	Hardware Encryption
SNAP	Tiny ECC

Evolutionary methods are successful in solving different problems, but there are some disadvantages associated with them [30], [33]. For example, in some problems that have an extensive search space, it is possible for algorithms to converge to local optimums so that the results can only be improved by increasing the initial population that is no feasible with one processor. The other problem is the speed of algorithms, i.e., finding solutions may take a long time.

Parallel algorithms can improve the quality and timing overhead of obtaining results [12], [13], [30]. One of the well-known evolutionary methods is a Genetic Algorithm (GA) [12], [13]. Genetic algorithms are population-based search methods that mimic the process of natural selection and evolution (i.e. each GA is started with initializing a population and executes frequent operations, such as selection, crossover, mutation, and replacement). All operations of GAs are repeated until arriving at a suitable result or ending in a certain generation. Parallel Genetic Algorithms are implemented in four categories [12]: Master-Slave, Fine Grain, Multi-Population (Coarse Grain), and Hybrid methods. A GA detects a good solution when it has a suitable selection pressure. When there are several processors with several memories, multi-population methods are useful for GAs, because they could have a bigger population. In a multi-population method, there is a set of processors, such that each processor hosts an independent population and independently runs a serial GA on this population. The key feature in this method is the migration operation. After several iterations, each processor selects some of the best chromosomes and sends them to the other processors. This operation shares the best solution of each processor among the others and helps to find the best solution in lower iterations while providing a higher accuracy [12], [13], [26], [30].

To the best of our knowledge, no algorithm has been implemented so far considering both security and speed-up aspects of ECC. Therefore, we intend to propose an efficient algorithm to select a secure elliptic curve and to improve both security and performance of ECC by using parallel evolutionary algorithms. This algorithm leads to a secure and efficient elliptic curve that would be useful in different applications for example providing a secure key exchange in TLS when used for e-commerce. In this work, we specially motivate its usage in the E-health domain as the algorithm is fast, lightweight and highly secure and thus perfectly match to the requirement of wearable devices.

IV. THE PROPOSED METHOD

In this paper, we propose an efficient framework to select a secure elliptic curve for lightweight cryptography objectives. The algorithm is called SEECC.

ECC works on a discrete space; therefore, a Genetic Algorithm (GA) is a suitable choice to solve the secure elliptic curve selection problem. In addition, a parallel GA is more efficient than a sequential GA for these problems. Therefore, we have utilized a parallel multi-population genetic algorithm to achieve the best results for selecting a secure curve in ECC according to the selection conditions given in Section II.

In this work, there are some important aspects concerning the multi-population strategy that have been efficiently matched with the elliptic curve selection problem and enhanced to obtain more accurate solutions. The key points of multi-population GAs, regarding their efficient usage in this problem, are:

- Increasing the diversity of initial population
- Increasing the selection pressure
- Considering the migration operator

The multi-population strategy leads to better diversity of population which improves the search space of curves. This diversity also enhances the selection pressure to obtain more secure curves. Furthermore, the migration operator enables processors to exchange their best genetic material, thereby improving their genetic populations.

The proposed multi-population method uses a ring topology because of its simplicity and efficiency. The related pseudocode and flowchart are illustrated in Figure 4 and 6, respectively. Our approach can run on both the shared-memory and the message passing architectures. The algorithm obtains a result which satisfies all conditions of a secure elliptic curve.

Processor P_i :
1. Creates independently populations (initializing).
2. Evaluates the fitness function
3. Runs selection operator.
4. Runs crossover operator.
5. Runs mutation operator.
6. Sends the best chromosome to the master processor.
7. Runs replacement operator.
8. If this is the migration time now, sends the best chromosome to the next processor, and receives a chromosome from the previous processor, and replaces it on worth chromosome of

itself.
9. If fitness function of the best chromosome is equal to three go to step 10
   Otherwise go to step 3.
10. End.

