Applied Soft Computing 132 (2023) 109847

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Comparative study of metaheuristic optimization algorithms for image steganography based on discrete Fourier transform domain

Anna Melman, Oleg Evsutin *

HSE University, 20 Myasnitskaya Ulitsa, Moscow 101000, Russia

A R T I C L E   I N F O

Article history:
Received 2 April 2022
Received in revised form 10 October 2022
Accepted 14 November 2022
Available online 24 November 2022

Keywords:
Information security
Data hiding
Image steganography
Image processing
Metaheuristic optimization

A B S T R A C T

Image steganography and watermarking have demonstrated their efficiency in creating a covert communication channel and in digital image authentication respectively. The importance of cybersecurity tasks sets high efficiency requirements for information embedding methods. Many studies in this area focus on finding new ways to improve the embedding quality. One of them is to use metaheuristic optimization algorithms. The use of classical metaheuristics is common enough for a data hiding area. However, a lot of new metaheuristics have been proposed over the past few years. Their applicability has not yet been evaluated in the field of steganography and digital watermarking. In this paper, we present a comparative study on the efficiency of seven metaheuristic optimization algorithms for finding the best embedding options in the phase spectrum of the Discrete Fourier Transform (DFT). One of the contributions of our research is an improved DFT-based data hiding algorithm, which is a development of the error-free embedding algorithm we obtained earlier. The new algorithm also provides error-free extraction of embedded data, and it is distinguished by the high invisibility of embedding through the use of metaheuristic optimization. The use of metaheuristic optimization led to an increase in the PSNR value by an average of 2%–6% and an increase in the capacity value by an average of 16%–25%. Another important contribution of our research is the original formulation of the optimization problem for different classes of metaheuristics. The experimental results demonstrate the efficiency of metaheuristic optimization for solving data hiding problem. The Differential Evolution and the Particle Swarm Optimization showed the best values of embedding indicators among the classical metaheuristic optimization algorithms. The Gradient-Based Optimizer showed the highest efficiency among modern algorithms. Thus, our research indicates the relevance of further studies of other modern optimization algorithms for a data hiding area.

1. Introduction

One of the promising areas of cybersecurity is data hiding technology including digital steganography and digital watermarking. Steganography algorithms are designed to covertly transfer confidential information between users by embedding the information into digital containers. Watermarks are used to authenticate digital objects and control their integrity. In both cases, data hiding in various multimedia objects, primarily images, is the most common.

There are a large number of data hiding algorithms that differ significantly in various key features. The efficiency of data embedding is often assessed by imperceptibility and capacity criteria. The simultaneous achievement of high values of these indicators is challenging. This is because these indicators are inverse. Embedding information into a digital object is carried out by making changes to its data elements. Each change distorts the digital object. A larger embedding capacity requires more changes to be made, and this results in a higher distortion level. This worsens the embedding imperceptibility.

Thus, researchers in the field of data hiding are faced with the problem of embedding a given amount of information into a digital object with the least distortion. The general way to solve this problem is to select the most appropriate data elements for embedding message fragments.

There are two main methodologies that are used to select the data elements of a cover image. Many researchers propose to choose the best location of message fragments based on local features of the image. Such features include the smoothness of pixel blocks, brightness, contrast, statistical characteristics in the spatial and frequency domains, etc. [1–4]. However, it is difficult to find ideal feature values that provide the best embedding for all images. It is possible to select them for each image, but such a solution often results in additional information that must be transmitted along with the stego image in order to successfully extract the message.

* Corresponding author.
E-mail address: evsutin.oo@gmail.com (O. Evsutin).

https://doi.org/10.1016/j.asoc.2022.109847
1568-4946/

Table 1
Abbreviations used in the paper.

Abbreviations			
GA	Genetic Algorithm	SSIM	Structural Similarity Index Measure
PSO	Particle Swarm Optimization	CI	Cohort Intelligence
DE	Differential Evolution	ForOA	Forest Optimization Algorithm
ABC	Artificial Bee Colony	GWO	Gray Wolf Optimizer
ACO	Ant Colony Optimization	LSA	Lightning Search Algorithm
DFT	Discrete Fourier Transform	HGSO	Henry Gas Solubility Optimization
DCT	Discrete Cosine Transform	ROA	Rain Optimization Algorithm
DWT	Discrete Wavelet Transform	SMA	Slime Mould Algorithm
LSB	Least Significant Bit	MA	Mayfly Algorithm
PSNR	Peak Signal-to-Noise Ratio	HHO	Harris Hawks Optimization
MSE	Mean Squared Error	WCO	World Cup Optimization
QR	Quick Response	CoSO	Community of Scientists Optimization
IWT	Integer Wavelet Transform	AISA	Adolescent Identity Search Algorithm
SVD	Singular Value Decomposition	FBI	Forensic-Based Investigation
BER	Bit Error Rate	SPBO	Student Psychology Based Optimization
FOA	Fruit Fly Optimization Algorithm	GTOA	Group Teaching Optimization Algorithm
SOA	Seeker Optimization Algorithm	SFS	Stochastic Fractal Search
FrFT	Fractional Fourier Transform	LFD	Lévy Flight Distribution
FA	Firefly Algorithm	GBO	Gradient-Based Optimizer

A methodology based on setting and solving an optimization problem is more efficient. The location of the data elements to be changed is considered as a set of arguments of some objective function that has an optimum to be found. This methodology achieves better results, but its application can be difficult in real-time applications since optimization can be computationally expensive.

Within the framework of the second methodology, good results can be obtained by applying metaheuristic optimization algorithms. Metaheuristics direct the search process in the correct direction and allow us to explore the entire search space. The main advantage of metaheuristics is the ability to configure an algorithm for a variety of optimization problems.

The motivation for our research is as follows. There are many metaheuristic optimization algorithms with different features. Authors of data hiding studies often refer to the use of classical algorithms such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Differential Evolution (DE), Artificial Bee Colony (ABC) algorithm, and Ant Colony Algorithm (ACO). However, there is no in-depth analysis of their efficiency for data embedding problems and there is no analysis of the parameter value influence on efficiency in most studies. Besides, many new metaheuristics have been proposed over the past few years. Their efficiency has not yet been evaluated in the field of steganography and digital watermarking. Therefore, an urgent task is to study the applicability of both classical and new metaheuristics of different classes to the problems of increasing the efficiency of information embedding into digital images.

In this paper, we present an extensive experimental study on the efficiency of metaheuristic optimization algorithms for finding the best embedding options in the phase spectrum of the Discrete Fourier Transform (DFT). The main contributions of our research are as follows:

(1) We experimentally evaluate and compare the efficiency of using various metaheuristics to improve the performance of information embedding in the phase spectrum of DFT, both classical ones and those previously not used for the problems of embedding information into digital images; we also experimentally evaluate the influence of the main parameters of metaheuristics on the optimization efficiency.

(2) We formulate our optimization problem in two different ways using binary and real number vectors as individuals of population;

(3) We propose an improved algorithm for embedding information into the DFT phase spectrum using metaheuristics, which is distinguished by error-free extraction of embedded data and high imperceptibility of embedding.

The rest of the paper is organized as follows. Section 2 provides a brief overview of research combining metaheuristic optimization and data hiding. Section 3 describes our algorithm for embedding information into the DFT phase spectrum proposed earlier. In Section 4, we provide a brief overview of metaheuristic optimization algorithms and present the details of those we use for comparison. In Section 5, we propose our scheme for increasing the efficiency of information embedding into the DFT phase spectrum of images using metaheuristic optimization. Section 6 contains the results of the experiments and Section 7 contains their discussion. The Conclusion summarizes the research. Table 1 defines the abbreviations used in the paper.

2. Related work

All algorithms for hiding information in digital images are divided into two classes: spatial domain embedding and frequency domain embedding. Spatial domain embedding algorithms work with the spatial domain of images and frequency domain embedding algorithms hide information in the coefficients of frequency transformations, for example, Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), DFT and others.

The use of metaheuristic optimization is a fairly popular method of increasing the efficiency of data hiding. In this section, we separately consider examples of current research on embedding information in the spatial and frequency domain of images using metaheuristic optimization. We examine how other researchers use metaheuristics in various steganographic embedding and watermarking scenarios. The reviewed studies are grouped according to the metaheuristics used within Sections 2.1 and 2.2. Tables 2 and 3 summarize the results of the review.

2.1. Approaches used with spatial domain embedding

Many spatial domain embedding schemes use metaheuristic optimization techniques to improve embedding efficiency. For example, in [5], an image steganography scheme in the spatial domain based on the Least Significant Bit (LSB) operation is proposed. GA is used to find the best sequence of operations during embedding, such as pixel scanning, pixel shifting flipping secret bits, and others. Peak Signal-to-Noise Ratio (PSNR) is considered as the fitness function. This scheme ensures the optimal arrangement of message bits in the cover image.

In [6], a reversible embedding scheme combining histogram shifting and GA techniques is proposed. The authors consider

Table 2
Approaches used with spatial domain embedding.

Ref. no	Purpose of the study	Research methodology	Findings
[5]	High embedding capacity and imperceptibility	• Steganography; • LSB-embedding; • GA.	A novel way for finding the best LSB matching between the cover and the stego images; An optimized scheme for hiding secret data in a cover image to increase the quality and the embedding capacity.
[6]	High embedding capacity and imperceptibility	• Steganography; • Histogram shifting embedding; • GA.	The rate and distortion model for histogram shifting based multiple embedding; A GA based multiple embedding reversible scheme.
[7]	High embedding capacity and imperceptibility	• Steganography; • Chaotic maps; • GA.	An algorithm that uses GA to find the indices of the pixel values that makes minimum changes on the cover image.
[8]	High embedding capacity and imperceptibility.	• Steganography; • Pixel-value differencing embedding; • Modulus function; • PSO.	An algorithm that uses PSO to choose an optimal individual from more choices resulting from solutions of modulus function.
[9]	High embedding capacity and imperceptibility.	• Steganography; • LSB-embedding; • PSO.	A method of using PSO-based pixel selection for embedding a secret image.
[10]	High imperceptibility and robustness.	• Steganography; • LSB-embedding; • ABC.	An algorithm, in which the ABC is utilized to optimize the block assignment with the objective of minimizing the sum of the different bits in each block between the cover image and the secret image.
[11]	High embedding capacity and imperceptibility.	• Steganography; • LSB-embedding; • Edge detection; • ACO.	An algorithm that uses ACO for complex region detection and hides a variable amount of secret data in the different pixels of the complex region of a cover image.

histogram shifting based multiple embedding as the rate and distortion optimization problem in terms of multiple peak and zero bin pairs. The difference between the possible maximal distortion and the minimal distortion is used as the objective function.

In [7], the authors use chaotic maps to improve the data hiding scheme based on GA. Randomness of genetic is provided by different chaotic maps, for example, gauss, logistic and tent maps. PSNR is used as the fitness function.

PSO is another popular optimization algorithm. In [8], PSO is used to improve the quality of pixel-value differencing. The optimization algorithm selects the ideal pixel gray values among numerous modulus function solutions. The first objective function selects suitable pixels, and the second one finds the best solution among the optimization results for the first objective function.

The authors of [9] use PSO to provide high embedding capacity in the spatial domain of images. PSO finds the most suitable positions for embedding. In particular, optimization is used to find the best embedding starting point and pixel scan direction.

Study [10] uses ABC to optimize the block assignment for secret image embedding. The Mean Squared Error (MSE) metric is used as a fitness function. The authors demonstrate that their algorithm is better at resisting some noise attacks than similar steganographic schemes.

In [11], the authors use ACO to classify cover image pixels into a complex and smooth regions. Complex regions are used for embedding to reduce the distortion level of the stego image, while smooth regions remain unchanged.

A brief description of each of the reviewed studies is presented in Table 2.

2.2. Approaches used with frequency domain embedding

The authors of frequency domain embedding schemes also use metaheuristic optimization to improve performance indicators of information embedding in digital images.

The study [12] presents an algorithm for embedding data into a Fresnelet transform or discrete ripplet transform domain. To increase the capacity, a Quick Response (QR) code corresponding to a secret message is used as the embedded information. A modified version of GA is used to find the best direction, the best beginning point, and the best frequency coefficients for embedding.

In [13], a scheme for secure transmission of medical data is proposed. It combines steganography and encryption and is based on a bit mask oriented GA. The bit mask oriented GA successfully solves the premature convergence problem. The authors of [20] use this algorithm to encrypt sensitive data before embedding it into the DWT domain of medical images.

Study [14] presents a steganographic method that hides information into LSB-s of Integer Wavelet Transform (IWT) coefficients. The authors apply PSO to find the optimal substitution matrix for converting secret data into their substituted forms. The embedding imperceptibility metric PSNR is used as a fitness function.

In [15], the authors propose a robust blind watermarking scheme in a hybrid DWT-DCT and Singular Value Decomposition (SVD) domain. A multi-dimensional PSO is used to select the most appropriate DCT coefficients and to find the optimal value of the embedding parameter, which provides a good balance between robustness and imperceptibility. The objective function combines the PSNR and Bit Error Rate (BER) metrics.

Some studies use other metaheuristics. For example, the authors of [16] use DE for their watermarking scheme in the wavelet transform domain. They use DE twice: first to find suitable embedding locations, and then to choose the best embedding parameters. The main goal of using metaheuristic optimization is the high quality of watermarked images, so the objective function is based on the PSNR metric.

The authors of [17] use a Fruit Fly Optimization Algorithm (FOA) and an improved Seeker Optimization Algorithm (SOA) to find optimal locations for hiding data in the spatial domain. Additional information is embedded in the DCT coefficients of the most suitable pixels. The objective function combines the edge, entropy and intensity for a given pixel point.

The study [18] presents a reversible steganographic algorithm based on the Fractional Fourier Transform (FrFT). The task of finding the best pixel positions for embedding is solved using the Firefly Algorithm (FA), and information embedding is performed

Table 3
Approaches used with frequency domain embedding.

