The ISC Int'l Journal of Information Security

ISeCure

January 2015, Volume 7, Number 1 (pp. 47–58)

http://www.isecure-journal.org

Optimizing Image Steganography by Combining the GA and ICA

Faramarz Sadeghi [1,*], Fatemeh Zarisfi Kermani [1], and Marjan Kuchaki Rafsanjani [1]

[1] *Department of Computer Science, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran*

ARTICLE INFO.

Article history:
Received: 27 December 2014
Revised: 20 June 2015
Accepted: 29 July 2015
Published Online: 1 August 2015

Keywords:
Combinatorial Optimization, Discrete Imperialist Competitive Algorithm, Genetic Algorithm, Steganography, Pair-wise Least Significant Bit matching, Space Filling Curves.

ABSTRACT

In this study, a novel approach which uses combination of steganography and cryptography for hiding information into digital images as host media is proposed. In the process, secret data is first encrypted using the mono-alphabetic substitution cipher method and then the encrypted secret data is embedded inside an image using an algorithm which combines the random patterns based on Space Filling Curves (SFC) and the optimal pair-wise LSB matching method. We employ a modified Imperialist Competitive Algorithm by Genetic Algorithm operations, namely Discrete Imperialist Competitive Algorithm (DICA), to perform the optimal pair-wise LSB matching method and find the suboptimum adjustment list. The performance of the proposed method is compared with other methods with respect to Peak Signal to Noise Ratio (PSNR). The PSNR value of the proposed method is higher than the state-of-the-art methods by almost 4dB to 5dB.

1 Introduction

Nowadays, with the growth of computer networks and internet-based applications, digital communication has become an essential part of most infrastructure. The convenient and timely usage of on-line services and easy data sending through the Internet is useful for individuals as well as organizations. However, the public networks such as Internet are not reliable because of copyright violation, counterfeiting, forgery and fraud. Therefore,confidentiality and security of digital communications are serious issues in sharing top secret information through a public channel. Cryptography and information hiding are two major techniques for safe communication. Cryptography is a method of secret communication in which sensitive information are written in meaningless forms so that only the trusty receiver can recover them [1–4], but it may cause suspicion. Information hiding, including watermarking and steganography schemes [5], consists concealing secret data into digital multimedia such as text, image, sound and video files (called cover/host media) so that the secret data is invisible and undetectable [6]. Watermarking is the practice of imperceptibly altering a piece of data in order to embed information [7]. The steganography is the art and science of using each digital multimedia to embed secret data in an undetectable way. We note that the secret data does not need to relate to the host media [8–10].

In image steganography, a digital image is selected as a digital media cover to hide a secret message. The image containing a secret message is called stego-image. Steganography methods are categorized into spatial domain and frequency domain methods. Embedding in frequency domain methods like BSS [11] is carried out by modifying suitably selected coefficients

* Corresponding author.
Email addresses: farsad@uk.ac.ir (F. Sadeghi), fzarisfi@mail.uk.ac.ir (F. Zarisfi Kermani), kuchaki@uk.ac.ir (M. Kuchaki Rafsanjani).

ISSN: 2008-2045

in a transform domain such as DFT [1], DWT [2], and DCT [3]. On the other hand, methods like OLSB [12], OPAP [13], IP [14], HSV [15], PVD [16] and SM [17] in spatial domains work directly with pixels. The most common and simple method for spatial domains is to manipulate the bit planes of each pixel of the cover image. Because these methods mostly use the least significant bit plane of each pixel for the embedding process, they are called LSB-based steganography methods. We can divide LSB methods into two sets: 1) LSB replacement, and 2) LSB matching. In the first set, the LSBs of each pixel store the bits of secret message. The methods of the first set can be easily implemented, but the distortion of the stego-image produced by them, comparing with other methods, is very high. This is a very important problem because the quality of stego-image is an important criterion for evaluating the performance of a steganography method. Methods such as OLSB have been proposed to solve this problem using different optimization algorithms like GA [4] [12] or CSO [5] [18] to obtain an optimal LSB substitution matrix. OPAP [13] is another method that improves the quality of stego-image by adjusting the pixel value after hiding secret data and in IP [14] each section of secret data is either inverted or not, in order to achieve the best quality of stego-image. However, they are not very successful because they use the raster scan to select the cover pixel. The ARIP method [19] is a combination of OPAP and IP and uses the random scan based on Space Filling Curves (SFC) to select suitable cover pixels for the embedding process. ARIP has the best quality stego-image and higher embedding capacity than the older methods but in compression with LSB matching methods does not perform very well.

In this article, we propose a novel steganography method that uses LSB matching method and random scan based on Space Filling Curves to improve the performance of the ARIP method based on the MSE and PSNR parameters. In this method, the cover image is divided into a non-overlapping blocks of equal size and then the part of ciphered secret data, using the mono-alphabetic substitution cipher algorithm, is the same data embedded repeatedly according to four random scans based on Space Filling Curves (Hilbert-SFC, Moore-SFC, ZigZag1-SFC, Z-SFC) and the optimal pair-wise LSB matching method. The best scan with minimum distortion i.e. minimum Mean Square Error (MSE) and maximum Peak Signal to Noise Ratio (PSNR) for each block of cover image are identified and fixed. A binary code (00, 01, 10, and 11) is assigned to each of the four random scans.

The binary code of the fixed scan and the suboptimum adjustment list, which is obtained by using the Discrete Imperialist Competitive Algorithm (DICA), for each particular block of cover image are stored as the key. Although encrypting the secret data has no effect to the visual quality of the stego-image but in order to make access to secret data for hackers difficult, it is better to add the cryptography step before the embedding stage in steganography methods. Thus we combine a cryptography method and our proposed steganography method in this article. The rest of this paper is organized as follows: in Section 2, some works that are relevant to our propose are mentioned. Also, three necessary preliminaries are introduced and described in Section 3. In Section 4, the proposed steganography algorithm according to the combinational method based on random scans and DICA is discussed. The experimental results are given in Section 5 and in Section 6, the complexity and performance of the proposed method is analyzed. Finally, Section 7 and 8 summarize the main conclusions and suggest the further works, respectively.

2 Related Work

Since the proposed method is categorized in the spatial domain, in this section we go over some steganography methods that try to improve the methods on this domain like LSB replacement and LSB matching.