Figure 4. Pseudo code of SEECC

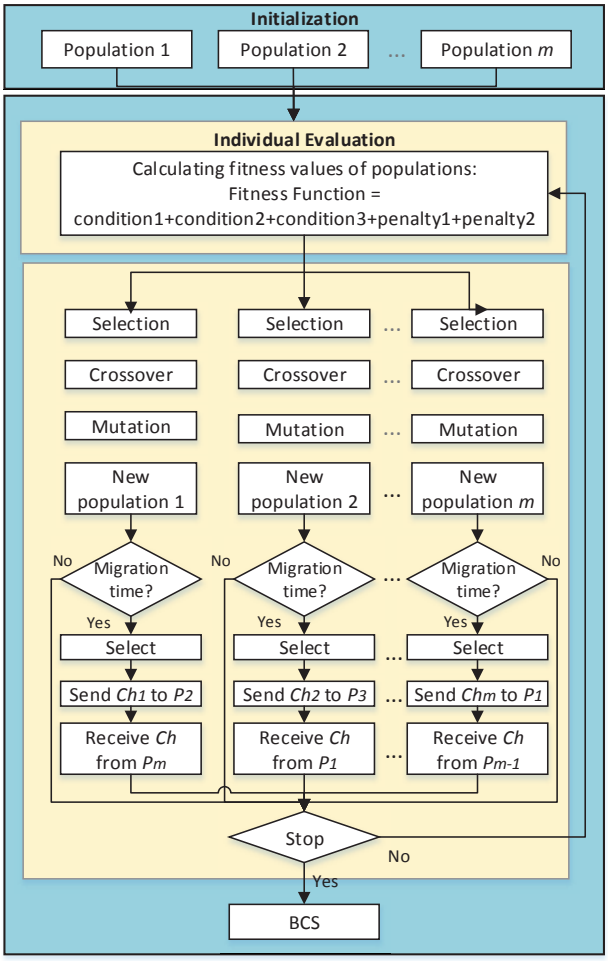


Figure 5. The algorithm flowchart of SEECC

The main phases of our algorithm, shown in Figure 5, are as follows:

1) Initialization

The primary population is created randomly. Figure 6 illustrates the structure of the chromosomes.

First, the parameters *a*, *x*, and *y* are randomly initialized, then *b* and *r* are calculated. The parameters *a*, *b* are the coefficients of Equation 4, *r* is the order of the base point, and *x*, *y* are the coordinates of the base point. All the genes (*a, b, r, x, and y*) of this chromosome have integer values because the search space of this problem is discrete. The involved genetic operators are also of the integer type.

a	*b*	*r*	*x*	*y*

Figure 6. Structure of the chromosome

2) Chromosome Evaluation

Three mathematical selection conditions, presented in Section II, must be satisfied in the evaluation phase:

Condition_1: $N \mid \#E(F_q)$, where N is a large prime number, $N > 2^{160}$.

Condition_2: $(q^k - 1) \nmid \#E(F_q)$, $1 \leq k \leq C$, where $C=20$.

Condition_3: $\#E(F_q) \neq q$.

Each condition has a positive score of 1. Furthermore, there are two potential penalties for checking whether the proposed curve is a non-singular curve (Equation 5) or whether the proposed curve group is a cyclic one ($\#E(F_q)$ must be a prime number to form a cyclic group). Each of these penalties has a negative score of -1 if the curve is no a non-singular one or if it is no in a cyclic group. Therefore, the value of the fitness function can be considered an integer in the range [-2, +3]. Since our method is a maximization problem, in the best case, this value is equal to three, if all the three selection conditions are satisfied and the curve is non-singular in a cyclic group.

$$Fitness\ Function = Condition_1 + Condition_2 + Condition_3 + Penalty_1 + Penalty_2 \quad (10)$$

3) Selection operator

The fundamental idea of the selection operator is that it gives preference to better chromosomes and allows them to pass on their genes to the next generation. The proposed algorithm adopts the tournament method with three members to select the best chromosome. This operator has been selected among different selection methods, such as roulette wheel selection, tournament selection, rank selection, and steady state selection [34], because of its simplicity compared to the other methods. First, three chromosomes are randomly selected, and then the best one is selected for the next generation in every cycle.