Ref. no	Purpose	Methodology	Findings
[12]	High embedding capacity and imperceptibility.	• Steganography; • Fresnelet transform or discrete ripplet transform; • LSB-embedding; • QR-code; • Modified GA.	An algorithm that uses GA for selecting the optimal transform coefficients to enhance the system performance.
[13]	High embedding capacity and imperceptibility.	• Steganography; • DWT; • Data encryption; • Modified GA.	A bit mask oriented GA; A method of secure medical data transmission using GA-encryption and data embedding into medical images through 1- and 2-level DWT.
[14]	High security, imperceptibility, and robustness.	• Steganography; • IWT; • LSB-embedding; • Substitution matrix; • Optimal pixel adjustment; • PSO.	An algorithm that uses PSO to find the optimal substitution matrix and optimal pixel adjustment procedure to minimize the variations in the modified coefficient values from the original values.
[15]	High imperceptibility and robustness.	• Watermarking; • DWT, DCT, SVD; • Linear combination embedding; • Intertwining logistic map; • Modified PSO.	An algorithm that combines DWT, DCT and SVD, chaos encryption, and multi-dimensional PSO to optimize embedding strength and location of the appropriate DCT coefficients.
[16]	High imperceptibility.	• Watermarking; • DWT; • Linear combination embedding; • DE.	An algorithm that uses DE to find an appropriate location for each of watermark blocks and the optimal values for Alpha-blending coefficients.
[17]	High embedding capacity and imperceptibility.	• Steganography; • DCT; • LSB-embedding; • Rivest cipher cryptosystem; • FOA, improved SOA.	An algorithm of medical data protection that combines two metaheuristics to provide optimal placement for embedding the secret data into a cover image.
[18]	High imperceptibility and robustness.	• Steganography; • FrFT; • Histogram shifting embedding; • FA.	A reversible embedding scheme that transforms the cover image into a time–frequency domain and uses FA to find the optimal locations for data hiding.
[19]	High imperceptibility and robustness.	• Watermarking; • Translation invariant wavelet transform, SVD; • Linear combination embedding; • GWO.	A non-blind image watermarking scheme that uses enhanced GWO to optimize the scaling factor.
[20]	High embedding capacity, imperceptibility, and robustness.	• Watermarking; • DWT, SVD; • Linear combination embedding; • Arnold transform and gyrator transform; • PSO, GWO.	A multi-watermarking scheme that selects the optimal embedding regions and embedding strengths by the higher contrast values of blocks using the hybrid PSO-GWO algorithm.
[21,22]	High embedding capacity and imperceptibility.	• Steganography; • DCT; • Substitution matrix; • CI.	A reversible steganography algorithm that uses CI optimization to reduce the computational time of the optimal substitution matrix selection.

using the histogram shifting technique. The objective function combines the Structural Similarity Index Measure (SSIM) and BER metrics to achieve a balance between imperceptibility and embedding robustness.

In [19], the authors use an enhanced Gray Wolf Optimizer (GWO) in their watermarking algorithm based on SVD in the translation invariant wavelet domain. The choice of this frequency transform is due to its higher robustness to image processing operations.

In [20], GWO is combined with PSO to effectively implement adaptive color multi-watermarking embedding. A hybrid optimization scheme is applied to find optimal embedding regions and embedding strengths. The value of the objective function depends on the correlation coefficients between the original image and the watermarked image, as well as between the original watermark and the watermark extracted from the image after applying some attacks.

In [21,22], the authors suggest using Cohort Intelligence (CI) optimization for compressed image steganography. The PSNR metric is used as an objective function. The quantized DCT coefficients are subject to changes during the embedding process.

A brief description of each study is presented in Table 3.

Summarizing the review, it can be noted that the use of metaheuristics to improve the quality of various data hiding schemes is common. However, the schemes described in previously published studies mostly refer to classical, well-known metaheuristics. In our previous study [23], we also discuss the efficiency of classical metaheuristics in relation to the problem of DFT-based information embedding in digital images. In recent years, researchers have proposed a large number of new metaheuristics that are more efficient for different optimization problems. Therefore, the assessment of the applicability of modern metaheuristic optimization algorithms is a promising area of research, contributing to the development of the data hiding field.

3. Embedding algorithm using DFT transform

In this study, we experimentally evaluate the applicability of classical and modern metaheuristics to improve the quality of adaptive information embedding into the DFT phase spectrum of images using the embedding algorithm that we proposed earlier in [24]. We briefly describe the algorithm below.

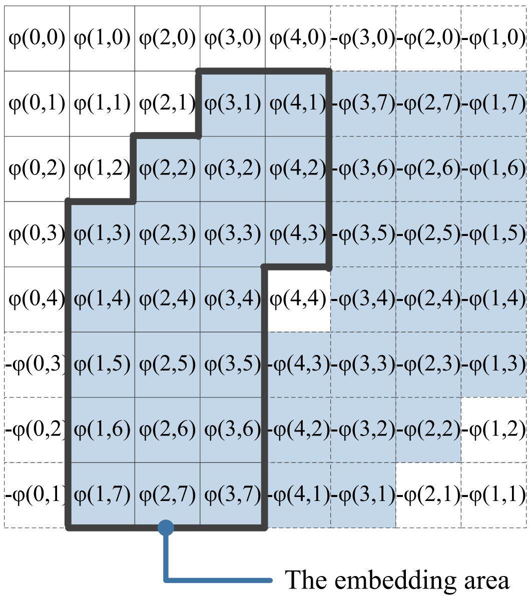

Fig. 1. Embedding area in DFT block.

The algorithm divides the image into non-overlapping 8 × 8 pixel blocks. The result of applying DFT to a pixel block is a complex DFT coefficient block of the same size. We use the phase component of the DFT block for embedding. It should be noted that the phase spectrum of the DFT is an odd function, therefore, when changing the element of the phase spectrum $\varphi(u, v)$, it is necessary to change the symmetrically located element as follows:

$$\varphi(-u, -v) = -\varphi(u, v), u = \overline{0, 7}, v = \overline{0, 7}. \quad (1)$$

The embedding area consists of 21 block elements. It includes all high-frequency and mid-frequency components of the DFT spectrum of an 8 × 8 pixel block, as shown in Fig. 1. The dotted line marks the symmetrical elements. The size of the embedding area was chosen experimentally. Reducing the embedding area is impractical since this also leads to a decrease in the embedding capacity. At the same time, it is necessary to use low-frequency components to increase the embedding area. Experiments show that this leads to noticeable distortions of the stego image pixel block in many cases.

The most homogeneous image blocks are used for embedding. Their average amplitude value is less than the threshold value A_{crit}. The information is embedded into phase values of coefficients. The phase values of the DFT coefficients are in the range of $(-\pi; \pi]$. It is necessary to select two values φ_0 and φ_1 from this range that $\varphi_0 = -\varphi_1$. Embedding of each message bit b into a phase value φ occurs according to the formula

$$\varphi' = \begin{cases} \varphi_0, \text{ if } b = 0, \\ \varphi_1, \text{ if } b = 1, \end{cases} \quad (2)$$

where φ' is a phase value of DFT coefficient after embedding.

A feature of the [24] algorithm is the iterative embedding procedure. It aims at combating extraction errors caused by rounding. Iterative information embedding is performed as follows. Information is extracted and extraction errors are checked after embedding a secret message fragment into the block of DFT coefficients. At the stage of data extraction, an inverse DFT is performed, the values are rounded to obtain integer pixels, and a re-DFT is performed. The phase values of the coefficients can be distorted after these procedures. Therefore, when extracting, an interval of values is used, the width of which is determined by the algorithm parameter ε. The extraction formula is

$$b = \begin{cases} 0, \text{ if } \varphi'' \in (\varphi_0 - \varepsilon, \varphi_0 + \varepsilon), \\ 1, \text{ if } \varphi'' \in (\varphi_1 - \varepsilon, \varphi_1 + \varepsilon), \end{cases} \quad (3)$$

where φ'' is a phase value of DFT coefficient during extracting.

If errors are found, they are corrected. This process continues until an option is obtained that provides error-free information extraction. If this cannot be achieved in a given number of iterations τ, then the block is considered empty. In this case, all phase values from the embedding area are changed so that they do not fall into the extraction intervals indicated in formula (3).

Another feature of the [24] algorithm is the minimization of the number of variable elements of the phase spectrum. Some blocks can require a very large number of iterations to ensure error-free information extraction. As a result, the imperceptibility of embedding is reduced. Using a small number of iterations significantly reduces capacity, since most of the blocks become empty. We vary the message fragment length and select a specific location of the message bits in the embedding area by checking possible options to solve this problem. Initially (before embedding), any block of phase values F contains a certain sequence of bits and empty elements that do not fall into the extraction intervals (3). Therefore, the main idea is to convert the sequence F to the embedding option X with the highest capacity and the least number of changes. The general scheme of the algorithm is shown in Fig. 2.

An important feature of the described scheme is that we can embed a message bit in any element of the embedding area, but we cannot make a random element empty. This idea is illustrated in Fig. 3. For clarity, we provide an example for an embedding area with a small number of elements. Let there be a block of phase values. Applying formula (3) to it before embedding, we get the sequence F. The "–" character matches empty elements that do not fall into the extraction intervals. It is necessary to embed in this block as many message bits M as possible with few changes. In the first step, we fix the elements containing 0 or 1 and check all possible options for the number of bits that can be embedded into empty elements. We estimate the coefficient Q using the formula

$$Q = \frac{s - d}{s + d}, \quad (4)$$

where s is a number of same elements of sequences F and X, and d is a number of their different elements.

In Fig. 3, we get 6 options (X1–X6), and the best options are X1 and X5. The number of options used for further search is determined by the algorithm parameter. Option X1 does not need further consideration since empty elements from X1 match empty elements of F. However, option X5 assumes 1 bit placement in a two-element empty space (the second empty space in F). This can be done in two ways: X5.1 and X5.2. Thus, X1, X5.1, X5.2 are good options for embedding.

The described scheme makes it possible to take into account the initial values of the DFT phase spectrum elements. As a result, the amount of distortion in the final stego image is reduced. Experimental results demonstrate a high level of imperceptibility and good capacity value.

However, this algorithm significantly limits the search space for embedding options. Researching more different embedding options may reveal better embedding options and, as a result, increase imperceptibility and capacity indicators. Therefore, in this study, we propose to consider the search for the best embedding option for each block as an optimization problem. A promising research direction is the use of metaheuristic optimization algorithms. To date, researchers have proposed a wide variety of

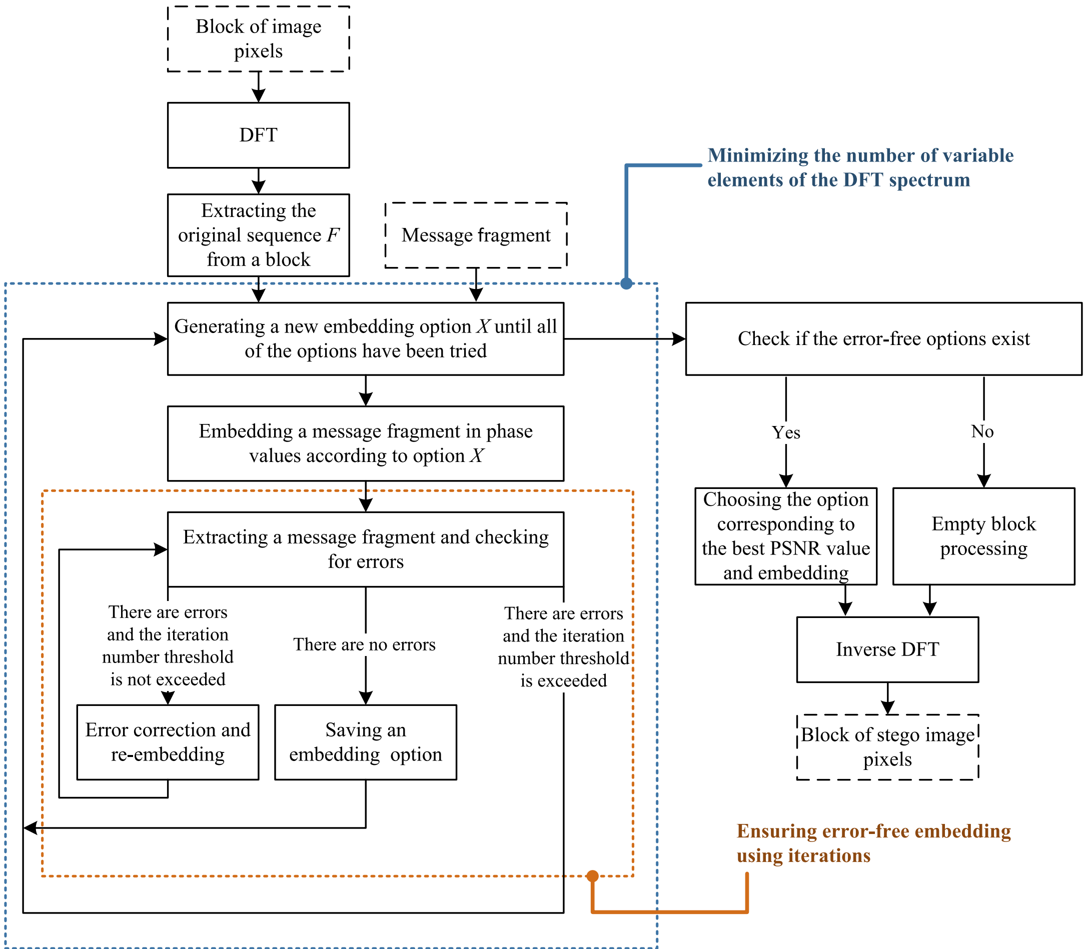

Fig. 2. The general scheme of [24].

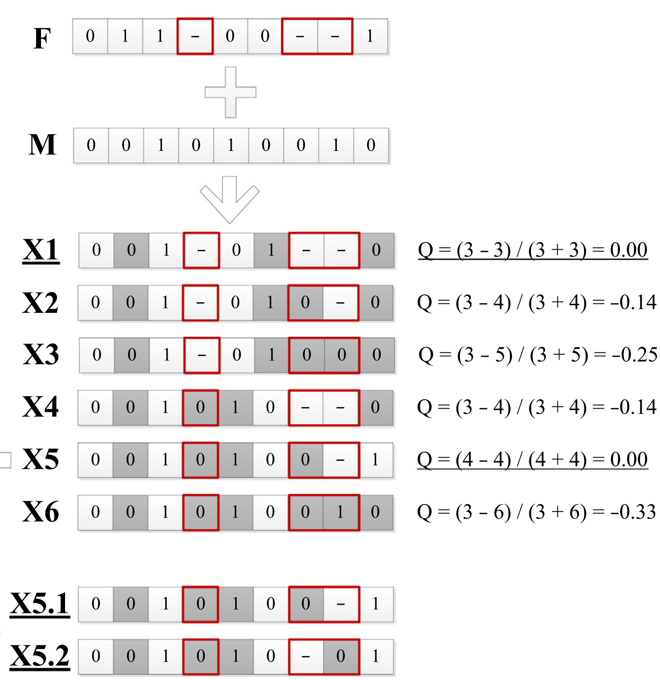

Fig. 3. Search scheme for embedding options from [24].

metaheuristic algorithms. The use of classical metaheuristics is common enough for data hiding schemes. However, in recent years, many studies have proposed whole new metaheuristics. Therefore, it is an urgent task is to study the applicability of new metaheuristics to the problem of increasing the efficiency of embedding information into digital images, in particular, embedding information into the DFT phase spectrum. We provide a detailed description in the next section.

