CHINESE OPTICS LETTERS

Vol. 19, No. 1 | January 2021

Intelligent algorithms: new avenues for designing nanophotonic devices [Invited]

Lifeng Ma (马丽凤)[1], Jing Li (李 靖)[1], Zhouhui Liu (刘舟慧)[1], Yuxuan Zhang (张宇轩)[1], Nianen Zhang (张念恩)[1], Shuqiao Zheng (郑澍乔)[1], and Cuicui Lu (路翠翠)[1,2*]

[1] Key Laboratory of Advanced Optoelectronic Quantum Architecture and Measurements of Ministry of Education, Beijing Key Laboratory of Nanophotonics and Ultrafine Optoelectronic Systems, School of Physics, Beijing Institute of Technology, Beijing 100081, China

[2] Collaborative Innovation Center of Light Manipulations and Applications, Shandong Normal University, Jinan 250358, China

*Corresponding author: cuicuilu@bit.edu.cn

Received June 10, 2020 | Accepted September 4, 2020 | Posted Online December 14, 2020

The research on nanophotonic devices has made great progress during the past decades. It is the unremitting pursuit of researchers that realize various device functions to meet practical applications. However, most of the traditional methods rely on human experience and physical inspiration for structural design and parameter optimization, which usually require a lot of resources, and the performance of the designed device is limited. Intelligent algorithms, which are composed of rich optimized algorithms, show a vigorous development trend in the field of nanophotonic devices in recent years. The design of nanophotonic devices by intelligent algorithms can break the restrictions of traditional methods and predict novel configurations, which is universal and efficient for different materials, different structures, different modes, different wavelengths, etc. In this review, intelligent algorithms for designing nanophotonic devices are introduced from their concepts to their applications, including deep learning methods, the gradient-based inverse design method, swarm intelligence algorithms, individual inspired algorithms, and some other algorithms. The design principle based on intelligent algorithms and the design of typical new nanophotonic devices are reviewed. Intelligent algorithms can play an important role in designing complex functions and improving the performances of nanophotonic devices, which provide new avenues for the realization of photonic chips.

Keywords: intelligent algorithms; nanophotonic devices; deep learning methods; swarm intelligence; genetic algorithm.
DOI: 10.3788/COL202119.011301

1. Introduction

The various technical challenges that traditional electronic devices have faced in recent years suggest that Moore's law is becoming increasingly difficult to maintain[1]. As a promising successor to electronic devices, nanophotonic devices have become the focus of optical research, and the design of devices has always been one of the core topics of nanophotonic devices. Traditional design methods for nanophotonic devices, which rely on experience and physical inspiration for structural design and parameter optimization, cannot find the best performances of devices each time, and usually require a long calculation period with parameter sweeps[2,3]. With the increasing demand for device performance in practical applications, deep learning methods, the gradient-based inverse design method, swarm intelligence algorithms, individual inspired algorithms, and some other intelligent algorithms have been proposed to overcome the shortcomings of traditional design methods. These intelligent algorithms transfer the "method orientation" to "problem orientation," which is more suitable for solving various kinds of optimization problems.

Intelligent algorithms are, in many cases, practical alternative techniques for solving varieties of challenging engineering problems[4]. Intelligent algorithms are also methods inspired by natural phenomena or laws, and people learn and imitate natural principles and use intelligent algorithms to solve practical problems. For example, the deep learning method, which can realize the recognition and classification of speech and images close to human learning behavior, may replace humans as the operator in the optimization process; the swarm intelligence algorithms, which simulate the behavior of animal groups in searching for food and constantly change the search direction through learning experience, can help people find the optimal path. In many practical applications, intelligent algorithms are practical techniques to deal with various challenging problems, such as the design of widely used structures in nanophotonic devices: photonic crystals[5], optical materials[6], etc.

Intelligent algorithms can help us solve these design problems. Whether it is an intricate continuous problem or a discrete problem to be optimized, intelligent algorithms can be applied under both cases and find feasible solutions in a short time. It shows that the avenues of designing nanophotonic devices based on intelligent algorithms will be an important direction for the future development of nanophotonics. Intelligent algorithms are important core techniques for parameter tuning and computer-aided design of devices, which can establish a clear and intuitive physical scene for the device's working principle. The design efficiency can be greatly improved by using the appropriate algorithm, and the best performance of the device can be expected. Therefore, the study of intelligent algorithms is of great practical significance to the design of nanophotonic devices. The design of nanophotonic devices based on intelligent algorithms will play a more significant role in improving manufacturing capacity and levels in future.

In this review article, the deep learning method, the gradient-based inverse design method, swarm intelligence algorithms [including genetic algorithm (GA), particle swarm optimization (PSO), and ant colony algorithm (ACA)], individual inspired algorithms [including the simulated annealing algorithm (SAA), the hill climbing algorithm, and tabu search (TS)], and some other algorithms [including the direct binary search (DBS) algorithm, topology optimization, and Monte Carlo method] are introduced from research background or concept to applications for designing nanophotonic devices. A summary of the intelligent algorithms and their applications for designing nanophotonic devices is shown in Fig. 1. Corresponding application examples of nanophotonic devices are listed under each mentioned intelligent algorithm. The advances in the design of nanophotonic devices using various intelligent algorithms may bring new inspiration for further research of nanophotonic structures and devices. Recently, our group has developed an intelligent algorithm by combining GA and the finite element method (FEM), and we have realized on-chip wavelength routers[7] and polarization routers[8], which are the smallest ones to date based on optimization algorithms. We also introduce the SAA into GA and FEM and have realized cascaded nanophotonic devices of broadband filter and wavelength routers. Based on this, more novel structures are predicted, and the device performances will be more excellent.

This review includes seven sections. The first section is the introduction, where we illustrate the purpose of writing this review. The second section is about the deep learning method, especially the artificial neural network, where the history, principle, and applications are demonstrated. In the third section, the gradient-based inverse design is introduced, including the adjoint algorithm for optimizing the parameters of nanophotonic devices, which is a further improvement on the gradient-based inverse design method and solves the problems of the system that follows the known laws of physics. The fourth section focuses on swarm intelligence algorithms, introducing GA and PSO, which have been widely used in recent years, as well as ACA, which is often used to optimize the design of solar devices. The fifth section is the individual inspired algorithms including the SAA, the hill climbing algorithm, and the TS algorithm, which are introduced from the aspects of concept, development process, and application. The sixth section is some other intelligent algorithms, including DBS, topology optimization, and Monte Carlo method, which play an important role for designing the multiplexer, band structures, optical imaging, etc. The last section is the summary, which summarizes the advantages of intelligent algorithms in designing complex

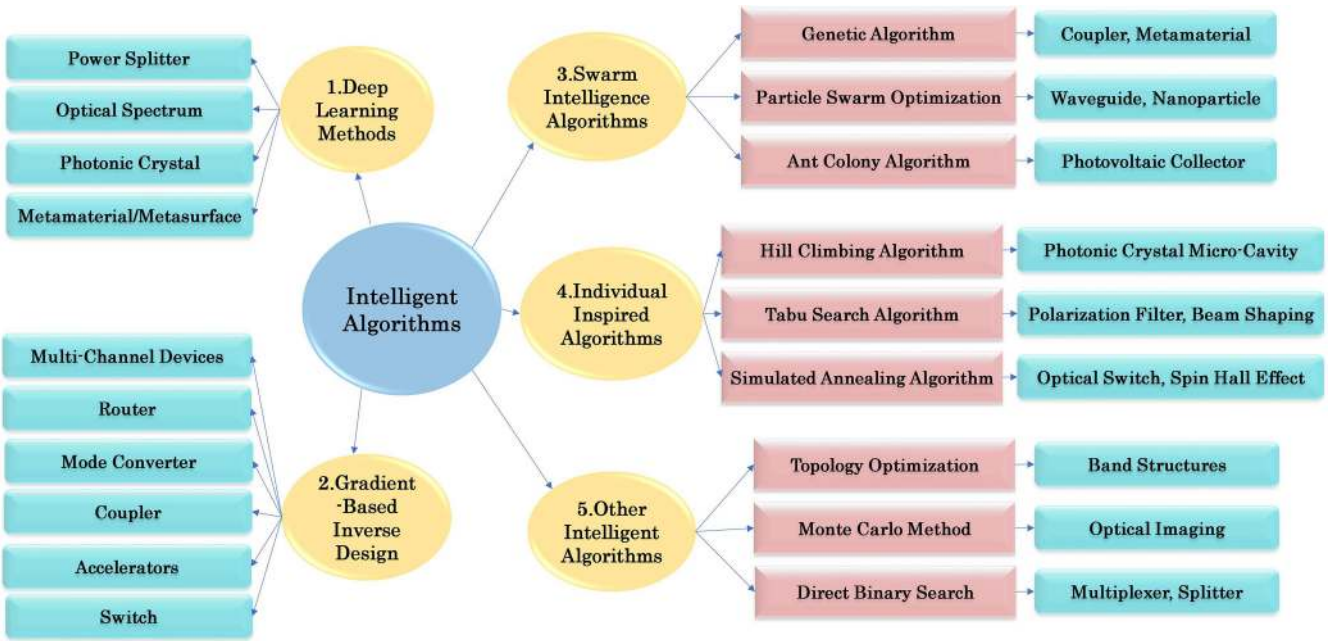

Fig. 1. Summary of intelligent algorithms and their applications for designing nanophotonic devices in this review.

functions and improving device performance for designing nanophotonic devices, and explains the development trend of using intelligent algorithms, especially in the design of nanophotonic devices in future.

2. Nanophotonic Devices Based on Deep Learning Methods

In 2016, after the artificial intelligence (AI) "AlphaGo" defeated the go world champion Lee Sedol, a terminology called "deep learning" was then firmly printed in people's minds[9]. In fact, deep learning is a form of machine learning, and machine learning is a branch of AI. Machine learning is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. However, hampered by the poor hardware performance at that time, early machine learning approaches such as connectionism were not suitable for complex learning tasks, and often suffered from problems like overfitting.

In order to get rid of these troubles, researchers try to develop an algorithm that integrates the process of feature learning into the process of machine learning, which is so-called representation learning. Deep learning is a typical kind of representation learning (see Fig. 2 for the inclusion relation of these three with AI). Deep learning experienced a long time before AlphaGo was a blockbuster, and in the past people almost gave it up. It was not until 2006, when Hinton *et al.* proposed a model called the "deep confidence network" that deep learning was back in the spotlight[10]. It can be said that it is the increment of effective data, the implementation of high-performance computing hardware, and the improvement of training methods that make deep learning play a significant role in the field of artificial intelligence. As a kind of machine learning, the emergence of deep learning has greatly promoted the development of artificial intelligence. Deep learning methods are algorithms closer to human learning behavior, so deep learning has made remarkable achievements in many fields, such as speech recognition, image recognition, and classification.

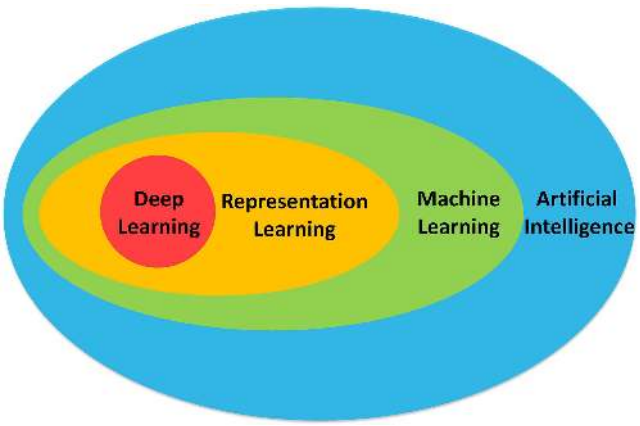

Fig. 2. Inclusion relation of machine learning, representation learning, deep learning, and artificial intelligence.

The core of deep learning is the design of the artificial neural network (ANN). As the term suggests, the structure of the ANN is based on the simulation of the neural network of the human brain. Some of the neurons activate the messages received from somewhere else and then pass them onto other neurons. That is to say, deep learning methods are representation learning methods with multiple levels of representation, obtained by composing simple but non-linear modules that each transforms the representation at one lower level into a representation at a higher and slightly more abstract level. With the composition of enough transformations, complex functions can be learned[11].

2.1 Introduction to deep-learning method

In this part, the utilization in nanophotonic devices through the deep-learning method will be introduced and illustrated. First of all, when applying the deep learning method, a certain number of training data need to be generated and then the quantitative characteristics of a group of data in the form of a one-dimensional vector $\boldsymbol{x}$ are input to the neural network. The input information is processed in the first layer (i.e., the layer after the input layer) and then transferred to the next layer. Taking the neurons in the lth layer as an example (see Fig. 3), the neural network in the lth layer has n_l neurons (the number of layers, the number of neurons in each layer, and other preset parameters before training the neural system are called hyperparameters). Z and A are used to represent the information before and after processing with the activation function $g = g(Z)$, respectively. Then the processing of information by neurons in the lth layer can be expressed as follows:

$$Z_{[l]} = \Theta_{[l-1]}^{T} A_{[l-1]}, \tag{1}$$

$$A_{[l]} = g(Z_{[l]}) \tag{2}$$

Here, $\Theta_{[l-1]}$ is an $n_{l-1} \times n_l$ matrix. The function g is the activation function, and the commonly used non-linear activation functions are the sigmoid function, the ReLu function, the Tanh function, etc. When the information is transferred to the last layer and activated, the so-called prediction value $\hat{Y}$ is obtained. After selecting the appropriate cost function, the chain rule is used for backpropagation and the stochastic gradient descent (SGD) algorithm is used to update the value of the parameters (weight and bias) of each neuron in the neural network, which ends a process of training. After feeding a large number of training data, it is expected that Θ in ANN will be updated to a value more suitable for dealing with similar problems, i.e., the cost function converges to a local minimum. By using the test set after the training process, problems such as underfitting and overfitting can be detected. Specifically, this is done by calculating variance and bias, and drawing the learning curve during the testing process.

Usually, ANN can deal with two kinds of problems: regression problems and classification problems. The time consumed to

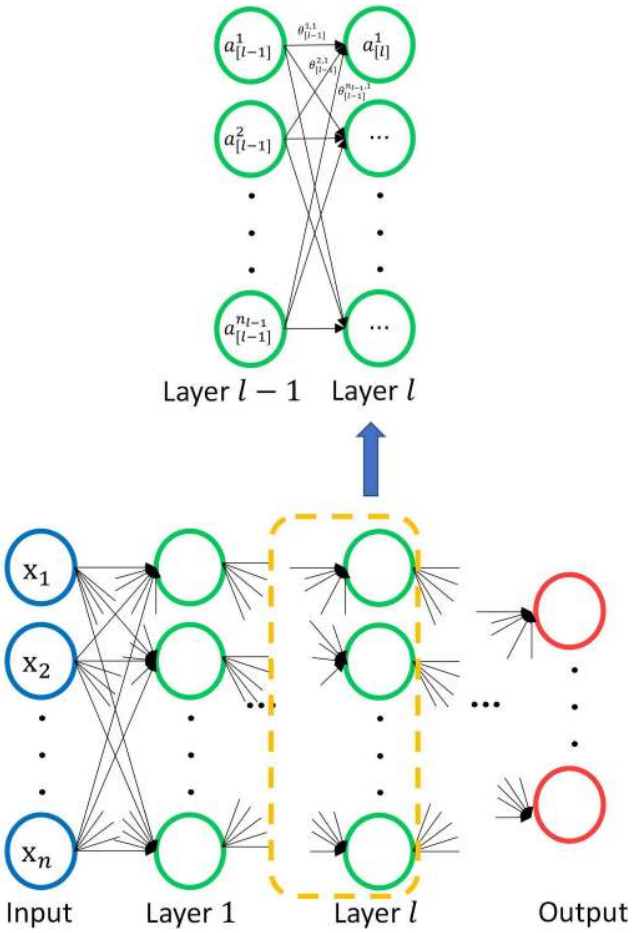

Fig. 3. Neurons in each layer process and transfer data in the form of column vectors, and the weights of neural networks are expressed as matrices. The θ represents the element of the matrix Θ. It is worth noting that we did not list all the weights of layer $l-1$, but only the information processing of the first neuron in the lth layer is shown here.

train an ANN is a reference to evaluate an ANN. Also, when evaluating the performance of a neural network in classification problems, parameters such as precision and recall are often introduced, even though sometimes it is necessary to trade off these two parameters. Some ways can be used to improve the performance. The main ways are as follows.

a. Longer training time and more training data. (Sometimes it does not work or even makes the ANN perform worse.)
b. Select an appropriate architecture of ANNs.
c. Select an appropriate algorithm (such as the Adam algorithm and dropout strategy)
d. Select the appropriate activation functions.
e. Fine tune the hyperparameters (e.g., the number of layers and the number of neurons per layer).

Deep learning methods have been applied in many fields, including the design of nanophotonic devices. In order to design and evaluate a nanophotonic device, it is necessary to predict the optical response, and the prediction is usually implemented by solving Maxwell's equations using dedicated numerical methods[12], which is rather time-consuming. The trained ANN can be used to predict the optical response quickly by forward propagation. The trained neural network can also be used to design nanophotonic devices with high efficiency.

2.2 Typical architectures of ANNs

In this part, typical architectures of ANNs will be introduced and illustrated. There are several typical architectures of ANNs that are often adopted to design and optimize nanostructures with different functions.

Malkiel *et al.* trained and tested a bidirectional deep-learning architecture with the capability of predicting the geometry of nanostructures solely based on the far-field response of the nanostructures, and the prediction is accurate[13]. Once this deep neural network (DNN) is trained, the geometry of the nanostructure can be obtained by querying the inverse network according to the measured/expected transmission spectrum. Then the obtained geometry is input into the direct network after training, and the direct network calculates the predicted transmission spectrum [see Fig. 4(a)]. When dealing with the inverse scattering problem using neural networks, it often suffers from a typical non-uniqueness problem, which makes it rather difficult to train neural networks on a training set with a large amount of data. Liu *et al.* demonstrated a tandem network (TN) that tolerates both explicit and implicit nonunique training instances [see Fig. 4(b)]. The forward modeling network is trained in advance, so during the training process weights in the pretrained forward modeling network are fixed and the weights in the inverse network are adjusted to reduce the value of the cost function (i.e., the error between the predicted response and the target response). The outputs of the intermediate layer M are the designed parameters of the device. It provides a method for training large-scale neural networks for the inverse design of complex photonic structures[14]. Metasurfaces are versatile and novel platforms for manipulating the scattering, color, phase, or intensity of light. Lei *et al.* also utilized a TN to optimize a metasurface in order to reduce the computational cost significantly[15]. They proved that the metasurfaces can achieve up to 400 times the third harmonic enhancement after optimization.

In most cases, neural networks with more layers perform better, whereas fully connected deep neural networks (FCDNNs) generally suffer from the problem of vanishing gradients. As a result, increasing the depth of an FCDNN does not necessarily improve the performance. Kojima *et al.* solved this problem by using a residual deep neural network [ResNet, see Fig. 4(d)] to improve the depth of training up to 8 hidden layers for both the forward and inverse problem[16]. It takes them about two weeks to complete collecting the 20,000-simulation data by numerical simulations while approximately 22 min to train the neural network. Once again, it reflects the high efficiency of the deep learning method.

As a typical neural network structure, the convolutional neural network (CNN) has been successfully applied in the field

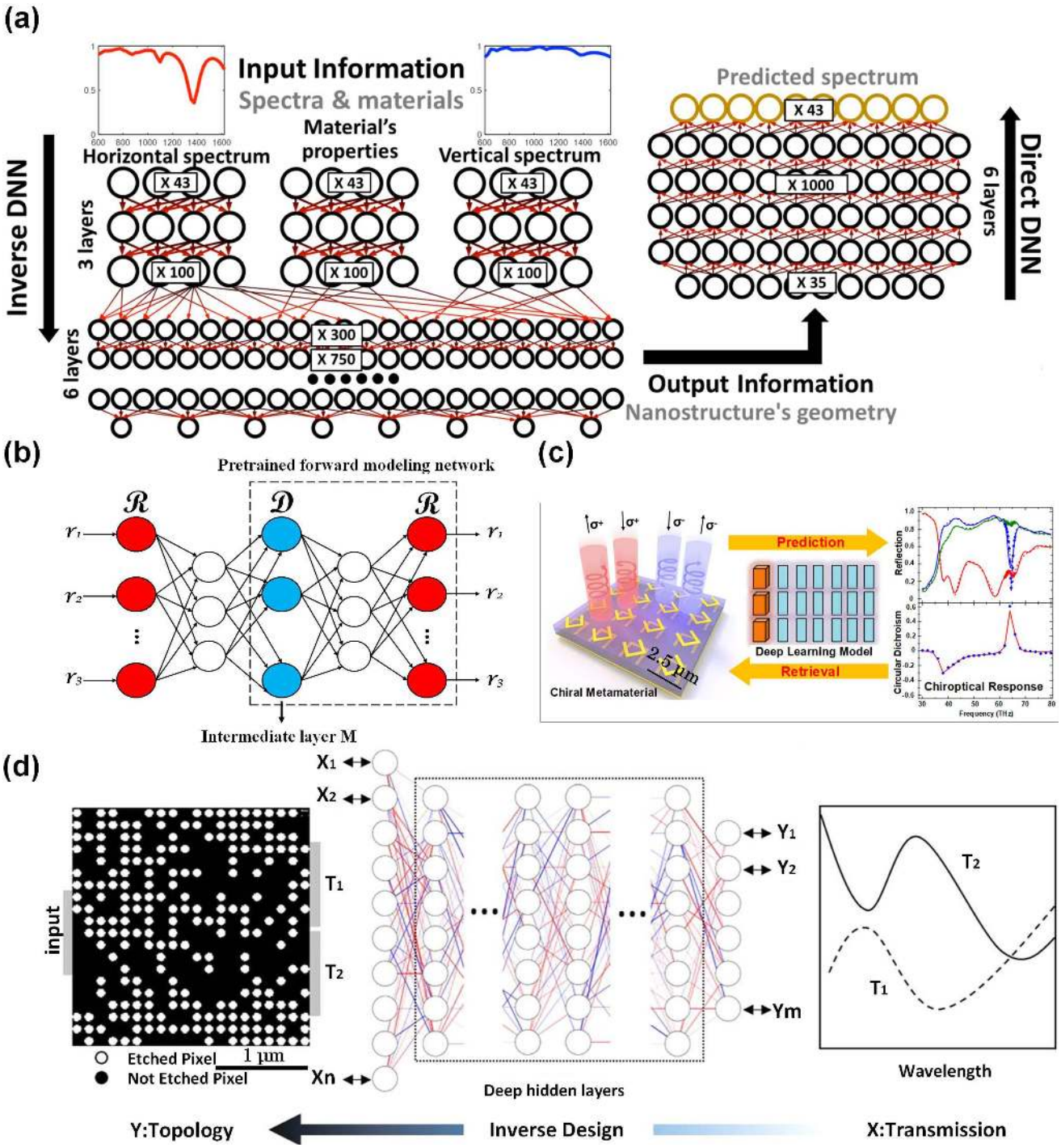

Fig. 4. (a) Bidirectional network used for inverse design[13]. (b) The TN consists of an inverse design network and a forward modeling network[14]. (c) A CNN consists of two bidirectional neural networks, and it is capable of automatically designing and optimizing three-dimensional (3D) chiral metamaterials with strong chiral-optical responses at specified wavelengths[17]. (d) A DNN for forward and inverse design of a power splitter[16].

of image recognition and is now also used in the design of nanophotonic devices. The two main advantages of CNNs over FCDNNs are parameter sharing and sparsity of connections (i.e., in each layer, each output value depends only on a small number of inputs, which somewhat avoids the problem of overfitting and is more suitable to deal with the design problems with more parameters). Ma *et al.* reported a CNN model comprising of two bidirectional neural networks assembled by a partial stacking strategy [see Fig. 4(c)], to automatically design and optimize 3D chiral metamaterials with strong chiral-optical responses at predesignated wavelengths[17]. Wu *et al.* used CNN to predict the topological invariant of a 1D photonic

crystal (PC) for geometric configurations that lie outside the parameter space of the training dataset[18], as shown in Fig. 5(a). Zhao *et al.* designed an optical fiber imaging system based on a deep CNN that can transmit real-time non-artificial cell images through a one-meter-long Anderson localizing optical fiber[19]. They showed that trained neural networks could learn to retrieve the images of cells with very different shapes and categories that have never been "seen" during training.