In the LSB replacement methods, the LSBs of the cover image pixel bits are exchanged directly by the secret information bits. In VRS [20, 21] has been used a pixel's dependency on its neighborhood to find out the smooth area and sharp area in cover image and to estimate the amount of secret data to be embedded into an input cover pixel. Also in [22], all of the image pixels have been classified into 8 regions and then the 8 distinct ordering coding have been applied to each region by the developed partial optimization encoder. Thus, the most effective ordering, means that the minimum number of bits has been altered to each region, is obtained. These methods try to improve the security and guarantee to construct the final stego image with minimum distortion in compare with simple LSB replacement. Another way, which has been proposed to improve the performance of simple LSB replacement method, is the method of randomized process and optimal LSB substitution [12, 18]. Farther, finding an optimal substitution matrix in this method, which can help decrease the differences and increase similarities with cover image, can be considered as an optimization problem. In [23] we proposed one compound meta-heuristic method, by

[1] Discrete Fourier Transform
[2] Discrete Wavelet Transform
[3] Discrete Cosine Transform
[4] Genetic Algorithm
[5] Cat Swarm Optimization

using both PSO [6] and SA [7] algorithms, to find optimal substitution matrix.

Therefore, first we found an optimal substitution matrix by use of PSO-SA algorithm and encoded secret image by using the obtained matrix. Finally, we embedded the encoded secret image into cover image by replacing. Also in [24] a semi-reversible data hiding method has been proposed. So that, firstly interpolation methods are used to scale up and down the cover image before hiding secret data for a higher capacity and quality. Secondly, the LSB substitution method is used to embed secret data.

In the LSB matching methods, if the secret bit does not match the LSB of the cover pixel, the value of the cover pixel is randomly increased/decreased and otherwise is not changed. In the last years, the more researchers tried to improve and modify this strategy with maintaining its principles. In [25] the complexity of each cover image pixels is calculated by a complexity measure based on a local neighborhood analysis and a threshold. Then suitable pixel is chosen and if its LSB value is not matched with secret bit then the pixel value is randomly changed. In this article we tried to find and determine the secure location of the cover images that are more suitable for embedding. In [26] the Markov chain distance based on the second-order statistics has been chosen as the security metric, for this reason two improved LSB matching method have been proposed to make this security metric as small as possible. In the first method, the empirical Markov transition matrix of a cover image and the pseudorandom number generated by a pseudorandom number generator are used to determine the modification directions of randomly added/subtracted by 1. In the second method, the Genetic Algorithm (GA) is used to find the optimum matching vector. In [27] the LSB matching revised image steganography has been expanded and an edge adaptive scheme which can select the embedding regions according to the size of secret message and the difference between two consecutive pixels in the cover image has been proposed. Also in [28, 29] researchers tried to modify pair-wise LSB matching method (Mielikainen's method) to improve the visual quality of the stego images and then increase the security. Therefore, they applied the quantum gravitational search algorithm (QGSA) and Genetic Algorithm to search an optimal solution among all the permutation orders.

3 Preliminaries

In this section, we discuss three main requirement processes. Section 3.1 presents the data hiding method based on simple LSB matching [4] and improved methods of its like Mielikainen method [30] (is called pair-wise LSB matching) and optimal pair-wise LSB matching [6], respectively. Section 3.2 describes the Imperialist Competitive Algorithm (ICA) and Section 3.3 introduces the Space Filling Curves (SFC).

3.1 Simple LSB Matching

In this method, instead of always replacing the LSBs of the cover image pixels with secret bits, if the secret bit does not match the LSB of the cover pixel, the value of the cover pixel is randomly increased/decreased, otherwise the pixel value of the cover image is not change. Because, the numbers of pixels that are changed in this method are large, the analytic methods such as the proposed method in [31], which are based on the center of mass (COM) of the histogram characteristic function (HCF), can detect the existence of embedded secret data in stego-images. To solve this problem, the pair-wise LSB matching method is proposed.

3.1.1 Pair-wise LSB Matching

Mielikainen in [30] modified the simple LSB matching method which simultaneously considers two pixels of the cover image (x_i, x_{i+1}) and two secret bits (m_i, m_{i+1}) as described in Figure 1 and each time obtains two pixels of the stego-images (y_i, y_{i+1}) so that, the value of the i^{th} secret bit, m_i, is compared with the LSB of the cover image's i^{th} pixel, x_i, and the value of the $(i+1)^{th}$ secret bit, m_{i+1}, is compared by the function of x_i and x_{i+1} as follows:

$$f(x_i, x_{i+1}) = LSB\big(\lfloor x_i/2 \rfloor + x_{i+1}\big) \qquad (1)$$

The visual quality of stego-image obtained by the pair-wise LSB matching method is better than that obtained by simple LSB matching method, but it can be improved by finding the optimum adjustment list of secret bit pairs that has the best matching with cover pixel pairs. That it is explained in the following section.

3.1.2 Optimal Pair-wise LSB Matching

This method assigns a score to each case of the ordinary pair-wise LSB matching method as follows:

- Case 1: If the pixel pairs of the cover image is copied to the stego-image. The score is T_1.
- Case 2: If one pixel of the cover image is unchanged and the other one is modified randomly then, the score is T_2.
- Case 3: If one pixel of the cover image is unchanged and the other one is modified deterministically then, the score is T_3.

[6] Particle Swarm Optimization
[7] Simulated Annealing

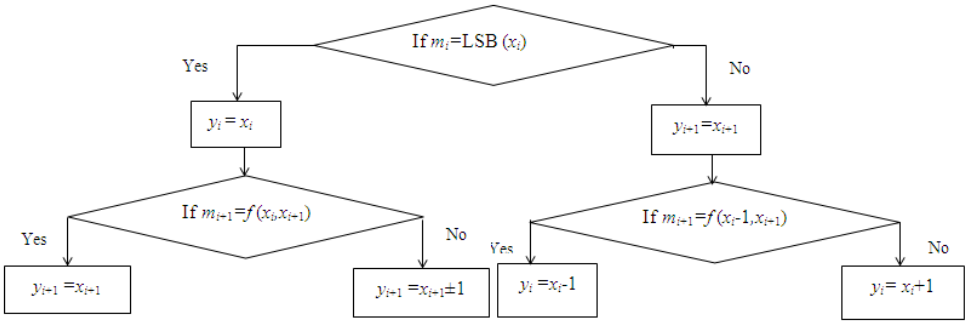

Figure 1. The flowchart of Meilikainen's method.