4. Metaheuristic optimization algorithms

In this section, we present the main features of metaheuristic optimization algorithms and a brief overview of metaheuristics of different classes. Next, we list the metaheuristics chosen for our study.

4.1. A brief overview of metaheuristics

Metaheuristic optimization algorithms or metaheuristics is a vast class of optimization algorithms that allow us to find an optimal, or close to it solution for a wide range of problems. They direct the search process and explore the entire search space.

The basic steps of the algorithm are the same for most metaheuristics. The initial stage forms an initial set of solutions (population) and calculates the value of the objective function for each of them. Next, the algorithm performs an iterative search for the best solution. New solutions are obtained by changing and combining previously obtained solutions. The implementation depends on the mathematical model of each specific algorithm. As a rule, new solutions replace their predecessors if the objective function value improves. Reaching the stop condition stops the optimization algorithm. In our study, the stop condition is the achievement of a given number of objective function evaluations. As a result, the best solution becomes the one corresponding to the best value of the objective function.

There are many different metaheuristics [25–27]. The authors classify the metaheuristics differently [28] depending on the basic idea. In our study, we highlight Nature-based, Social-based and Mathematic-based algorithms (Table 4).

Nature-based algorithms are based on various natural phenomena: animal behavior, physical and chemical processes, weather phenomena, etc. This is the most common class of metaheuristics. For example, Forest Optimization Algorithm (ForOA) [29] is based on the trees' seeding procedure, Lightning Search Algorithm (LSA) [30] is based on the natural phenomenon of lightning, Henry Gas Solubility Optimization (HGSO) [31] imitates the huddling behavior of gas, Rain Optimization Algorithm (ROA) [32] is inspired by the movement of raindrops, and Slime Mould Algorithm (SMA) [33] is based on the oscillation mode of slime mould in nature. Evolutionary-based and Swarm-based subclasses are worth noting, since they are most often used to solve various scientific and applied problems. Evolutionary-based algorithms, inspired by evolutionary processes and natural selection, include such metaheuristics as GA [34] and DE [35]. Swarm-based algorithms are inspired by the group behavior (swarm intelligence) of individuals in nature. These include PSO [36], ACO [37], ABC [38], FA [39], Mayfly Algorithm (MA) [40], and Harris Hawks Optimization (HHO) [41].

Social-based algorithms are based on various aspects of human interaction. This class of algorithms has been actively studied in recent years. For example, World Cup Optimization Algorithm (WCO) [42] is inspired by football tournaments and competitions among the countries to win the World Cup, Community of Scientists Optimization (CoSO) [43] is inspired by the behavior of a community of scientists, Adolescent Identity Search Algorithm (AISA) [44] is inspired by the process of identity development of adolescents, Forensic-Based Investigation (FBI) algorithm [45] is inspired by the suspect investigation–location–pursuit process that is used by police officers, Student Psychology Based Optimization (SPBO) [46] is inspired by the psychology of a student who is trying to obtain the highest marks in the examination, Group Teaching Optimization Algorithm (GTOA) [47] is inspired by group teaching mechanism, and CI [48] is inspired by the candidates' self supervised learning behavior in a cohort.

Mathematic-based algorithms include metaheuristics whose mathematical models were not inspired by any specific natural or social phenomenon. This class is the least represented among the studies on metaheuristics. These algorithms include, for example, Stochastic Fractal Search (SFS) [49] that uses a mathematical concept called the fractal, Lévy Flight Distribution (LFD) [50] which is inspired by the Lévy flight random walk for exploring unknown large search spaces, and Gradient-Based Optimizer (GBO) [51] inspired by the gradient-based Newton's method.

In this section, we do not provide an exhaustive list of known metaheuristics, since compiling such a list is beyond the scope of the study. However, we tried to note both classical metaheuristics and those that have been recently proposed and published in leading scientific journals. Our task was to select a set of algorithms for conducting an experimental study aimed at evaluating the efficiency of metaheuristics of different classes in our optimization problem. We relied on the classification presented in Table 4 when forming this set. We chose modern algorithms that are published in high-ranking journals and have a number of citations. This indicates the reproducibility of the results declared by the authors.

The largest number of algorithms belongs to the nature-based class in the resulting set. This is because this class includes all classical metaheuristics that are traditionally used to optimize the data embedding. Experiments with these algorithms are an important part of our research since the comparison of classical and modern metaheuristics is an essential aspect of our motivation.

4.2. The description of chosen metaheuristics

In this subsection, we present a brief description of the selected algorithms. A detailed description of these algorithms can be found in the relevant studies.

GA [34] is one of the most famous evolutionary algorithms inspired by nature. It simulates the process of natural selection. The first stage generates an initial population randomly and calculates an objective function for each individual resulting in a binary vector. Next, GA performs crossover and mutation operations with probabilities p_{cross}^{GA} and p_{mut} accordingly. The crossover operation is the creation of new individuals by combining parts of the chromosomes of two different individuals. A mutation changes an individual's randomly selected gene. A new generation consists of individuals with the best values of the objective function.

DE [35] also belongs to nature-based evolutionary algorithms. DE works with real numbers, so the population's individuals are vectors of real numbers. The DE algorithm crosses each individual of the population with another randomly selected individual with a probability p_{cross}^{DE} during the optimization process. Before crossing, the parent is mutated using scaling factor α. If a descendant individual provides a better value of the objective function, it replaces the parent. An individual with the best value of the objective function becomes the solution after the completion of the algorithm.

PSO [36] is inspired by the behavior of flocks of birds in nature. This is one of the classic metaheuristic algorithms related to swarm intelligence. This algorithm also works with real numbers. At the initial stage, it is necessary to generate not the population of individuals itself, but their location and velocity vectors. At each iteration, PSO determines the best known position of each particle and the entire swarm of particles as a whole to use them for calculating new particle velocity values. As a result, the best swarm state becomes the solution to the optimization problem.

Table 4
Metaheuristic classes.

Metaheuristic class	Inspiration	Examples
Nature-based	Natural phenomena such as animal behavior, physical and chemical processes, weather phenomena.	ForOA, LSA, HGSO, ROA, SMA, GA, DE, PSO, ACO, ABC, FA, MA, HHO
Social-based	The behavior of people in different social groups and situations.	WCO, CoSO, AISA, FBI, SPBO, GTOA, CI
Mathematic-based	Mathematical concepts, functions, classical optimization algorithms.	SFS, LFD, GBO

HHO [41] is an example of modern swarm-based algorithms. It is inspired by the cooperative behavior and chasing style of Harris' hawks in nature. Each individual in the population is a hawk, while the best solution at each step is considered as the position of a prey. The energy of the prey E decreases considerably during the escaping behavior. If $|E| \geq 1$, the hawks search different regions to explore the prey location (the exploration phase). If $|E| < 1$, the algorithm tries to exploit the neighborhood of the solutions during the exploitation steps (the exploitation phase). The exploitation phase is inspired by the different hunting strategies of Harris' hawks that depend on the escaping behaviors of the prey.

AISA [44] is an example of a social-based algorithm. It is inspired by the process of identity development of adolescents in the peer group. Each individual of the population is an adolescent's identity. The authors consider 3 different cases of adolescents' identity formation. In the first case, the adolescent is supposed to adopt some of the best features of the peer group (the best identity vector). In the second case, the adolescent chooses someone special from the peer group as a role model (the best individual). In the third case, the adolescent adopts the negative traits of his peers (negative identity vector).

FBI [45] is inspired by the suspect investigation–location–pursuit process that is used by police officers. FBI algorithm has two phases: the investigation phase A (exploration), which is implemented by the team of investigators, and the pursuit phase B (exploitation), which is implemented by the team of police agents. Step A1 represents the "interpretation of findings". Each possible suspect location is investigated while taking into account the effects of other locations. Step A2 is the "direction of inquiry". This step evaluates the probability that the suspect location should be investigated in more detail. Step B1 represents the "actions". After the investigation team has identified the best location, agents approach the location that has the best objective value. Step B2 extends the process of "actions".

GBO [51] is an example of mathematic-based algorithm. It is inspired by the gradient-based Newton's method. GBO combines the power of gradient and metaheuristic optimization techniques. Each member of the population is called "a vector". The gradient search rule employs the gradient-based method to enhance the exploration tendency and accelerate the convergence rate. The local escaping operator enables the proposed GBO to escape from local optima.

5. Proposed embedding scheme

5.1. Benefits of metaheuristic optimization

In this paper, we use metaheuristic optimization to find the best embedding options and compare the efficiency of various metaheuristics.

The placement of message bits in the embedding area can be different and it depends on the embedding option. The embedding option determines which elements are used to embed the message bits and which ones remain unchanged. This has a significant effect on the block distortion level and on the block capacity. Obviously, the best solution can be found by checking all possible options. However, this search is redundant. In Fig. 4, we show that such a search consistently explores all options, including bad ones. Even if the best option has already been found, we cannot know for sure until we have checked all the possible options. This is time consuming and computationally intensive.

Metaheuristic optimization method explores the search space in a different way. Each iteration of the optimization algorithm brings us closer to the best solution. Fig. 4 shows that in this case the worst options are not considered. It should be noted that metaheuristics do not guarantee that the best option will be found, but they can find the best option with a high probability. The quality of the found solution depends on the number of individuals in the population and iterations of the optimization algorithm, as well as on additional parameters of each algorithm.

Earlier in [24] we used a limited search, but it was not directed. The search algorithm immediately cut off some of the options according to the capacity criterion, among which there could be better options than the found one. Therefore, in this study, we use metaheuristic optimization and form the objective function simultaneously taking into account several effectiveness criteria. Next, we describe the objective function in more detail.

5.2. Statement of optimization problems

In this section, we consider our optimization problem in detail. We describe the objective function and explain how the transition from the embedding option to the block of DFT phase coefficients is carried out when embedding information.

5.2.1. Objective function

The most important characteristics of the steganographic embedding quality are capacity and imperceptibility, so we use them when constructing the objective function [52].

Capacity is the number of bits of embedded information. Since the search for the best embedding option is performed for each separate image block, we use the capacity of the current block $C_{block} \geq 1$ for the objective function. The maximum block capacity $C_{\max}$ is 21 bits. We convert the capacity value to be in the range of (0; 1] so the capacity component is expressed by the formula

$$C_f = \frac{C_{block}}{C_{\max}}. \quad (5)$$

We use the PSNR metric to evaluate the embedding imperceptibility. For a block of 8 × 8 pixels, this is calculated as follows:

$$\mathrm{PSNR} = 10 \times \log 10 \left(\frac{255^2}{\mathrm{MSE}} \right), \quad (6)$$

where $\mathrm{MSE} = \frac{1}{8 \times 8} \sum_{i=1}^{8 \times 8} (I_i - S_i)^2$, I_i is a cover image pixel value, and S_i is a stego image pixel value.

It is impossible to know in advance the maximum possible PSNR value of each block. However, empirical research has shown that the value $PSNR_{block} \leq 50$ dB in practice, therefore $PSNR_{\max} = 50$ dB. Thus, the imperceptibility component is formed as follows:

$$PSNR_f = \frac{PSNR_{block}}{PSNR_{\max}}. \quad (7)$$

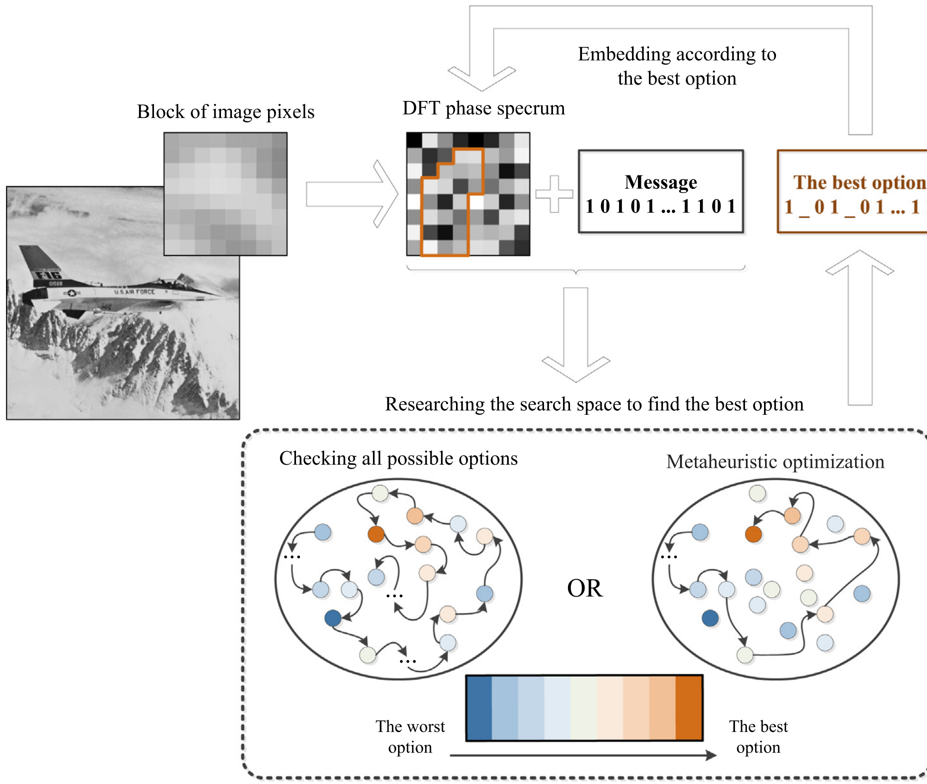

Fig. 4. General scheme of the optimization process.

To preserve the main feature of the [24] algorithm, it is necessary to ensure the error-free extraction of information from the image block. For the objective function, we use the value E_f, obtained according to the following rule:

$$E_f = \begin{cases} 0, \text{ if there are extraction errors,} \\ 1, \text{ if there are no extraction errors.} \end{cases} \quad (8)$$

Then the objective function can be described by the following expression:

$$F = k_C C_f + k_{PSNR} PSNR_f + k_E E_f, \quad (9)$$

where k_C is a capacity weighting factor, k_{PSNR} is a PSNR value weighting factor, k_E is a weighting factor for the value E_f, $F \in (0; 1]$.