Another type of neural network whose application range is extended rapidly is the generative adversarial network (GAN)[20]. Generating samples is a harder problem compared to discriminative models. Given a training set, the GAN learns to generate new data with the same statistics as the training set. This feature enables the GAN to be applied to inverse design. Fan *et al.* showed that the GAN can be trained from periodic and topologically optimized metagratings images to produce efficient and topologically complex devices that can operate over a wide range of deflection angles and wavelengths with a one-time computational cost[21]. However, GAN models sometimes suffer from problems such as mode collapse, non-convergence, and diminished gradient. Ma *et al.* presented a probabilistic model of a variational auto-encoder (VAE) for the design of devices[22]. A semi-supervised learning strategy is used in this work, which improves the performance of the model. The GAN can be combined with other generation model methods of deep neural networks, such as auto-encoder (AE), to improve the stability of the network. For example, Tang *et al.* utilized a novel conditional variational auto-encoder (CAVE) [see Fig. 5(b)] for their power splitter design application[23]. They succeeded in using only binary-level nanophotonic datasets to generate a power splitter with an arbitrary ratio in the bandwidth between 1250 nm and 1800 nm. The FDTD simulations confirm that the overall transmission is close to 90%.

Recently, benefiting from the development of deep learning itself and the open source software libraries such as TensorFlow, there are more and more reports of applying neural networks to the design of nanophotonic devices, and the overall trend is that the structure of neural networks is more advanced and complex. In addition, taking advantage of the "black box" characteristics of deep learning (i.e., people do not care about its internal structure, but only its input and output), some novel algorithms have been invented by modifying the deep learning method. Zhou *et al.* designed two programmable optical signal processing chips with a learning ability based on the idea of the deep

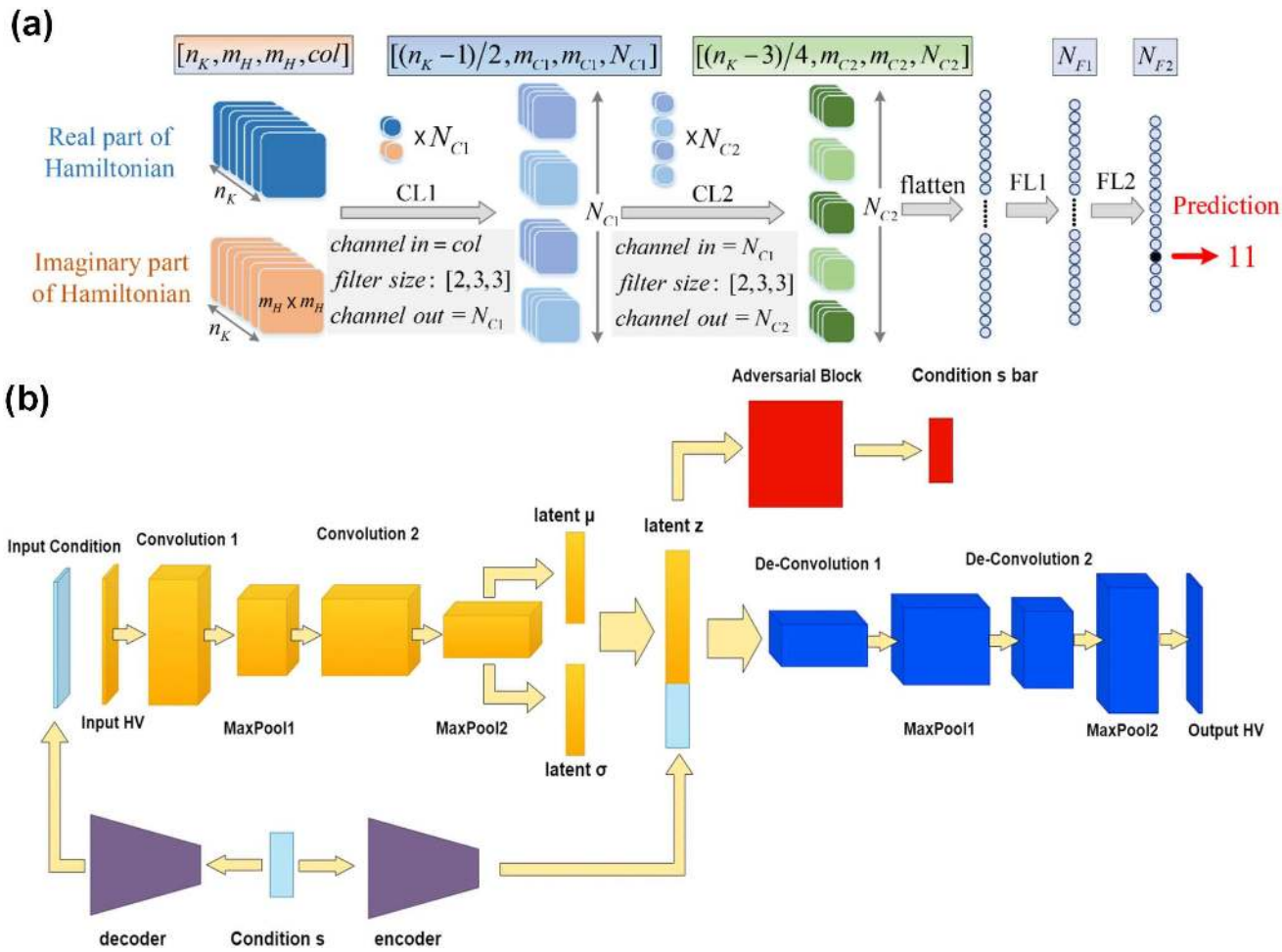

Fig. 5. (a) CNN used to predict the invariance of 1D photonic crystal[18]. (b) A novel CAVE for the design of a power splitter[23].

learning method[24–26]. The chip can be trained to perform the desired function by the gradient descent method and can be treated as a black box without having to know the internal information.

2.3 Discussion and outlook

Deep learning methods have many advantages over traditional algorithms. First, one advantage of deep learning is that once trained it costs less time than traditional algorithms (i.e., less computational cost) and is more likely to find better local optimal solutions. For example, using neural networks to predict the spectrum of a nanoscale optical device tends to be more accurate than traditional algorithms. Hammond *et al.* trained ANNs to model both strip waveguides and chirped Bragg gratings, and they found that the trained ANNs decreased the computational cost relative to the traditional design methodologies by more than 4 orders of magnitude[27]. As a result of the higher efficiency of designing well-performed devices compared to the traditional algorithm, deep learning methods have also been successfully employed in other areas such as high-energy physics[28], condensed matter[29], chemical physics[30], and holography[31]. Second, compared with traditional optimization algorithms, deep learning methods can realize inverse design more easily and the deep learning methods for discovering optical structures based on desired functional characteristics have made rapid progress[13,14,16,27,32–42]. Third, the deep learning neural network has many typical structures and strong flexibility. We can choose the appropriate neural network for optimal design according to the needs of design devices, and many problems in the training process can be solved by adjusting the hyperparameters or structure of the neural network appropriately. Last, as one of the frontiers of computer science, deep learning is still in development with its wide-ranging applications. The one that gets the most attention is the all-optical neural network. Optical computing systems have attracted more and more attention due to the importance of cutting down computing costs. Optical computing has the advantages of low energy consumption, scalability, no photoelectric conversion, and broad bandwidth, and can be used as special accelerating hardware for AI algorithms (such as DNN). Shen *et al.* proposed a new architecture for an all-optical neural network that greatly improved the computational speed of dealing with conventional learning tasks[43]. Feldmann *et al.* showed the development of a new kind of all-optical neural network[44], and Lin *et al.* also realized all-optical machine learning through a diffraction deep neural network[45]. Some other architectures have been proposed as well such as Mach–Zehnder interferometers[43], single-pixel imaging[46], nanophotonic medium[47], and Fourier optics[48]. The development of all-optical neural networks may revolutionize the field of computing.

However, deep learning methods also have some limitations and drawbacks. First, since the design of nanophotonic devices is a non-convex problem, it is impossible to guarantee that the designed devices are optimal. Jiang *et al.* presented a global optimizer that performs a global search for the optimal device within the design space, but the final devices may not be the optimal[49]. Second, it takes a lot of computational cost and time cost to prepare the training set and train the ANN, especially when dealing with complex learning tasks. In order to train the CNN to predict the optical responses of arbitrary structures by 2D cross-section images, Qu *et al.* spent 15 days preparing input data and nearly a week training the network[50]. However, the introduction of unsupervised learning and transfer learning algorithms can help to release the burden on data[39,51]. Third, it is sometimes difficult to exploit the trained neural network for further analysis because the learning mechanisms of ANN (note that these mechanisms can be useful sometimes) are operating as black boxes, whereas, the useful information about the features of photonic structures can be extracted with proper techniques, such as introducing latent space[22]. Fourth, stronger ability of transfer learning is needed to cope with the changeable situations[52], although this in-development ability has shown its power in the design of nanophotonic devices[51]. Last, when the number of training samples is small, conventional methods may perform better than deep learning methods. Jiao *et al.* found that linear-regression-based methods may outperform the deep learning approaches for two black-box optical imaging problems[53].

At the end of this section, some prospects for neural networks are given. Based on the outstanding performance of the deep learning method in the nanophotonic field and the analysis of a number of papers, we can confidently predict that there will be less nanophotonic device design works in the future without a deep learning algorithm. Its flexibility also facilitates it to be an excellent candidate for handling other nanophotonic problems[54–56]. Additionally, as deep learning methods have a better transfer learning capability than the traditional machine learning methods, deep transfer learning also shines in other fields[57,58]. We believe that in the future the trained neural network can be used not only for designing specific devices, but also for designing new devices, which means less time required to design devices with different functions and a wider parameter space to search for the optimal solution based on a pre-trained ANN.

3. Nanophotonic Devices Based on the Gradient-Based Inverse Design

Between the 1870s and 1880s, the importance of inverse problems has grown considerably in many fields. The mathematical expression of a physical law is a rule that defines a mapping T of a set of functions ξ called the parameters into a set of functions δ called the results. According to the above expression, to find inverse mappings of δ into ξ, inverse problems can be defined in a precise mathematical form that excludes the so-called "fitting procedure" in which models depending on a few parameters and giving a good fit of the experimental results are obtained by trial and error or any other techniques[2,59]. In the field of nanophotonic devices, formulas for inverse problems have been widely understood, and the application of computational

methods based on inverse design for nanophotonic devices has recently grown considerably[2,3]. There are two central thrusts of inverse problems in nanophotonics, which are the determination of solution characteristics and the discovery of effective algorithms for working from desired characteristics to physical systems.

3.1 Introduction to the gradient-based inverse design

The Vuckovic group at Stanford University reported an inverse design algorithm and there are a variety of nanophotonic devices designed by the algorithm, such as multi-channel devices, power splitter (router)[60–62], grating coupler[63], TE or TM mode converters, mode routers such as TE/TM and wavelength router, spatial mode[3,60,62,64,65], dielectric laser accelerators[66], and non-reciprocal pulse router and switch[67,68]. Considering the general formulation of the inverse design problem for optical devices, the gradient-based inverse design algorithm specifies device functionality by describing the mode conversion efficiency between a set of input modes and output modes[60,62–65]. The input and output modes are specified by the user and keep fixed during the optimization process. The input modes $i = 1, \ldots, M$ are at frequencies ω_i, and can be represented by an equivalent current density distribution J_i. The fields E_i produced by each input mode satisfy Maxwell's equations

$$\nabla \times \mu_0^{-1} \nabla \times E_i - \omega_i^2 \varepsilon E_i = -i\omega_i J_i, \tag{3}$$

where ε is the permittivity distribution and μ_0 is the permeability of free space. For each input mode i, a set of output modes $j = 1, \ldots, N_i$ are specified, whose amplitudes are bounded between α_{ij} and β_{ij}. If the output modes are guided modes of waveguides with modal electric fields ε_{ij} and magnetic fields H_{ij}, this constraint can be written using a mode orthogonality relationship

$$\alpha_{ij} \le \left| \iint (E_i \times H_{ij} + \varepsilon_{ij} \times H_i) n \mathrm{d}r_\perp \right| \le \beta_{ij}. \tag{4}$$

Here, n is a unit vector pointing in the propagation direction and $r_\perp$ denotes the coordinates perpendicular to the propagation direction. Faraday's law

$$\nabla \times E_i = -i\omega\mu_0 H_i \tag{5}$$

can be used to rewrite Eq. (4) purely in terms of the electric field:

$$\alpha_{ij} \le \left| \iint (E_i \times H_{ij} + \varepsilon_{ij} \times \frac{i}{\omega\mu_0} \nabla \times E_i) n \mathrm{d}r_\perp \right| \le \beta_{ij}. \tag{6}$$

More generally, the output mode amplitude can be specified in terms of a linear function L_{ij} of the electric field E_i:

$$\alpha_{ij} \le \left| \iint L_{ij}(E_i) \right| \le \beta_{ij}, \tag{7}$$

where $V = \{E : R^3 \to C^3\}$ is the space of all possible electric field distributions and $L_{ij} : V \to C$ maps the electric field distributions to a complex scalar. R is the set of real numbers, C is the set of complex numbers, and V is the electric field distributions.

After the problem formulation, the gradient-based inverse design algorithm solves Maxwell's equations numerically and employs numerical optimization techniques to design devices. It uses two methods to solve this problem: the 'objective first' method and a 'steepest descent' method. In the objective first method, the algorithm constrains the electric fields E_i to satisfy the performance constraints in Eq. (7). Then the algorithm minimizes the violation of physics using the alternating directions method of multipliers (ADMM) optimization algorithm[60,62–65].

3.2 Application of the gradient-based inverse design

The gradient-based inverse design algorithm is a relatively general computational method for nanophotonic design that is widely used in the design of nanophotonics devices. In this section, we present some typical nanophotonic devices designed by the inverse design algorithm. The multi-channel device, which is called a hub by its designers, is shown in Fig. 6(a)[62]. The device has two input waveguides, two output waveguides, and two wavelengths, hence called a $2 \times 2 \times 2$ hub. The performance specification of the multi-channel device is that the input and output modes all consist of the fundamental TE-polarized mode at either the 1550 nm or 1310 nm wavelength. This hub directs input arms 1 and 2 into output arms 1 and 2 at 1550 nm, but swaps them at 1310 nm. The electromagnetic energy density distribution of the hub is shown in Fig. 6(b).

In addition to the multi-channel device in a single polarization mode, the algorithm is also used to design devices that can exhibit different functionality for different input excitations, such as the mode converters. Figure 6(c) is a schematic diagram of the TE mode converter, which is a mode conversion device operating in TE polarization[62]. The footprint of the TE mode converter is only 1.6 μm × 2.4 μm, and the operating wavelength is 1550 nm. The above examples are all designed on the plane structure. The gradient-based algorithm is also used to design a variety of three-dimensional waveguide-coupled devices on a silicon photonics platform[60]. Figures 6(d) and 6(e) show the 1×3 power router, whose power in the fundamental traverse-electric (TE) mode of the input waveguide is equally split into the fundamental TE mode of the three output waveguides, with at least 95% efficiency. The structure of the power router consists of a single fully etched 220 nm thick Si layer with SiO_2 cladding. Broadband performance is achieved by simultaneous optimization at 6 equally spaced wavelengths from 1400 nm to 1700 nm, and the total footprint is 3.8 μm × 2.5 μm[60].

The researchers then improved the algorithm further, and they introduced the adjoint method to compute the gradient efficiently by using a single time-reversed electromagnetic simulation. Usually, when we optimize the parameters of a system, we know the laws of physics (usually expressed as PDE) that the system follows. This type of problem, called PDE-constrained optimization, has many application scenarios[69,70]. A class of methods to solve this kind of problem is called the adjoint method[71].

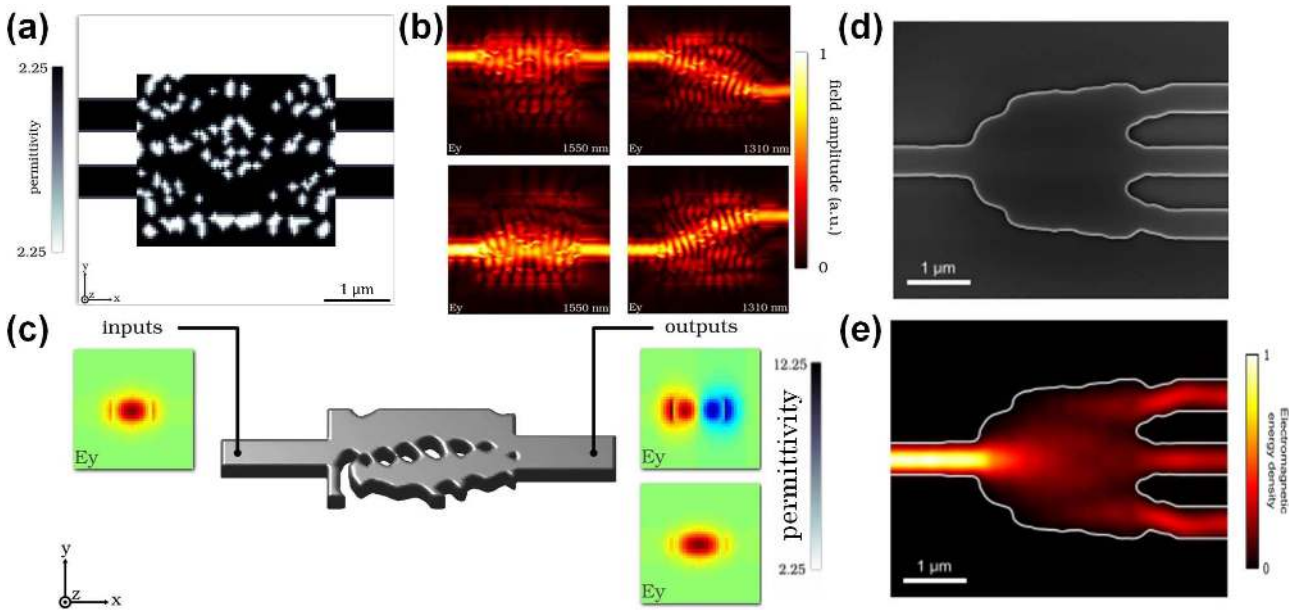

Fig. 6. Nanophotonic devices designed by the gradient-based inverse design. (a) The structure diagram of $2\times2\times2$ hub[62]. (b) The electromagnetic energy density of the hub about the fundamental TE-polarized mode at either 1550 nm or 1310 nm. (c) Performance specification of the TE mode converter[62]. (d) and (e) 1×3 power router with 500 nm wide input and output waveguides[60]: (d) SEM image of the fabricated router; (e) the electromagnetic energy density of the power router at 1550 nm.

The adjoint method can well optimize parameters and solve practical problems, is relatively mature, and can be widely used. The time-dependent adjoint method can also be used to solve optimal control problems. If the state transition of the control problem itself is too complex to form a closed solution, gradient descent is a good choice considering the constraints. On the other hand, in the optimal control, we can consider the randomness of system transfer, so the adjoint method can obviously take the similar randomness into account. Given t for time, we consider the optimization within the time period $0 \le t \le T$.

System: $g[x(0),p]=0, h(x,\dot{x},p,t)=0$. Determine the initial state $x(0)$, and the subsequent evolution of x follows an ODE.

Loss function: $F(x,p)=\min_p\int_0^T f(x,p,t)\mathrm{d}t$. This is an integral over time.

Similarly, we define the Lagrange function:

$$\mathscr{L}=\int_0^T[f(x,p,t)+\lambda^T h(x,\dot{x},p,t)]\mathrm{d}t+\mu^T g[x(0),p]. \tag{8}$$

After taking the derivative, we can get

$$d_p\mathscr{L}=\int_0^T[\partial_x f d_p x+\partial_p f+\lambda^T(\partial_x h d_p x+\partial_{\dot{x}} h d_p\dot{x}+\partial_p h)]\mathrm{d}t + \mu^T[\partial_{x(0)} g d_p x(0)+\partial_p g]. \tag{9}$$

And then we simplify $d_p\dot{x}$ and $d_p\mathscr{L}$.

The end result is

$$d_p\mathscr{L}=\int_0^T[\partial_x f+\lambda^T\partial_x h-\dot{\lambda}^T\partial_{\dot{x}}h-\lambda^\tau d_t(\partial_{\dot{x}}h)]d_p x\mathrm{d}t + \int_0^T(\partial_p f+\lambda^T\partial_p h)\mathrm{d}t+\lambda^T\partial_{\dot{x}}h d_p x|_T + (\mu^T\partial_{x(0)}g-\lambda^T\partial_{\dot{x}}h d_p x)|_0 d_p x(0)+\mu^T\partial_p g. \tag{10}$$

We can take the multipliers x and y so that both of the terms in the brackets are zero.

In a word, during the whole optimization process, only three steps are needed for a single gradient descent.

1. For current p, calculate the $x(0)$ according to the $g[x(0),p]=0$, and solve all the $x(t)$ through the $h(x,\dot{x},p,t)=0$.
2. Write out the ODE that the multiplier satisfies, and solve for $\lambda(t)$ and $\mu(t)$ based on the condition that the terms in the two brackets are zero. (Note here that solving the ODE is in reverse time.)
3. Calculate the gradient $d_pF=\int_0^T(\partial_p f+\lambda^T\partial_p h)\mathrm{d}t+\mu^T\partial_p g$, where the multiplier has been calculated, and then the gradient goes down.

Thus, for each step of gradient descent, we only need to do a few simulations and then solve a few ODEs. The computation is greatly reduced.

The application of the adjoint method in the optimization problem with constraints has two main aspects[71]. First, a system with unknown parameters can output data by collecting input, and then estimate the parameters of the system. Loss

reflects the difference between the system output and the actual output measured. Second, in order to design a system with a certain function, loss reflects whether the system and the target function fit, and then we optimize the parameters of the system to complete the inverse design.

In the improved gradient-based inverse design algorithm, the adjoint algorithm is used to calculate the gradient and optimize the parameters and the structure. With boundary parameterization and structure optimization, a broadband optimization to produce a robust device can be performed[63–65]. Other types of routers like the TE/TM mode and wavelength routers are designed by the improved gradient-based inverse design algorithm, which shows that different photonic signals can be split[60,65]. The final result of the TE/TM router is shown in Figs. 7(a) and 7(b)[62]. The footprint of the device is 2.8 μm × 2.8 μm, and the conversion efficiencies into the upper and lower output arms are 87.6% and 88.8%, respectively. Figure 7(c) is a measured transmission of the three-channel wavelength router designed by the algorithm[65]. Applying gradient descent directly can lead to structures with weak dielectric constant modulation between the two, which will lead to poor performance in the discrete phase. In the process of designing the wavelength router, Su *et al.* mitigated this issue through a specific variant of penalty functions, which they called biasing. The peak average measured transmission of the three-channel wavelength router is −2.82 dB at 1471 nm, −2.55 dB at 1512 nm, and −2.29 dB at 1551 nm, and the peak insertion loss is −2.29 dB with −10.7 dB crosstalk. The simulated electromagnetic density at the operating wavelengths is shown in Fig. 7(d). As can be seen from the figure, a continuous topography is generated in the design process of the algorithm. In addition, the device was designed in approximately 60 h on a single computer with an Intel Core i7 −5820K processor, 64 GB of RAM, and three Nvidia Titan Z graphics cards. The design process needs to be further improved in order to increase efficiency and adapt to the development of high density and high-speed integration.