4) Crossover operator

GA has two main operators, which are named crossover and mutation. Since GA is a semi-random optimization method, its operators do not always occur. In other words, they happen with a probability. The crossover operator selects genes from chromosomes, that are the parents, and creates a new offspring. In other words, this operator exploits the search space to find more accurate solutions. The crossover chromosomes are chosen randomly from the population according to a probability that is named crossover rate (P_c). The crossover rate controls the frequency with which the crossover operator is applied [34]. In our algorithm, the real type crossover among the different crossover techniques, such as one-point crossover, two-point crossover, uniform crossover, and real type crossover [34], [35], is utilized on the first gene of a chromosome (*a*). The other genes of a chromosome are generated similarly as in the initialization phase. This crossover operator has been selected here, because the selection of a secure elliptic curve in a finite field is a discrete problem. The crossover rate in our algorithm is set to 0.8 ($P_c = 0.8$).

5) Mutation operator

The mutation operator occurs with a probability that is named mutation rate (P_m) [34]. If this operator happens on a chromosome, it randomly changes the new offspring according to the mutation rate. In other words, this operator explores the search space to discover a new search area and prevents all solutions in a population from falling into a local optimum. The mutation rate is a measure for determining when mutations occur over time [35]. In our algorithm, the first gene is replaced with a random number, and the other genes of the chromosome are generated similarly as in the initialization phase. The mutation rate in this algorithm is set to 0.2 (P_m = 0.2).

6) Replacement operator

The current generation of chromosomes is replaced by the recently generated offspring based on a particular replacement technique. In our algorithm, the steady-state strategy is utilized for the replacement operator. The operation compares each chromosome of the current population with the last generation. If a chromosome in the current generation is better than its corresponding chromosome in the last generation, the new chromosome is replaced on the old one.

7) Migration operator

During the migration process, some of the best chromosomes, in each processor, are chosen and sent to the next processor in the ring at each migration time point. Concurrently, each processor receives the chromosomes sent by the previous processor and replaces its worst chromosomes with the received ones.

8) Stopping strategy

In this algorithm, there are two stopping strategies. The first one is to find a chromosome with a fitness value that equals three. The second one is to stop at the 100th generation.

V. Experiment And Results

In this section, implementation and experimental results are reported concerning the accuracy, speedup and efficiency.

A. Implementation details

The proposed method is implemented in Visual C++ using the MPI library for parallelization. The proposed algorithm was run with MPICH2 on a shared memory structure. It is worth mentioning that our implementation works on both shared memory and the message passing architectures. In a message passing system, we need a ring connection. In a shared memory system there is no such limitation. All implementations and experiments have been performed on an Intel Core i3-330m Processor 2.13 GHz, running Windows 7 Home Premium (64-bit) and equipped with 4 GB of main memory. Two cores have been utilized in the parallel implementation.

B. Case studies

The proposed parallel algorithm has been tested on two prime numbers. The primes have randomly been generated where the order of the curve ($\#E(F_q)$) becomes a prime number to have a cyclic group. In addition, the experiments have been performed 30 times for each number in both serial and parallel cases.