Optimization solves the problem of maximizing the objective function value. Error-free information extraction takes precedence over high imperceptibility and capacity, therefore $k_E = 0.5$. If $F \leq 0.5$ then there is no error-free extraction, and the block is empty. The weighting factors k_C and k_{PSNR} are selected empirically. A larger value of the coefficient k_C provides a higher capacity, and a larger value of the coefficient k_{PSNR} provides a higher level of embedding invisibility. We use $k_C = 0.1$ and $k_{PSNR} = 0.4$ for our experiments as these coefficient values provide a good balance between imperceptibility and embedding capacity for different images, while increasing the PSNR value is given priority.

5.2.2. Binary vectors as individuals

GA uses binary vectors as individuals of the population. Some GA versions uses real numbers, but we use binary GA to study different variants of optimization problem.

Let us describe the embedding procedure as a function

$$E(\Phi, M, V) = \Psi, \quad (10)$$

where $\Phi = \left(\varphi_{i,j}\right)_{i=1,j=1}^{n,n}$, $\varphi_{i,j} \in (-\pi, \pi]$ is an initial block of phase values, $\Psi = \left(\psi_{i,j}\right)_{i=1,j=1}^{n,n}$, $\psi_{i,j} \in (-\pi, \pi]$ is a new block of phase values after embedding, $M \in \{0, 1\}^l$, $0 < l < \frac{n^2}{2}$ is a message fragment, V is an embedding option that determines the message bits location in the block of phase values, $n = 8$, and $l = 21$.

We define the embedding option V as follows. The embedding algorithm provides for the choice of one of three possible actions for each phase value a_1 = "to embed 0 bit", a_2 = "to embed 1 bit", and a_3 = "to create an empty element". Let us define these actions as an alphabet $A = \{a_1, a_2, a_3\}$. Then the sequence of actions that transforms Φ into Ψ can be written as an l-character string V of the alphabet A elements.

Obviously, the optimization problem is to find the optimal embedding option in accordance with the objective function F in formula (9). However, an individual of the GA population cannot be an embedding option. The initial stage of the GA-optimization algorithm generates population individuals randomly, while the arrangement of the string V symbols is not random and depends on the message M bits. Therefore, we use a truncated alphabet $B = \{b_1, b_2\}$, where b_1 = "to embed a bit", and b_2 = "to create an empty element". We set a one-to-one correspondence between the embedding option V and the pair of the auxiliary sequence $X \in B^l$ and the message M. Auxiliary sequences X can be represented as GA individuals.

Now let us formalize the optimization problem:

$$X^* = \arg\max_{X \in B^l} \left[F\left(I(\Phi), I(\Psi)\right)\right], \quad (11)$$

where $I(\Phi)$ and $I(\Psi = E(\Phi, M, V^*))$ are pixel blocks corresponding to phase value blocks before and after embedding, respectively, and V^* is the optimal embedding option, which is determined by the formula

$$v_i^* = \begin{cases} a_1, & \text{if } x_i^* = b_1 \text{ and } m_i = 0, \\ a_2, & \text{if } x_i^* = b_1 \text{ and } m_i = 1, \\ a_3, & \text{if } x_i^* = b_2. \end{cases} \quad (12)$$

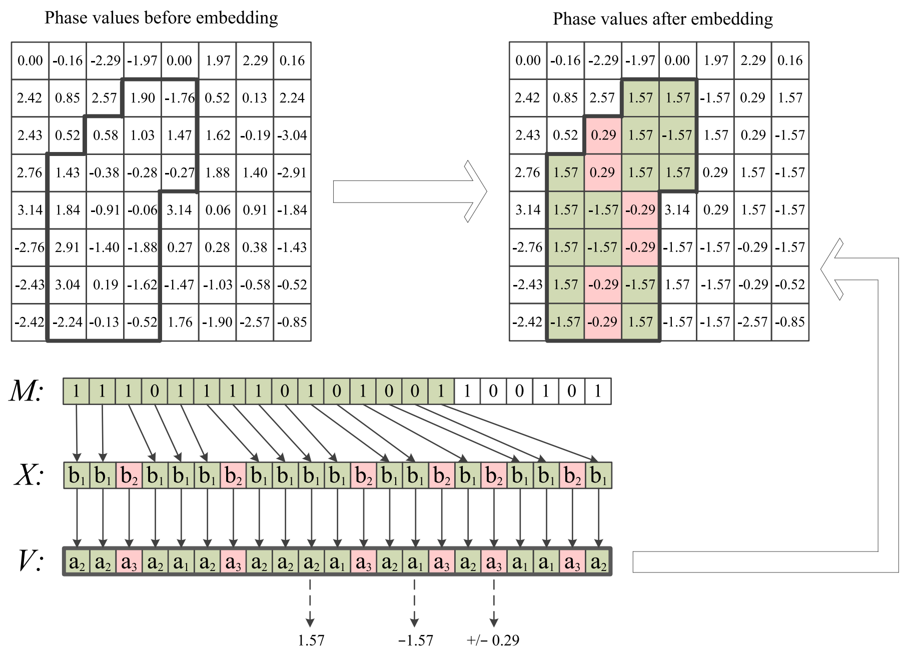

Fig. 5. Embedding a message fragment into a block of phase values when using a binary vector as a population individual.

Fig. 5 illustrates the transition process. The figure shows the message fragment M. The transition from the auxiliary sequence X (this was previously found by GA) to the embedding option V is carried out according to the formula (12). The embedding option allows us to embed 15 bits into the block of phase values. Message bit embedding is done using the formula (2). If $v_i^* = a_3$ then the element of the phase spectrum randomly takes a value $\frac{\varphi_1 - \varepsilon}{2}$ or $\frac{\varphi_0 + \varepsilon}{2}$.

5.2.3. Real number vectors as individuals

GA is well suited for solving the described problem. However, it can also be solved by other metaheuristics that use vectors of real numbers as individuals. Therefore, below we consider an alternative formulation of the optimization problem.

Phase values of DFT coefficients are real numbers. An optimization algorithm result is also a vector of real numbers. These numbers can be used as new values for the phase spectrum elements. Let us describe the embedding procedure as a function

$$E(\Phi, M, W) = \Psi, \tag{13}$$

where the variables Φ, Ψ, M correspond to the analogous ones in the formula Eq. (10), W is an embedding option, $W = (w_1, w_2, \ldots, w_l)$, and $w_i \in (-\pi, \pi]$.

However, just like in the case of a binary vector, we cannot generate values from the entire range of phase spectrum values without any restrictions, since the formula (3) establishes a one-to-one correspondence between the message bit values and the extraction intervals. It is necessary to ensure the correspondence between the secret message fragment and the new phase spectrum values.

Therefore, we define an auxiliary sequence $Y = (y_1, y_2, \ldots, y_l)$, $y_i \in [0, \pi]$ that can be used as a population individual. Then the optimization problem is formulated as follows:

$$Y^* = \arg\max_{Y} \left[F\left(I(\Phi), I(\Psi)\right)\right], \tag{14}$$

where $I(\Phi)$ and $I(\Psi = E(\Phi, M, W^*))$ are pixel blocks corresponding to phase value blocks before and after embedding, respectively, and W^* is the optimal embedding option, which is determined by the formula

$$w_i^* = \begin{cases} (-1)^{m_i+1} y_i^*, & \text{if } |\varphi_0| - \varepsilon < w_i < |\varphi_0| + \varepsilon, \\ (-1)^r y_i^*, & \text{otherwise,} \end{cases} \tag{15}$$

where r is a random binary number.

Our proposed approach is shown in Fig. 6. The sequence Y obtained by some metaheuristic contains 21 values from the range of $[0, \pi]$. If an element of the sequence Y falls within the extraction intervals specified in the formula (3) then it contains the message bit, otherwise, the corresponding element is empty. It is necessary to change the signs of the sequence Y elements in accordance with the message fragment M to form the embedding option W. Embedding 1 preserves the positive sign of Y element and embedding 0 makes the value of Y element negative. Signs of empty elements are randomly selected. As a result, values of the vector W sequentially replace the original values of the phase spectrum elements from the embedding area.

5.3. An improved information embedding scheme

Thus, we propose an improved scheme for embedding information into the DFT phase spectrum of digital images based on metaheuristic optimization. We presented the initial embedding scheme [24] in Fig. 2. Fig. 7 shows the distinguishing features of the new scheme. Note that the abbreviation OF means the objective function. It is also worth noting that the new scheme does not provide for separate handling of the cases "There are no errors" and "There are errors and the iteration number threshold is exceeded" at the stage of applying the metaheuristic, since these conditions are taken into account when calculating the objective function. If the best value of the objective function is less than 0.5, then no embedding option provides error-free extraction, and an empty block processing is required. The presented scheme can use different metaheuristics for optimization within the framework of this embedding scheme. At the same time, the statement of optimization problems provides for the possibility of working both with binary sequences and with real ones. Therefore, the set of potentially applicable metaheuristics is very wide and is not limited to those considered in this study.

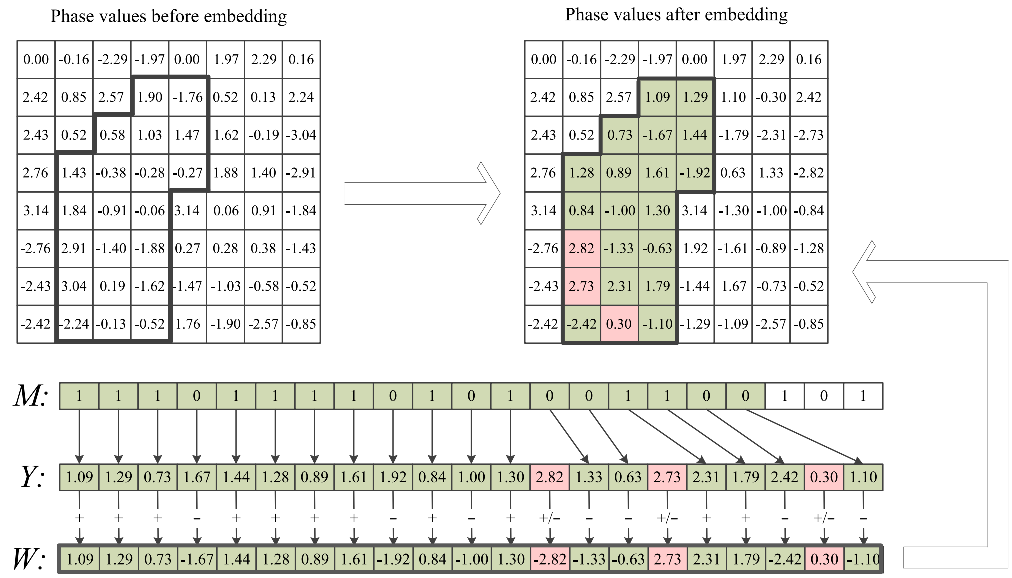

Fig. 6. Embedding a message fragment into a block of phase values when using a real number vector as a population individual.

The information extraction scheme corresponding to the proposed embedding scheme is quite simple and does not differ in any way from the extraction scheme [24]. To extract the embedded message, the stego image is split into blocks of 8 × 8 pixels. The DFT is performed and the phase spectrum of the DFT is calculated for each block that was used for embedding. The extraction of a message fragment from the phase spectrum of the DFT block is carried out according to formula (3). If there is no value in the block that falls within the embedding intervals, it is skipped and the transition to the next block occurs. The extracted fragments are combined into one message after extracting the embedded data from all blocks of the stego image.

6. Experimental results

In this section, we provide an experimental comparison of the different metaheuristics efficiency when embedding information into the DFT phase spectrum in accordance with the proposed scheme. For the main series of experiments, we used eight 512 × 512 grayscale images such as "Airplane", "Lena", "Baboon", etc. [53]. We embedded as much information as possible into each image and measured the PSNR value for the entire cover image and stego image. The embedding was repeated 30 times for each image and for each metaheuristic. In addition, we conducted an additional series of experiments with modern color images sized 512 × 768 [54].

All the simulations have been carried out on an Intel i5, 2.9 GHz desktop computer with 16 GB of RAM (Windows 10 platform). The program was written in the C++ programming language.

The embedding parameters were as follows: parameters of the embedding intervals $\varphi_0 = -1.57$ and $\varphi_1 = 1.57$; the number of iterations $\tau = 5$; the width of the extracting intervals $\varepsilon = 1$, the threshold value $A_{crit} = 0.9$. The search for the best embedding option was performed for each image block. The population size was $P = 40$ individuals in all experiments. The optimization process continued until $G = 1600$ estimates of the objective function were reached. The dimension corresponds to the maximum capacity of the block $C_{\max}$ and was equal to 21.

Optimization algorithms GA, DE and PSO provide for varying parameters. Table 5 shows the parameter values used in the experiments for these algorithms. The results of the parameter selection experiments are shown in Figs. 8–10. For the rest of the metaheuristics, we used the parameter values recommended by the authors of the respective studies; they are also indicated in Table 5.

Table 5
Metaheuristic parameters.

Algorithm	Parameter	Value
GA	Crossover probability p_{cross}^{GA} Mutation probability p_{mut}	1.0 0.4
DE	Crossover probability p_{cross}^{DE} Scaling factor α	0.6 0.2
PSO	Inertia factor w Social component c_1 Cognitive component c_2	0.4 1.4 1.4
HHO	–	–
AISA	Number of Chebyshev polynomials k	3 (Default)
FBI	–	–
GBO	Parameter $\beta_{\min}$ Parameter $\beta_{\min}$ Probability pr	0.2 (Default) 1.2 (Default) 0.5 (Default)

Fig. 8 shows the results of experiments on the choice of GA parameters. An experimental study has shown that sufficiently large values of the algorithm parameters are required to achieve the highest indices of imperceptibility and capacity: $p_{cross}^{GA} = 1.0$ and $p_{mut} = 0.4$. This is because only a small part of embedding options provides error-free information extraction, and performing the crossover and mutation operators with a high probability is required to find such options faster. Otherwise (for small values of the parameters), the algorithm goes through many options, among which there is no error-free ones. Fig. 8a shows that the best embedding quality is achieved with the maximum possible value of the parameter p_{cross}^{GA}. Fig. 8b shows that increasing the parameter p_{mut} above 0.4 leads to a decrease in the PSNR value with a small increase in capacity. Since embedding invisibility takes precedence over capacity in our study, we chose the parameter value that provides the highest PSNR value.