With the development of artificial intelligence and information technology, more and more types of nanophotonic devices have been recently designed by the inverse design algorithm. It tends to be used in the design of multifunctional devices and cascaded devices, such as laser-driven particle accelerators, resonators, interfacing grating couplers of conceptual photonic circuits, and switches[61,66–68]. A scanning electron microscope (SEM) image of cascaded nonlinear resonances is shown in Fig. 8(a), which is implemented on a silicon-on-insulator platform[67]. The enlarged images show reflectors designed by the inverse design algorithm on the silicon waveguide in the resonator–waveguide coupling region. The R = 94% reflector used to implement a Lorentzian resonator is shown in Fig. 8(b). The top is the optimization trajectory to obtain the desired non-resonant high reflection, the bottom is the low-power transmission of a single device with non-resonant reflection R = 94%, and the red line is a fit with a Lorentzian line shape. A conceptual circuit comprised of three components with completely different

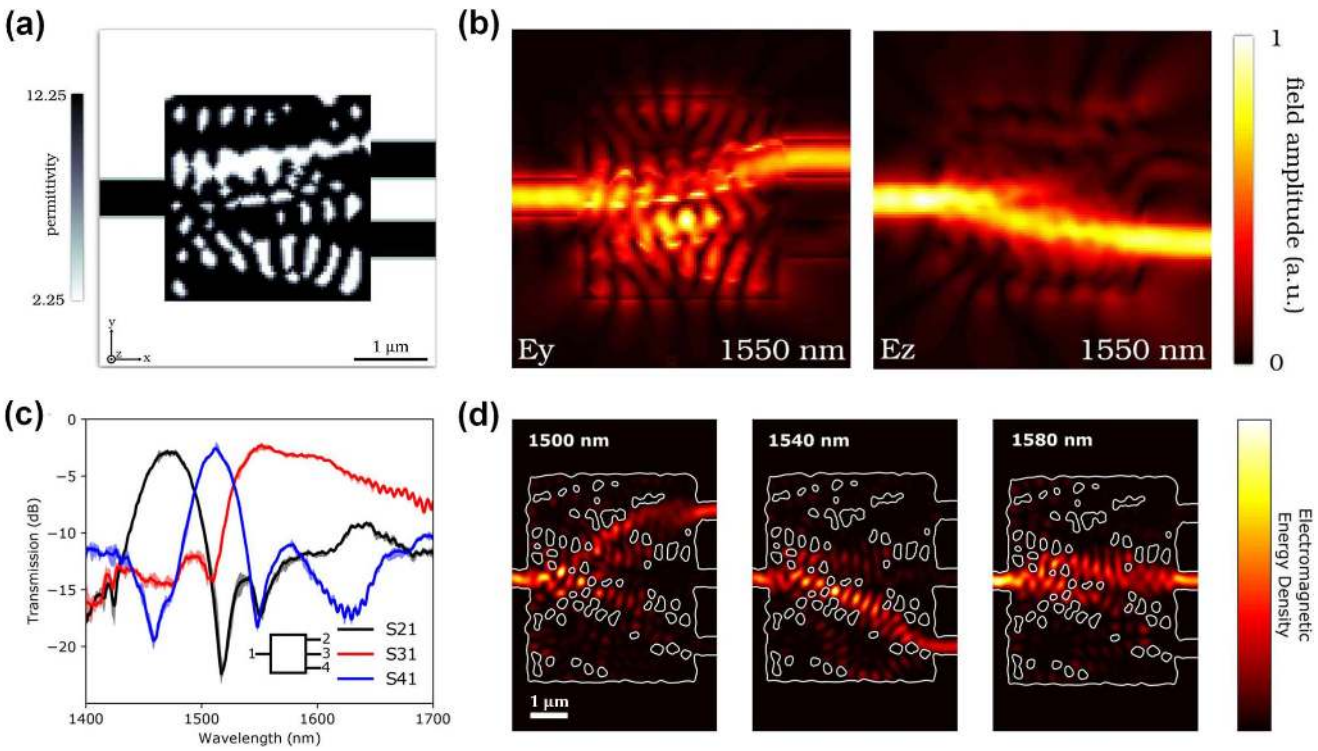

Fig. 7. Nanophotonic devices designed by the gradient-based inverse design. (a) The structure diagram of TE/TM router[62]. (b) The Electromagnetic energy density of the TE/TM router at 1550 nm. (c) Measured transmission of the three-channel router[65]. (d) Simulated electromagnetic energy density of the three-channel router at the three operating wavelengths.

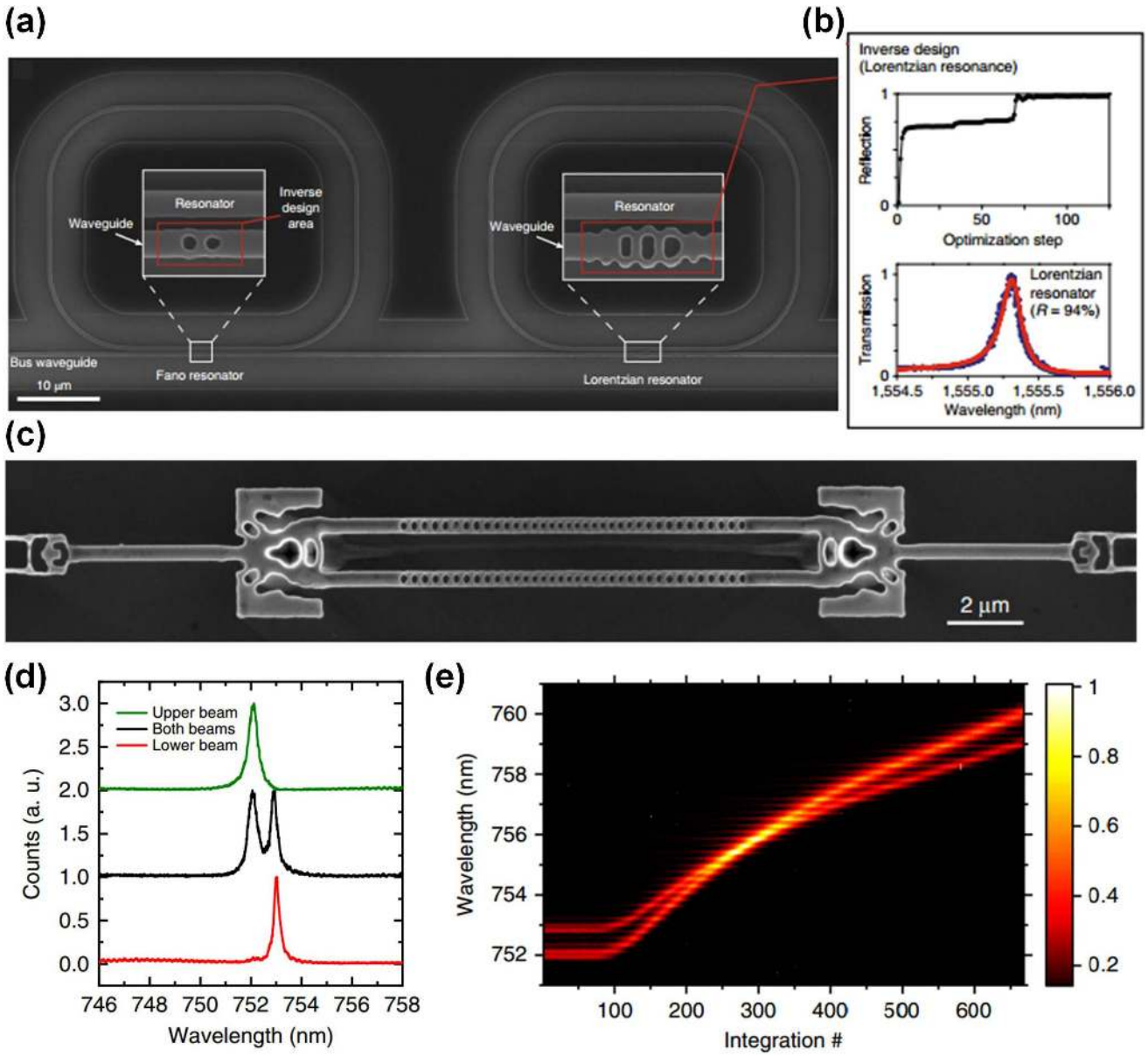

Fig. 8. Nanophotonic devices designed by the gradient-based inverse design. (a) SEM image of cascaded Fano–Lorentzian resonators implemented on a silicon-on-insulator platform[67]. (b) The $R = 94\%$ reflector used to implement a Lorentzian resonator. Top: optimization trajectory to obtain the desired non-resonant high reflection; bottom: low-power transmission of a single device with non-resonant reflection $R = 94\%$ and red line is a fit with a Lorentzian line shape. (c) A conceptual photonic circuit that consists of a grating coupler followed by a waveguide-splitter and two resonators, and the outputs of them are then recombined in a waveguide-splitter and coupled off-chip through a grating coupler[61]. (d) Spectra of the nano-beams from the device shown in (a). The green, black, and red data correspond to the upper, both, and the lower nano-beam, respectively. (e) Demonstration that cavities with a fabrication-induced frequency offset can be tuned in resonance via gas tuning; the color bar corresponds to normalized counts.

geometries – vertical couplers, waveguide-splitters, and nano-beam PhC cavities – is also designed by the gradient-based inverse design algorithm, which is shown in Fig. 8(c)[61]. The device is designed to interfere with the transmission of two nano-beam PhC cavities at an inverse-designed waveguide splitter with a 50:50 splitting ratio and simulated efficiencies of 95%. The cavities are addressed separately or simultaneously by top-down excitation with a supercontinuum source focused on the cavities directly, as presented in Fig. 8(d). The two cavities are tuned into and out of resonance via gas condensation, as shown in Fig. 8(e). Comparing the amplitudes of the cavity on and off resonance, which draws the conclusion of constructive interference, indicates that the cavities are approximately in phase and have the same polarization. The multifunctional cascaded devices designed based on the gradient-based inverse design algorithm presented above have achieved good performance, but

the size of the devices is large, and the method may still be further improved to meet the requirements of high density integration.

3.3 Discussions

The gradient-based inverse design can automatically design photonic devices, which is an automated photonics design, and only requires the user to input high-level parameters. The algorithm can afford large parameter space, and design devices that exploit the full space parameters of fabricable devices. It tends to require fewer simulations than genetic or particle swarm optimization as they do not rely on parameter sweeps or random perturbations to find their minima. The gradient-based inverse design algorithm can be used to design photonic devices with any passive and linear photonic element. However, the design achieved by the inverse design algorithm typically exhibits a continuous topography, and some very small components in structures may be formed during the inverse designing process, which brings challenges for sample fabrication. Moreover, the gradient-based inverse design method usually produces a local optimal solution, and it cannot realize the true global optimization.

4. Nanophotonic Devices Based on Swarm Intelligence Algorithms

Swarm intelligence refers to 'the non-intelligent subject shows the characteristics of intelligent behavior through cooperation', which is a kind of computing technology based on the laws of biological group behavior. In recent years, there have been various algorithms in the research field of swarm intelligence theory, such as the genetic algorithm (GA), particle swarm optimization (PSO), and the ant colony algorithm (ACA). It is proved that swarm intelligence algorithms are effective methods through the research of the theory and application method. It can effectively solve most optimization problems.

4.1 Genetic algorithm

GA is an adaptive optimization global search algorithm that simulates the genetic and evolutionary process of organisms in natural environments[72]. In essence, it is a parallel, efficient, and global search method that can automatically acquire and accumulate knowledge when searching space automatically and controlling the search process adaptively to obtain the optimal solution. GA has been successfully applied to various academic and industrial applications, such as communication and photonics[73–75]. GA is a stochastic optimization technique based on natural selection and evolutionary biology. It is well suited for complex problems where many of the system parameters must be optimized simultaneously and for other practical problems where there may not be a unique and well-defined optimal value.

According to individual fitness and certain rules, some individuals with excellent traits are selected from the nth generation group and passed on to the next generation $(n+1)$ population. In this selection process, the greater the fitness of an individual, the greater the chance of being selected to the next generation. For the fitness of individual i of f_i and the population size of N, the probability formula of i being selected is

$$P_i = \frac{f_i}{\sum_{i=1}^{N} f_i}. \tag{11}$$

Individuals selected from population are randomly matched and, for each individual, a certain probability (crossover probability 0.25–1.0) is used to swap parts of their chromosomes (partial position of the encoding bit string). The search ability of GA is extended better. Figure 9 is the flow chart of the GA.

To obtain the desired optical properties, Huntington *et al.* designed a lattice evolution algorithm that allows lattice optical materials to exhibit simple properties or focus light on discrete points[6]. It is shown in Fig. 10(a). Using multiple scattering and GA to determine the photonic crystal structure to be optimized is reliable and can complete specified optical tasks. GA is used to operate on a set of candidate structures to find new candidate structures with stronger performance during the whole iteration. David *et al.* used GA to design an optimized antireflection coating with broadband and omnidirectional characteristics[76]. The simulated reflection characteristics of the antireflection coating are shown in Fig. 10(b). The simulation results of the optimized three-layer coating show that the performance of the coating is significantly improved compared with that of the traditional coating.

Yu *et al.* used GA to optimize the design of the prevalent thin-film-on-insulator platform for reflectors[77]. The structure is

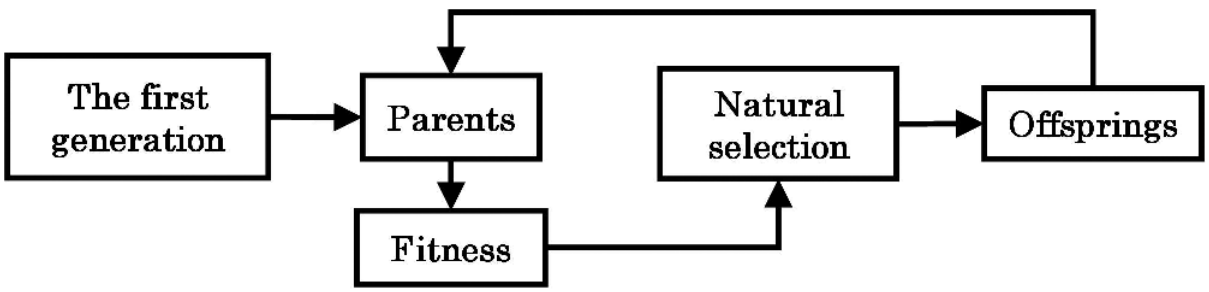

Fig. 9. The flow chart of GA[77].

composed of randomly distributed pixels, and the manufacturing process is compatible with CMOS, requiring only one-step lithography and etching. The structure and electric field distribution before and after the optimized design are shown in Fig. 10(c). Sanchis *et al.* designed a coupler device capable of introducing light generation from optical fibers into a photonic crystal-based waveguide[78]. As is shown in Fig. 10(d), the optimized integrated device (waveguide coupler) and its electric field modulus diagram are illustrated.

Our group has constructed an intelligent algorithm by combining GA and FEM to design a wavelength[7] and polarization router[8] that can realize the beam splitting of wavelength and the recognition of TE and TM polarization states. The footprint is only 1.4 μm × 1.8 μm for the wavelength router and 0.97 μm × 1.24 μm for the polarization router in the experiment around the optical communication range. These are the smallest ever demonstrated experimentally. A broad operation band, transmission up to 98%, and various output ports can be simultaneously achieved, and it is convenient to realize various routers with different materials (both dielectric and metal), different configurations, different channels, and different structure cell quantities or sizes. In addition, the average position error tolerance for each cell structure is about ±20 nm for all the wavelength or polarization routers designed by the intelligent algorithm, which satisfies the current nanofabrication technology. Figures 11(a) and 11(b) are the structures of the wavelength router and the transmission spectrum of its upper, right, and lower ports, respectively; the structure of the polarization router is shown in Fig. 11(c) and the transmission spectra of its right and lower ports are shown in Figs. 11(d) and 11(e), respectively.

Chen *et al.* proposed a method[79] combining field emission (FE) modeling and GA to optimize the focused quality of integrated gated carbon nanotubes. The design effect is shown in Figs. 12(a) and 12(b). It is challenging to overlap the radiation power spectrum between the magnetic dipole moment and the electric dipole moment of nanoparticles in a wideband way. Liu *et al.* combined GA, Maxwell's equation, and electromagnetic multipole expansion[80] to design a nanoparticle that supported resonant broadband forward light scattering. The result is shown in Fig. 12(c).

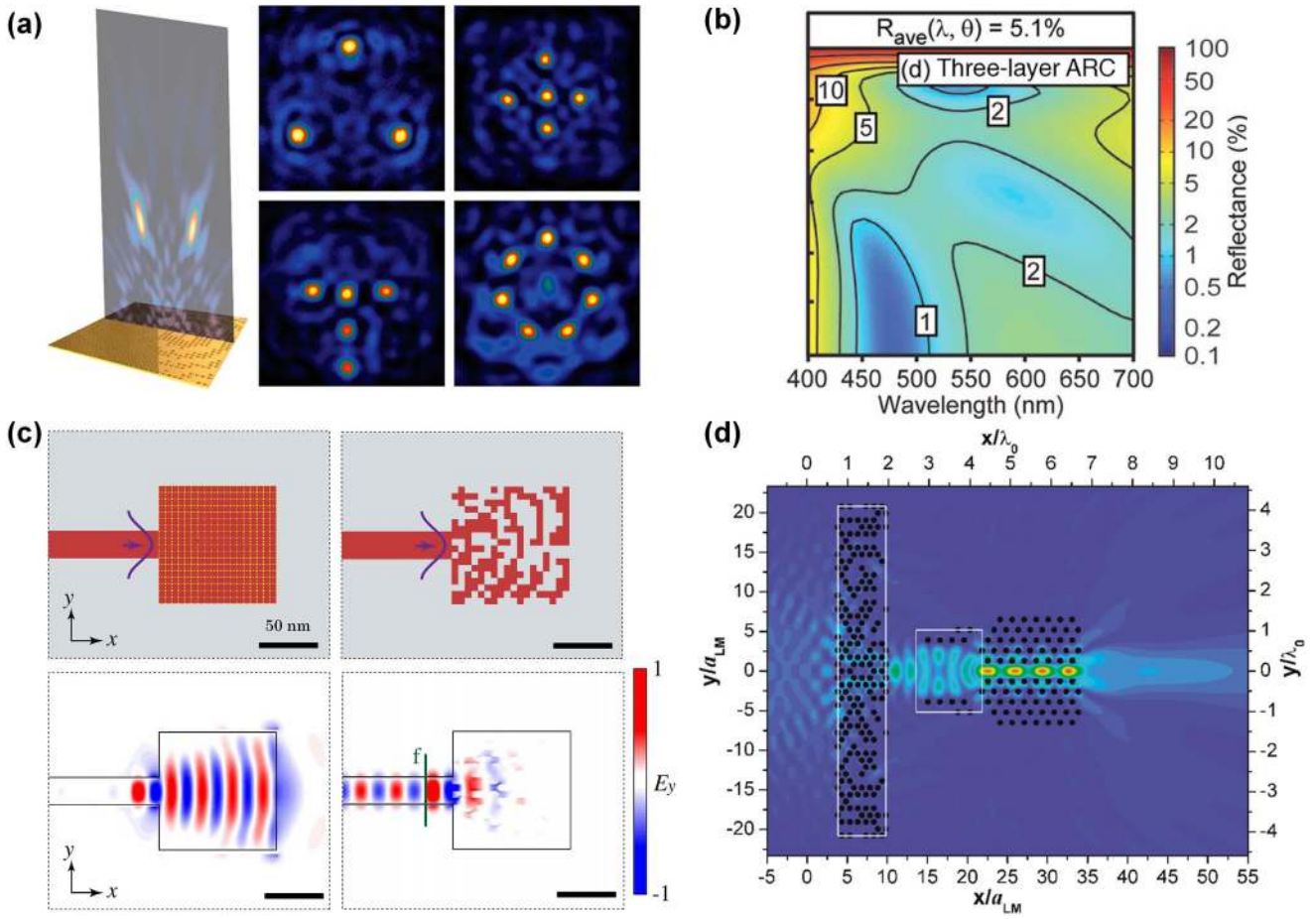

Fig. 10. Nanophotonic devices designed by GA. (a) Lattice optical materials capable of focusing light into several different focal points in the far field. The left is a schematic diagram of the experimental device. The right shows light focused on several different points through a lattice of lattice optical materials[6]. (b) Simulated reflection characteristics of antireflection coatings[76]. (c) The left is the initial silicon plate and the corresponding electric field distribution before optimization, and the right is the structure and electric field distribution of the reflector after optimization[77]. (d) The structure obtained after GA and simulated transmittance spectrum[78].

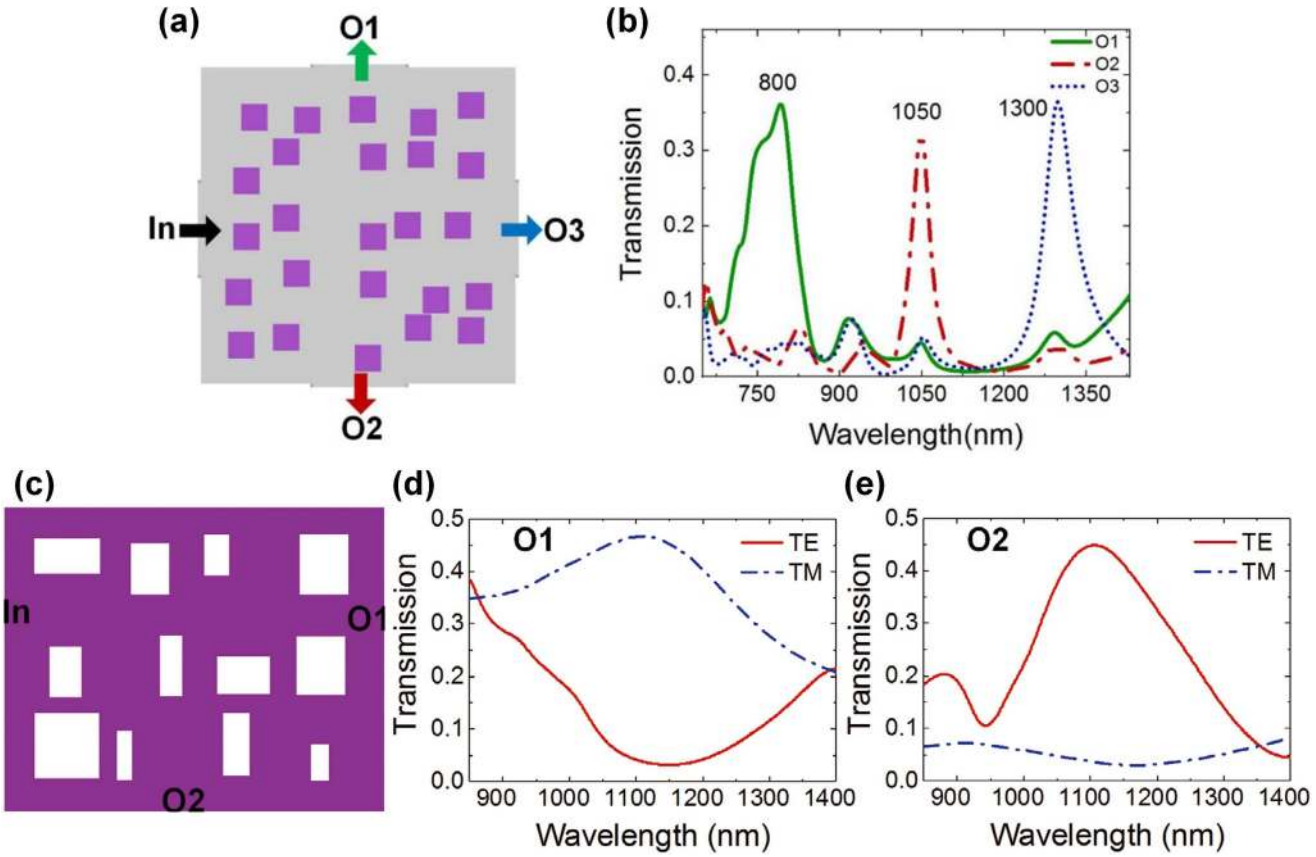

Fig. 11. Nanophotonic devices designed by GA. (a) The structure diagram of wavelength router and (b) the simulated transmittance[7]. (c) The optimized structure of the polarization router. (d) and (e) are the simulated transmission spectra of the polarization router's O1 and O2 ports[8].