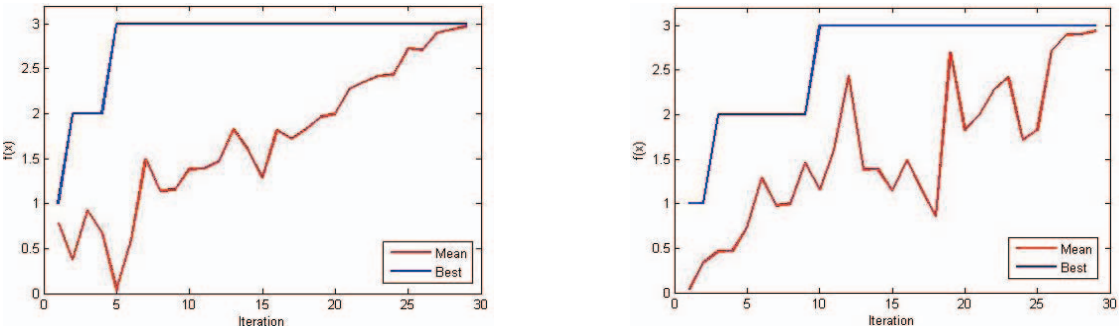
Prime Number q_1: 249989

According to the convergence diagram (Figure 7), the parallel implementation for the prime number (Figure 7 (a)) converges sooner to the optimum solution than the serial implementation (Figure 7 (b)). The Best graph in the convergence diagram shows the values of the best chromosome in each iteration. Also, the Mean graph shows the average value for the whole population in each iteration. The other important parameters that must be considered in these problems are stability and solution reliability, which can be illustrated by the stability diagram. In the stability diagram (Figure 8), lower fluctuations indicate that the proposed parallel algorithm (Figure 8 (a)) is more stable and reliable than the serial algorithm (Figure 8 (b)).

Prime Number : 11616799

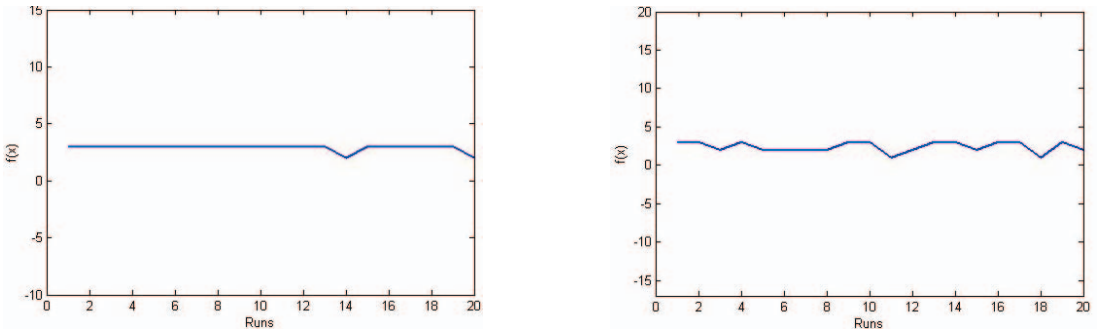
The second case study is a larger prime number than the first one. According to the convergence of the parallel and serial methods diagram (Figure 9), the speed of convergence decreases when the prime number gets larger. In addition, it can be simply observed that the distance between convergence of the parallel (Figure 9 (a)) and serial methods (Figure 9 (b)) increases when the search space of the problem grows. This is the best proof to justify the use of parallel methods, because cryptographic applications deal with the extremely large numbers. The other attractive point, in this example, is the stability diagram's quality changes that have been illustrated in Figures 10 (a) and 10 (b). In the stability diagram, like in the case of the smaller prime number, lower fluctuations indicate that the proposed parallel algorithm (Figure 10 (a)) is more stable and reliable than the serial algorithm (Figure 10 (b)).

The statistical results are listed in TABLE III. There are five important parameters in this table that are described as follows: The standard error (SE) is the standard deviation of the sampling distribution of a statistic, most commonly of the mean [36]. In addition, a method with the lowest SE value is the best one. The standard deviation (STD) is a measure that is utilized to quantify the amount of variation or dispersion of a set of data values [36]. A standard deviation close to 0 indicates that the data points tend to be very close to the mean (also called the expected value) of the set while a high standard deviation indicates that the data points are spread out over a wider range of values [36]. The mean is the average value of all the best results in all 20 runs. The best and the worse are the best and the worse values of all 20 runs. A method can be considered the best one when it has the lowest values of SE and STD and the mean and the highest values of the best and the worst. TABLE III demonstrates that our algorithm is more accurate with fewer errors than the existing serial methods. In TABLE IV, two important parameters, speedup and efficiency, are also compared for both serial and parallel approaches. The table shows that the efficiency of the proposed parallel implementation is outstanding, and it clearly outperforms the existing serial methods.