Fig. 9 shows how the DE parameter values affect its efficiency in our optimization problem. We chose $p_{cross}^{DE} = 0.6$ because this provides a balance between embedding capacity and imperceptibility. The change in the capacity value when the parameter p_{cross}^{DE} is increased to 0.8 is not significant, while the PSNR value shows

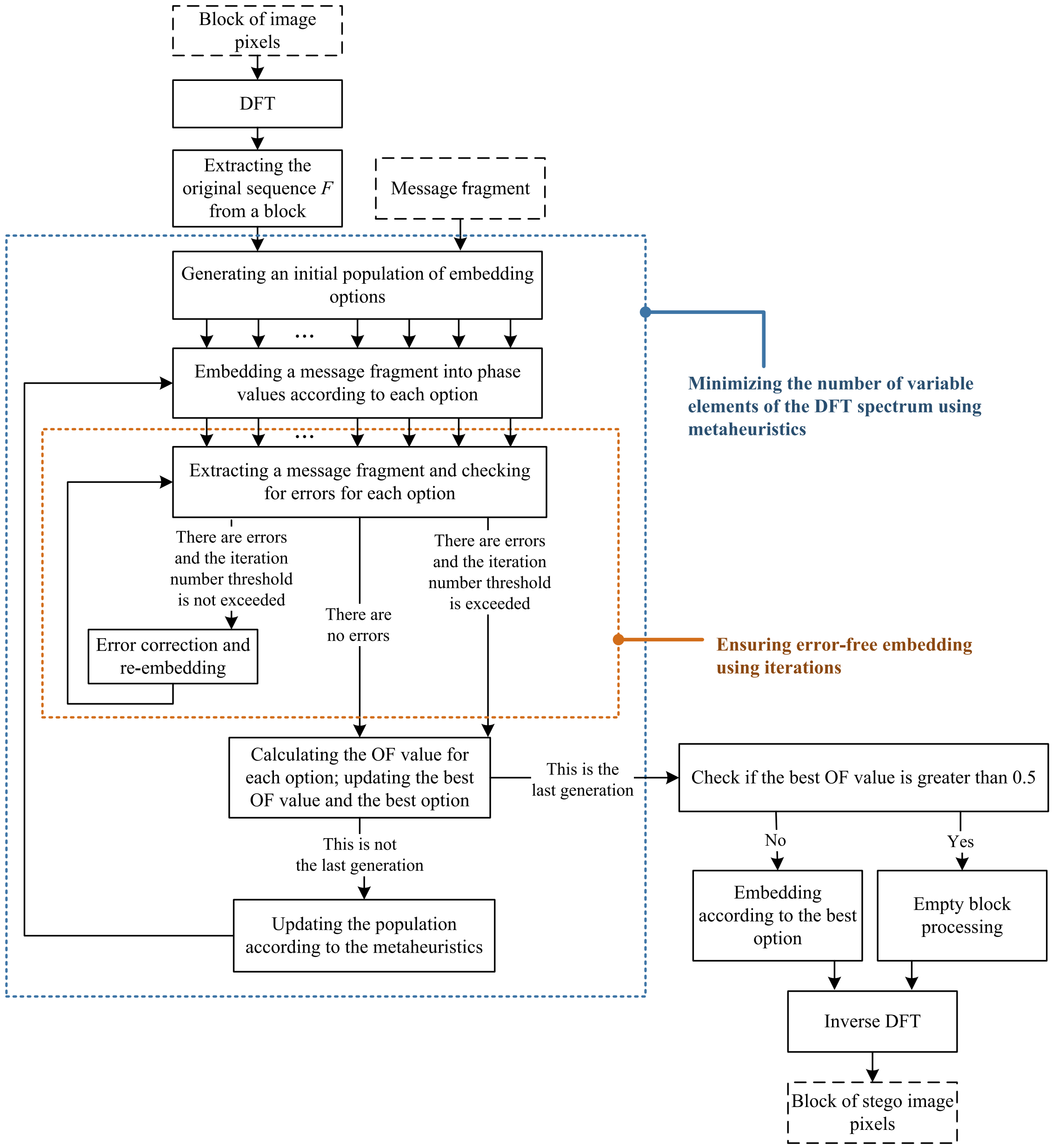

Fig. 7. The proposed embedding scheme.

a significant decrease. A further increase leads to an even more noticeable reduction in the invisibility of embedding. The value $\alpha = 0.2$ provides the best capacity and PSNR at the same time, so we chose this parameter value.

In Fig. 10, we illustrate the choice of parameter values for the PSO algorithm. Note that we set $c_1 = c_2$ in our study. We made this decision based on the analysis of studies on PSO-based embedding schemes. The values $w = 0.4$ and $c_1 = c_2 = 1.4$ provide the maximum PSNR value at high capacity. This allows us to provide better embedding imperceptibility.

A numerical comparison of the metaheuristic efficiency is presented in Tables 6 and 7. For clarity, we have added the performance indicators of the original algorithm [24] to the comparison. We use average results of 30 program runs. The best values are in bold. The use of metaheuristic optimization provided an increase in the PSNR value by an average of 2.05–6.19%. HHO did not have a significant positive effect on the level of embedding imperceptibility, and the use of GA led to a decrease in the PSNR value. The increase in capacity when using metaheuristics was 16.22–24.71%.

Figs. 11–12 illustrate the comparison of metaheuristics by PSNR criteria and capacity for different images. We use boxplots to demonstrate the variability of the results of each algorithm. All implemented metaheuristics demonstrate a high level of robustness. The robustness of the algorithms is slightly worse on the capacity criterion due to the lower priority of this criterion when constructing the objective function. It can be noted that GA and AISA metaheuristics demonstrated the highest robustness in terms of the PSNR criterion, and DE and AISA in terms of capacity. Algorithms DE and GBO have the worst robustness in terms of the PSNR criterion, GA and FBI have the worst robustness in terms of the capacity criterion.

Figs. 13–14 show the quality of stego images generated using different metaheuristics. Fig. 13 represents the original Goldhill image as well as seven stego images corresponding to each of the metaheuristics. Fig. 14 contains enlarged fragments of these images.

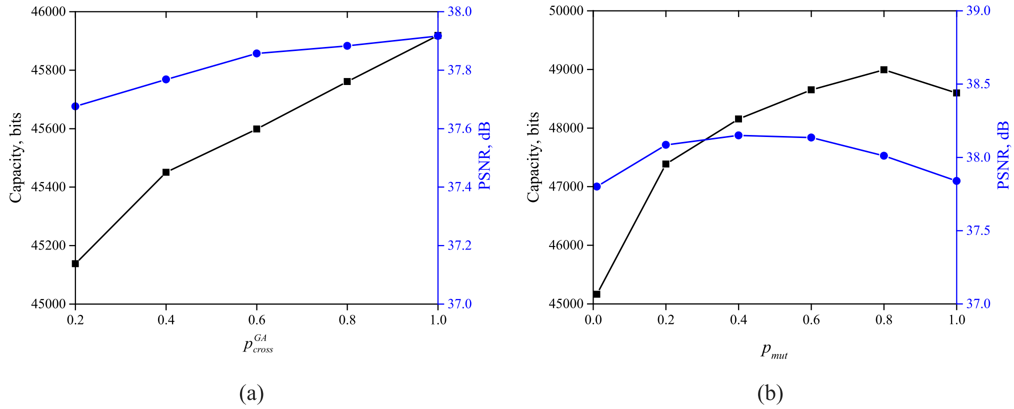

Fig. 8. Choosing the best parameter value for GA: (a) p_{cross}^{GA}, (b) p_{mut}.

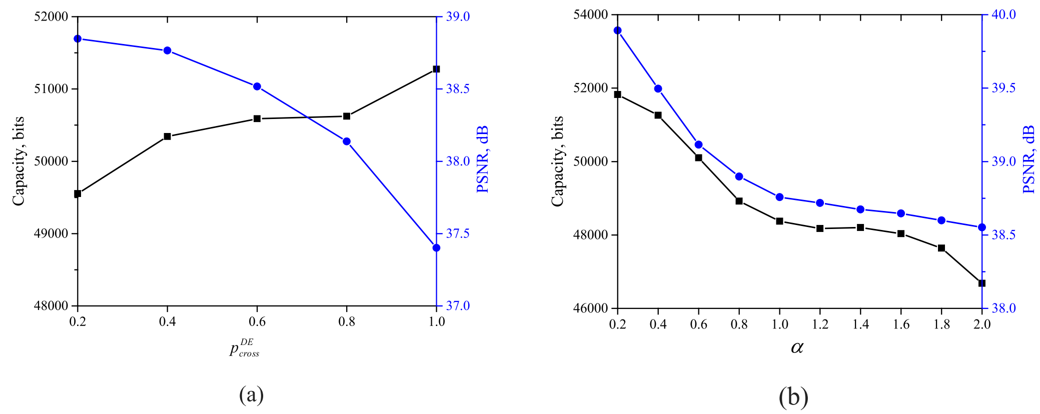

Fig. 9. Choosing the best parameter value for DE: (a) p_{cross}^{DE}, (b) α.

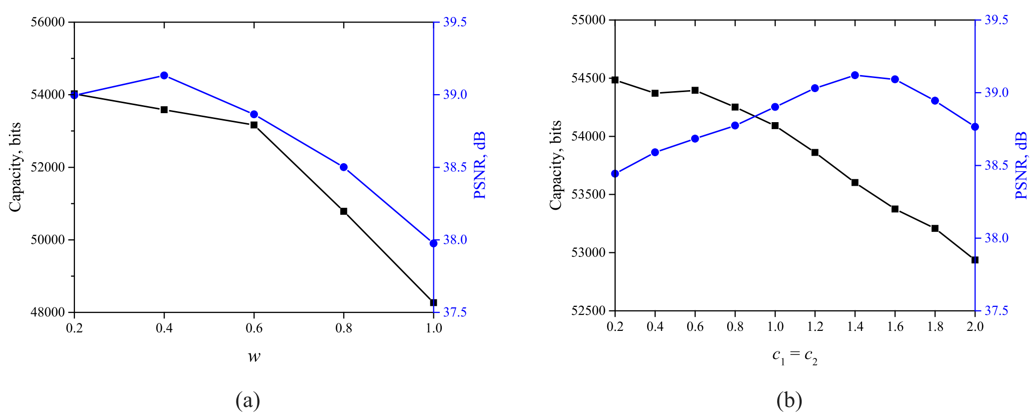

Fig. 10. Choosing the best parameter value for PSO: (a) w, (b) c_1 and c_2.

Table 6
Comparison of metaheuristics by PSNR criterion (dB).

Algorithm	Image							
	Airplane	Baboon	Barbara	Boat	Goldhill	Lena	Peppers	Stream
[24]	40.37	40.75	40.01	37.26	37.24	39.33	37.62	39.54
GA	39.84	40.06	39.34	36.83	36.68	38.54	37.07	39.12
DE	**42.76**	**43.14**	**42.55**	**39.94**	**39.76**	**41.45**	**39.73**	**42.11**
PSO	41.75	41.79	41.41	38.59	38.49	40.31	38.47	40.89
HHO	40.67	40.56	40.33	37.38	37.27	39.25	37.28	39.66
FBI	41.35	41.37	41.07	38.23	38.07	39.96	38.03	40.43
AISA	41.55	41.59	41.24	38.42	38.27	40.17	38.28	40.63
GBO	42.52	42.66	42.25	39.44	39.30	41.10	39.26	41.70

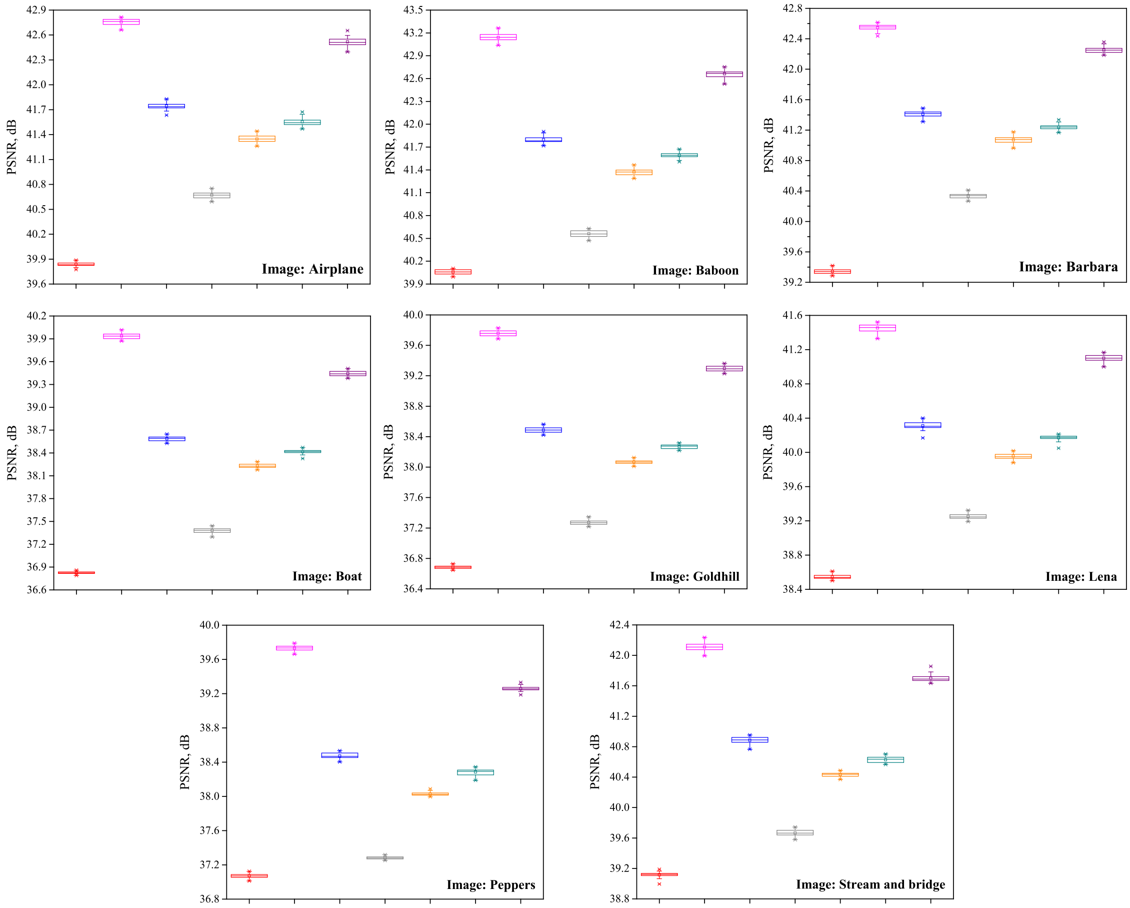

Fig. 11. Comparison of metaheuristics by the PSNR criterion for different images.