GA is well suited for complex problems, such as having to optimize many system parameters at the same time, and some other application problems may not have well-defined and unique optimal values. GA can not only solve the single objective optimization problem, but can also play a more important role in the multi-objective optimization problem. The common selection method of multi-objective GA is to define individual fitness through different methods. Although the local search ability of GA is poor, it is often used in combination with other algorithms to improve the performance of the algorithm by taking advantage of its easy parallel implementation. In many works[6–8,76–82], researchers apply GA to the field of photonics in order to achieve the optimal design of devices.

Just like GA, based on biological evolution, the cultural algorithm (CA) uses cultural or social evolution to simulate human society and solve optimization problems by using domain

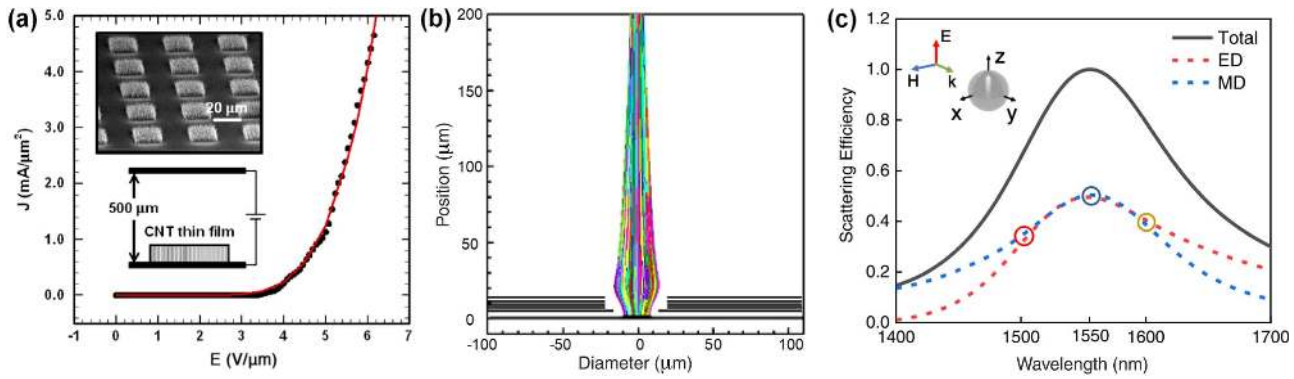

Fig. 12. Nanophotonic devices designed by GA. (a) Measured data and calculated results (red solid line), the illustration is a schematic of carbon nanotube films and diode FE measurements. (b) Optimized electron beam trajectories for type of FE device[79]. (c) The total scattering efficiency of normalization (black line), and the contribution of induced electric dipole (ED) and magnetic dipole (MD) moments of core-shell nanoparticles[80].

knowledge to reduce the search space[83]. CA is an evolutionary algorithm of social or cultural evolution proposed by Reynolds in 1994. It is a flexible technology that is easy to implement. Khorrami *et al.* first introduced a guided mode resonance (GMR) grating filter designed using CA and explained that CA can be used to design electromagnetic time problems, such as devices with more than three target parameters.

4.2 Particle swarm optimization

The PSO algorithm is derived from the simulation study of migration and aggregation behavior in the foraging process of birds. The basic idea is to find the optimal solution through the cooperation and information sharing among individuals in the group. It contains the characteristics of evolutionary calculation and swarm intelligence. It is essentially a kind of random search algorithm[84] that can converge to an optimal solution with a large probability.

In PSO, the velocity and position of each particle in the solution space are initialized, including the entire possible solution set[85]. The fitness function acts as a guide to get these particles to the target value of the fitness function.

The whole process can be represented by the equations

$$V_{id} = \omega V_{id} + C_1 \text{random}(0,1)(P_{id} - X_{id}) + C_2 \text{random}(0,1)(P_{gd} - X_{id}), \quad (12)$$

$$X_{id} = X_{id} + V_{id}, \quad (13)$$

where ω is the nonnegative inertia weight factor. When it is larger, the global optimization ability is stronger, while the local optimization ability is weak; when it is smaller, the global optimization ability is weak and the local optimization ability is strong. C_1 and C_2 are acceleration constants. C_1 is the individual learning factor of each particle, and C_2 is the social learning factor of each particle. Usually, $C_1 = C_2 = 2$, but it does not have to be 2. In general, $C_1 = C_2 \in [0, 4]$. Random (0, 1) represents the random number in the interval [0, 1], P_{id} represents the dth dimension of the individual extremum of the ith variable, and P_{gd} represents the dth dimension of the global optimal solution.

Using the PSO algorithm to optimize the parameters, Djavid *et al.* proposed an evolutionary design approach of the photonic crystal notch filter[86]. The designed filter and the recording of the electric field intensity are shown in Figs. 13(a) and 13(b), respectively. Kumar *et al.* used the simulated PSO algorithm to optimize the structure of photonic crystals and studied a waveguide terminal realizing directional emission of photonic crystals[5]. Figure 13(c) shows the optimized PSO structure and their electric field distributions. Forestiere *et al.* used the PSO algorithm to optimize the array of plasma nanoparticles[87], resulting in a non-periodic structure and an enhanced broadband field across the entire visible spectrum. They also found that the broadband field enhancement in nanoplasmas can be achieved by designing aperiodic arrays, and aperiodic arrays provide the necessary interactions between distant diffraction interactions at multiple scales and near-field quasi-static couplers within small nanoparticle clusters. The optimized array

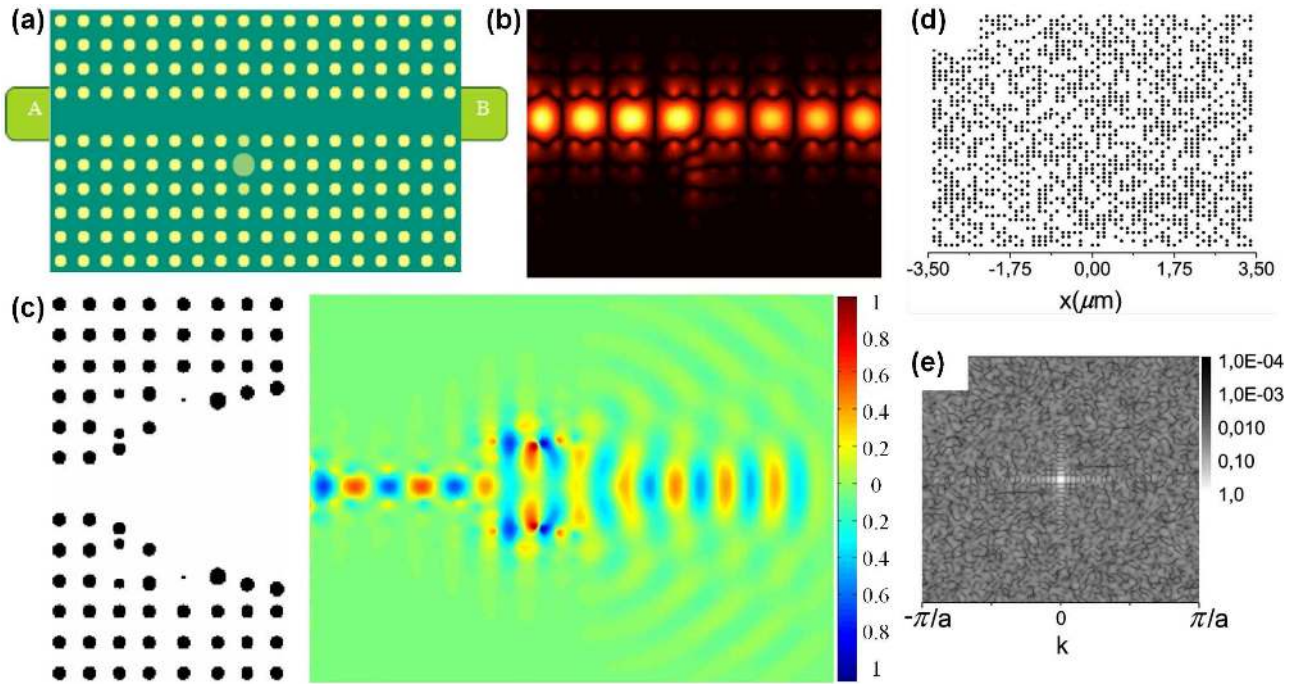

Fig. 13. Nanophotonic devices designed by PSO. (a) A notch filter based on microcavity and (b) single frame extract video recording of the electric field intensity of the notch filter at the wavelength of 1500 nm[86]. (c) The structure of the tapered PSO and the distribution inside the electric field[5]. (d) The optimized geometry of the silver nanoparticles array and (e) the magnitude of its Fourier transform[87].

of silver nanoparticles and its Fourier transform magnitude are shown in Figs. 13(d) and 13(e), respectively.

In order to design a binary mask, Rogers *et al.* used a binary PSO algorithm to optimize the mask[88]. The whole swarm consists of a specified number of particles that move continuously in the N-dimensional search space to find the optimal solution. Compared with the number of fixed annuli, the number here will be different during optimization. They defined the optimized merit function as the size of the center point. In the application of this algorithm, they used a group of 60 particles and performed 10,000 iterations with $N = 100$, resulting in a super-oscillatory lens (SOL) design [Fig. 14(a)]. The clustering feature is displayed in the Fig. 14(b). It is well known that multi-objective optimization strategies have great advantages over single-objective optimization methods in finding a well-distributed set of solutions, which ensures post-processing and decision-making extremely convenient. Most of the designed grating couplers used in silicon photonics match a nearly 10 μm mode-field diameter (MFD) of single-mode telecommunications fibers[89]. Passoni *et al.* analyzed grating couplers on silicon on insulator (SOI) platforms applied to MFD (4–100 μm) and gained a physical understanding of the efficiency of the corresponding coupled spectra and the spectral trend. Mak *et al.* used a binary particle swarm optimization algorithm to optimize the binary configuration of cells, and the region with a size of 4.8 μm × 4.8 μm was optimized[90]. By processing binary variables and thresholding, the continuous configuration space is transformed into a discrete configuration space. They studied the application of a small power divider based on two-dimensional (2D) grid binary particle swarm optimization in casting a standard silicon photon platform. The design results are shown in Figs. 14(c) and 14(d).

Ha *et al.* proposed a design method for the ultra-compact small footprint lens. Combining the PSO algorithm with spatial technology[91], the two-step and four-step zoom lenses are integrated into the SOI chip with the footprint of 35 μm × 35 μm to achieve the zoom of 2.5 × and 3.4 ×, which provides a new idea for the designing of small on-chip display devices. The design results are shown in Figs. 14(e) and 14(f). With the use of PSO and GA, Wohlfeil *et al.* proposed a fast and flexible optimization method for fiber grating couplers[92]. A kind of one-dimensional fiber grating coupler is derived from a waveguide with random etching. The resulting theoretical coupling efficiency of the grating is up to 1.1 dB and provides clear design rules for the layout of the efficient fiber grating couplers.

PSO has a fairly fast speed of approaching the optimal solution, which can effectively optimize the parameters of the system. The advantage of PSO is that it can be applied to continuous function optimization problems. The main drawback of this method is that it is easy to produce premature convergence, especially in dealing with complex multiple optimal value search problems, and its local optimization ability is poor. PSO falls into local minimum, which is mainly attributed to the loss of diversity of population in search space. To further improve it, we can either combine it with other algorithms or

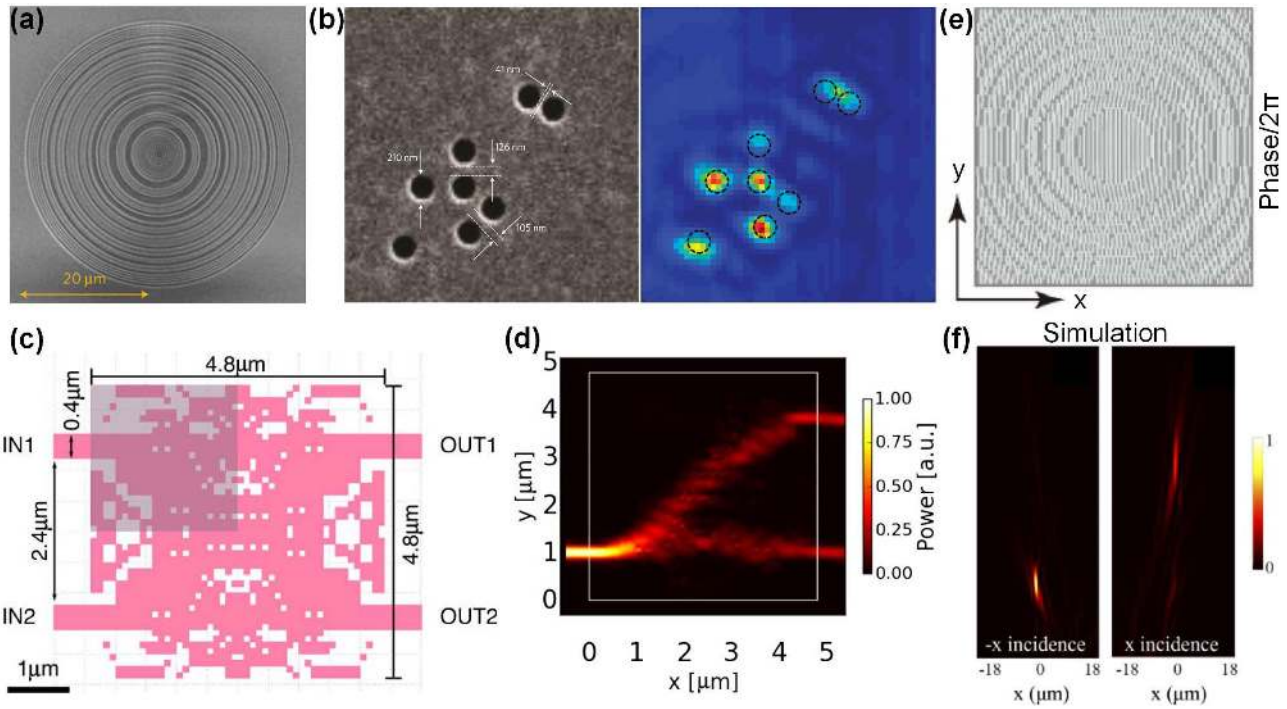

Fig. 14. Nanophotonic devices designed by PSO. (a) The SEM image of SOL and (b) the SEM image of the cluster of nanoholes on the metal membrane. The SOL image shows all the main features of the cluster[88]. (c) Optimized power splitter device and (d) normalized strength[90]. (e) The white rectangle represents the spatial distribution of the nanometer aperture of the two-channel multiplexing lens. (f) The simulated intensity profiles of the radiated beam of the two-channel multiplexing metalens in the *xz* plane[91].

add mutation operation. PSO has been used to optimize nanostructures and design nanophotonic devices. It can be used to optimize multidimensional problems. Although PSO has a high requirement for parameter setting, its process is easy to understand and its convergence speed is fast.

4.3 Ant colony algorithm

ACA is derived by simulating the process of ants finding their way in nature, and it is an intelligent algorithm to search the shortest path. ACA has the advantages of strong robustness[93] and easy integration with other algorithms.

The basic ACA is expressed as follows: at the initial moment, m ants are randomly placed, and the initial amount of pheromone on each path is equal. At the moment t, the probability of the kth ant moving from node i to node j is

$$p_{ij} = \frac{\tau_{ij}\eta_{ij}}{\sum \tau_{ij}^{\alpha}\eta_{ij}^{\beta}}, \tag{14}$$

where τ_{ij} is the intensity of biopheromone on the line of ij at time t; η_{ij} is the heuristic factor indicating the expected degree of ant moving from node i to node j, usually taking the reciprocal of the distance between i and j. α and β represent the degree of relative importance of the pheromone and expectancy heuristic factors, respectively. After each ant traverses once, the pheromone update on each path is

$$\tau_{ij} = (1-\rho)\tau_{ij} + \Delta\tau_{ij}, \tag{15}$$

where ρ represents the loss level of the total amount of pheromone on the path, and $1-\rho$ represents the residual factor of the pheromone. $\Delta\tau_{ij}$ represents the increment of the pheromone on path ij after the completion of this iteration, which can be represented as

$$\Delta\tau_{ij} = \begin{cases} \frac{1}{L_k}, & \text{the } k\text{th ant goes by } ij, \\ 0, & \text{others}, \end{cases} \tag{16}$$

where L_k represents the length of the path traveled by the kth ant in this traversal. Figure 15 is the flow chart of the optimization process of ACA.

Using ACA, Saouane *et al.* obtained the setting of the optimal inclination angle for the photovoltaic collector through simulation and improved the efficiency of the collector[94]. Guo *et al.* proposed a method to optimize the anti-reflective coating of silicon solar cells with ACA in the range of 400 nm to 1000 nm wavelength[95]. Figures 16(a) and 16(b) show the schematic diagram of the antireflection coating system designed on the silicon substrate by the ACA-based calculation method and the simulation results of the reflectance performance of the optimized antireflection coating with a wavelength ranging from 400 nm to 1100 nm and an angle from 0° to 90°.

However, if the parameters are not set properly, the solution speed will be very slow and the quality of the solution will be particularly poor. In the early stage, it takes a long search time and a large amount of calculation, which leads to a long time for the overall solution. In the design of nanophotonic devices, ACA is suitable for combinatorial optimization and continuous function optimization. The whole process of the algorithm is intuitive, but it takes a long time to solve.

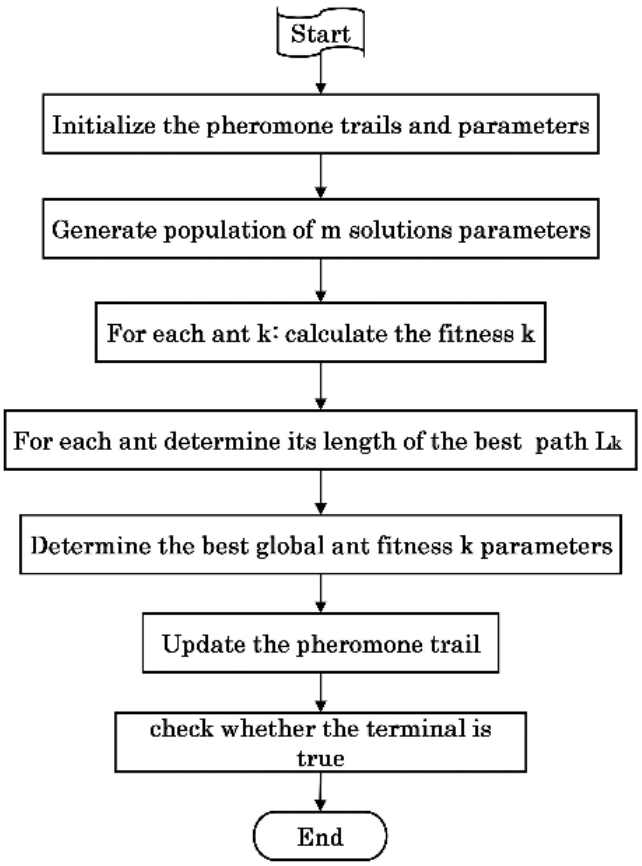

Fig. 15. The flow chart of ACA optimization process[94].

For swarm intelligence algorithms, the overhead of each individual in the system is very small, and the functions that each individual can achieve are very simple, which leads to the short execution time of each individual. Therefore, the implementation is relatively simple and convenient for researchers to implement programming and parallel processing on the computer. However, parameter sensitivity is a problem that needs to be paid attention to, because improper selection will increase the time cost and complexity of subsequent calculations.

5. Nanophotonic Devices Based on Individual Inspired Algorithms

5.1 Simulated annealing algorithm

The SAA was first introduced by Kirkpatrick *et al.* in 1983 to mainly apply to discrete optimization problems. Originating from the physical process in which a crystalline solid slowly cools down from a relatively high temperature and gradually forms a regular crystal configuration during the annealing

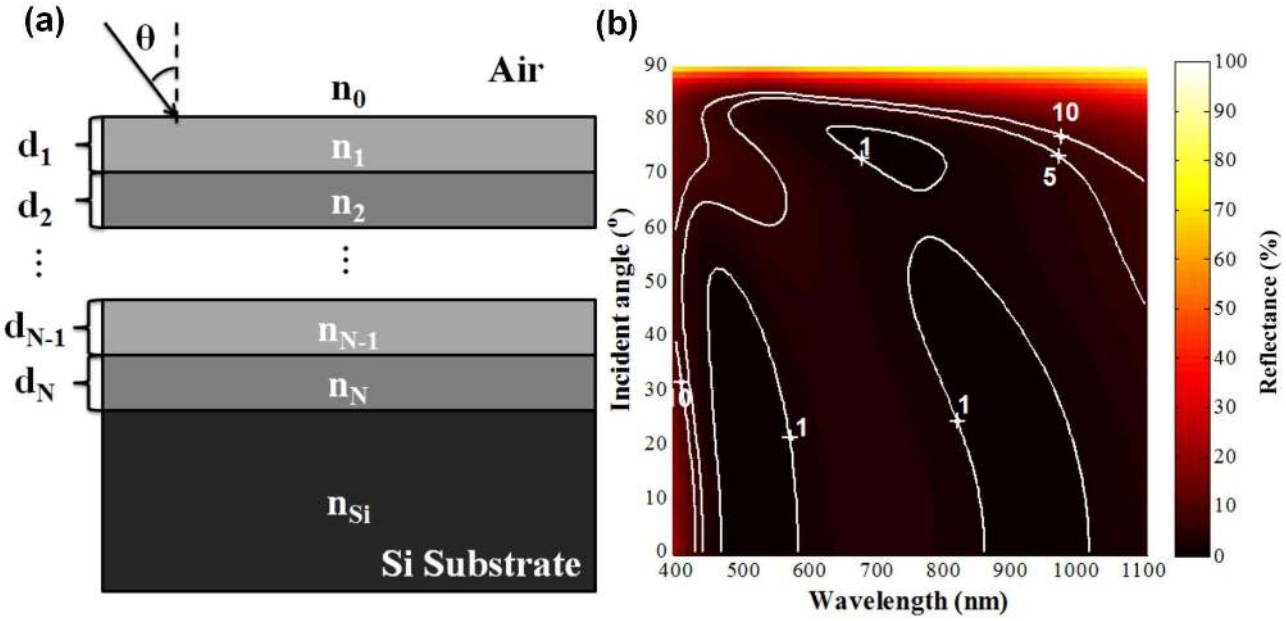

Fig. 16. Nanophotonic devices designed by ACA. (a) The ACA-based method was used to calculate the reflection coefficient of the antireflection coating system on silicon substrate and (b) the simulation results show that the reflectivity of the antireflection coating system is changed with wavelength and incident angle by ACA[95].

process, the algorithm provides a strategy to escape local optima, hoping to achieve the global optimum[96].