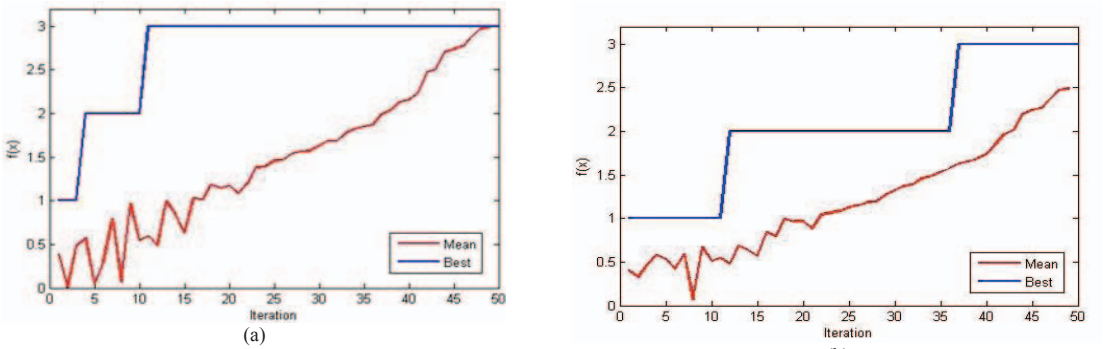
(a) (b)

Figure 7. Convergence diagrams for F_{249989} (a) parallel implementation (b) serial implementation



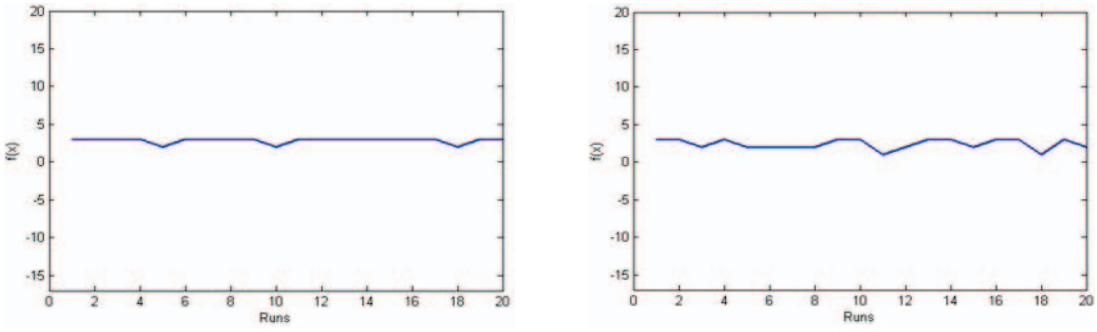
(a) (b)

Figure 8. Stability diagrams for F_{249989} (a) parallel implementation (b) serial implementation.



(a) (b)

Figure 9. Convergence diagrams for $F_{11616799}$ (a) parallel implementation (b) serial implementation.



(a) (b)

Figure 10. Stability diagram for $F_{11616799}$ (a) parallel implementation (b) serial implementation

TABLE III. STATISTICAL RESULTS FOR $\mathbf{F_{11616799}}$ AND $\mathbf{F_{249989}}$

Methods & Case Studies	SE	STD	Mean	Median	Best	Worst
Serial F_{249989}	0.152177182050536	0.680557047378721	2.4000000	2.50000000	3	1
Parallel F_{249989}	0.068824720161169	0.307793505625546	2.9000000	3	3	2
Serial $F_{11616799}$	0.152177182050536	0.680557047378721	2.4000000	2.50000000	3	1
Parallel $F_{11616799}$	0.081917802190913	0.366347548532523	2.850000	3	3	2