Table 7
Comparison of metaheuristics by capacity criterion (bits).

Algorithm	Image							
	Airplane	Baboon	Barbara	Boat	Goldhill	Lena	Peppers	Stream
[24]	51 307.00	19 707.00	39 080.00	48 561.00	49 542.00	55 811.00	54 748.00	23 324.00
GA	59 286.73	23 020.90	45 530.23	56 424.47	56 964.70	65 725.03	63 846.47	26 752.37
DE	62 452.20	24 200.67	47 989.50	59 175.43	60 097.73	68 767.63	66 976.00	28 386.83
PSO	63 312.30	24 638.87	48 673.10	60 209.00	61 135.57	69 655.37	67 951.50	28 879.90
HHO	62 135.23	24 601.10	48 313.07	60 155.57	60 943.93	69 312.67	68 050.27	28 788.83
FBI	60 178.87	23 740.03	46 711.17	57 968.97	58 893.30	67 076.87	65 757.07	27 797.13
AISA	**63 673.77**	**24 736.43**	**48 935.10**	**60 499.90**	**61 414.97**	**69 975.73**	**68 328.17**	**29 052.80**
GBO	63 175.63	24 532.83	48 551.83	59 943.50	60 912.67	69 453.53	67 727.00	28 766.20

Additionally, we conducted an experiment on embedding information in color images. We used each of the RGB channels to hide the message. Table 8 presents the numerical results. Figs. 15–16 demonstrate the example of cover image and the stego images. Experimental results show that there are no significant differences when embedding information in grayscale and color images.

In this section, we have presented the results of experiments with our algorithm for embedding information into the DFT phase spectrum of digital images based on metaheuristic optimization. In the next section, we analyze the obtained results and carry out their statistical analysis.

7. Discussion

Based on the results of the experiments (Figs. 11–12, Tables 6–8), there are a number of conclusions. All the metaheuristics demonstrated a high level of robustness of the results obtained with multiple reproductions for different images. In all cases, the use of metaheuristic optimization made it possible to increase the embedding capacity in comparison with [24]. The AISA, PSO, GBO, and HHO algorithms provided the highest capacity values. All the metaheuristics, except GA, also improved the embedding imperceptibility. The highest PSNR values were provided by the DE, GBO, and PSO algorithms. Thus, the GBO and PSO algorithms demonstrated the highest embedding efficiency in terms of imperceptibility and capacity at the same time. The DE algorithm provides the best PSNR value at a sufficiently high capacity. We preserved the error-free information extraction in all cases.

7.1. Statistical study

We conducted a statistical study for a deeper analysis of the results. First of all, we performed pairwise comparisons of metaheuristics using a nonparametric sign test [55]. The comparison results are presented in Tables 9–10. The results are presented as

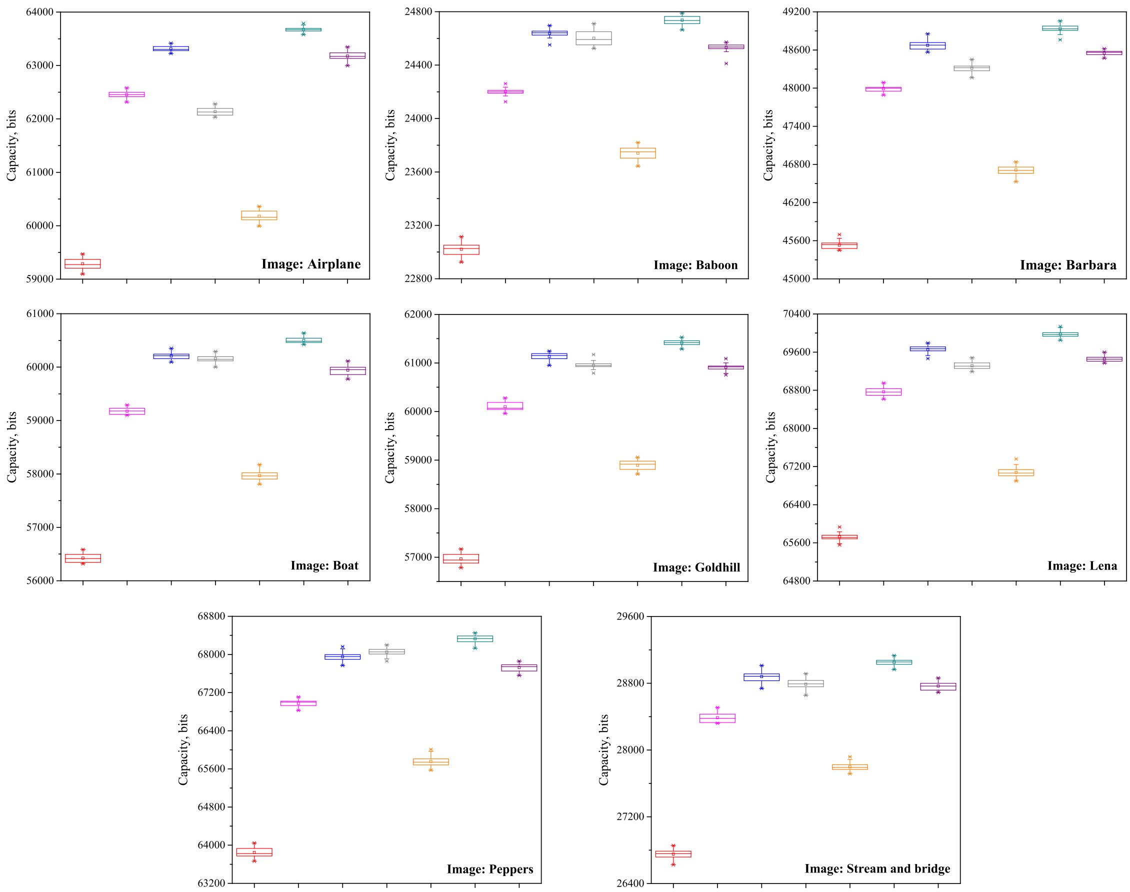

Fig. 12. Comparison of metaheuristics by the capacity criterion for different images.

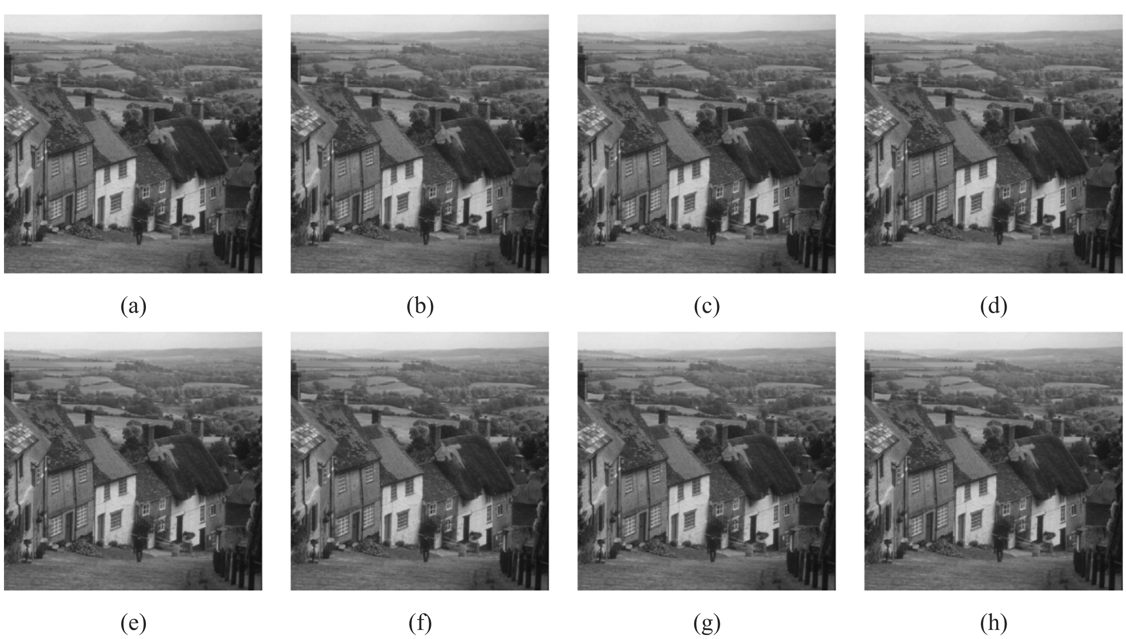

Fig. 13. Image examples: (a) cover; (b) GA; (c) DE; (d) PSO; (e) HHO; (f) FBI; (g) AISA; (h) GBO.

Wins/*Loses* and detected difference α is indicated if the algorithm performs better on at least 7 cases. It follows from Table 9 that the best results in terms of the PSNR criterion are provided by the DE algorithm, GBO has the second result, and PSO has the third one. It follows from Table 10 that AISA is the best algorithm in terms of capacity, PSO comes in second, HHO and GBO come in third.

We also performed multiple comparisons among all algorithms. Friedman's test [55] showed the existence of significant differences among the algorithms (Table 11). Shaffer static

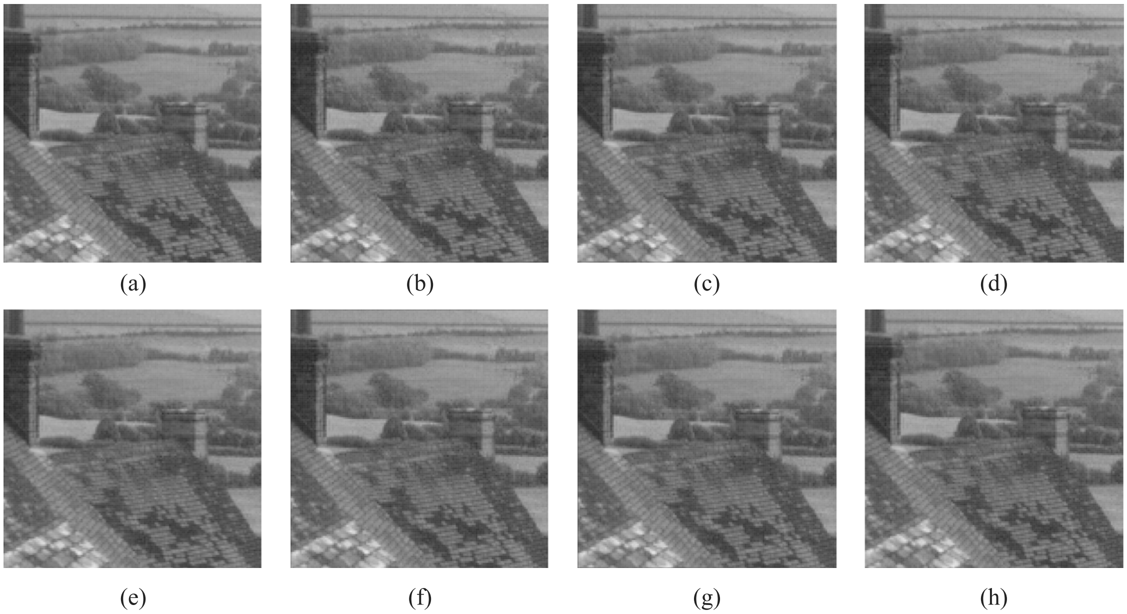

Fig. 14. Enlarged fragments of images: (a) cover; (b) GA; (c) DE; (d) PSO; (e) HHO; (f) FBI; (g) AISA; (h) GBO.

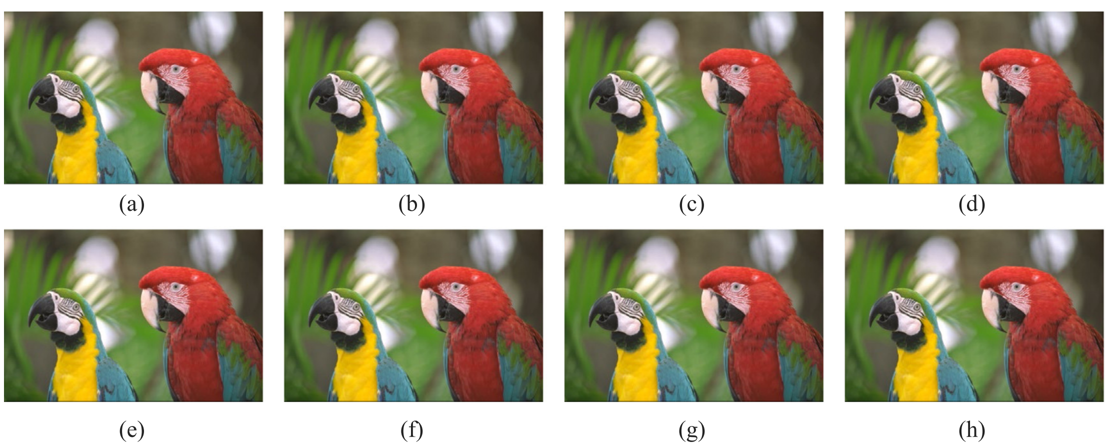

Fig. 15. Color image examples: (a) cover; (b) GA; (c) DE; (d) PSO; (e) HHO; (f) FBI; (g) AISA; (h) GBO.

Table 8
Comparison of metaheuristics by PSNR (dB) and capacity (bits) criteria for color images.

Algorithm	Image							
	Kodak image 5		Kodak image 8		Kodak image 21		Kodak image 23	
	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity
[24]	40.48	108 528	39.32	124 046	41.20	187 789	40.98	271 168
GA	39.98	125 175	38.94	144 236	40.61	219 535	40.46	316 880
DE	**43.29**	132 104	**41.86**	151 267	**43.28**	229 841	**43.07**	332 665
PSO	42.13	134 410	40.55	153 946	42.26	232 635	42.178	336 369
HHO	40.95	133 083	39.33	153 492	41.05	231 391	41.12	330 420
FBI	41.74	128 733	40.13	148 201	41.79	224 013	41.73	320 598
AISA	41.82	**135 119**	40.35	**154 815**	42.02	**233 955**	42.02	**338 362**
GBO	42.92	133 860	41.40	153 252	42.92	232 031	42.84	335 884

procedure and Bergmann–Hommel procedure were chosen for multiple comparisons as recommended in [55]. Tables 12–13 show p-values for all hypotheses for PSNR and capacity in decreasing order. Hypothesis of the form "Algorithm 1 vs. Algorithm 2" suggests that Algorithm 1 is better than Algorithm 2. Using a level of significance $\alpha = 0.1$, 10 hypotheses (no. 1–10) are rejected by both methods for PSNR values. In other cases (no. 11–21), the statistical differences between the results are less expressed. In the case of capacity, 7 hypotheses (no. 1–7) are rejected by Shaffer method and Bergmann–Hommel procedure rejects an additional hypothesis number 8, thus confirming the improvement of PSO over DE. In other cases (no. 9–21), the statistical differences between the results of applying metaheuristics are less expressed. In general, the statistical significance of differences between capacity values for different metaheuristics is less than for PSNR values.