Since it is better to keep the cover pixels unchanged, we must have $T_1 > T_2 > T_3$.

The score of each cover pixel pairs and secret bits is now calculated and the score matrix is obtained as follows:

$$M_{L_s \times L_h} = \{m(i,j) | m(i,j) \in \{T_1, T_2, T_3\}, 1 \leq i \leq L_s, 1 \leq j \leq L_h\} \tag{2}$$

where L_s and L_h are whole numbers of secret stream pairs and also cover stream pairs, respectively and $m(i,j)$ is the score matching of i^{th} pair of secret streams and j^{th} pair of cover streams. In what follows we assume that $L_s = L_h$.

After obtaining the score matrix M, the adjustment list $J = \{j_1, j_2, \ldots, j_k, \ldots, j_{L_s}\}$ is generated by selecting L_s elements and so that just one element of each row and column of score matrix M is selected. After generating the adjustment list, the total score is calculated as follows:

$$f(j) = \frac{1}{f^M} \sum m(i,j), \qquad f^M = L_S \times T_1 \tag{3}$$

where L_s is whole numbers of secret stream pairs, $m(i,j)$ is the score matching of i^{th} pair of secret streams and j^{th} pair of cover streams and T_1 is the maximum assigned score to each case of the ordinary pair-wise LSB matching method.

It is obvious that different adjustment lists can be produced by selecting elements of score matrix M differently so an adjustment list with higher total score is the best adjustment list that is called suboptimum adjustment list. The stego-image produced by using this suboptimum adjustment list is closer to the cover image and has higher visual quality in comparison with two previous methods. Therefore, the suboptimum adjustment list can be obtained by applying an optimizing algorithm. The method proposed by Xu [6] uses an Immune Programming (IP) strategy to achieve the suboptimum adjustment list using total score as the criterion for optimization.

3.2 Imperialist Competitive Algorithm

The Imperialist Competitive Algorithm (ICA) is a new socio-politically motivated global search strategy that has recently been introduced for dealing with different optimization tasks [32]. This evolutionary optimization strategy performs well with regard to both convergence rate and global optimum achievement [33] so, ICA as an optimization algorithm is used in many different situations such as industrial planning, resource allocation, scheduling, pattern recognition and image encryption.

As with other evolutionary algorithms, this algorithm starts with an initial population. Each individual of the population is called a *country*. There are two types of countries: imperialist states and colonies. Initially all the colonies are divided among the imperialists based on their power. Imperialists with more power possess more colonies. The imperialist states together with their colonies form empires.

After forming initial empires, the colonies in each empire start moving toward their relevant imperialist country (assimilation policy). A sudden change in power, organizational structures or other socio-political characteristics in a country is called a revolution. This process often occurs after assimilation and causes some colonies in each empire randomly to change their positions. The percentage of colonies in each empire that are involved in the revolution process is called the revolution rate. Consequently, after executing two steps i.e. assimilation and revolution, all colonies take up new positions. It is possible that a colony of an empire reaches a better position than the imperialist so that their positions are reversed and the algorithm is continued by the imperialist in the new position.

The total power of an empire depends on both the power of the imperialist country and the power of its colonies. This fact is modelled by defining the total power of an empire as the power of the imperialist plus a percentage of mean power of its colonies and is

calculated according to Equation 4.

$$T.C_n = cost(imperialist_n) + \zeta mean\{cost(colonies of empire_n)\} \quad (4)$$

where $T.C_n$ is the total cost of the n^{th} empire and ζ is a small positive number in $[0, 1]$. In the final step, we have a competition among the imperialists. In this step, all empires try to take the possession of colonies of other empires and control them. The imperialistic competition gradually brings about a decrease in the power of the weaker empires and an increase in the power of the more powerful ones. This is modelled by picking some (usually one) of the weakest colonies of the weakest empire and starting a competition among all empires to possess these colonies. Based on their total power, each empire will have a likelihood of taking possession. It is not certain that these colonies will be possessed by the most powerful empires, but these empires will be more likely to possess them.

Several criteria can be used to stop the algorithm. One idea is to use a number of maximum iterations of the algorithm, called maximum decades, to stop the algorithm. Or, the end of the imperialistic competition, when there is only one empire, can be considered as the stop criterion of the ICA. The algorithm can also be stopped when its best solution in different decades cannot be improved for several consecutive decades.

The flowchart of the Imperialist Competitive Algorithm is shown in Figure 2

3.3 Space Filling Curves

A Space Filling Curve is a one-to-one mapping from an N-dimension space to a one-dimension space which traverses just once through all points in an arbitrary dimensional space. SFCs are used in algorithms of image processing based on the spatial coherence of nearby pixels [35] such as compression [36], encryption [37] and hiding information [37].

In the field of image processing, scanning with space filling curves can be identified as a mapping from a two dimensional (2-D) image plane to a 1-D pixel sequence [35] so that, the resulting sequence of pixels are processed as required the final image is obtained when the (possibly modified) pixel sequence is placed back in a frame alongside the same SFC [37]. Thus, every curve has two free ends which may be joined with other curves. The basic curve is the order 1. To derive a curve of order i; each vertex of the basic curve is replaced by the curve of order $i-1$; which may be appropriately rotated and/or reflected to fit the new curve [39]. In this article, we use four famous SFC in orders 2 and 3, namely Hilbert-SFC, Peano-SFC (Z-SFC), Zigzag-SFC and Moore-SFC, in the embedding phase. The Hilbert curve traverse three points in a straight line and then are turned around. These curves are suitable for approximating square images. Peano or Z-SFC is another kind of Space Filling Curves introduced by Peano in 1890 [40]. Moore-SFC is similar to the Hilbert curve, but the rotation of basic curve in each quadrant of higher orders is different. Also, Zigzag-SFC is similar to the Peano curve but traverses an image in the neighborhood of leading diagonal. Figure 3 depicts the order 3 of Hilbert, Peano, Moore and Zigzag SFCs, respectively.