TABLE IV. SPEEDUP AND EFFICIENCY VALUES FOR PARALLEL IMPLEMENTATIONS

The Problem	# Processors	Parallel Time	Serial Time	Efficiency	Speedup
Parallel F_{249989}	2	18.2	27.66	0.76	1.52
Parallel $F_{11616799}$	2	44.03	64.67	0.73	1.46

VI. CONCLUSION

The wireless and mobile technologies have stimulated a great advance in promoting the development of electronic healthcare. One of the important challenges in the E-health is providing security and privacy.

In this paper, to enhance the security in E-health applications, an efficient secure Cryptosystem (SEECC) was presented. An efficient selection method was proposed, which is based on the secure curve selection mathematical principles using parallel genetic algorithm. The proposed algorithm was evaluated on two case studies. The obtained results proved that the proposed method has obtained a higher accuracy in a lower time.

References

[1] E. Jovanov, A. Milenkovic, C. Otto, and P. C.D. Groen, "A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation", Journal of NeuroEngineering and Rehabilitation, vol. 2, no. 6, 2005.

[2] K. Malasri and L. Wang, "Addressing Security in Medical Sensor Networks," HealthNet '07, pp. 7-12, San Juan, Puerto Rico, USA, 2007.

[3] M. Mana, M. Feham, and B. A. Bensaber, "Trust Key Management Scheme for Wireless Body Area Networks," International Journal of Network Security, Vol. 12, No. 2, PP. 75, 2011.

[4] Y. Ren, R. W. N. Pazzi, A. Boukerche, "Monitoring Patients via a Secure and Mobile Healthcare System," Wireless Communications, IEEE, Vol 17, pp. 59-65, 2010.

[5] W. Stalling, "Network Security Essentials: Applications and Standards," Fourth edition, Pearson, ISBN 13: 978-0-13-610805-4, 2011.

[6] C. Karlof, N. Sastry, and D. Wagner, "TinySec: A link layer security architecture for wireless sensor networks," Second ACM Conference on Embedded Networked Sensor Systems, pp. 162-175, Nov. 2004.

[7] L. Uhsadel, A. Poschmann, and C. Paar, "Enabling full-size public- key algorithms on 8-bit sensor nodes," Proceedings of European Workshop on Security in Ad-Hoc and Sensor Networks, LNCS 4572, pp. 73-86, Springer-Verlag, 2007.

[8] A. Liu and P. Ning, "TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks," Proceedings of the International Conference on Information Processing in Sensor Networks, pp. 245-256, 2008.

[9] S. S. M. Meingast and T. Roosta, "Security and privacy issues with health care information technology," Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 5453-5458, Aug. 2006.

[10] E. Barker, D. Johnson, M. Amid, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography," NIST Special Publication 800-56A, 2007.

[11] IEEE P1363/D13, Standard Specifications for Public-Key Cryptography, 1999. http://grouper.ieee.org/groups/1363/

[12] E. Cantú-Paz, " A Survey of Parallel Genetic Algorithms," Department of Computer Science and Illinois Genetic Algorithms Laboratory University of Illinois at Urbana-Champaign, 1997.

[13] E. Alba, F. Luna, A. J. Nebro, and J. M. Troya, "Parallel Heterogeneous Genetic Algorithms for Continuous Optimization," Parallel Computing, Vol. 30, pp. 699–719, ELSEVIER, 2004.

[14] Md. M. Haque, A. S. K. Pathan, and C. S. Hong, "Securing U-healthcare sensor networks using public key based scheme," ICACT '08, pp. 1108-1111, Feb. 17-20, 2008.

[15] N. Koblitz, "Elliptic Curve Cryptosystems Mathematics of Computation," pp. 203- 309, 1987.

[16] M. Wang, G. Dai, H. Hu, L. Pen, "Selection of Security Elliptic Curve Based on Evolution Algorithm," International Conference on Computational Intelligence and Natural Computing, IEEE, pp. 55-57, 2009.