As shown by the results of experiments and statistical research, GA has the lowest efficiency among all the metaheuristics.

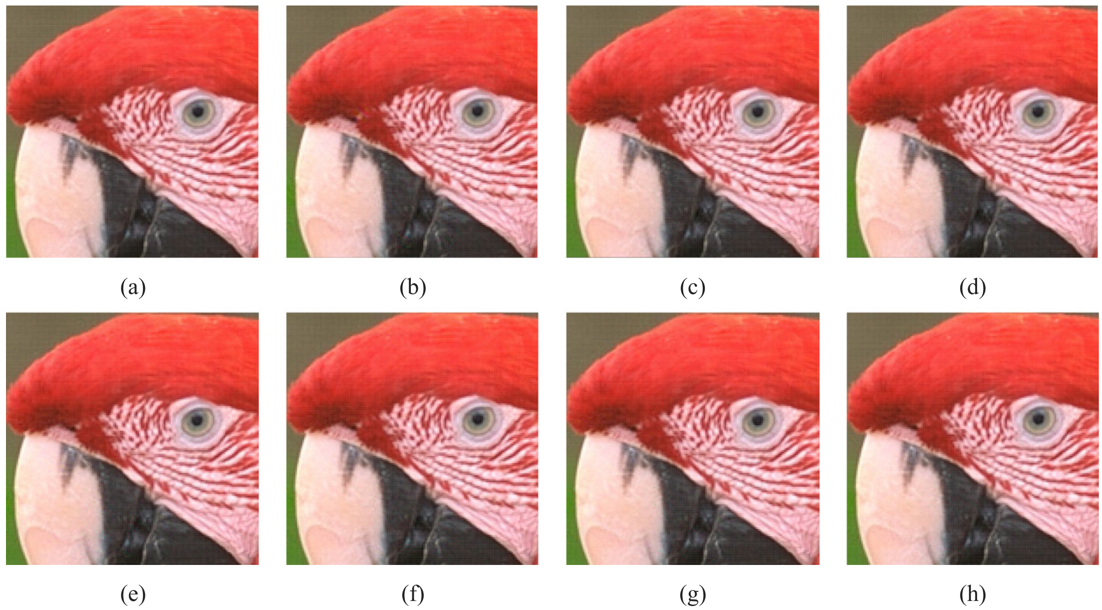

Fig. 16. Enlarged fragments of color images: (a) cover; (b) GA; (c) DE; (d) PSO; (e) HHO; (f) FBI; (g) AISA; (h) GBO.

Table 9
Pairwise comparisons of metaheuristics by sign test for PSNR.

	GA	DE	PSO	HHO	FBI	AISA	GBO
GA	×	0/8	0/8	0/7	0/8	0/8	0/8
DE	8/0 ($\alpha = 0.05$)	×	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)
PSO	8/0 ($\alpha = 0.05$)	0/8	×	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	0/8
HHO	8/0 ($\alpha = 0.05$)	0/8	0/8	×	0/8	0/8	0/8
FBI	8/0 ($\alpha = 0.05$)	0/8	0/8	8/0 ($\alpha = 0.05$)	×	0/8	0/8
AISA	8/0 ($\alpha = 0.05$)	0/8	0/8	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	×	0/8
GBO	8/0 ($\alpha = 0.05$)	0/8	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	×

Table 10
Pairwise comparisons of metaheuristics by sign test for capacity.

	GA	DE	PSO	HHO	FBI	AISA	GBO
GA	×	0/8	0/8	0/8	0/8	0/8	0/8
DE	8/0 ($\alpha = 0.05$)	×	0/8	1/7	8/0 ($\alpha = 0.05$)	0/8	0/8
PSO	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	×	7/8 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	0/8	8/0 ($\alpha = 0.05$)
HHO	8/0 ($\alpha = 0.05$)	7/8 ($\alpha = 0.05$)	1/7	×	8/0 ($\alpha = 0.05$)	0/8	5/8
FBI	8/0 ($\alpha = 0.05$)	0/8	0/8	0/8	×	0/8	0/8
AISA	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	×	8/0 ($\alpha = 0.05$)
GBO	8/0 ($\alpha = 0.05$)	8/0 ($\alpha = 0.05$)	0/8	3/8	8/0 ($\alpha = 0.05$)	0/8	×

Table 11
The Friedman test.

Algorithm	Ranks achieved by the Friedman test							Results	
	GA	DE	PSO	HHO	FBI	AISA	GBO	Statistic	*p*-value
PSNR	7.00	1.00	3.00	6.00	5.00	4.00	2.00	48.00	1.1843×10^{-8}
Capacity	7.00	4.875	2.125	3.375	6.00	1.00	3.625	45.9643	3.0130×10^{-8}

This allows us to conclude that for the problem under study, the use of vectors of real numbers as individuals of the population is more efficient than binary vectors. This is because the solution found using GA (embedding option) is only a change map, and the changes are performed in the same way for all image blocks. Other metaheuristics choose the best values of the phase coefficients directly, therefore, the individual characteristics of each block are preserved, and a lower level of distortion is achieved.

7.2. Comparison with other schemes

In Table 14, we compare the performance of our scheme with other studies on steganographic embedding of information into the coefficients of other frequency transforms belonging to the same family of transforms using metaheuristic optimization. The proposed embedding scheme provides a lower capacity than analogues, but it has two important advantages. First, embedded information is extracted without errors in all cases due to a combination of iterative embedding and metaheuristic optimization. Secondly, our scheme does not require the transfer of additional information unique to each particular image for successful data extraction. The transmission of such information, for example, a location map, can compromise the presence of a hidden data transmission channel and make the use of steganography methods meaningless. Our scheme also does not require re-optimization at the extraction stage.

Table 12
Multiple comparisons among all algorithms for PSNR values.

No.	Hypothesis	Unadjusted *p*-values	Adjusted *p*-values	
			Shaffer	Bergmann–Hommel
1	GA vs. DE	2.7774×10^{-8}	5.8325×10^{-7}	5.8325×10^{-7}
2	GA vs. GBO	3.6726×10^{-6}	5.5089×10^{-5}	5.5089×10^{-5}
3	DE vs. HHO	3.6726×10^{-6}	5.5089×10^{-5}	5.5089×10^{-5}
4	GA vs. PSO	2.1283×10^{-4}	0.0032	0.0023
5	DE vs. FBI	2.1283×10^{-4}	0.0032	0.0023
6	HHO vs. GBO	2.1283×10^{-4}	0.0032	0.0023
7	GA vs. AISA	0.0055	0.0822	0.0493
8	DE vs. AISA	0.0055	0.0822	0.0493
9	PSO vs. HHO	0.0055	0.0822	0.0493
10	FBI vs. GBO	0.0055	0.0822	0.0493
11	GA vs. FBI	0.0641	0.7049	0.3845
12	DE vs. PSO	0.0641	0.7049	0.3845
13	PSO vs. FBI	0.0641	0.7049	0.3845
14	HHO vs. AISA	0.0641	0.7049	0.3845
15	AISA vs. GBO	0.0641	0.7049	0.3845
16	GA vs. HHO	0.3545	2.1272	1.0636
17	DE vs. GBO	0.3545	2.1272	1.0636
18	PSO vs. AISA	0.3545	2.1272	1.0636
19	PSO vs. GBO	0.3545	2.1272	1.0636
20	HHO vs. FBI	0.3545	2.1272	1.0636
21	FBI vs. AISA	0.3545	2.1272	1.0636

Table 13
Multiple comparisons among all algorithms for capacity values.

No.	Hypothesis	Unadjusted *p*-value	Adjusted *p*-values	
			Shaffer	Bergmann–Hommel
1	GA vs. AISA	2.7774×10^{-8}	5.8325×10^{-7}	5.8325×10^{-7}
2	FBI vs. AISA	3.6726×10^{-6}	5.5089×10^{-5}	5.5089×10^{-5}
3	GA vs. PSO	6.3805×10^{-6}	9.5707×10^{-5}	9.5707×10^{-5}
4	DE vs. AISA	3.3380×10^{-4}	0.0050	0.0037
5	PSO vs. FBI	3.3380×10^{-4}	0.0050	0.0037
6	GA vs. HHO	7.9051×10^{-4}	0.0119	0.0087
7	GA vs. GBO	0.0018	0.0267	0.0160
8	DE vs. PSO	0.0109	0.1199	0.0763
9	HHO vs. FBI	0.0151	0.1660	0.1056
10	AISA vs. GBO	0.0151	0.1660	0.1358
11	HHO vs. AISA	0.0279	0.3068	0.1673
12	FBI vs. GBO	0.0279	0.3068	0.1673
13	GA vs. DE	0.0491	0.4423	0.2948
14	DE vs. HHO	0.1649	1.1544	0.8246
15	PSO vs. GBO	0.1649	1.1544	0.8246
16	DE vs. GBO	0.2472	1.4830	0.8246
17	PSO vs. HHO	0.2472	1.4830	0.8246
18	DE vs. FBI	0.2976	1.4830	0.8929
19	PSO vs. AISA	0.2976	1.4830	0.8929
20	GA vs. FBI	0.3545	1.4830	0.8929
21	HHO vs. GBO	0.8170	1.4830	0.8929

Table 14
Comparison with other frequency schemes based on metaheuristic optimization.

Ref. no.	Freq. transform	Metaheuristic	Capacity, bpp (per color)	PSNR, dB	Extraction errors	Additional information
[17]	DCT	FOA + improved SOA	≈0.61–0.99	58.05–63.06	Yes	No, optimization is needed to extract
[18]	FrFT	FA	0.13–1.00	40.12–53.40	Yes	Yes, transmitted separately
[22]	DCT	CI	1.13	29.94–42.40	No	Yes, transmitted separately
Proposed	DFT	PSO, GBO, DE	0.09–0.27	38.47–43.14	No	No

7.3. Computational complexity

We also evaluated the impact of using metaheuristic optimization on the computational complexity of the algorithm. Searching for a suitable embedding option using exhaustive search requires a large number of embedding cycles, including the execution of the DFT and inverse DFT. This number can be determined by the following formula (for grayscale images):

$$O_{ES} = \left(2^{C_{\max}} - 1\right) \times \tau \times \frac{M \times N}{8 \times 8}, \quad (16)$$

where $M \times N$ is a cover image size.

In the case of exhaustive search, about 4.30×10^{10} embedding operations are needed for an image of size 512×512 and the number of iterations $\tau = 5$ in the worst case. Estimation of embedding operation number when using metaheuristic optimization is performed by the formula

$$O_{MO} = (G + 1) \times P \times \tau \times \frac{M \times N}{8 \times 8}. \quad (17)$$

When using the optimization parameters defined at the beginning of Section 6, in the worst case, about 3.38×10^{7} embedding operations are required. Thus, the use of metaheuristics significantly reduces the computational complexity of the algorithm.

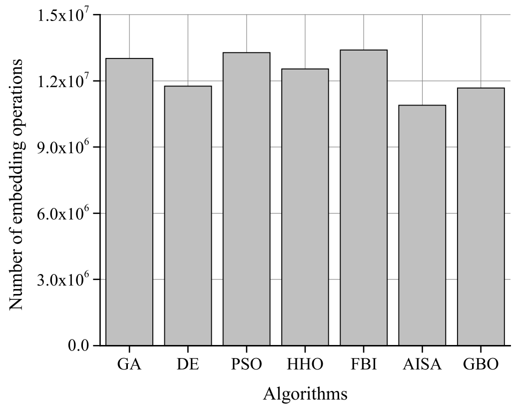

Fig. 17. Number of embedding operations for the metaheuristics.

The average number of embedding operations for different metaheuristics is shown in Fig. 17. In this case, we used all image blocks for embedding and did not use special processing for empty blocks. It can be seen that AISA is the most efficient in terms of the number of embeddings required.

7.4. Summary and future plans

Thus, all considered metaheuristics demonstrated a high level of robustness both when the algorithm is repeatedly executed for the same image, and when information is embedded in different images. The DE algorithm provided the best invisibility of embedding. The second and third places were taken by GBO and PSO, respectively. AISA was the best algorithm in terms of embedding capacity, PSO was the second, HHO and GBO were the third.

We presented an original formulation of an optimization problem for different classes of metaheuristics. The results showed that for solving this problem a population of real number vectors is much better than a population of binary vectors. Direct replacement of the phase values with the found real numbers preserves the features of a particular block and reduces the level of stego image distortion.

We proposed an improved algorithm for embedding information into the DFT phase spectrum using metaheuristics. The results of the experiments showed that the combination of the iterative embedding procedure and metaheuristic optimization ensures high embedding imperceptibility and error-free information extraction. Our algorithm does not require additional information unique to each image to extract information or re-optimization at the extraction phase. Searching for the best embedding options using metaheuristics significantly reduces the computational complexity of the algorithm compared to an exhaustive search.

The study has not shown a link between the class of metaheuristics and embedding efficiency. However, the high quality indicators obtained using GBO confirm the relevance of research on the efficiency of modern metaheuristics in the field of image steganography and watermarking. The results can be taken into account by researchers in the field of data hiding and applied to DFT and other frequency transformations.

Further research plans are aimed at scaling this study. We plan to use metaheuristic optimization to improve the quality of information embedding in the domains of other frequency transformations, such as DCT and DWT. In future research, we will evaluate the applicability of other metaheuristics, not considered in this article, to solving problems in the field of data hiding and the impact of the objective function design on the algorithm's performance.

8. Conclusion

In this study, we proposed an improved algorithm for data hiding in the DFT domain using metaheuristics, experimentally evaluated and compared the efficiency of different metaheuristics for embedding information into the DFT phase spectrum. We compared 7 metaheuristics, including well-known ones such as GA, DE, PSO, as well as fairly new ones such as HHO, FBI, AISA, and GBO. We also experimentally evaluate the influence of the main parameters of classical metaheuristics on optimization efficiency. We presented an original formulation of an optimization problem for different classes of metaheuristics. The results showed that for solving this problem a population of real number vectors is much better than a population of binary vectors. Direct replacement of the phase values with the found real numbers preserved the features of a particular block and reduced the level of stego image distortion. DE and PSO showed the best values of embedding indicators among the classical metaheuristic optimization algorithms. GBO showed the highest efficiency among modern algorithms. A statistical study that included the nonparametric sign test, the Friedman test, the Shaffer static procedure, and the Bergmann–Hommel procedure showed that the results were statistically significant.