4 The Proposed Method

In this article, we combine the advantages of Space Filling Curves and optimal pair-wise LSB matching method to obtain a stego-image with a quality better than the existing methods. In additional, we use an encryption stage over the secret bit sequence before embedding stage to increase hardness of accessing to secret data. In the other words, the proposed steganography method in this article can be divided into three phases:

- Phase 1: Ciphering secret bit sequence by mono-alphabetic substitution cipher algorithm.
- Phase 2: Embedding the ciphered secret bit sequence according to optimal pair-wise LSB matching method and Space Filling Curves.
- Phase 3: Extracting the secret bit sequence from the stego-image and deciphering it.

4.1 Ciphering Phase

In this phase, the secret bit sequence S is ciphered by using the mono-alphabetic substitution cipher algorithm [41] which is a bijective (i.e. one-to-one and onto) mapping function. So that, the bit locations in S are numbered sequentially from 0 to $n-1$, where n is the secret data sequence size. Finally, the new stream S' is made by changing the each of bit location x in S to the new location $g(x)$ according to Eq. (5)

$$g(x) = (k_0 + k_1 x)\ mod\ n \qquad gcd(k_1, n) = 1 \quad (5)$$

where k_0 and k_1 are two constants used as keys, and $gcd(.,.)$ means the greatest common divisor. This randomization process can produce a meaningless stream S' which is difficult to detect unless the receivers understand the cipher process and own the two keys (i.e., k_0 and k_1).

4.2 Embedding Phase

In this phase, we combine the SFCs and optimal pair-wise LSB matching method to embed the ciphered secret bit sequence S' at a cover image C by using the algorithm that is shown in pseudo code of Algorithm 1.

According to the brief discussion in the Section 3.1,

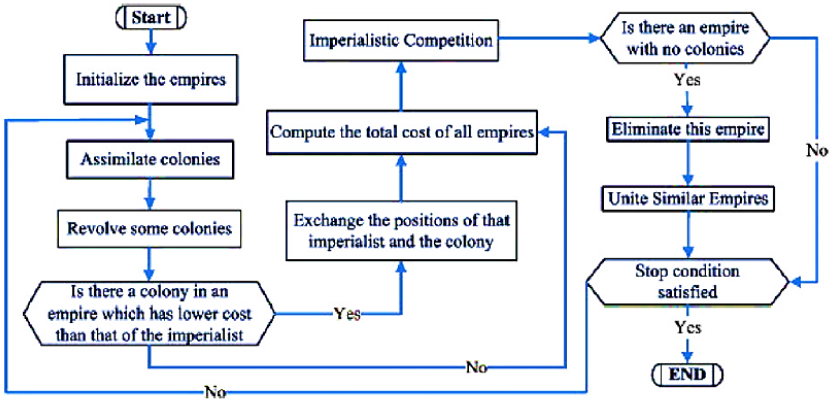

Figure 2. Flowchart of the Imperialist Competitive Algorithm [34].

Algorithm 1 Pseudo code of embedding phase.

Function EMBEDDING(array $C[m][n]$, array $S'[1][65536]$,array $RP[1][k]$)
  Divide the cover image C with size $m \times n$ into N blocks with size k. // according to the order of SFCs, block size is 4×4 or 8×8
  for $i = 1$ to $i = N$ **do** // i is the block counter.
    for $RP = 1$ to $RP = 4$ **do** // RP identifies the random pattern corresponding to the one of the SFCs
      Traverse the i^{th} block according to random pattern RP and obtain sequence of pixels, $C'[k]$.
      Divide $C'[k]$ and k bits of the ciphered secret bit sequence (i.e. $S'[k]$) into non-overlapping pairs and obtain the sequence of pixel pairs $C''[k/2]$ and the sequence of secret bit pairs $S''[k/2]$.
      Compute the score matrix $M[k/2][k/2]$, as described in Section 3.1, by using the two sequences $C''[k/2]$ and $S''[k/2]$.
      Recall the described DICA in Section 4.2.2 with $C''[k/2]$, $S''[k/2]$ and $M[k/2][k/2]$ as its inputs and return the $best-order[RP][k/2]$, as its output.
      Sort the sequence $S''[k/2]$, according to $best-order[RP][k/2]$ and obtain the sorted sequence $S'''[k/2]$.
      Perform the pair-wise LSB matching method, as shown in Figure 1, for sequences $C''[k/2]$ and $S'''[k/2]$and obtain the sequence of stego pixels $ST[RP][k]$.
      Retrieve the sequence of stego pixels $ST[RP][k]$ according to random pattern RP to achieve the i^{th} block of stego-image.
      Compute $MSE[i][RP]$.
    end for
    $MSE[i] = min(MSE[i][RP])$.
    Assign the value of $key[i,1]$ and $key[i,2]$ based on Table 1;
    Assign key $[i, 3 \ldots 2+k/2] = best-order[RP][k/2]$; // best-order is the output of DICA for the RP that is corresponding to the SFC with minimum MSE value.
    Obtain the pixels of the i^{th} block of stego-image H based on $ST[RP][k]$ and random pattern RP;
    $P = P - k$; // P is equal to the remaining length of ciphered secret bit sequence after embedding k bit.
    if $P > 0$ **then**
      continue;
    else
      break;
    end if
  end for
end Function

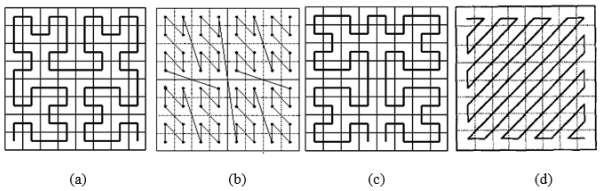

Figure 3. Space Filling Curves (a) Hilbert-SFC, (b) Peano-SFC, (c) Moore-SFC and (d) Zigzag-SFC.

the matching method is based on the adaptation of the least significant bit of pixel pairs of the cover image with bit pairs of the secret bit sequence. For this purpose, it is enough that the 2-D image (i.e. cover image) is transformed to a 1-D pixel sequence. In this article, we use four famous SFCs in orders 2 and 3 for scanning the cover image and finding the best arrangement of cover pixel pairs that are suitable for applying the optimal pair-wise LSB matching method. Thus outset, we divide cover image into blocks so that the size of blocks are equal with order of SFCs and then for identifying that which type of space filling curves that are shown in Figure 3 is suitable for each block of cover image, we assign the binary codes 00, 01, 10, 11 to the each of them, respectively. Finally, a particular binary code will be fixed for each block based on finding the best arrangement of pixel pairs for each of them.