[17] R. Schoof, "Elliptic Curves Over Finite Fields and the Computation of Square Roots mod p," Mathematics of Computation, Vol 44, 483-494, 1985.

[18] Pohlig, S. C. and Hellman, M. E, "An improved algorithm for computing logarithm over GF (p) and its cryptographic significance" IEEE Trans. Inf. Theory, pp. 106-110, 1978.

[19] J. Pollard, "Monte Carlo methods for index computation mod p. Mathematics of Computation," Mathematics of Computation, Vol32, pp. 918-924, 1978.

[20] A. Joux, "The Weil and Tate Pairings as Building Blocks for Public Key Cryptosystems," Algorithmic Number Theory, Vol. 2369, Springer, pp. 20-32, 2002.

[21] A. K. Lenstra, "Efficient Identity Based Parameter Selection for Elliptic Curve Cryptosystems," Springer- Verlag, LNCS 1587, pp. 294-302, 1999.

[22] H. Baier, "Efficient Algorithms for Generating Elliptic Curves over Finite Fields Suitable for Use in Cryptography," PhD Thesis, Dept. of Computer Science, Technical Univ. of Darmstadt, 2002.

[23] E. Konstantinou, Y. Stamatiou, and C. Zaroliagis, "On the Efficient Generation of Elliptic Curves over Prime Fields," In Cryptographic Hardware and Embedded Systems– CHES 2002, Springer-Verlag, LNCS 2523, pp. 333-348, 2002.

[24] E. Savas, T. A. Schmidt, and C. K. Koc, "Generating Elliptic Curves of Prime Order," In Cryptographic Hardware and Embedded Systems –CHES 2001, Springer-Verlag, LNCS 2162, pp. 145-161, 2001.

[25] T. N. Shankar and G. Sahoo, "Cryptography with Elliptic Curves," International Journal Of Computer Science And Applications Vol. 2, pp. 38-42, 2009.

[26] A. Majd, Sh. Lotfi, and G. Sahebi, "Review on Parallel Evolutionary Computing and Introduce Three General Framework to Parallelize All EC Algorithms," The 5th Conference on Information and Knowledge Technology, IEEE, pp. 61-66, 2013.

[27] E. Konstantinou, and C. Stamatiou, "Efficient Generation of Secure Elliptic Curves," International Journal of Information Security, Springer, pp. 47-63, 2006.

[28] W. Stein, "Elementary Number Theory: Primes, Congruences, and Secrets," ISBN 9 78-0-387-85525-7, Springer, 2008.

[29] J. Hoffstein, J. Pipher, J. H. Silverman, "An Introduction to Mathematical Cryptography," ISBN: 978-0-387-77993-5, springer, 2008.

[30] A. Majd and G. Sahebi, "A Survey on Parallel Evolutionary Computing and Introduce Four General Frameworks to Parallelize All EC Algorithms and Create New Operation for Migration," Journal of Information and computing Science, Vol. 9, pp. 97-105, 2014.

[31] M. Amara and A. Siad, "Elliptic Curve Cryptography and its Applications," 7th international Workshop on Systems, Signal Processing and their Applications (WOSSPA), 2011.

[32] C. A. Coello, G. B. Lamont, and D. A. Van Veldhuizen, "Evolutionary Algorithms for Solving Multi-Objective Problems" ISBN 9 78-0-387-33254-3, Springer, 2007.

[33] S. Nesmachnow, H. Cancela, and E. Alba, "A parallel micro evolutionary algorithm for heterogeneous computing and grid scheduling," Applied Soft Computing 12, Elsevier, pp. 626-639, 2012.

[34] D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning, "Machine Learning" ", 1988.

[35] T. Back, "Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms", Oxford University Press Oxford, 1996.

[36] B.S. Everitt, "The Cambridge Dictionary of Statistics", ISBN 0-521-81099-X, 2003.