Figure 17 illustrates the flow chart of simulated annealing. The algorithm begins with a customized initial temperature, which lowers at a given speed at the end of each iteration. At each temperature, a new-found solution is compared with the current one based on a given objective function. A better solution will be accepted consistently, while a worse solution with a higher objective function value can also be employed according to the Metropolis criteria, that is, the algorithm accepts a worse solution with the probability

$$P(S_i,S_j) = \begin{cases} 1, & f(S_j) < f(S_i), \\ \exp\{[f(S_i) - f(S_j)]/T\}, & f(S_j) \geq f(S_i), \end{cases} \tag{17}$$

where $f(S_i)$ and $f(S_j)$ stand for values of the current solution and new-found solution from the objective function. T, represents the parameter temperature. With the descendance of temperature, the algorithm lowers the tolerance of the distance between two solutions and the frequency of accepting a worse solution, simultaneously. The expression of the objective function deserves thinking twice before application as it determines whether the final result fits the goal. Owing to the simple structure, SAA gets weak when facing a large number of parameters to be optimized, as it randomly selects a new solution from the solution space. The searching efficiency and possibility of finding an optimum decrease simultaneously with more unknown parameters.

Different from swarm intelligence algorithms, SAA has a simple structure that allows application of SAA under various circumstances. As another advantage, SAA requires no knowledge of the specific problem and thus guarantees the robustness of a random initial guess. The convergence of SAA was promised with strict mathematical demonstration[97], while it does not promise a global optimum, like other heuristic algorithms. As a defect, the performance of SAA is sensitive to the customized parameters, especially the initial temperature.

The analogous physical annealing process inspires us to set a high initial temperature in avoidance of an insufficient cooling process, that is, loss of ability to escape local minima. But it introduces a waste of computing budget as the algorithm loses the ability to judge the quality of the new-found solution and accepts all of them until the excessively high initial temperature cools to a critical temperature. A critical temperature represents a balance point at which objective function values are preferred, but the temperature is warm enough to tunnel through such solutions. We have no idea about the appropriate value for the initial temperature when the algorithm needs no knowledge of the problem. In that case, experiments are expected to identify the initial temperature and such a method was proposed by Basu *et al.*[98].

Due to the mechanism of SAA, a large computing budget is always expected to search for the optimum. The situation deteriorates even more with an excessive initial temperature. Considering the efficiency and computing time required in the field of nanophotonic devices, it is not appropriate to employ such a time-consuming algorithm alone, which might be the reason for SAA's not being widely used to design nanophotonic devices. But strategies like combining SAA with other algorithms to develop its efficiency could still be a good option when it comes to devices with discrete parameters to be optimized.

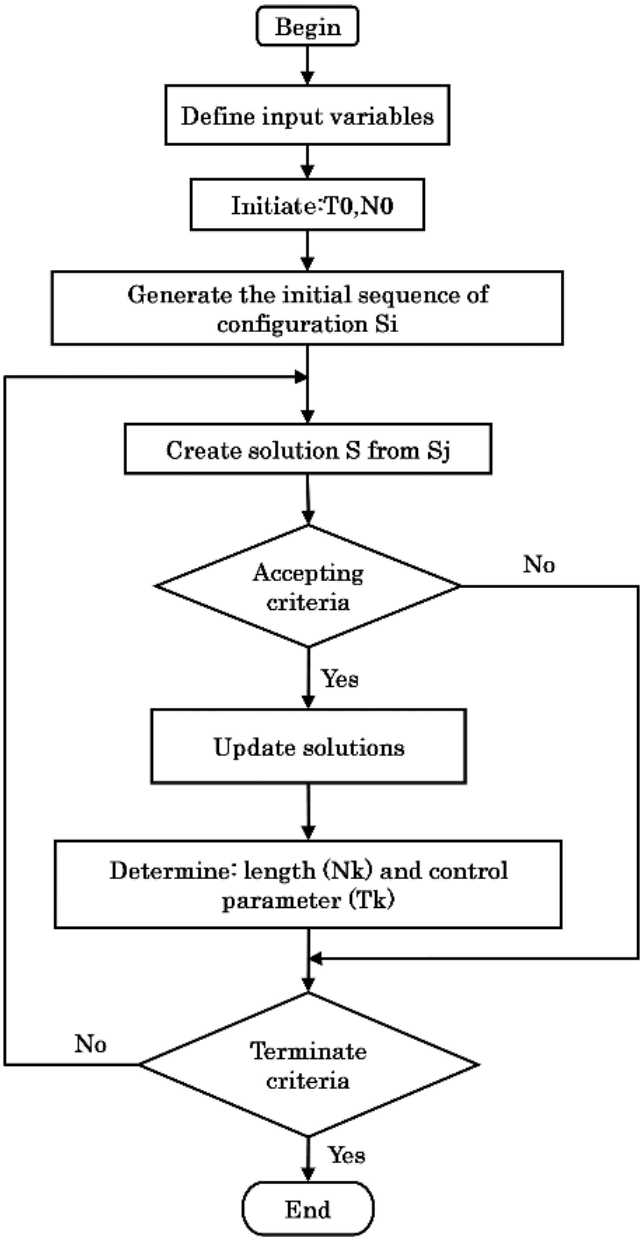

Fig. 17. The flow chart of SAA[98].

SAA was proposed in the field of optical inverse design by Hara[99] as early as 1996. It was used in the design of a pnpn differential optical switch and generated a good result taking fabrication errors of structural parameters into account. For other devices like a wedge filter, binary diffractive element, planar attenuator[100], and multilayer microwave absorber[101], SAA was also once employed as the optimization tool. The possibility of the application of SAA to any arbitrary optoelectronic devise was thus assured. In more recent times, Xie *et al.* designed a broadband twisted light emitter based on a combination of SAA and GA[102]. As shown in Fig. 18(b), the core component of the device is a circular area divided into 288 pixels, which are filled with either air or silicon block. The element design was thus transformed to a discrete optimization. Distribution of these blocks helps control the phase modulations aroused by propagation and resonances, which realizes the functionality of the orbital angular momentum (OAM) emitter. Figure 18(a) gives a schematic of the device. It receives optical frequency combs as input from the left (right) waveguide and generates output in the form of an OAM with a state number -1 $(+1)$. Compared to the inherent narrow bandwidth of whispering-gallery-mode-enabled OAM emitters, the bandwidth of this device is expanded to 200 nm between 1450 and 1650 nm, for the use of developed SAA. Emission efficiency and mode purity are also promised by the algorithm and reach the values of 35% and 97%, respectively. Simulation results and experiment data of emission efficiency for the OAM emitter are depicted in Fig. 18(c). Through a similar method, a broadband on-chip photonic spin Hall element was designed by Xie *et al.*[103]. Figure 19(a) gives a schematic of the element working as a photonic spin detector when it couples light into different waveguides according to the polarization states. In Fig. 19(b), the output power measured at the left/right port (port 1/port 2) varies with the polarization of incident light. The output power of different ports under different polarization degrees of incident light measured is shown in Fig. 19(c). As an emitter, the element generates circularly polarized light when light is coupled into two waveguides. SAA improves the efficiency of the device as a detector or emitter of spin light to 22% and 35%, and expands the bandwidth to 200 nm for the detector.

5.2 Hill-climbing algorithm

The hill-climbing algorithm is a local search algorithm. Its advantage is that it does not need a traversal process to reach the highest point of the solution space; instead, it selects nodes with a higher value through heuristics where the efficiency is highly improved[104]. The process does not require memories of the previous steps, which makes it save storage space when searching for the optimal solution in a large parameter space.

The optimization of nanophotonic devices is often a complex problem. It is necessary to find the optimal solution in the full parameter space, and the form of the objective function is often complicated. Therefore, hill climbing is not an excellent method for designing nanophotonic devices. However, when the initial structure has been proved to possess an effective function, using hill climbing can further improve the performance of the device. In the design of a photonic crystal-based nanocavity[105], the optimization is based on an initial structure designed using the deterministic method. As is shown in Fig. 20(a), the initial structure is a one-dimensional photonic crystal, and the filling fractions are quadratically tapered. This structure has been proved to be effective in their previous work[106]. However, the hill-climbing algorithm has several unavoidable disadvantages. As is shown in Fig. 21(b), when a node is higher than any of its neighbors, but it is not the highest point of the whole function, this node will be regarded as the optimal result, which makes us unable to get satisfactory results in the calculation. In addition, this algorithm is easily limited by certain functional

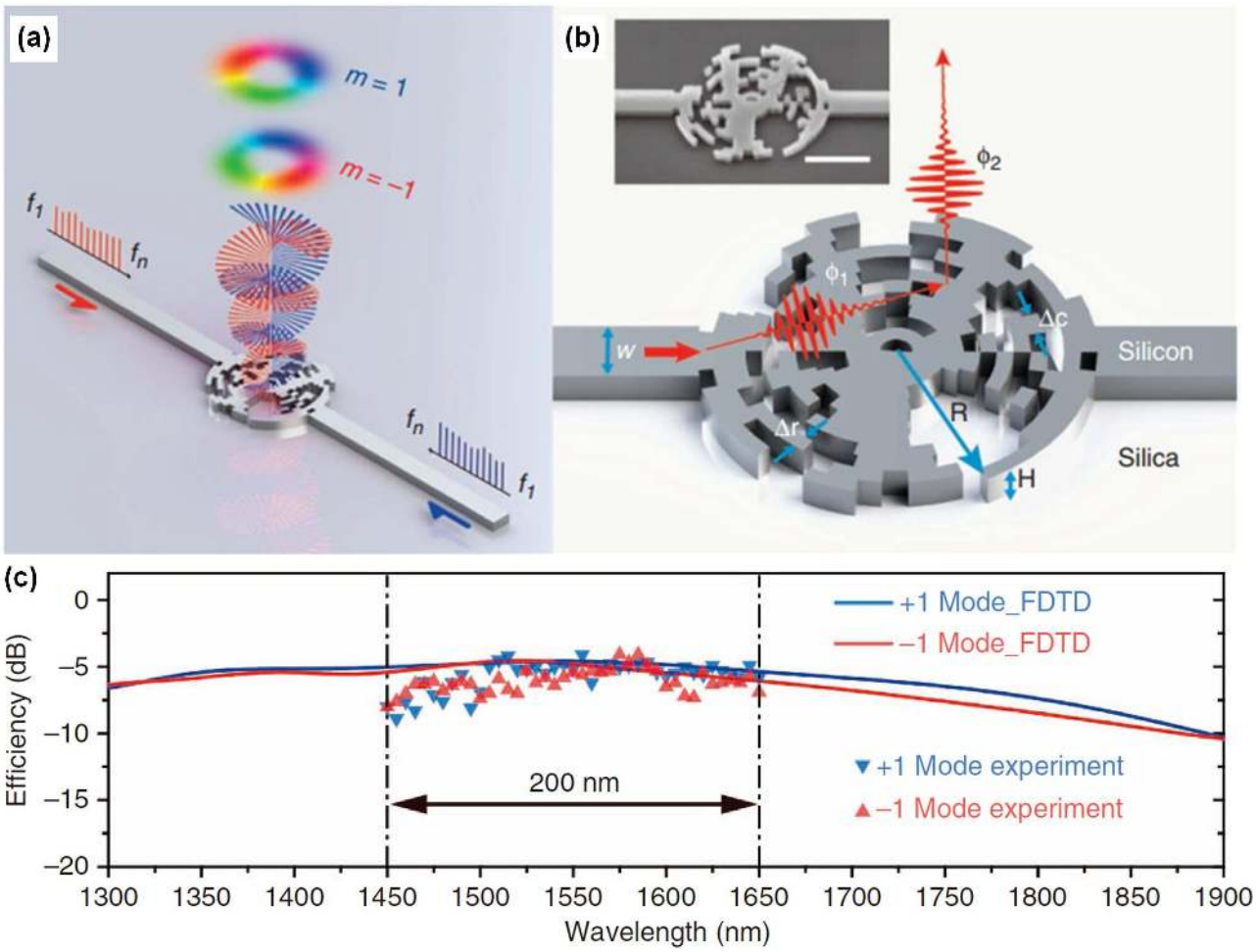

Fig. 18. Nanophotonic devices designed by SAA. (a) A schematic of the twisted light emitter. (b) Details of structure parameters. R ($R = 1200$ nm) stands for the radius of the device, H ($H = 220$ nm) the height and W ($W = 440$ nm) the width of the waveguide. The red arrow represents light from the left waveguide. Φ_1 and Φ_2 represents the propagating phase modulation and resonance modulation, respectively. Here, a scanning electron microscope image of the fabricated OAM is presented[102]. (c) FDTD simulation result and experiment data of the OAM emitter.

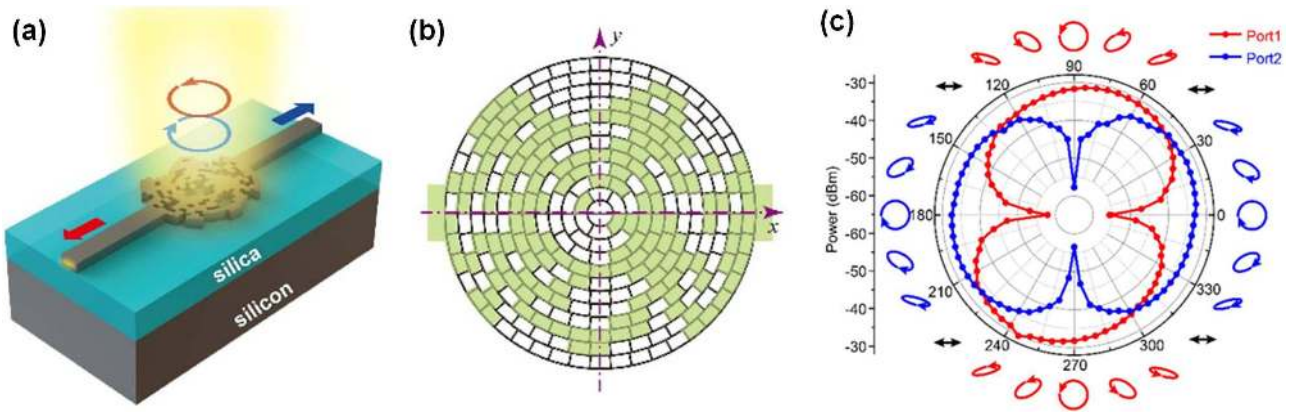

Fig. 19. Nanophotonic devices designed by SAA. (a) Schematic of the photonic spin element. Incident light is coupled into different waveguides according to the spin states. (b) The core component of an optical element. The design area is divided into 288 pixels. The green blocks stand for optimized structures filled with silicon and the white blocks stand for air. (c) The measured output power at different ports when the polarization of incident light varies[103].

shapes. When encountering a "plateau" (the function value remains unchanged in a certain range) during the searching process, the search direction cannot be determined, and will move randomly, which greatly reduces the efficiency; when encountering a "ridge" (the function is steep near the maximum), it may oscillate repeatedly around the optimal solution, and the forward speed will slow down. The hill-climbing algorithm is used to adjust the radius of the three lattices near the nanocavity [Fig. 20(a)]. Starting with random values of R_1, R_2, and R_3, structures with different parameters are evaluated by FDTD, and selected by the method of hill climbing. As a result, the quality factor of the second-order TE mode is improved to 1.99×10^4 [Fig. 20(b), 20(c), and 20(d)].

The hill-climbing algorithm is a relatively basic algorithm that is easy to start with. However, with the development of intelligent algorithms, more complex algorithms have obvious advantages in the design of nanophotonic devices and are more widely adopted.

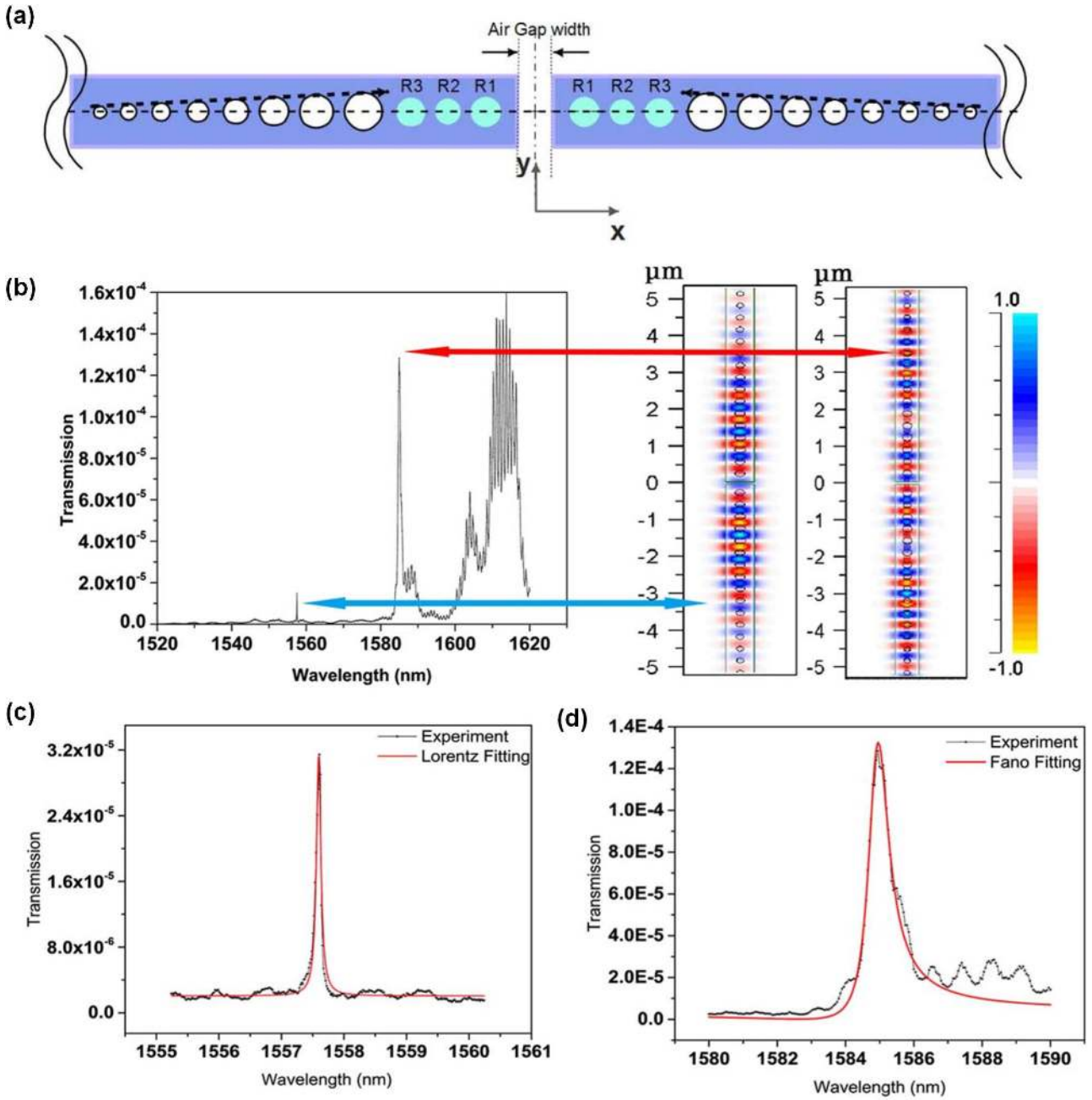

Fig. 20. Nanophotonic devices designed by the hill-climbing algorithm. (a) An example of the target function in which the difficulties of hill climbing are shown. (b) The schematic of the photonic crystal split-beam nanocavity. R_1, R_2, and R_3 are optimized by the algorithm. Experimental transmission spectrum of the split-beam cavity under 0.6 mW input power respectively in the whole measurement range, (c) the 2nd TE mode individually and (d) the 4th TE mode individually[105].

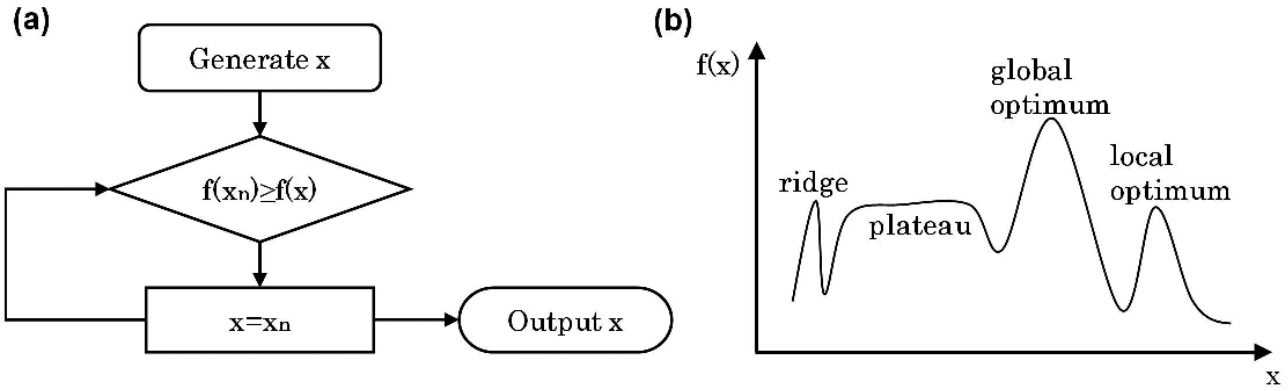

Fig. 21. (a) Flowchart of the hill climbing algorithm. (b) An example of the target function in which the difficulties of hill climbing are shown.

5.3 Tabu search

The TS algorithm[107] is a meta-heuristic random search algorithm first proposed by Glover, which is based on the improvement of the hill-climbing algorithm, and usually used to solve combinatorial optimization problems. It starts from an initial feasible solution, and then identifies the one from a series of specific search directions where the value of the objective function has the greatest improvement.

In order to avoid repeated searching, a flexible "memory" technique, the establishment of the tabu list, is used in the TS search to record and select the optimization process that has been performed to guide the next search direction. The tabu list has an associated size, which can be a fixed size or change during the iterative process and can be visualized as a window on accepted moves. The moves that tend to undo moves within this window are forbidden[108]. During the iterative process of the TS algorithm, it is possible that all moves in the candidate set are in the tabu list, or the current move is in the tabu list, but the target value will improve significantly if the tabu is lifted. In this case, in order to break through the limitation, some tabu objects will be made reselectable. This method is called aspiration, and the corresponding rule is called aspiration criterion. In the end, the TS still will probably fall into a loop; in that case, a stopping criterion is needed. Normally, the program is assigned to stop when a fixed number of iterations are reached. The flowchart of the process of TS is presented in Fig. 22.

The advantage of TS is that it provides a very effective solution to jump out of the local optimal solution, and it has fast convergence speed, finding the optimal solution with less iterations. Since TS is not guaranteed to traverse the full parameter space, it is still possible to find a local optimal solution. The search path is determined by the direction of the current solution to the neighborhood, so the structure of the neighbors, that is, the mapping relationship between the initial solution and its neighbors, is particularly important.