Table 1. Allotted binary codes to each of space filling curves.

RP	key$[i,1]$	key$[i,2]$
1	0	0
2	0	1
3	1	0
4	1	1

In this article to find the suboptimum adjustment list that can produce the best quality stego-image, we

modify the standard Imperialist Competitive Algorithm into discrete version of it as is explained in the coming section.

4.2.1 Discrete ICA (DICA)

We know that the standard ICA, which was described in Section 3.2, is widely used for solving continuous optimization problems. Because it is a new optimization algorithm and has higher convergence speed than existent optimization algorithms, researchers have tried to modify this method for solving discrete optimization problems such as the problem described in this study. So, for making discrete ICA that is suitable for solving this problem, we applied two operators of Genetic Algorithm i.e. double point crossover and mutation instead of assimilation and revolution in standard ICA, respectively.

4.2.2 Finding the Suboptimum Adjustment List by Using DICA

Based on the brief discussion in the Section 3.1, a suboptimum adjustment list has the highest total score as calculated according to Equation 3. Since the arrangement of the secret bit pairs are changed according to this list, therefore it is possible that the produced stego-image according to an adjustment list with higher total score has more distortion in comparison with the produced stego-image according to an adjustment list with lower total score. The visual quality of gray level images can be computed based on two parameters namely PSNR[8] and MSE[9] as follows:

$$PSNR = 10 \times \log_{10}\left(\frac{255^2}{MSE}\right) \tag{6}$$

$$MSE = \frac{1}{W \times H}\sum_{i=1}^{W}\sum_{j=1}^{H}\left(S_{ij} - C_{ij}\right)^2 \tag{7}$$

where, S_{ij} and C_{ij} are respectively the intensity of ij pixel of the stego-image and the cover image and W and H are the pixel numbers for the width and the height of the cover image, respectively.

The proposed discrete ICA in this study tries to optimize two objective functions simultaneously: the total score of an adjustment list and the visual quality of stego-image that is produced by using that adjustment list. Therefore, fitness function of DICA will contain both. Since the DICA is a multi-objective optimization algorithm, one of the traditional approaches called weighting method, which aggregates the various objectives into single objective, is used. Hence, we shall optimize:

$$F = 0.5 \times f(J) + 0.5 \times PSNR \tag{8}$$

where, $f(J)$, the total score of an adjustment list is the first objective and $PSNR$, one of the criteria for measuring the visual quality of the produced stego-image, is the second objective. Since both objectives have the same importance, the assigned weight value for each of them is equal. In this subsection, the details of performing the steps of DICA are described.

1) Initialization

In order to construct the initial empires, at the outset by using one of the four SFCs that are shown in Section 3.3, we convert the cover image to sequence of pixels, then divide the sequence of pixels and sequence of secret bits into non-overlapping pairs and generate the score matrix M with dimensions $L_s \times L_h$ according to Section 3.1. So a country is defined as follows:

$$country = [c_1, c_2, \ldots, c_l], \qquad l = 1, 2, \ldots L_h \tag{9}$$

where c_l is the position of the l^{th} secret bit pair in the score matrix M.

After generating all countries by randomly selecting different elements of the score matrix M, we compute the fitness value according to Equation 8. A number of the countries with high fitness value are selected to be imperialist states while, the rest of countries form the colonies. These colonies are shared between imperialists based on their power, so that the more powerful imperialist state has more colonies.

2) Assimilation Policy

The assimilation strategy in DICA is completely different from the standard ICA. In each empire, a double point crossover operator of Genetic Algorithm is applied to each imperialist and colonies as the parents resulting in off-spring. Notice that the off-springs are always distinct but they may not be valid. To illustrate, a simple example is presented as follows:

Imperialist country:	1	8	7	4	6	2	3	5
Colony country:	1	7	8	4	5	3	6	2

Parents

First new country:	1	8	8	4	6	2	6	2
Second new country:	1	7	7	4	5	3	3	5

Off-springs

As shown above, both new countries as the off-springs are invalid because there are elements that are repeated while others are missing. In order to solve this problem, we select a repeated element randomly and replace it with a missing element as follows:

First modified new country:	1	**7**	8	4	6	**5**	**3**	2
Second modified new country:	1	7	**8**	4	5	**6**	3	**2**

Checking and modifying as a validation procedure

[8] Peak Signal to Noise Ratio
[9] Mean Square Error

are performed after generating off-spring. Finally, the fitness value of the new countries (off-springs) is computed based on Equation 8 and then the off-spring with higher fitness value is selected as the new colony country, replacing its parent colony.

3) Revolution

As has been said, the revolution policy in DICA is different from the standard ICA. In each empire, colonies are randomly selected according to the revolution rate and the mutation operator of Genetic Algorithm is applied. In this step, in order to inhibit generation of invalid countries, two random points are selected and the values are interchanged. This strategy is shown below.

1	7	8	4	6	2	3	5

6	7	8	4	1	2	3	5

4) Exchange the Position of Imperialist and Colonies

During any movement corresponding to assimilation policy or revolution process, it is possible that a colony of the empire reaches a better position than the imperialist. In this case, the algorithm will continues based on a new imperialist.

5) Imperialist Competition

After computing the total power of each empire according to Equation 4, the weakest colony (or colonies) of the weakest empire is selected by other empires and the competition to possess this colony begins. Each imperialist participating in this competition, according to its power, has a chance of possessing the colony. To start the competition, at first, the weakest empire is selected and then the possession probability of each empire is estimated based on the normalized total power of an empire according to Equation 10 and Equation 11.

$$N.T.C_n = max_i\{T.C_i\} - T.C_n \tag{10}$$

$$P_{p_n} = \left| N.T.C_n / \sum_{i=1}^{N_{imp}} N.T.C_i \right| \tag{11}$$

where $T.C_n$ is the total power of the n^{th} empire and $N.T.C_n$ is the normalized total power of n^{th} empire. The possession probability of each empire is computed according to Equation 11.