CRediT authorship contribution statement

Anna Melman: Methodology, Writing – original draft, Writing – review & editing. **Oleg Evsutin:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work is an output of a research project implemented as part of the Basic Research Program at the National Research University Higher School of Economics (HSE University). We are very grateful to the anonymous referees for their constructive comments and helpful suggestions to improve the quality of this paper.

References

[1] D. Laishram, T. Tuithung, A novel minimal distortion-based edge adaptive image steganography scheme using local complexity, Multimedia Tools Appl. 80 (2021) 831–854, http://dx.doi.org/10.1007/s11042-020-09519-9.
[2] I.J. Kadhim, P. Premaratne, P.J. Vial, High capacity adaptive image steganography with cover region selection using dual-tree complex wavelet transform, Cogn. Syst. Res. 60 (2020) 20–32, http://dx.doi.org/10.1016/j.cogsys.2019.11.002.
[3] P. Garg, R. Rama Kishore, Secured and multi optimized image watermarking using SVD and entropy and prearranged embedding locations in transform domain, J. Discrete Math. Sci. Cryptogr. 23 (2020) 73–82, http://dx.doi.org/10.1080/09720529.2020.1721875.
[4] M. Baziyad, T. Rabie, I. Kamel, Toward stronger energy compaction for high capacity dct-based steganography: a region-growing approach, Multimedia Tools Appl. 80 (2021) 8611–8637, http://dx.doi.org/10.1007/s11042-020-10008-2.

[5] R. Wazirali, W. Alasmary, M.M.E.A. Mahmoud, A. Alhindi, An optimized steganography hiding capacity and imperceptibly using genetic algorithms, IEEE Access 7 (2019) 133496–133508, http://dx.doi.org/10.1109/ACCESS.2019.2941440.

[6] J. Wang, J. Ni, X. Zhang, Y.-Q. Shi, Rate and distortion optimization for reversible data hiding using multiple histogram shifting, IEEE Trans. Cybern. 47 (2017) 315–326, http://dx.doi.org/10.1109/TCYB.2015.2514110.

[7] Ş. Doğan, A new data hiding method based on chaos embedded genetic algorithm for color image, Artif. Intell. Rev. 46 (2016) 129–143, http://dx.doi.org/10.1007/s10462-016-9459-9.

[8] Z. Li, Y. He, Steganography with pixel-value differencing and modulus function based on PSO, J. Inf. Secur. Appl. 43 (2018) 47–52, http://dx.doi.org/10.1016/j.jisa.2018.10.006.

[9] A.H. Mohsin, A.A. Zaidan, B.B. Zaidan, O.S. Albahri, A.S. Albahri, M.A. Alsalem, K.I. Mohammed, S. Nidhal, Nawar.S. Jalood, A.N. Jasim, Ali.H. Shareef, New method of image steganography based on particle swarm optimization algorithm in spatial domain for high embedding capacity, IEEE Access 7 (2019) 168994–169010, http://dx.doi.org/10.1109/ACCESS.2019.2949622.

[10] A. Banharnsakun, Artificial bee colony approach for enhancing LSB based image steganography, Multimedia Tools Appl. 77 (2018) 27491–27504, http://dx.doi.org/10.1007/s11042-018-5933-5.

[11] S. Khan, M.A. Irfan, K. Khan, M. Khan, T. Khan, R.U. Khan, M.F. Ijaz, ACO based variable least significant bits data hiding in edges using IDIBS algorithm, Symmetry 12 (2020) http://dx.doi.org/10.3390/SYM12050781.

[12] D. Jude Hemanth, J. Anitha, D.E. Popescu, L.H. Son, A modified genetic algorithm for performance improvement of transform based image steganography systems, J. Intell. Fuzzy Syst. 35 (2018) 197–209, http://dx.doi.org/10.3233/JIFS-169580.

[13] H.M. Pandey, Secure medical data transmission using a fusion of bit mask oriented genetic algorithm, encryption and steganography, Future Gener. Comput. Syst. 111 (2020) 213–225, http://dx.doi.org/10.1016/j.future.2020.04.034.

[14] P.K. Muhuri, Z. Ashraf, S. Goel, A novel image steganographic method based on integer wavelet transformation and particle swarm optimization, Appl. Soft Comput. 92 (2020) 106257, http://dx.doi.org/10.1016/j.asoc.2020.106257.

[15] X. Kang, Y. Chen, F. Zhao, G. Lin, Multi-dimensional particle swarm optimization for robust blind image watermarking using intertwining logistic map and hybrid domain, Soft Comput. 24 (2020) 10561–10584, http://dx.doi.org/10.1007/s00500-019-04563-6.

[16] L. Salimi, A. Haghighi, A. Fathi, A novel watermarking method based on differential evolutionary algorithm and wavelet transform, Multimedia Tools Appl. 79 (2020) 11357–11374, http://dx.doi.org/10.1007/s11042-019-08455-7.

[17] R. Roselin Kiruba, T. Sree Sharmila, Secure data hiding by fruit fly optimization improved hybridized seeker algorithm, Multidimens. Syst. Signal Process. 32 (2021) 405–430, http://dx.doi.org/10.1007/s11045-019-00697-w.

[18] A. Amsaveni, M. Bharathi, Use of firefly optimization algorithm for fractional fourier transform based reversible data hiding, J. Intell. Fuzzy Syst. 40 (2021) 415–425, http://dx.doi.org/10.3233/JIFS-191911.

[19] B. Dappuri, M.P. Rao, M.B. Sikha, Non-blind RGB watermarking approach using SVD in translation invariant wavelet space with enhanced Grey-wolf optimizer, Multimedia Tools Appl. 79 (2020) 31103–31124, http://dx.doi.org/10.1007/s11042-020-09433-0.

[20] Y. Shen, C. Tang, M. Xu, M. Chen, Z. Lei, A DWT-svd based adaptive color multi-watermarking scheme for copyright protection using AMEF and PSO-GWO, Expert Syst. Res. Appl. 168 (2021) 114414, http://dx.doi.org/10.1016/j.eswa.2020.114414.

[21] D.K. Sarmah, A.J. Kulkarni, JPEG based steganography methods using Cohort Intelligence with Cognitive Computing and modified Multi Random Start Local Search optimization algorithms, Inform. Sci. 430–431 (2018) 378–396, http://dx.doi.org/10.1016/j.ins.2017.11.027.

[22] D.K. Sarmah, A.J. Kulkarni, Improved Cohort Intelligence—A high capacity, swift and secure approach on JPEG image steganography, J. Inf. Secur. Appl. 45 (2019) 90–106, http://dx.doi.org/10.1016/j.jisa.2019.01.002.

[23] A. Melman, O. Evsutin, On the efficiency of metaheuristic optimization for adaptive image steganography in the DFT domain, in: 2021 XVII International Symposium Problems of Redundancy in Information and Control Systems, REDUNDANCY, 2021, pp. 49–54, http://dx.doi.org/10.1109/REDUNDANCY52534.2021.9606459.

[24] O. Evsutin, A. Kokurina, R. Meshcheryakov, O. Shumskaya, The adaptive algorithm of information unmistakable embedding into digital images based on the discrete Fourier transformation, Multimedia Tools Appl. 77 (2018) 28567–28599, http://dx.doi.org/10.1007/s11042-018-6055-9.

[25] X. Fan, W. Sayers, S. Zhang, Z. Han, L. Ren, H. Chizari, Review and classification of bio-inspired algorithms and their applications, J. Bionic Eng. 17 (2020) 611–631, http://dx.doi.org/10.1007/s42235-020-0049-9.

[26] A.S. Azad, M.S.A. Rahaman, J. Watada, P. Vasant, J.A.G. Vintaned, Optimization of the hydropower energy generation using Meta-Heuristic approaches: A review, Energy Rep. 6 (2020) 2230–2248, http://dx.doi.org/10.1016/j.egyr.2020.08.009.

[27] R.R. Sahoo, M. Ray, Metaheuristic techniques for test case generation: A review, J. Inf. Technol. Res. (JITR) 11 (2018) 158–171, http://dx.doi.org/10.4018/JITR.2018010110.

[28] H. Stegherr, M. Heider, J. Hähner, Classifying metaheuristics: Towards a unified multi-level classification system, Nat. Comput. (2020) http://dx.doi.org/10.1007/s11047-020-09824-0.

[29] M. Ghaemi, M.-R. Feizi-Derakhshi, Forest optimization algorithm, Expert Syst. Appl. 41 (2014) 6676–6687, http://dx.doi.org/10.1016/j.eswa.2014.05.009.

[30] H. Shareef, A.A. Ibrahim, A.H. Mutlag, Lightning search algorithm, Appl. Soft Comput. 36 (2015) 315–333, http://dx.doi.org/10.1016/j.asoc.2015.07.028.

[31] F.A. Hashim, E.H. Houssein, M.S. Mabrouk, W. Al-Atabany, S. Mirjalili, Henry gas solubility optimization: A novel physics-based algorithm, Future Gener. Comput. Syst. 101 (2019) 646–667, http://dx.doi.org/10.1016/j.future.2019.07.015.

[32] A.R. Moazzeni, E. Khamehchi, Rain optimization algorithm (ROA): A new metaheuristic method for drilling optimization solutions, J. Petrol. Sci. Eng. 195 (2020) 107512, http://dx.doi.org/10.1016/j.petrol.2020.107512.

[33] S. Li, H. Chen, M. Wang, A.A. Heidari, S. Mirjalili, Slime mould algorithm: A new method for stochastic optimization, Future Gener. Comput. Syst. 111 (2020) 300–323, http://dx.doi.org/10.1016/j.future.2020.03.055.

[34] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, first ed., Addison-Wesley Longman Publishing Co., Inc., USA, 1989.

[35] R. Storn, K. Price, Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (1997) 341–359, http://dx.doi.org/10.1023/A:1008202821328.

[36] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, 1995, pp. 39–43, http://dx.doi.org/10.1109/MHS.1995.494215.

[37] M. Dorigo, T. Stützle, Ant Colony Optimization, A Bradford Book, Cambridge, MA, USA, 2004.

[38] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Global Optim. 39 (2007) 459–471, http://dx.doi.org/10.1007/s10898-007-9149-x.

[39] X.-S. Yang, Firefly algorithms for multimodal optimization, in: O. Watanabe, T. Zeugmann (Eds.), Stochastic Algorithms: Foundations and Applications, Springer, Berlin, Heidelberg, 2009, pp. 169–178, http://dx.doi.org/10.1007/978-3-642-04944-6_14.

[40] K. Zervoudakis, S. Tsafarakis, A mayfly optimization algorithm, Comput. Ind. Eng. 145 (2020) 106559, http://dx.doi.org/10.1016/j.cie.2020.106559.

[41] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: Algorithm and applications, Future Gener. Comput. Syst. 97 (2019) 849–872, http://dx.doi.org/10.1016/j.future.2019.02.028.

[42] N. Razmjooy, M. Khalilpour, M. Ramezani, A new Meta-Heuristic optimization algorithm inspired by FIFA world cup competitions: Theory and its application in PID designing for AVR system, J. Control Autom. Electr. Syst. 27 (2016) 419–440, http://dx.doi.org/10.1007/s40313-016-0242-6.

[43] A. Milani, V. Santucci, Community of scientist optimization: An autonomy oriented approach to distributed optimization, AI Commun. 25 (2012) 157–172, http://dx.doi.org/10.3233/AIC-2012-0526.

[44] E. Bogar, S. Beyhan, Adolescent Identity Search Algorithm (AISA): A novel metaheuristic approach for solving optimization problems, Appl. Soft Comput. 95 (2020) 106503, http://dx.doi.org/10.1016/j.asoc.2020.106503.

[45] J.-S. Chou, N.-M. Nguyen, FBI inspired meta-optimization, Appl. Soft Comput. 93 (2020) 106339, http://dx.doi.org/10.1016/j.asoc.2020.106339.

[46] B. Das, V. Mukherjee, D. Das, Student psychology based optimization algorithm: A new population based optimization algorithm for solving optimization problems, Adv. Eng. Softw. 146 (2020) 102804, http://dx.doi.org/10.1016/j.advengsoft.2020.102804.

[47] Y. Zhang, Z. Jin, Group teaching optimization algorithm: A novel metaheuristic method for solving global optimization problems, Expert Syst. Appl. 148 (2020) 113246, http://dx.doi.org/10.1016/j.eswa.2020.113246.

[48] A.J. Kulkarni, I.P. Durugkar, M. Kumar, Cohort intelligence: A self supervised learning behavior, in: IEEE International Conference on Systems, Man, and Cybernetics, Vol. 2013, 2013, pp. 1396–1400, http://dx.doi.org/10.1109/SMC.2013.241.

[49] H. Salimi, Stochastic Fractal Search: A powerful metaheuristic algorithm, Knowl.-Based Syst. 75 (2015) 1–18, http://dx.doi.org/10.1016/j.knosys.2014.07.025.

[50] E.H. Houssein, M.R. Saad, F.A. Hashim, H. Shaban, M. Hassaballah, Lévy flight distribution: A new metaheuristic algorithm for solving engineering optimization problems, Eng. Appl. Artif. Intell. 94 (2020) 103731, http://dx.doi.org/10.1016/j.engappai.2020.103731.

[51] I. Ahmadianfar, O. Bozorg-Haddad, X. Chu, Gradient-based optimizer: A new metaheuristic optimization algorithm, Inform. Sci. 540 (2020) 131–159, http://dx.doi.org/10.1016/j.ins.2020.06.037.
[52] I.J. Kadhim, P. Premaratne, P.J. Vial, B. Halloran, Comprehensive survey of image steganography: Techniques, evaluations, and trends in future research, Neurocomputing 335 (2019) 299–326, http://dx.doi.org/10.1016/j.neucom.2018.06.075.
[53] SIPI Image Database, <http://sipi.usc.edu/database>.
[54] Kodak Lossless True Color Image Suite, <http://r0k.us/graphics/kodak/index.html>.
[55] J. Derrac, S. García, D. Molina, F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, Swarm Evol. Comput. 1 (2011) 3–18, http://dx.doi.org/10.1016/j.swevo.2011.02.002.