Gagnon *et al.* used the TS algorithm to solve inverse design problems in integrated photonics[109]. The proposed method is called parallel tabu search (PTS), which starts with a diverse population of solution individuals, and each individual goes through a TS process. This work provides a solution to the

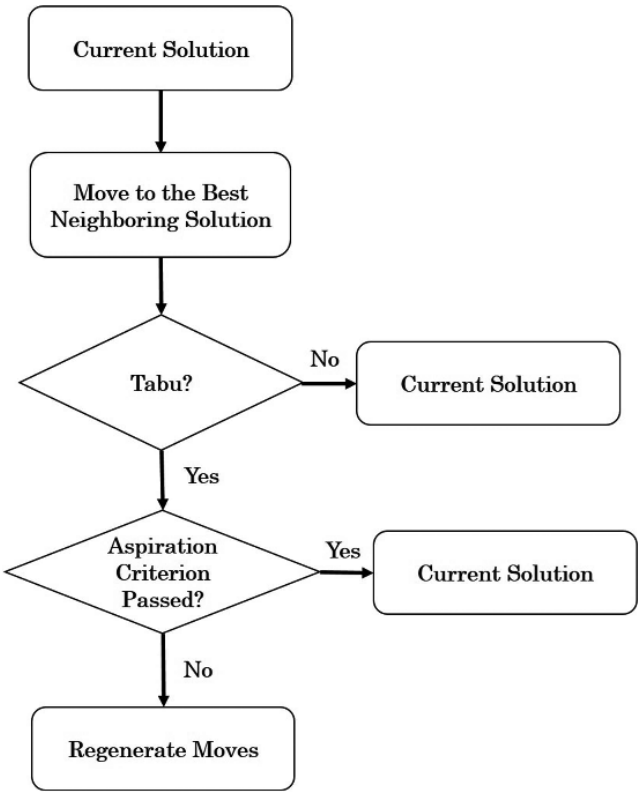

Fig. 22. The flowchart of TS.

coherent beam shaping problem, which is a multi-objective optimization problem, considering both the amplitude and the phase profile of the beam. The device is based on a 2D photonic crystal [Fig. 23(a)], and intelligent algorithms are implemented to determine the location of lattice defects. In order to compare GA and PTS, they use both methods for optimization. The best possible tradeoff between the amplitude and phase of the beam is shown in Fig. 23(b). In comparison with GA, the PTS can produce comparable or better solutions while requiring less computation time and fewer adjustable parameters. This approach is also applied to the design of integrated polarization filters[110], which is shown in Figs. 23(c) and 23(d). After the

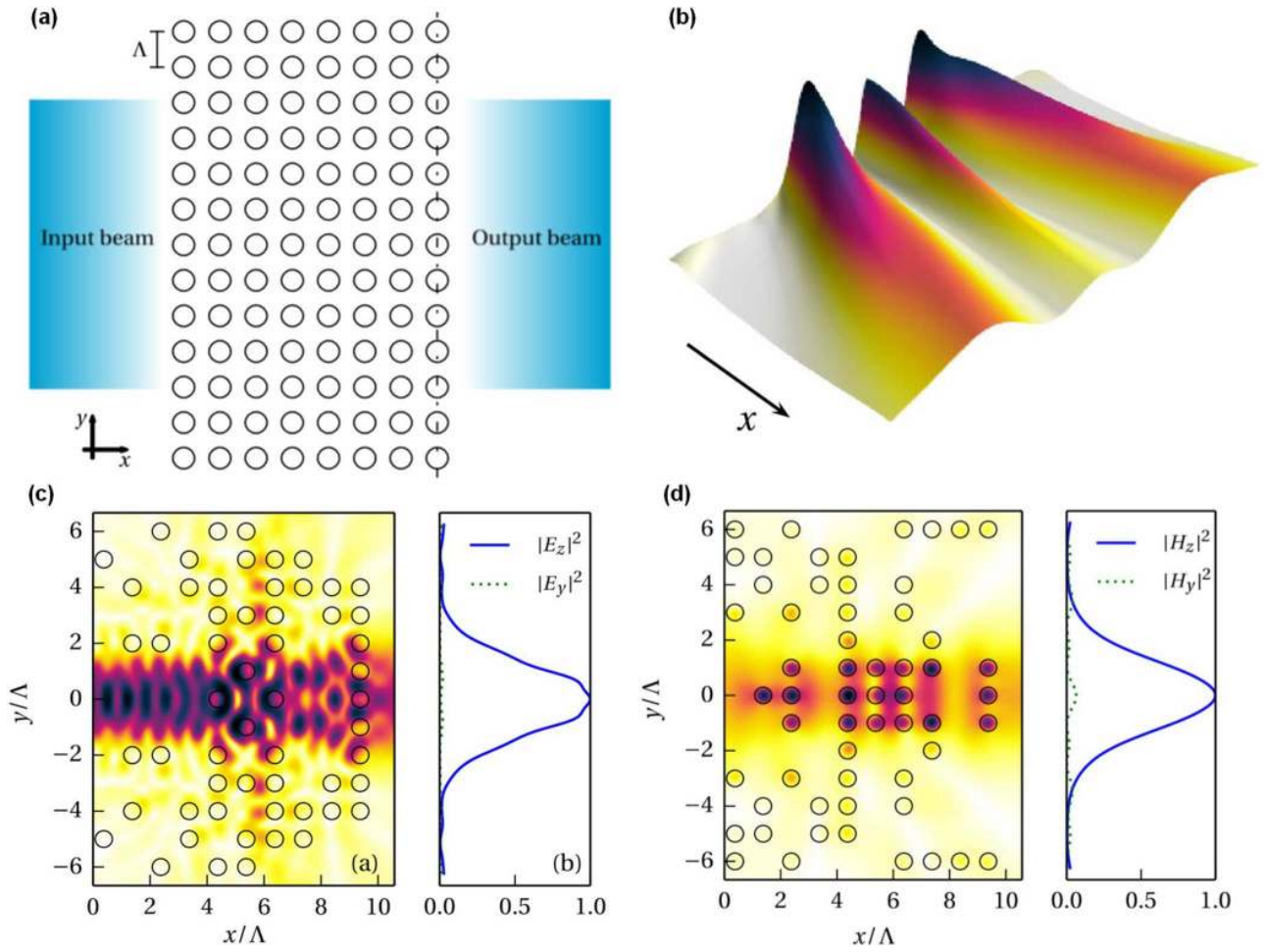

Fig. 23. Nanophotonic devices designed by TS. (a) Basic photonic lattice configuration for the beam shaping problem. (b) Best possible trade-off between the amplitude and the phase profile of the beam in the beam shaping problem[109]. (c) The $|E_z|$ field profile (arbitrary units) and comparison of orthogonal polarization components along target plane of optimized TM polarized Gaussian beam. (d) The $|H_z|$ field profile (arbitrary units) and comparison of orthogonal polarization components along target plane of optimized TE polarized Gaussian beam[110].

multi-objective optimization process, the devices allow for simultaneous polarization filtering and amplitude beam shaping. The average degree of polarization of the output beam is improved to 98% with a transmission efficiency over 75% for the TM polarizer and 80% for the TE polarizer.

TS has its opportunities in optimizing nanophotonic devices, especially when the parameter space is finite with discrete numeric values. However, due to relatively few reports, the prospects of this field need to be further explored.

6. Nanophotonic Devices Based on Other Algorithms

6.1 Direct binary search

As mentioned above, intelligent algorithms are beneficial to the design of compact devices and calculate the full parameter space, compared with conventional approaches. As one of the crucial algorithms, the DBS algorithm has drawn more and more attention recently. DBS is an iterative search algorithm that was first used for the synthesis of digital holograms[111]. The basic problem in the synthesis of binary digital holograms is to find a binary-valued transmittance function for the hologram. The DBS algorithm is used to manipulate the hologram transmittance directly to produce the best reconstruction and find a binary transmittance function that minimizes the mean squared error between the reconstructed image and original object, which is illustrated in the flow chart of Fig. 24[112].

With the development of intelligent optimization algorithms, the DBS algorithm has found more application domains, and there are some improved versions of the DBS algorithm. The modified version of the DBS algorithm operates in an iterative fashion. In the application of this method, the device should be discretized into "pixels" first. The possible pixel states are two different materials, and the two states are denoted by 1 and 0. During each iteration of the DBS algorithm, the pixel is toggled between these two states and the pixel to be perturbed is chosen at random. Then, a figure-of-merit (FOM) or objective function is calculated for the resulting device. If the FOM is improved, the perturbation is kept and the next parameter is perturbed, and the FOM is evaluated. If the FOM is not improved, the perturbation

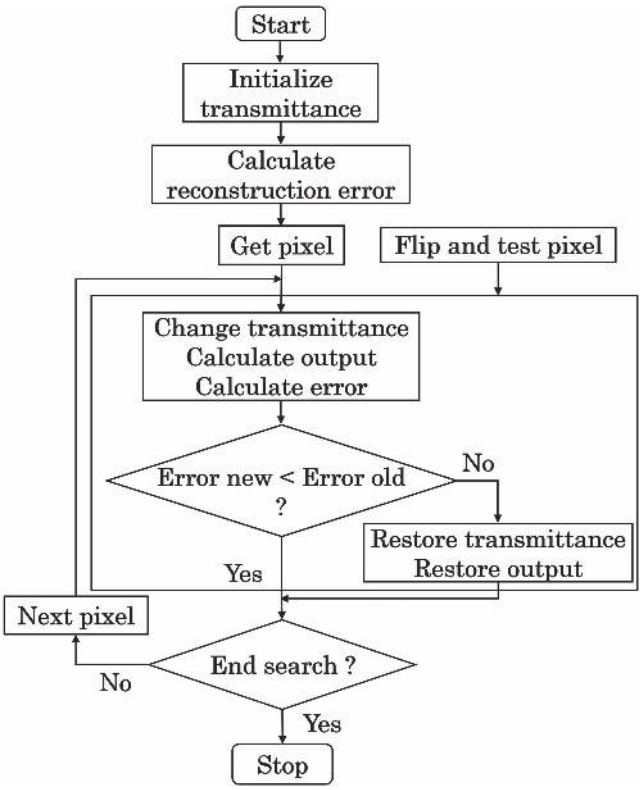

Fig. 24. The flow chart of DBS algorithm[112].

is discarded. At this time, an alternate perturbation (of the opposite sign) may be applied and the FOM is re-evaluated. This perturbation cycle continues until all the parameters have been addressed. This completes one iteration of the DBS algorithm. Such iterations are continued until the FOM converges to a stable value. An upper bound on the total number of iterations and a minimum change in FOM are defined to enforce numerical convergence[113–117].

The algorithm provides an effective approach to designing on-chip nanophotonic devices, such as the design of diffractive optics[117–119], nanophotonics for light trapping[114,116], couplers[120], computational microscopy[121], free-space polarizers[115], polarization beam splitter[113,122], optical modulator[123], the integrated cloak[124], mode router[125–127], and power splitters[128,129]. Several typical devices designed by the modified version of the DBS algorithm are introduced below. Shen *et al.* applied the nonlinear optimization to design a free-space-to-waveguide coupler, polarization beam splitter, and cloak[113,120,124], as shown in Fig. 25. Figure 25(a) is a free-space to multi-mode waveguide coupler and polarization splitter, and panels a, b, and c in Fig. 25(a) are the structure diagram, simulated time-averaged intensity distribution for light polarized along X and that polarized along Y, respectively[120].

In the device designs, they made use of the concept of free-form metamaterials, and found that allowing the geometry of the metamaterials to be freely optimized enables devices that can be highly functional. Moreover, nanopatterning enables one to engineer the refractive index in space at a deep sub-wavelength scale. In this way, devices that achieve high-efficiency mode conversion in an extremely small area become feasible. Then they designed a polarization beam splitter with a footprint of 2.4 μm × 2.4 μm in the same way, which is shown in Fig. 25(b), and the simulated steady-state intensity distributions for TE and TM polarized light at the design wavelength of 1550 nm are shown in Figs. 25(c) and 25(d), respectively[113].

With the development of the photonic integrated circuit, a higher density integration is required. One of the options to increase integration density is to decrease the spacing between the individual devices. An optical waveguide in the plane of the photonic integrated circuit is one of the most fundamental structures. However, the integration density of the waveguide is limited by the leakage of light from one waveguide to its neighbor, if the spacing between them is too small. The DBS algorithm is employed to design the integrated cloak with a footprint of just a few micrometers to decrease this spacing without considerably increasing cross talk[124]. Take the nanophotonic cloak that can render a waveguide invisible to a neighboring micro-ring resonator, for example. In most applications, light is coupled into the resonator via a waveguide that is placed in close vicinity to the ring. However, if another waveguide is placed close to the micro-ring, the two optical components would work as a coupled system with functionality that is different from that of either one working independently, as is shown in Fig. 25(e). Shen *et al.* designed a nanophotonic cloak that allows a waveguide to be placed at a gap of only 300 nm from the micro-ring and essentially renders the waveguide invisible to the micro-ring. The structure diagram of the device and the steady-state intensity distribution are shown in Fig. 25(f). Another option to increase integration density is to combine the function of multiple devices into a single compact device. Liu *et al.* designed a mode-division multiplexing circuit consisting of a multiplexer, a crossing, and a demultiplexer[126].

The DBS algorithm was used to optimize the structure of nanohole distribution. The microscope image of the circuit and a four-stage cascaded crossing circuit is shown in Fig. 26(a), and the zoom-in SEM image of the nanostructured crossing is shown in Fig. 26(b). The transmission spectra of the cascaded crossing are measured and normalized, as shown in Fig. 26(c). Moreover, Han *et al.* theoretically designed three 1 × 2 power splitters based on photonic-crystal-like metamaterial structure using the DBS algorithm[129]. The simulated results of the 1 × 4 power splitter are shown Figs. 26(d) and 26(e).

The DBS algorithm is a simpler iterative algorithm for the design of nanophotonic devices. The discrete structure generated by DBS algorithms is more favorable to the fabrication using traditional manufacturing techniques like focused ion beam milling or electron beam lithography. However, the DBS algorithm has some limitations. First, the algorithm is guaranteed to converge, but not necessarily to a global minimum. It inherently produces a suboptimal result, as the DBS algorithm converges to the first local minimum during the search process. Second, it is computationally expensive and suitable for discrete solution space and small parameter space. The cost of the

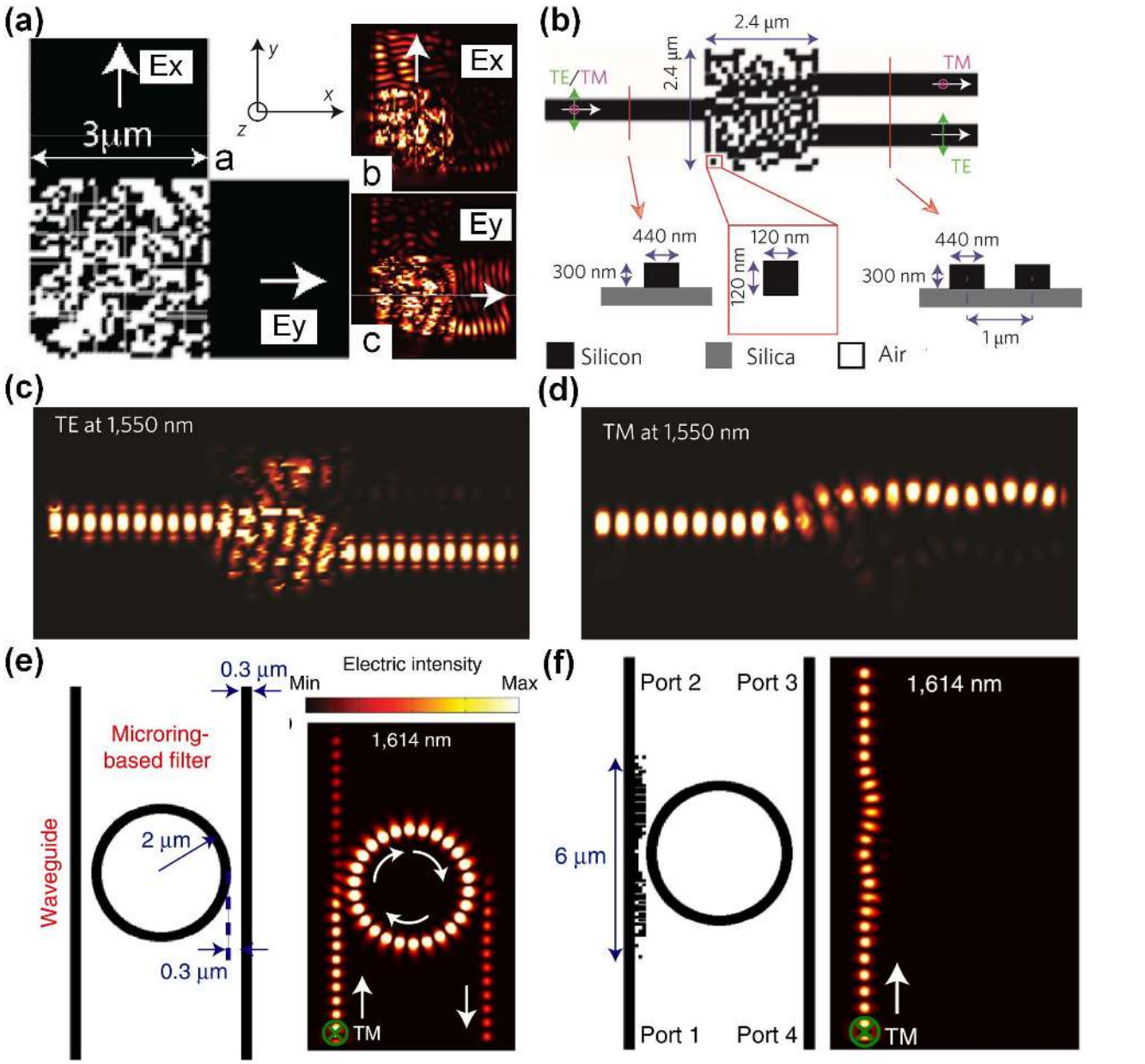

Fig. 25. Nanophotonic devices designed by DBS. (a) Panel a, structure diagram of a free-space to multi-mode waveguide coupler and polarization splitter; panels b and c are simulated time-averaged intensity distribution for light polarized along *X* and that polarized along *Y*, respectively[120]. (b) The structure diagram of a polarization splitter. (c) and (d) The simulated steady-state intensity distributions for TE and TM polarized light at the design wavelength of 1550 nm, respectively[113]. (e) and (f) Reference coupled system and the cloak for micro-ring resonator[124].

calculation and the probability of the DBS algorithm falling into the local optimal value will increase as the search space increases. Third, the algorithm is sensitive to the starting point. In view of the above analysis, there is an urgent need to develop an algorithm to design the optimal and multi-function integrated device.

6.2 Topology optimization

Topology optimization is a mathematical method for optimizing the distribution of materials in a given area according to given loads, constraints, and performance indicators. Topology optimization is one of the most promising aspects of structural optimization, with greater design freedom and space. Continuous topology optimization methods include the homogenization method, variable density method, level set method, etc. The homogenization method uses the finite element method to discretize the design area, and assumes that the entire design space is a microstructure unit (unit cell) similar to the "stomata distribution". The unit cells are evenly distributed and of the same size before the optimization starts. In the process of topology optimization, the unit cell density distribution changes; that

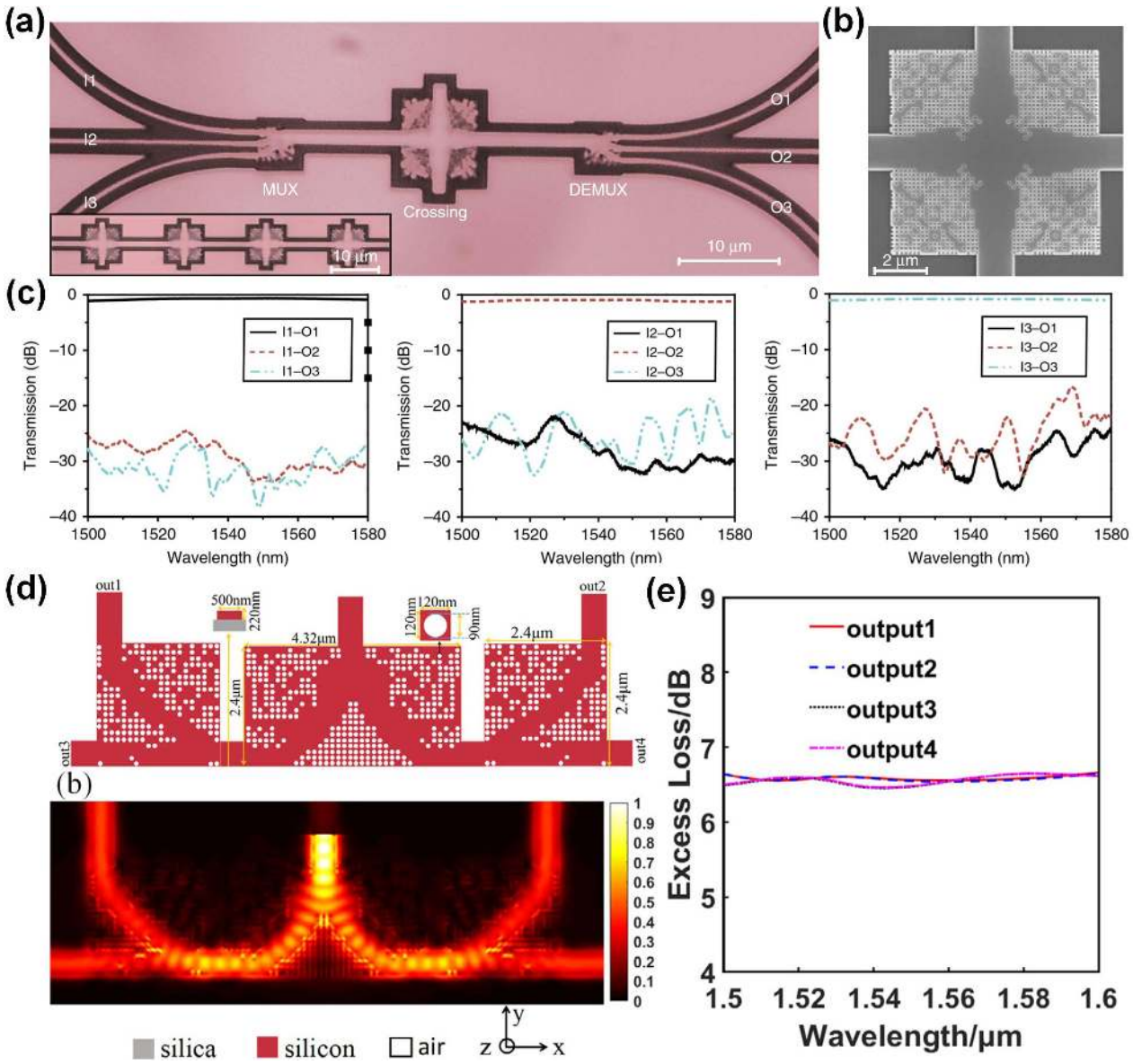

Fig. 26. Nanophotonic devices designed by DBS. (a) The top-view microscope image of the mode-division multiplexing circuit (top), and the lower left corner is the microscope image of the four-cascaded crossing[126]. (b) The scanning electron microscope image. (c) The measured transmission spectra for the mode-division multiplexing circuit. (d) The top view of the 1 × 4 power splitter (top), and the bottom is optical field distribution[129]. (e) Excess loss of each output port.

is, the unit cell density in the high stress area becomes larger while the unit cell density in the low stress area becomes smaller. A load-bearing structure was formed during the optimization process. This structure is "dense" in high-stress areas and "sparse" in low-stress areas. When the iterative calculations are all completed, define a reasonable minimum density, and then remove the area in the design space where the unit cell density is lower than this minimum to produce a weight-optimized load-bearing structure with the highest material effect. The variable density method expresses the corresponding relationship between the relative density of the element and the elastic modulus of the material in the form of a density function of continuous variables, seeks the best force transmission route of the structure, and optimizes the distribution of materials in the design area. It has the advantages of easy program implementation, high calculation efficiency, and calculation accuracy. However, the result of this method has a fuzzy boundary. The level set method is discussed below. The topology optimization of discrete structures is mainly based on the basic structure method, using different algorithms to solve the problem. Topology optimization is more and more widely used due to its advantages[130,131]. In the work of Chen *et al.*, topology optimization was performed, as shown in Fig. 27(a), and the metal nanoparticle dimer was designed in reverse, with the goal of optimizing the near-field enhancement factor in the gap below 10 nm. By optimizing the material layout within a given design

space, the topology optimization algorithm can generate a plasma nanodimer of two heart-shaped particles with convex and concave features, as shown in Fig. 27(b)[132].