In order to divide the given colonies among the empires, a vector P is formed as follows:

$$P = [p_{p_1}, p_{p_2}, \dots p_{p_{N_{imp}}}] \tag{12}$$

Now vectors R and D same size as P are defined. Each element of R is random number between 0 and 1.

$$R = [r_1, r_2, \dots r_{N_{imp}}] \quad r_1, r_2, \dots r_{N_{imp}} \approx U(0,1) \tag{13}$$

$$D = P - R = [D_1, \dots, D_{N_{imp}}] = [p_{p_1} - r_1, \dots p_{p_{N_{imp}}} - r_{N_{imp}}] \tag{14}$$

The empire with maximum $p_{p_i} - r_i$ takes control of the given colony.

During the imperialistic competition, weak empires will slowly lose their power and get weaker. At the end of process, just one empire will remain that governs all colonies.

6) Convergence

Steps 2-5 are repeated until the iteration number is equal to given maximum iteration number or we have found one empire.

4.3 Extraction Phase

In this phase, we try to extract secret bits by using the following algorithm that is shown in pseudo code of Algorithm 2. Finally, for obtaining the original secret bit sequence, we decipher the extracted sequence of this algorithm by using the mono-alphabetic algorithm in Equation 5 and keys k_0 and k_1.

Algorithm 2 Pseudo code of the extraction phase.

Function EXTRACTION(array $H[m][n]$, array $key[N][2+k]$)
  Divide the stego-image H with size $m \times n$ into N blocks with size k. // according to the order of SFCs, block size is 4×4 or 8×8
  i=1 // i is the block counter.
  while $i \leq N$ **do**
    Obtain the RP value according to values of $key[i,1]$ and $key[i,2]$ which have been shown in Table 1.
    Traverse the i^{th} block of stego-image H according to random pattern RP and obtain a pixel sequence $H'[k]$.
    Divide $H'[k]$ into non-overlapping pairs and obtain the sequence of pixel pairs $H''[k/2]$ and perform the procedure shown in Figure 4 and obtain the sequence of secret bit pairs $E[k/2]$.
    Sort the array $E[k/2]$ according to values of $key[i, 3 \dots 2+k/2]$ and obtain k bits of the embedded ciphered secret sequence in i^{th} block.
    $i = i + 1$
  end while
end Function

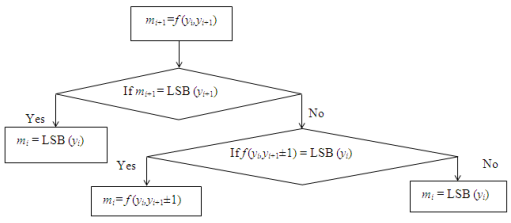

Figure 4. Extraction procedure.

5 Experimental Results

According to our researches, the steganography methods based on LSB matching are better than the methods based on LSB replacement in embedding the confidential data in cover images. To prove this claim, we use four images of 256 × 256 pixels with 8-bit gray level that is shown in Figure 5 as cover images and the random stream of 0s and 1s as secret data. As we know, the visual quality of stego-image and embedding capacity of cover image are two important criteria in evaluating a steganography method. Because the embedding capacity of the mentioned methods are identical so, our comparison is focused on the visual quality of stego-image. To perform this comparison,

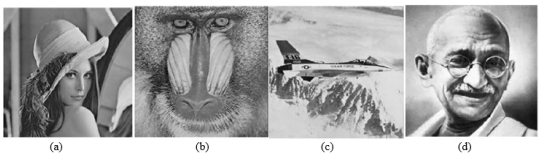

Figure 5. The cover images in our experiment, (a) Lena, (b) Baboon, (c) Jet and (d) Gandhi.

we divide the cover image into 4 × 4 or 8 × 8 blocks and use the Z-SFC, Hilbert-SFC, Zigzag-SFC and Moor-SFC scans and the adaptive selection of them for a particular block of cover image. After selecting the best scan for the particular block, we apply the each of three steganography methods based on LSB matching. So, to demonstrate the enhanced quality of the stego-image through the steganography methods based on LSB matching, the obtained results of them are compared with the results of the OPAP [13], IP [14] and ARIP [19] methods based on LSB replacement, as shown in Table 2.

In Table 2, it can be shown the MSE and PSNR values produced by ARIP method are better than the OPAP and IP methods. This enhancement in the quality of the produced stego-image is because of the adaptive selection of a SFC scan for each of blocks based on the nature of the secret data. But finding the suitable SFC scans, cannot be useful. The important reason of this event is absolute changes of the LSBs of the cover pixels that are different from the secret bits. So, trying to reduce these changes will be absolutely effective in the visual quality of the produced stego-image. Two methods namely simple LSB matching and pair-wise LSB matching were designed to certify this suggestion. Thus it can be seen that ARIP method in comparison with these methods is weaker.

Although the matching procedures produce the stego-image with the almost good visual quality, but finding the pixels of cover image which the LSBs of them have the best matching with the secret bits will be surely useful for obtaining the stego-image with the more good visual quality. So finding the suitable SFC scan for each particular block and as well as applying the matching procedures for embedding phase is the base idea of this article, so this idea looks as an optimization problem. So, to solve this optimization problem we use the modified Imperialist Competitive Algorithm namely DICA and set the parameters of it as shown in Table 3. As a result, detecting the best list of the most consistent of the cover pixels with the secret bits, in addition to finding the suitable SFC scans and applying the matching procedures for each particular block, can be caused the significant enhancement in the obtained PSNR values of the proposed method in comparison with the other mentioned methods in this article.

In general, we observe in Table 2 that embedding the data in the cover image with more details like image Baboon, produce the stego-image with higher visual quality. So, decrease in details of the cover images like image Gandhi, make them unsuitable for embedding process. Also we see that increase in block size (of 4 × 4 and 8 × 8) causes decrease in the visual quality of stego-image.

6 Complexity and Performance Analysis

Although because of slow convergence of DICA, finding a suboptimum adjustment list by using DICA is time-consuming, but we can guarantee that the proposed stego technique provides the confidential image with highly secure protection. In order to validate this statement, the following analysis has been performed.

For the 4 × 4 blocks, the total number of same size and non-overlapping blocks for each image with 256 × 256 pixels are 4096. So a hacker with no information about the keys and by using the brute force strategy should be done 8! × 4 work to find the suboptimum adjustment list and the optimum random pattern, respectively for each block. Therefore, the total work that is done to extract the embedded secret information from received image without keys is 4096 × 8! × 4. Similarly, for the 8 × 8 blocks, the complexity level is 1024 × 32! × 4. So this security level estimation reveals the robustness of the proposed stego against hackers.

7 Conclusion

In this paper, we have proposed a steganography method which uses a combination of the four random patterns based on Space Filling Curves with optimal pair-wise LSB matching that tries to find an optimum random pattern for each block of pixels of cover image and confidential data. The optimal pair-wise LSB matching method works according to the idea,

Table 2. The implementation results.

Cover image	Method	MSE	PSNR	MSE	PSNR
		4× 4	4 × 4	8 × 8	8 × 8
Lena	OPAP [13]	—	—	0.5018	51.1251
	IP [14]	0.4002	52.1080	0.4481	51.6171
	ARIP [19]	0.3873	52.2503	0.4346	51.7499
	Simple LSB matching	0.3869	52.2553	0.4371	51.7251
	Pair-wise LSB matching	0.2999	53.3607	0.3354	52.8745
	The proposed method	0.1137	57.5732	0.1349	56.8316
Baboon	OPAP [13]	—	—	0.4996	51.1446
	IP [14]	0.4019	52.0896	04538	51.5622
	ARIP [19]	0.3877	52.2458	0.4456	51.6414
	Simple LSB matching	0.3864	52.2603	0.4359	51.7371
	Pair-wise LSB matching	0.3006	53.3514	0.3369	52.8560
	The proposed method	0.1126	57.6174	0.1340	56.8621
Jet	OPAP [13]	—	—	0.5007	51.1350
	IP [14]	0.4002	52.1080	0.5004	51.1376
	ARIP [19]	0.3851	52.2751	0.4447	51.6501
	Simple LSB matching	0.3889	52.2324	0.4398	51.6985
	Pair-wise LSB matching	0.3013	53.3402	0.3350	52.8799
	The proposed method	0.1149	57.5298	0.1340	56.8589
Gandhi	OPAP [13]	—	—	0.4975	51.1629
	IP [14]	0.3904	52.2157	0.4964	51.1725
	ARIP [19]	0.3789	52.3456	0.4567	51.5345
	Simple LSB matching	0.3839	52.2892	0.4385	51.7110
	Pair-wise LSB matching	0.3006	53.3514	0.3364	52.8617
	The proposed method	0.1156	57.5014	0.1370	56.7654

Table 3. Parameters setting for DICA algorithm.

Parameter	Value
Number of variables (Dimension)	8 and 32 for blocks 4 × 4 and 8 × 8, respectively.
Number of initial country	20
Number of empires	5
Revolution rate	0.1
Iteration number	30

that considering the similarity between secret bits and LSBs of cover pixels and obtaining the best order with highest degree of similarity, can produce a stego-image with high visual quality in comparison with simple LSB matching and pair-wise LSB matching methods. Therefore in this article, we proposed a modified Imperialist Competition Algorithm, namely DICA, which was suitable for solving discrete problems and applying it to find a suboptimum adjustment list. This algorithm considers blocks of the cover image and the suboptimum adjustment list corresponding to each block is reserved as key and is sent to receiver to extract the original secret bits stream. In addition, ciphering the secret data using mono-alphabetic substitution cipher algorithm before embedding process, increases its security.

8 Future Works

The steganography methods can be categorized in the receiver side based on the additional necessary information apart from the stego image. For this reason, the steganography methods are classified into blind and non-blind types. In the blind type, the receiver with only having the stego image can be easily extracted the secret bit sequence. Whereas, in the non-

blind type, for the correct executing the extraction phase in the receiver side, additional information is needed. The steganography methods in blind category are more secure than the other category. As mentioned in Section 6, our proposed method guarantees that stego image has the high security protection, but can convert it into the one of the blind category methods by combining it with one of the blind methods or the lossless compression algorithms and or embedding the key file in the cover image. Also, by altering the optimization method with one of the evolutionary methods such as PSO, COA and SFCs with the other curves such as Spiral, Sweep, Graymight improve the results.

References

[1] Kanso, A., & Own, H. S. (2012). Steganographic algorithm based on a chaotic map. Commun Nonlinear Sci Numer Simulat, 17, 3287-3302.

[2] Chu, Y.H., & Chang, S. (1999). Dynamical cryptography based on synchronized chaotic systems. Electronics Letters, 35, 974975.

[3] Highland, H.J. (1997). Data encryption: a non-mathematical approach. Computer & Security, 16, 369386.

[4] Sharp, T. (2001). An implementation of key-based digital signal steganography. Lecture Notes in Computer Science, 2137, 1326.

[5] Petitcolas, F.A.P., Anderson, R.J., & Kuhn, M.G. (1999). Information hiding a survey. Proc. IEEE, 87, 1062-1078.

[6] Xu, H., Wang, J., & Kim, H. J. (2010). Near-optimal solution to pair-wise LSB matching via an immune programming strategy. Information Sciences, 180, 1201-1217.

[7] Cox, I., Miller, M., Bloom, J., Fredrich, J., & Kalker, T. (2007). Digital Watermarking and Steganography. (2nd Ed.). Burlington: Morgan Kaufmann Publishers, MA.

[8] Simmons, G.J. (1984). The prisoners' problem and the subliminal channel. Proc, Crypto, 51-67.

[9] Cheddad, A., Condell, J., Curran, K., & Kevitt, P.M. (2010). Digital image steganography: Survey and analysis of current methods. Signal Processing, 90, 727-752.

[10] Lou, D.C., Wu, N.M., Lin, Z.H., & Tsai, C.S. (2010). A novel adaptive steganography based on local complexity and human vision sensitivity. the Journal of Systems and Software, 83, 1236-1248.

[11] Sajadi, H., & Jamzad, M. (2010). BSS: Boosted steganography scheme with cover image preprocessing. Expert systems with application, 37, 7703-7710.

[12] Wang, R.Z., Lin, C.F., & Lin, J. (2001). Image hiding by optimal LSB substitution and genetic algorithm. Pattern Recognition, 34, 671-683.

[13] Chan, C.K., & Chen, L.M. (2004). Hiding data in images by simple LSB substitution. Pattern Recognition. 37, 469-474.