The level set method is a numerical technique for interface tracking and shape modeling. One of the advantages of the level set method is that the curves and surfaces can be numerically calculated on a Cartesian grid without parameterizing curves and surfaces (this is the so-called Eulerian approach). Another advantage of the level set method is that it is easy to track the topology change of the object. For example, an object may be divided into two parts or combined into one, or a new cavity or new entity may be created. All of these make the level set method a powerful tool for time-changing objects modeling, such as expansion of airbag and oil droplets falling into the water. However, the level set equation needs to be updated with the PDE equation. During the process, the level set equation needs to be reset to ensure the continuous update of the PDE, which will greatly reduce the optimal convergence speed, or even fail to converge.

The optimal design of photonic bandgaps for 2D square lattices is considered[133]. The level set method can represent the interface between two materials with two company's dielectric constants[134].

Let $\epsilon = \begin{cases} \epsilon_1 & \text{for } \{x : \varphi(x) < 0\} \\ \epsilon_2 & \text{for } \{x : \varphi(x) > 0\} \end{cases}$ [135]. The level set function is updated by solving the Hamilton–Jacobi equation $\varphi_t + V|\nabla\varphi| = 0$, where the velocity V gives the correct direction to optimize the desired design. The optimization problems to be solved here are as follows.

1. Maximize the bandgap in TM: $\sup_\varphi\left(\inf_\alpha \omega_{\mathrm{TM}}^{m+1} - \sup_\alpha \omega_{\mathrm{TM}}^{n}\right)$.
2. Maximize the bandgap in TE: $\sup_\varphi\left(\inf_\alpha \omega_{\mathrm{TE}}^{m+1} - \sup_\alpha \omega_{\mathrm{TE}}^{n}\right)$.

The main approach is as follows[133].

1. First choose the initial ϵ and decide which bandgap we want to maximize.
2. For $i = 0, 1, 2, \ldots$, find the velocity V that gives an ascent direction and a step size t_i to yield an increase in the objective bandgap. Use the level set method to update φ and then obtain the new ϵ.

The evolution of the dielectric distribution is shown in Fig. 28(a). The change of bandgap as the number of iterations increases is shown in Fig. 28(b). The final band structure for maximizing the bandgap between ω_{TM}^1 and ω_{TM}^2 is shown in Fig. 28(c).

The level set method can calculate the curves and surfaces in the evolution process numerically on the Cartesian grid without parametric curves and surfaces. It has a larger application space, and it is believed that the level set algorithm can solve more problems.

6.3 Monte Carlo method

The Monte Carlo method, also known as a statistical simulation method, is a very important numerical calculation method guided by the theory of probability and statistics, which was proposed in the mid-1940s due to the development of science and technology and the invention of electronic computers. The Monte Carlo method is a method that uses random numbers to solve many computing problems. The Monte Carlo method

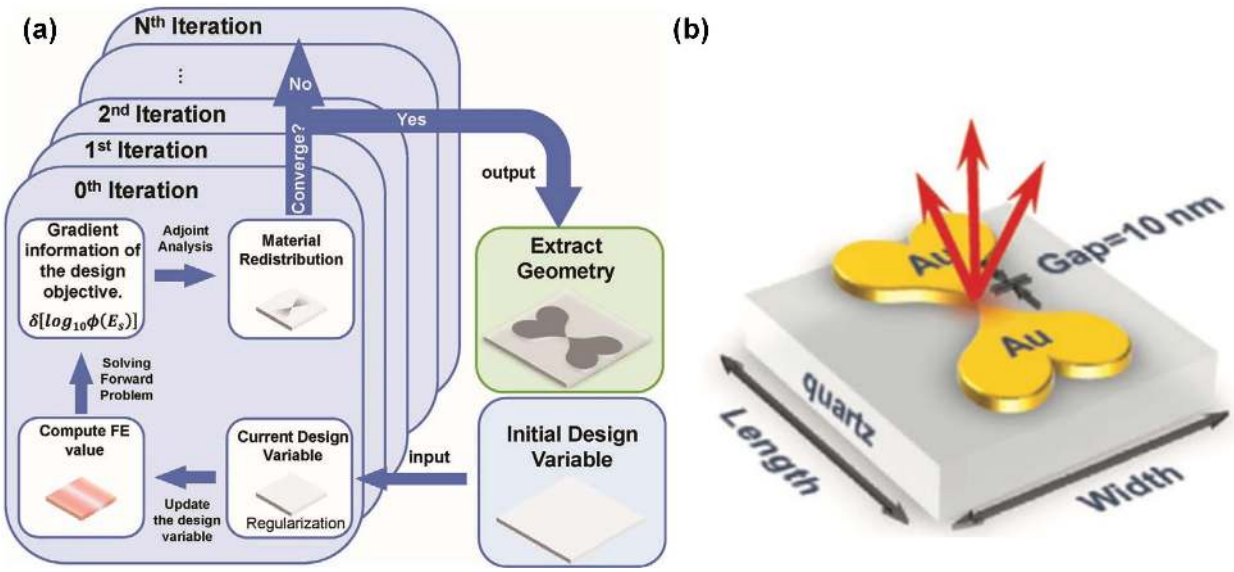

Fig. 27. (a) The structure of the topology optimization algorithm used in the work. (b) The 3D model of gold nanoparticle dimer with predefined key parameters in geometry and material.

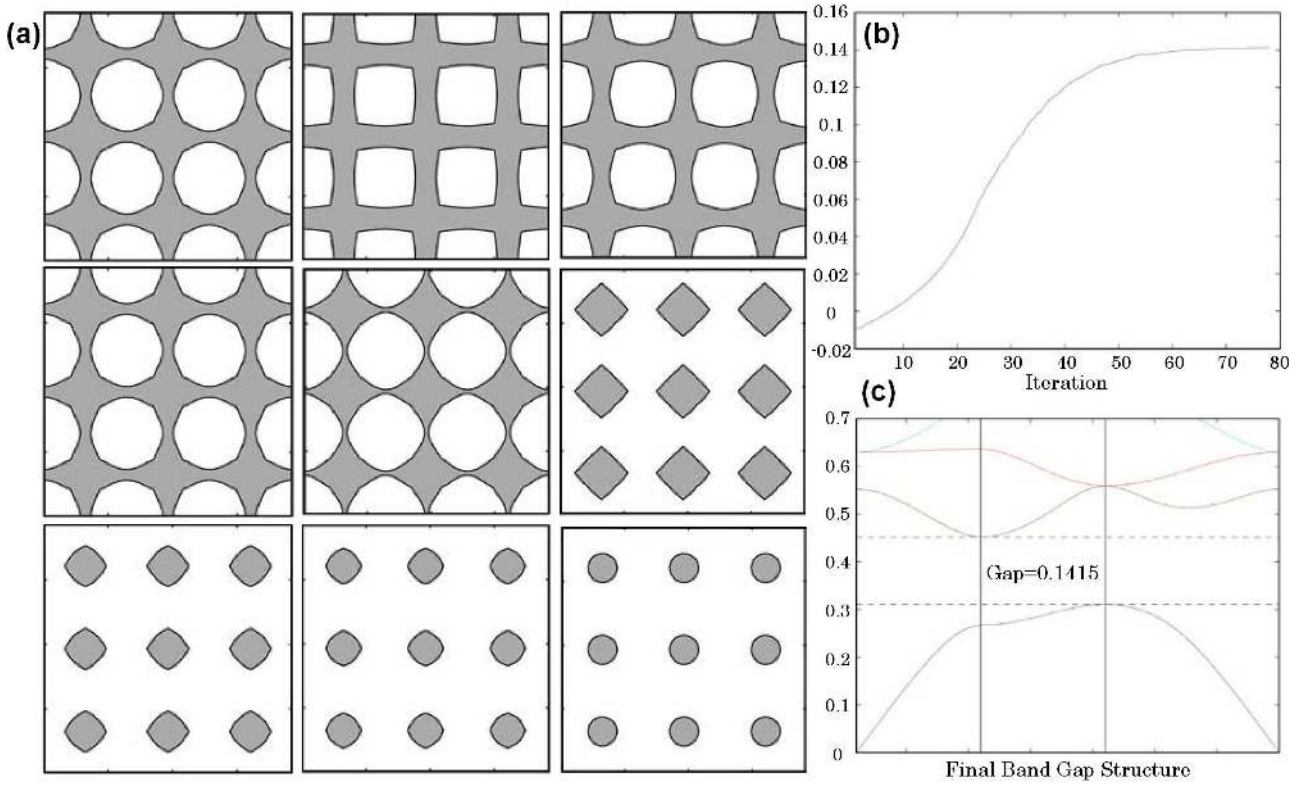

Fig. 28. Nanophotonic devices designed by the level set method. (a) The evolution of the dielectric distribution[133]. (b) The bandgap versus the iteration. (c) The final band structure with the largest bandgap between ω^1_{TM} and ω^2_{TM}.

is widely used in financial engineering, macroeconomics, computational physics, and other fields[136,137].

The Monte Carlo method usually solves mathematical problems by constructing random numbers that conform to certain rules. The Monte Carlo method is an effective method to find numerical solutions for those problems that are too complex to obtain analytical solutions or have no analytical solutions at all. The most common application of the Monte Carlo method in mathematics is the Monte Carlo integral[137].

Applying the Monte Carlo method to practical problems has two main parts.

1. When the Monte Carlo method is used to simulate a process, it is necessary to generate a random variable of a probability distribution.
2. The numerical characteristics of the model are estimated by the statistical method, and the numerical solutions of practical problems are obtained[138].

With the help of computer technology, the Monte Carlo method has many advantages; it is simple and fast, eliminating the need for complicated mathematical derivation and calculation. Moreover, the Monte Carlo method has a strong adaptability, and the complexity of the problem geometry has little influence on it. It is believed that the Monte Carlo method will have more applications in the field of photonic nanometers.

7. Summary and Outlook

In this review article, we extensively discuss a variety of intelligent algorithms including deep learning methods, the gradient-based inverse design method, swarm intelligence algorithms, individual inspired algorithms, and other intelligent algorithms, as well as nanophotonic devices designed using these algorithms. Some representative examples are used to analyze various intelligent algorithms for different situations. In many practical applications, intelligent algorithms are practical methods to deal with various challenging problems. The advantages, disadvantages, characteristics, and suitable devices of the algorithms discussed in this paper are presented in Table 1.

Compared with the traditional design method, the intelligent algorithm is universal and efficient. For example, the advantages of deep learning are that once trained it takes less time (i.e., less computational cost) than traditional algorithms, and is more likely to find better optimization solutions. In addition, compared with traditional algorithms, the deep learning method can realize inverse design more easily. ANN has many typical structures and strong flexibility. According to the design requirements of the equipment and many problems in the training process, we can choose the appropriate neural network for optimal design. First, the design of nanophotonic devices is nonconvex, and there is no guarantee that the designed devices are optimal. Second, preparing training sets and training neural networks require a lot of computing and time costs, especially when

Table 1. Comparison of Various Intelligent Design Algorithms.

Intelligent Design Algorithms	Advantages	Disadvantages	Unique Features	Suitable Design for Photonic Structures and Devices
1. Deep learning methods	Spend less time than traditional algorithms (after training). More likely to find better local optimal solutions. Many typical structures and strong flexibility. Realize inverse design more easily.	Take a lot of computational and time cost for preparation and training. Difficult to exploit the trained ANN for further analysis. Poor performance when dealing with problems with few samples.	Great for dealing with problems that training set is easy to generate. Ability of transfer learning (albeit immature). Some hyperparameters need to be tweaked.	Nanoparticle[51], power splitter[16,23], optical spectrum[34,37,50,52], metamaterial[17], metasuface[21,49]
2. Gradient-based inverse design algorithm	Large parameter space, high computational efficiency.	Exhibit a continuous topography, produce a local optimal solution.	Gradient-based, large parameter space.	Multi-channel devices[62], router[60–62,64,65,67], coupler[63], mode converter[62], accelerators[66], switch[68]
3. Genetic algorithm	Suitable for solving complex optimization problems, concurrency, extensibility.	Low search efficiency in late evolution, premature convergence.	Large coverage, self-organization, self-adaptation, self-learning.	Coupler[78], metamaterial[6], nanoparticle[80], router[79]
4. Particle swarm optimization	Fast convergence speed, easily understood, parallel computing.	High requirement for parameter setting.	Real-time change of perception.	Waveguide[5], nanoparticle[87]
5. Ant colony algorithm	Suitable for combinatorial and continuous function optimization, intuitive.	High time cost.	Usually combined with other algorithms.	Photovoltaic collector[95]
6. Simulated annealing algorithm	Robustness of a random initial guess, simple structure, parallel computing.	Sensitive to parameters, low efficiency.	Converge with the drop of temperature.	Coupler[103], switch[99], metamaterial[139]
7. Hill-climbing algorithm	Easily understood, avoid traversal in solution space.	Unable to break out the local optimum.	One of the most basic heuristic algorithms.	Nano-cavity[104]
8. Tabu search algorithm	Suitable for combinatorial optimization, fast convergence speed.	Premature convergence, high requirement for parameter setting.	Parallel tabu search can improve efficiency.	Polarization filter[110], beam shaping device[109]
9. Direct binary search	Discrete structure generated by DBS algorithms is more favorable to the fabrication.	Suitable for small parameter space, computationally expensive, sensitive to the starting point.	A simpler iterative algorithm.	Coupler[120], computational microscopy[121], polarizer[115], router[113,122,128,129], optical modulator[123], integrated cloak[124]
10. Topology optimization	Large degree of freedom in design, high sustainability.	Complex shapes are difficult to manufacture.	Optimize the material distribution.	Band structures[133]
11. Monte Carlo method	Strong adaptability.	Assumptions need to be fair.	Solve problems without analytical solutions.	Optical imaging[140]

dealing with complex learning tasks. Third, further analysis using trained neural networks is difficult because ANN's learning mechanisms (note that they are sometimes useful) operate as black boxes. However, the useful information about the features of photonic structures can be extracted by introducing proper techniques such as latent space[22]. Fourth, a stronger capacity for migration learning is needed to cope with changing situations, although this ability is being developed to demonstrate its power in the design of nanophotonic devices. Finally, in the case of fewer training samples, traditional methods may perform better than deep learning methods.

The gradient-based inverse design method can automatically design nanophotonic devices and only requires the user to input high level parameters. This method can provide large parameter space and design devices using full space parameters of manufacturable devices, which often requires less simulation than GA or PSO because they do not rely on parametric scanning or random perturbations to find their minima. This method can be used to design any passive, linear photonic device.

However, the implemented design usually presents a continuous terrain, and some very small structural components may be formed during the inverse design process, which presents a challenge to sample making. In addition, the gradient-based inverse design method usually produces only local optimal solutions and cannot realize the true global optimal solution.

Swarm intelligence algorithms have certain robustness and strong evolutionary or search ability. GA can not only solve single-objective optimization problem, but also play a greater role in multi-objective optimization problems. It has the characteristics of group search and is suitable for solving complex optimization problems, such as the need to optimize multiple system parameters at the same time, and some other application problems may not have clear and unique optimal values. Moreover, GA is scalable and easy to be combined with other algorithms. However, the search efficiency of GA in the later stage of evolution is slightly lower, and it is prone to premature convergence. Although the local search ability of the genetic algorithm is poor, it is often used in combination with other algorithms to improve its performance due to its easy parallel implementation. PSO has a fast speed of approaching the optimal solution, which can effectively optimize the parameters of the system, and the process is simple and easy to understand. The advantage of PSO is that it can be applied to continuous function optimization problems. The main drawback of this method is that it requires high parameters. It is easy to produce premature convergence when dealing with complex multiple optimal value search problems, and its local optimization ability is poor. PSO falls into local minimum, which is mainly due to the loss of diversity in the search space. We can improve it by combining it with other algorithms or adding mutation operations. PSO has been used to optimize nanostructures and design nanophotonic devices. It can be used to optimize multidimensional problems. The ACA is suitable for combinatorial optimization and continuous function optimization. The whole algorithm process is intuitive and easy to understand, but it takes a long time to solve. ACA has strong robustness in solving performance and is easy to be implemented in parallel. Therefore, other algorithms are usually combined with ACA to improve the performance of the algorithm, to design more ideal nanophotonic devices.

Individual inspired algorithms can give a better solution in a certain acceptable time, but cannot guarantee it is optimal. The calculation process of SAA is simple, and it has strong universality and robustness. However, it is very sensitive to customized parameters, especially the initial temperature. When faced with a large number of parameters that need to be optimized, SAA randomly selects new solutions from the solution space, thus making the performance weak. In the case of many unknown parameters, the search efficiency and the possibility of finding the optimal solution will decrease. The climbing algorithm is more intuitive, because the memory requirement is small. But it cannot solve the problem of large-scale multi-constraint. The TS algorithm has fast convergence speed and few iteration times, but the results depend on the initial solution and the adjacent.

The DBS algorithm is a simple iterative algorithm for designing nanophotonic devices. The discrete structure generated by the DBS algorithm is more conducive to the traditional manufacturing techniques such as focused ion beam milling or electron beam lithography. However, the DBS algorithm has some limitations. First, the algorithm guarantees convergence, but not necessarily to the minimum. When it converges to the first local minimum during the search, it inherently produces a suboptimal result. Second, it is suitable for discrete solution space and small parameter space due to its large computation. The calculation cost and the probability of falling into the local optimal value will increase with the increase of search space. Third, the algorithm is sensitive to the starting point. Topological optimization has more design freedom and design space, among which the level set method used in designing of nanophotonic devices can be used for numerical calculation of curves and surfaces in the evolution process on the Cartesian grid of parametric curves and surfaces, but the process is more complex and requires a certain mathematical foundation. The level set equation needs to be updated with a partial differential equation and, in the middle, the level set equation needs to be reset to ensure the continuous update of the partial differential equation, which greatly reduces the optimal convergence rate or even fails to converge. The Monte Carlo method has strong adaptability and can solve probability and statistics problems easily and quickly. However, the number of samples must be large enough, and the calculation process is long.

As the need for nanophotonic devices to achieve more functions is further strengthened, the intelligent algorithms, especially the more popular method – the deep learning method – with higher efficiency and better effect[54–56], will continue playing a significant role in the designing of nanophotonic devices to implement complex functions and improve the performance of nanophotonic devices. This will provide an avenue for the realization of photonic chips in the future. As for the utilization of intelligent algorithms, we think during the designing process of nanophotonic devices, multiple algorithms can be

adopted simultaneously to provide efficient and optimal solutions, rather than just one algorithm. In addition, when too many algorithms are difficult to choose from, the more reports some algorithms appear in, the more frequently they have been used, which may be a reference for similar problems.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (Nos. 11604378, 91850117, and 11654003) and the Beijing Institute of Technology Research Fund Program for Young Scholars.

References

1. Y. Chen, Y. Cheng, R. Zhu, F. Wang, H. Cheng, Z. Liu, C. Fan, Y. Xue, Z. Yu, J. Zhu, X. Hu, and Q. Gong, "Nanoscale all-optical logic devices," Sci. Chin. Phys. Mech. **62**, 44201 (2019).
2. K. Yao, R. Unni, and Y. Zheng, "Intelligent nanophotonics: merging photonics and artificial intelligence at the nanoscale," Nanophotonics **8**, 339 (2019).
3. S. Molesky, Z. Lin, A. Y. Piggott, W. Jin, J. Vucković, and A. W. Rodriguez, "Inverse design in nanophotonics," Nat. Photonics **12**, 659 (2018).
4. N. Palumbo, "525.770 – Intelligent algorithms," https://ep.jhu.edu/programs-and-courses/525.770-intelligent-algorithms.
5. M. S. Kumar, S. Menabde, S. Yu, and N. Park, "Directional emission from photonic crystal waveguide terminations using particle swarm optimization," J. Opt. Soc. Am. B **27**, 343 (2010).
6. M. D. Huntington, L. J. Lauhon, and T. W. Odom, "Subwavelength lattice optics by evolutionary design," Nano Lett. **14**, 7195 (2014).
7. Z. Liu, X. Liu, Z. Xiao, C. Lu, H. Wang, Y. Wu, X. Hu, Y. Liu, H. Zhang, and X. Zhang, "Integrated nanophotonic wavelength router based on an intelligent algorithm," Optica **6**, 1367 (2019).
8. C. Lu, Z. Liu, Y. Wu, Z. Xiao, D. Yu, H. Zhang, C. Wang, X. Hu, Y. C. Liu, X. Liu, and X. Zhang, "Nanophotonic polarization routers based on an intelligent algorithm," Adv. Opt. Mater. **8**, 1902018 (2020).
9. S. R. Granter, A. H. Beck, and D. J. Papke, "AlphaGo, deep learning, and the future of the human microscopist," Arch. Pathol. Lab. Med. **141**, 619 (2017).
10. G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," Science. **313**, 504 (2006).
11. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature **521**, 436 (2015).
12. A. F. Oskooi, D. Roundy, M. Ibanescu, P. Bermel, J. D. Joannopoulos, and S. G. Johnson, "Meep: a flexible free-software package for electromagnetic simulations by the FDTD method," Comput. Phys. Commun. **181**, 687 (2010).
13. I. Malkiel, M. Mrejen, A. Nagler, U. Arieli, L. Wolf, and H. Suchowski, "Plasmonic nanostructure design and characterization via deep learning," Light Sci. Appl. **7**, 555 (2018).
14. D. Liu, Y. Tan, E. Khoram, and Z. Yu, "Training deep neural networks for the inverse design of nanophotonic structures," ACS Photon. **5**, 1365 (2018).
15. L. Xu, M. Rahmani, Y. Ma, D. A. Smirnova, K. Z. Kamali, F. Deng, Y. K. Chiang, L. Huang, H. Zhang, S. Gould, D. N. Neshev, and A. E. Miroshnichenkoa, "Enhanced light-matter interactions in dielectric nanostructures via machine-learning approach," Adv. Photon. **2**, 026003 (2020).
16. M. H. Tahersima, K. Kojima, T. Koike-Akino, D. Jha, B. Wang, C. Lin, and K. Parsons, "Deep neural network inverse design of integrated photonic power splitters," Sci. Rep. **9**, 1368 (2019).
17. W. Ma, F. Cheng, and Y. Liu, "Deep-learning-enabled on-demand design of chiral metamaterials," ACS Nano. **12**, 6326 (2018).
18. B. Wu, K. Ding, C. T. Chan, and Y. Chen, "Machine prediction of topological transitions in photonic crystals," arXiv:1907.07996 (2017).
19. J. Zhao, Y. S. H. Zhu, Z. Zhu, J. E. Antonio-Lopez, R. A. Correa, S. Pang, and A. Schulzgen, "Deep-learning cell imaging through Anderson localizing optical fiber," Adv. Photon. **1**, 066001 (2019).
20. I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proceedings of the 27th International Conference on Neural Information Processing Systems* (2014), p. 1.
21. J. Jiang, D. Sell, S. Hoyer, J. Hickey, J. Yang, and J. A. Fan, "Free-form diffractive metagrating design based on generative adversarial networks," ACS Nano. **13**, 8872 (2019).
22. W. Ma, F. Cheng, Y. Xu, Q. Wen, and Y. Liu, "Probabilistic representation and inverse design of metamaterials based on a deep generative model with semi-supervised learning strategy," Adv. Mater. **31**, 1901111 (2019).
23. Y. Tang, K. Kojima, T. Koike-Akino, Y. Wang, P. Wu, M. TaherSima, D. Jha, K. Parsons, and M. Qi, "Generative deep learning model for a multi-level nano-optic broadband power splitter," in *Optical Fiber Communications Conference and Exhibition* (2020), paper Th1A.1.
24. H. Zhou, Y. Zhao, G. Xu, X. Wang, Z. Tan, J. Dong, and X. Zhang, "Chip-scale optical matrix computation for PageRank algorithm," IEEE J. Sel. Top. Quant. **26**, 8300910 (2020).
25. H. Zhou, Y. Zhao, X. Wang, D. Gao, J. Dong, and X. Zhang, "Self-configuring and reconfigurable silicon photonic signal processor," ACS Photon. **7**, 792 (2020).
26. H. Zhou, Y. Zhao, Y. Wei, F. Li, J. Dong, and X. Zhang, "All-in-one silicon photonic polarization processor," Nanophotonics **8**, 2257 (2019).
27. A. M. Hammond and R. M. Camacho, "Designing integrated photonic devices using artificial neural networks," Opt. Express **27**, 29620 (2019).
28. P. Baldi, P. Sadowski, and D. Whiteson, "Searching for exotic particles in high-energy physics with deep learning," Nat. Commun. **5**, 4308 (2014).
29. W. J. Brouwer, J. D. Kubicki, J. O. Sofo, and C. L. Giles, "An investigation of machine learning methods applied to structure prediction in condensed matter," arXiv:1405.3564 (2014).
30. M. Rupp, A. Tkatchenko, K. Müller, and O. A. von Lilienfeld, "Fast and accurate modeling of molecular atomization energies with machine learning," Phys. Rev. Lett. **108**, 058301 (2012).
31. S. Jiao, Z. Jin, C. Chang, C. Zhou, W. Zou, and X. Li, "Compression of phase-only holograms with JPEG standard and deep learning," Appl. Sci. **8**, 1258 (2019).
32. S. So, J. Mun, and J. Rho, "Simultaneous inverse design of materials and structures via deep learning: demonstration of dipole resonance engineering using core-shell nanoparticles," ACS Appl. Mater. Inter. **11**, 24264 (2019).
33. Y. Chen, J. Zhu, Y. Xie, N. Feng, and Q. H. Liu, "Smart inverse design of graphene-based photonic metamaterials by an adaptive artificial neural network," Nanoscale **11**, 9749 (2019).
34. T. Zhang, J. Wang, Q. Liu, J. Zhou, J. Dai, X. Han, Y. Zhou, and K. Xu, "Efficient spectrum prediction and inverse design for plasmonic waveguide systems based on artificial neural networks," Photon. Res. **7**, 368 (2019).
35. Z. Liu, D. Zhu, S. P. Rodrigues, K. Lee, and W. Cai, "Generative model for the inverse design of metasurfaces," Nano Lett. **18**, 6570 (2018).
36. J. Peurifoy, Y. Shen, L. Jing, Y. Yang, F. Cano-Renteria, B. G. DeLacy, J. D. Joannopoulos, M. Tegmark, and M. Soljacic, "Nanophotonic particle simulation and inverse design using artificial neural networks," Sci. Adv. **4**, eaar4206 (2018).
37. P. R. Wiecha and O. L. Muskens, "Deep learning meets nanophotonics: a generalized accurate predictor for near fields and far fields of arbitrary 3D nanostructures," Nano Lett. **20**, 329 (2019).
38. H. Ren, W. Shao, Y. Li, F. Salim, and M. Gu, "Three-dimensional vectorial holography based on machine learning inverse design," Sci. Adv. **6**, eaaz4261 (2020).
39. W. Ma and Y. Liu, "A data-efficient self-supervised deep learning model for design and characterization of nanophotonic structures," Sci. China Phys. Mech. Astron. **63**, 284212 (2020).
40. Z. Liu, D. Zhu, K. Lee, A. S. Kim, L. Raju, and W. Cai, "Compounding meta-atoms into meta-molecules with hybrid artificial intelligence techniques," Adv. Mater. **32**, 1904790 (2019).
41. O. Hemmatyar, S. Abdollahramezani, Y. Kiarashinejad, M. Zandehshahvar, and A. Adibi, "Full color generation with Fano-type resonant HfO_2 nanopillars designed by a deep-learning approach," Nanoscale **11**, 21266 (2019).
42. S. So, T. Badloe, J. Noh, J. Bravo-Abad, and J. Rho, "Deep learning with coherent nanophotonic circuits," Nanophotonics **9**, 1041 (2020).
43. Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, and M. Soljačić, "Deep learning enabled inverse design in nanophotonics," Nat. Photon. **11**, 441 (2017).