[14] Yang, C.H. (2008) Inverted pattern approach to improve image quality of information hiding by LSB substitution. Pattern Recognition, 41, 2674-2683.

[15] Provos, N., & Honeyman, P. (2003). Hide and seek: an introduction to steganography. IEEE Security & Privacy Magazine, 1, 32-44.

[16] Wu, D.C., & Tsai, W.H. (2003). A steganographic method for images by pixel-value differencing. Pattern Recognition Letters, 24, 1613-1626.

[17] Chang, C.C., & Tseng, H.W. (2004). A steganographic method for digital images using side match. Pattern Recognition Letters, 25, 1431-1437.

[18] Wang, Z.H., Chang, C.C., & Li, M.C. (2012). Optimizing least-significant-bit substitution using cat swarm optimization strategy. Information Sciences, 192, 98-108.

[19] Amirtharajan, R., & Bosco, J. (2012). An intelligent chaotic embedding approach to enhance stego-image quality. Information Sciences, 193, 115-124.

[20] Hossain, M., Haque, S. A., & Sharmin, F. (2010). Variable Rate Steganography in Gray Scale Digital Images Using Neighborhood Pixel Information, The International Arab Journal of Information Technology, 7, 34-38.

[21] Ioannidou, A., Halkidis, S. T., & Stephanides, G. (2012). A novel technique for image steganography based on a high payload method and edge detection, Expert Systems with Applications, 39, 1151711524.

[22] Akar, F., Yalman, Y., & Varol, H. S. (2013). Data hiding in digital images using a partial optimization technique based on the classical LSB method, Turkish Journal of Electrical Engineering & Computer Sciences, 21, 2037-2047.

[23] Sadeghi, F., Kuchaki, M., & Zarisfi, F. (2013). Hiding Information in Image by Compound Meta-Heuristic Algorithm PSO-SA, International Journal of Computer Science & Artificial Intelligence, 3, 125-133.

[24] Jung, K. H., & Yoo, K. Y. (2015). Steganographic method based on interpolation and LSB substitution of digital images, Multimed Tools Appl, 74, 21432155.

[25] Sabeti, V., Samavi, S., & Shirani, S. (2013). An adaptive LSB matching steganography based on octonary complexity measure, Multimed Tools Appl, 64, 777793.

[26] Liu, G., Zhang, Z., & Dai, Y. (2010). Improved LSB-matching Steganography for Preserving Second-order Statistics, JOURNAL OF MULTI-

MEDIA, 5, 458 -463.
[27] Luo, W., Huang, F., & Huang, J. (2010) Edge Adaptive Image Steganography Based on LSB Matching Revisited, Information Forensics and Security, IEEE Transactions, 5, 201-0214.
[28] Soleimanpour, M., Nezamabadi, H., M. Farsangi, M. & Mahyabadi, M. (2012).A more secure steganography method based on pair-wise LSB matching via a quantum gravitational search algorithm, 16th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP), Shiraz, Fars, 2-3 May.
[29] Khosravi, M., Soleymanpour, S., & Mahyabadi, M. (2012). Improved pair-wise LSB matching steganography with a new evaluating system, Sixth International Symposium on Telecommunications (IST), Tehran, 6- 8 November
[30] Mielikainen, J. (2006). LSB matching revisited. IEEE Processing Letters, 13, 285-287.
[31] Harmsen, J., & Pearlman, W. (2003). Steganalysis of additive-noise modelable information hiding. Proc, SPIE Security Watermarking Multimedia Contents, 131-142.
[32] Atashpaz, E., & Lucas, C. (2007). Imperialist Competitive Algorithm: An algorithm for optimization inspired by imperialistic competition. IEEE Congress on Evolutionary Computation, 46614667.
[33] Rajabioun, R., Atashpaz-Gargari, E., & Lucas, C. (2008). Colonial Competitive Algorithm as a Tool for Nash Equilibrium Point Achievement. Lecture notes in computer science, 5073, 680-695.
[34] https://en.wikipedia.org/wiki/Imperialist_competitive_algorithm
[35] Koga, T., & Suetake, N. (2013). Image coarsening by using space-filling curve for decomposition-based image enhancement. J. Vis. Commun. Image R, 24, 806-818.
[36] Konrad C, C. (2011). Two-constraint domain decomposition with Space Filling Curves. Parallel Computing, 37, 203-216.
[37] Bhatnagar, G., & Wu, Q.M.J. (2012). Selective image encryption based on pixels of interest and singular value Decomposition. Digital Signal Processing, 22, 648-663.
[38] Feng, G.R., Jiang, L.G., He, C., & Xue, Y. (2006). Chaotic spread spectrum watermark of optimal space-filling curves. Chaos Solitons and Fractals, 27, 580-587.
[39] Chen, H.L., & Chang, Y.I. (2005). Neighbor-finding based on space-filling curves. Information Systems, 30, 205-226.
[40] Peano, G. (1890). Sur une courbe, qui remplit toute une aire plane. Mathematische Annalen, 36, 157-160.
[41] Rhee, M.Y. (1994). Cryptography and Secure Communication. Singapore. McGraw-Hill Book Co Press.

Faramarz Sadeghi received his Ph.D. degree in Applied Mathematics with academic orientation in Computer Science from Shahid Bahonar University of Kerman, Iran in 2005. He is currently assistant professor at the Department of Computer Science in Shahid Bahonar University of Kerman, Iran. His main interests are computational theory, image processing, data mining and text mining.

Fatemeh Zarisfi Kermani received her B.S. Degree and M.S. Degree in Computer Science from Shahid Bahonar University of Kerman, Iran in 2011 and 2013, respectively. She is currently working on his Ph.D. degree in Computer Science at Shahid Bahonar University, Iran. Her research is concentrated on image processing, soft computing, data mining and text mining.

Marjan Kuchaki Rafsanjani received her Ph.D. in Computer Engineering, Iran in 2009. She is currently assistant professor at the Department of Computer Science in Shahid Bahonar University of Kerman, Iran. She published about 120 research papers in international journals and conference proceedings. Her main areas of interest are computer networks (wireless networks, mobile ad hoc networks (MANETs), wireless sensor networks (WSNs)), electronic commerce, artificial intelligence, grid & cloud computing.