44. J. Feldmann, N. Youngblood, C. D. Wright, H. Bhaskaran, and W. H. P. Pernice, "All-optical spiking neurosynaptic networks with self-learning capabilities," Nature **569**, 208 (2019).
45. M. Veli, Y. Luo, A. Ozcan, M. Jarrahi, Y. Rivenson, N. T. Yardimei, and X. Lin, "All-optical machine learning using diffractive deep neural networks," Science **361**, 1004 (2018).
46. S. Jiao, J. Feng, Y. Gao, T. Lei, Z. Xie, and X. Yuan, "Optical machine learning with incoherent light and a single-pixel detector," Opt. Lett. **44**, 5186 (2019).
47. E. Khoram, A. Chen, D. Liu, L. Ying, Q. Wang, M. Yuan, and Z. Yu, "Nanophotonic media for artificial neural inference," Opt. Lett. **7**, 823 (2019).
48. J. Chang, V. Sitzmann, X. Dun, W. Heidrich, and G. Wetzstein, "Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification," Sci. Rep. **8**, 12324 (2018).
49. J. Jiang and J. A. Fan, "Global optimization of dielectric metasurfaces using a physics-driven neural network," Nano Lett. **19**, 5366 (2019).
50. I. Sajedian, J. Kim, and J. Rho, "Finding the optical properties of plasmonic structures by image processing using a combination of convolutional neural networks and recurrent neural networks," Microsyst. Nanoeng. **5**, 27 (2019).
51. Y. Qu, L. Jing, Y. Shen, M. Qiu, and M. Soljačić, "Basic instincts," ACS Photon. **6**, 1168 (2019).
52. M. Hutson, "Migrating knowledge between physical scenarios based on artificial neural networks," Science. **360**, 845 (2018).
53. S. Jiao, Y. Gao, J. Feng, T. Lei, and X. Yuan, "Does deep learning always outperform simple linear regression in optical imaging?" Opt. Express **28**, 3717 (2020).
54. E. Goi, Q. Zhang, X. Chen, H. Luan, and M. Gu, "Perspective on photonic memristive neuromorphic computing," PhotoniX **1**, 3 (2020).
55. T. Chen, J. van Gelder, B. van de Ven, S. V. Amitonov, B. de Wilde, H. Ruiz Euler, H. Broersma, P. A. Bobbert, F. A. Zwanenburg, and W. G. van der Wiel, "Classification with a disordered dopant-atom network in silicon," Nature **577**, 341 (2020).
56. Y. Kiarashinejad, S. Abdollahramezani, and A. Adibi, "Deep learning approach based on dimensionality reduction for designing electromagnetic nanostructures," NPJ Comput. Mater. **6**, 12 (2020).
57. T. Han, C. Liu, W. Yang, and D. Jiang, "Deep transfer network with joint distribution adaptation: a new intelligent fault diagnosis framework for industry application," ISA Trans. **97**, 269 (2020).
58. M. Wang and W. Deng, "Deep visual domain adaptation: a survey," Neurocomputing **312**, 135 (2018).
59. K. Chadan, P. C. Sabatier, and R. G. Newton, *Inverse Problems in Quantum Scattering Theory* (Springer Science & Business Media, 1988).
60. A. Y. Piggott, J. Petykiewicz, L. Su, and J. Vučković, "Fabrication-constrained nanophotonic inverse design," Sci. Rep. **7**, 1786 (2017).
61. C. Dory, D. Vercruysse, K. Y. Yang, N. V. Sapra, A. E. Rugar, S. Sun, D. M. Lukin, A. Y. Piggott, J. L. Zhang, M. Radulaski, K. G. Lagoudakis, L. Su, and J. Vučković, "Inverse-designed diamond photonics," Nat. Commun. **10**, 3309 (2019).
62. J. Lu and J. Vučković, "Nanophotonic computational design," Opt. Express **21**, 13351 (2013).
63. N. V. Sapra, D. Vercruysse, L. Su, K. Y. Yang, J. Skarda, A. Y. Piggott, and J. Vuckovic, "Inverse design and demonstration of broadband grating couplers," IEEE J. Sel. Top. Quantum. **25**, 6100207 (2019).
64. A. Y. Piggott, J. Lu, K. G. Lagoudakis, J. Petykiewicz, T. M. Babinec, and J. Vučković, "Inverse design and demonstration of a compact and broadband on-chip wavelength demultiplexer," Nat. Photon. **9**, 374 (2015).
65. L. Su, A. Y. Piggott, N. V. Sapra, J. Petykiewicz, and J. Vučković, "Inverse design and demonstration of a compact on-chip narrowband three-channel wavelength demultiplexer," ACS Photon. **5**, 301 (2017).
66. N. V. Sapra, K. Y. Yang, D. Vercruysse, K. J. Leedle, D. S. Black, R. J. England, L. Su, R. Trivedi, Y. Miao, O. Solgaard, R. L. Byer, and J. Vučković, "On-chip integrated laser-driven particle accelerator," Science **367**, 79 (2020).
67. K. Y. Yang, J. Skarda, M. Cotrufo, A. Dutt, G. H. Ahn, M. Sawaby, D. Vercruysse, A. Arbabian, S. Fan, A. Alù, and J. Vučković, "Inverse-designed non-reciprocal pulse router for chip-based LiDAR," Nat. Photon. **14**, 369 (2020).
68. P. Camayd-Muñoz, G. Roberts, C. Ballew, M. Debbas, and A. Faraon, "Inverse designed shape-reconfigurable multifunctional photonics," in *CLEO* (2020), paper FW3B.2.
69. J.-K. Byun, J.-H. Ko, H.-B. Lee, J.-S. Park, and H.-S. Kim, "Application of the sensitivity analysis to the optimal design of the microstrip low-pass filter with defected ground structure," IEEE Trans. Mag. **45**, 1462 (2009).
70. J. Lehtinen, "A framework for precomputed and captured light transport," ACM Trans. Graphics **26**, 13 (2007).
71. J. C. Finlay and T. H. Foster, "Recovery of hemoglobin oxygen saturation and intrinsic fluorescence with a forward-adjoint model," Appl. Opt. **44**, 1917 (2005).
72. S. Forest, "Genetic algorithms-principles of natural-selection applied to computation," Science **261**, 872 (1993).
73. G. Pu, L. Yi, L. Zhang, and W. Hu, "Intelligent programmable mode-locked fiber laser with a human-like algorithm," Optica **6**, 362 (2019).
74. Z. Yu, H. Cui, and X. Sun, "Genetic-algorithm-optimized wideband on-chip polarization rotator with an ultrasmall footprint," Opt. Lett. **42**, 3093 (2017).
75. A. C. S. Chan, A. K. S. Lau, K. K. Y. Wong, E. Y. Lam, and K. K. Tsia, "Arbitrary two-dimensional spectrally encoded pattern generation—a new strategy for high-speed patterned illumination imaging," Optica **2**, 1037 (2015).
76. D. J. Poxson, M. F. Schubert, F. W. Mont, E. F. Schubert, and J. K. Kim, "Broadband omnidirectional antireflection coatings optimized by genetic algorithm," Opt. Lett. **34**, 728 (2009).
77. Z. Yu, H. Ccui, and X. Sun, "Optimal design of integrally gated CNT field-emission devices using a genetic algorithm," Photon. Res. **5**, B15 (2017).
78. L. Sanchis, A. Håkansson, D. López-Zanón, J. Bravo-Abad, and J. Sánchez-Dehesa, "Genetically optimized on-chip wideband ultracompact reflectors and Fabry–Perot cavities," Appl. Phys. Lett. **84**, 4460 (2004).
79. P. Y. Chen, C. H. Chen, J. S. Wu, H. C. Wen, and W. P. Wang, "Integrated optical devices design by genetic algorithm," Nanotechnology **18**, 395203 (2007).
80. M. Liu, Y. Xie, T. Feng, and Y. Xu, "Resonant broadband unidirectional light scattering based on genetic algorithm," Opt. Lett. **45**, 968 (2020).
81. K. Liao, T. Gan, X. Hu, and Q. Gong, "AI-assisted on-chip nanophotonic convolver based on silicon metasurface," Nanophotonics **9**, 3315 (2020).
82. Z. Lin, X. Li, R. Zhao, X. Song, Y. Wang, and L. Huang, "High-efficiency Bessel beam array generation by Huygens metasurfaces," Nanophotonics **8**, 1079 (2019).
83. Y. Khorrami, D. Fathi, and R. C. Rumpf, "Fast optimal design of optical components using the cultural algorithm," Opt. Express **28**, 15954 (2020).
84. N. Padhye, "Topology optimization of compliant mechanism using multi-objective particle swarm optimization," in *Proceedings of the 10th Annual Conference Companion on Genetic and Evolutionary Computation* (2008), p. 1831.
85. J. F. Schutte, J. A. Reinbolt, B. J. Fregly, R. T. Haftka, and A. D. George, "Parallel global optimization with the particle swarm algorithm," Int. J. Numer. Meth. Eng. **61**, 2296 (2004).
86. M. Djavid, S. A. Mirtaheri, and M. S. Abrishamian, "Photonic crystal notch-filter design using particle swarm optimization theory and finite-difference time-domain analysis," J. Opt. Soc. Am. B **26**, 849 (2009).
87. C. Forestiere, M. Donelli, G. F. Walsh, E. Zeni, G. Miano, and L. Dal Negro, "Particle-swarm optimization of broadband nanoplasmonic arrays," Opt. Lett. **35**, 133 (2010).
88. E. T. F. Rogers, J. Lindberg, T. Roy, S. Savo, J. E. Chad, M. R. Dennis, and N. I. Zheludev, "A super-oscillatory lens optical microscope for subwavelength imaging," Nat. Mater. **11**, 432 (2012).
89. M. Passoni, D. Gerace, L. Carroll, and L. C. Andreani, "Grating couplers in silicon-on-insulator: the role of photonic guided resonances on lineshape and bandwidth," Appl. Phys. Lett. **110**, 41107 (2017).
90. J. C. C. Mak, C. Sideris, J. Jeong, A. Hajimiri, and J. K. S. Poon, "Binary particle swarm optimized 2 × 2 power splitters in a standard foundry silicon photonic platform," Opt. Lett. **41**, 3868 (2016).
91. Y. Ha, Y. Guo, M. Pu, F. Zhang, X. Li, X. Ma, M. Xu, and X. Luo, "Minimized two- and four-step varifocal lens based on silicon photonic integrated nano-apertures," Opt. Express **28**, 7943 (2020).
92. B. Wohlfeil, L. Zimmermann, and K. Petermann, "Optimization of fiber grating couplers on SOI using advanced search algorithms," Opt. Lett. **39**, 3201 (2014).
93. N. Zhu, H. J. Sun, and N. F. Zhou, "Ant colony optimization for dynamic RWA in WDM networks with partial wavelength conversion," Photon. Network Commun. **11**, 229 (2006).

94. I. Saouane, A. Chaker, B. Zaidi, and C. Shekhar, "Optimal angle of polycrystalline silicon solar panels placed in a building using the ant colony optimization algorithm," Eur. Phys. J. Plus. **132**, 106 (2017).
95. X. Guo, H. Y. Zhou, S. Guo, X. X. Luan, W. K. Cui, Y. F. Ma, and L. Shi, "Design of broadband omnidirectional antireflection coatings using ant colony algorithm," Opt. Express **22**, A1137 (2014).
96. M. Lundy, "Convergence of an annealing algorithm," Math. Programm. **34**, 111 (1986).
97. M. Čepin, *Assessment of Power System Reliability* (Springer-Verlag London Limited, 2011).
98. A. Basu and L. N. Fraze, "Rapid determination of the critical temperature in simulated annealing inversion," Science **249**, 1409 (1990).
99. K. Hara, T. Iwamoto, and K. Kyuma, "Optimum design technique for optoelectronic devices using simulated annealing," Electron. Commun. Japan **79**, 22 (1996).
100. L. D. Khalaf and A. F. Peterson, "Performance of the simulated annealing and genetic algorithms for the design of periodic devices," Int. J. Microwave Millimeter-Wave Comput. Aided Eng. **7**, 108 (1997).
101. Y. T. Lu and Y. Q. Zhou, "Design of multilayer microwave absorbers using hybrid binary lightning search algorithm and simulated annealing," Photon. Network Commun. **78**, 75 (2017).
102. Z. Xie, T. Lei, F. Li, H. Qiu, Z. Zhang, H. Wang, C. Min, L. Du, Z. Li, and X. Yuan, "Ultra-broadband on-chip twisted light emitter for optical communications," Light: Sci. Appl. **7**, 18001 (2018).
103. Z. Xie, T. Lei, H. Qiu, Z. Zhang, H. Wang, and X. Yuan, "Broadband on-chip photonic spin Hall element via inverse design," Photon. Res. **8**, 121 (2020).
104. S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach* (Prentice-Hall, 2003).
105. T. Lin, F. Tian, P. Shi, F. S. Chau, G. Zhou, X. Tang, and J. Deng, "Design of mechanically-tunable photonic crystal split-beam nanocavity," Opt. Lett. **40**, 3504 (2015).
106. Q. Quan, P. B. Deotare, and M. Loncar, "Photonic crystal nanobeam cavity strongly coupled to the feeding waveguide," Appl. Phys. Lett. **96**, 203102 (2010).
107. E. Tabli, *Metaheuristics: From Design to Implementation* (Wiley, 2009).
108. H. Youssef, S. M. Sait, and H. Adiche, "Evolutionary algorithms, simulated annealing and tabu search: a comparative study," Eng. Appl. Artif. Intel. **14**, 167 (2001).
109. D. Gagnon, J. Dumont, and L. J. Dubé, "Multiobjective optimization in integrated photonics design," Opt. Lett. **38**, 2181 (2013).
110. D. Gagnon, J. Dumont, J. Déziel, and L. J. Dubé, "Optimization of integrated polarization filters," Opt. Lett. **39**, 5768 (2014).
111. M. A. Seldowitz, J. P. Allebach, and D. W. Sweeney, "Synthesis of digital holograms by direct binary search," Appl. Opt. **26**, 2788 (1987).
112. B. B. Chhetri, S. Yang, and T. Shimomura, "Stochastic approach in the efficient design of the direct-binary-search algorithm for hologram synthesis," Appl. Opt. **39**, 5956 (2000).
113. B. Shen, P. Wang, R. Polson, and R. Menon, "An integrated-nanophotonics polarization beamsplitter with $2.4 \times 2.4\ \mu m^2$ footprint," Nat. Photon. **9**, 378 (2015).
114. B. Shen, P. Wang, and R. Menon, "Optimization and analysis of 3D nanostructures for power-density enhancement in ultra-thin photovoltaics under oblique illumination," Opt. Express **22**, A311 (2014).
115. B. Shen, P. Wang, R. Polson, and R. Menon, "Ultra-high-efficiency metamaterial polarizer," Optica **1**, 356 (2014).
116. P. Wang and R. Menon, "Optimization of periodic nanostructures for enhanced light-trapping in ultra-thin photovoltaics," Opt. Express **21**, 6274 (2013).
117. G. Kim, J. A. Dominguez-Caballero, and R. Menon, "Design and analysis of multi-wavelength diffractive optics," Opt. Express **20**, 2814 (2012).
118. G. Kim, J. A. Dominguez-Caballero, H. Lee, D. J. Friedman, and R. Menon, "Increased photovoltaic power output via diffractive spectrum separation," Phys. Rev. Lett. **110**, 123901 (2013).
119. T. R. Sales and D. H. Raguin, "Multiwavelength operation with thin diffractive elements," Appl. Opt. **38**, 3012 (1999).
120. B. Shen, P. Wang, R. Polson, and R. Menon, "Integrated metamaterials for efficient and compact free-space-to-waveguide coupling," Opt. Express **22**, 27175 (2014).
121. G. Kim and R. Menon, "An ultra-small three dimensional computational microscope," Appl. Phys. Lett. **105**, 061114 (2014).
122. Y. Liu, S. Wang, Y. Wang, and W. Liu, "Subwavelength polarization splitter–rotator with ultra-compact footprint," Opt. Lett. **44**, 4495 (2019).
123. A. Majumder, B. Shen, R. Polson, T. Andrew, and R. Menon, "An ultra-compact nanophotonic optical modulator using multi-state topological optimization," arXiv:1712.02835 (2017).
124. B. Shen, R. Polson, and R. Menon, "Increasing the density of passive photonic-integrated circuits via nanophotonic cloaking," Nat. Commun. **7**, 13126 (2016).
125. H. Xie, Y. Liu, S. Wang, Y. Wang, Y. Yao, Q. Song, J. Du, Z. He, and K. Xu, "Highly compact and efficient four-mode multiplexer based on pixelated waveguides," IEEE Photon. Tech. Lett. **32**, 166 (2020).
126. Y. Liu, K. Xu, S. Wang, W. Shen, H. Xie, Y. Wang, S. Xiao, Y. Yao, J. Du, Z. He, and Q. Song, "Arbitrarily routed mode-division multiplexed photonic circuits for dense integration," Nat. Commun. **10**, 3263 (2019).
127. Y. Liu, W. Sun, H. Xie, N. Zhang, K. Xu, Y. Yao, S. Xiao, and Q. Song, "Very sharp adiabatic bends based on an inverse design," Opt. Lett. **43**, 2482 (2018).
128. H. Xie, Y. Liu, Y. Wang, Y. Wang, Y. Yao, Q. Song, J. Du, Z. He, and K. Xu, "An ultra-compact 3-dB power splitter for three modes based on pixelated meta-structure," IEEE Photon. Tech. Lett. **32**, 341 (2020).
129. H. Ma, J. Huang, K. Zhang, and J. Yang, "Arbitrary-direction, multichannel and ultra-compact power splitters by inverse design method," Opt. Commun. **462**, 125329 (2020).
130. Y. Deng, Z. Liu, C. Song, P. Hao, Y. Wu, Y. Liu, and J. G. Korvink, "Topology optimization of metal nanostructures for localized surface plasmon resonances," Struct. Multidiscip. Optim. **53**, 967 (2016).
131. J. Huang, J. Yang, D. Chen, W. Bai, J. Han, Z. Zhang, J. Zhang, X. He, Y. Han, and L. Liang, "Implementation of on-chip multi-channel focusing wavelength demultiplexer with regularized digital metamaterials," Nanophotonics **9**, 159 (2020).
132. Y. Chen, Y. Hu, J. Zhao, Y. Deng, Z. Wang, X. Cheng, D. Lei, Y. Deng, and H. Duan, "Topology optimization-based inverse design of plasmonic nanodimer with maximum near-field enhancement," Adv. Funct. Mater. **30**, 2000642 (2020).
133. C. Y. Kao, S. Osher, and E. Yablonovitch, "Maximizing band gaps in two-dimensional photonic crystals by using level set methods," Appl. Phys. B **81**, 235 (2005).
134. S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations," J. Comput. Phys. **79**, 12 (1988).
135. F. Stanley and J. Osher, "Level set methods for optimization problems involving geometry and constraints," J. Comput. Phys. **171**, 1 (2001).
136. A. K. S. Heinrich and H. Niederreiter, *Monte Carlo and Quasi-Monte Carlo Methods* (Springer, 2006).
137. K. Binder, *Applications of the Monte Carlo method in Statistical Physics* (Springer, 1987).
138. R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo Method* (Wiley, 2008).
139. Y. Zhao, X. Cao, J. Gao, Y. Sun, H. Yang, X. Liu, Y. Zhou, T. Han, and W. Chen, "Broadband diffusion metasurface based on a single anisotropic element and optimized by the Simulated Annealing algorithm," Sci. Rep. **6**, 23896 (2016).
140. L. Baranyai, "Laser induced diffuse reflectance imaging – Monte Carlo simulation of backscattering measured on the surface," MethodsX **7**, 100958 (2020).