Materials and Design 197 (2021) 109213

Contents lists available at ScienceDirect

Materials and Design

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

A directional Gaussian smoothing optimization method for computational inverse design in nanophotonics

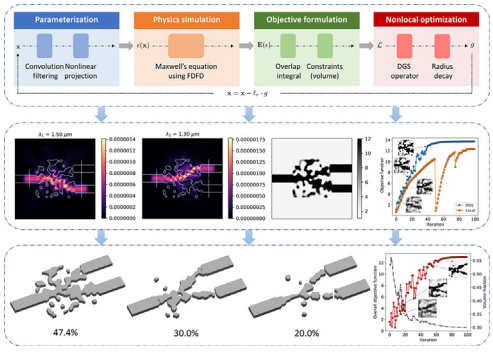
Jiaxin Zhang [a,*], Sirui Bi [b,c], Guannan Zhang [a]

[a] Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37830, USA
[b] Department of Civil and Systems Engineering, Johns Hopkins University, Baltimore, MD 21218, USA
[c] Computational Science and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37830, USA

HIGHLIGHTS

- A novel nonlocal optimization method for inverse design is proposed
- This approach employs directional Gaussian smoothing (DGS) to capture the nonlocal gradient and global structure of loss landscapes
- Dynamic mechanisms with adaptive decay improve the convergence and robustness in nonlocal optimization
- The methodology demonstrates higher performance and lower uncertainty compared with local gradient approaches given random initialization with noise
- The DGS gradient is combined with MMA optimizer for conducting nonlocal constrained optimization with a limited amount of material usage

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 18 July 2020
Received in revised form 20 September 2020
Accepted 7 October 2020
Available online 20 October 2020

Keywords:
Inverse design
Nanophotonics
Constrained optimization
Directional Gaussian smoothing
Robustness
Fabrication constraint

ABSTRACT

Local-gradient-based optimization approaches lack nonlocal exploration ability required for escaping from local minima in non-convex landscapes. A directional Gaussian smoothing (DGS) approach was proposed in our recent work and used to define a truly nonlocal gradient, referred to as the DGS gradient, in order to enable nonlocal exploration in high-dimensional black-box optimization. Promising results show that replacing the traditional local gradient with the nonlocal DGS gradient can significantly improve the performance of gradient-based methods in optimizing highly multi-modal loss functions. However, the current DGS method is designed for unbounded and unconstrained optimization problems, making it inapplicable to real-world engineering design optimization problems where the tuning parameters are often bounded and the loss function is usually constrained by physical processes. In this work, we propose to extend the DGS approach to the constrained inverse design framework in order to find a better design. The proposed framework has its advantages in portability and flexibility to naturally incorporate the parameterization, physics simulation, and objective formulation together to build up an effective inverse design workflow. A series of adaptive strategies for smoothing radius and learning rate updating are developed to improve the computational efficiency and robustness. To enable a clear binarized design, a dynamic growth mechanism is imposed on the projection strength in parameterization. The methodology is demonstrated by an example of wavelength demultiplexer. Our method shows superior performance compared to the state-of-the-art approaches. By incorporating volume constraints, the optimized design achieves an equivalently high performance but significantly reduces the amount of material usage.

* Corresponding author at: Computer Science and Mathematics Division, Oak Ridge
E-mail addresses: zhangj@ornl.gov, jiaxin.zhang@jhu.edu (J. Zhang).

https://doi.org/10.1016/j.matdes.2020.109213

1. Introduction

Photonic devices have been applied in a wide range of applications, including photonic integrated circuits [1], optical interconnects and sensors [2,3], augmented reality (AR) [4] and quantum computing [5]. As a growing number of applications in nanophotonic devices, photonic design is becoming increasingly demanding and challenging to optimize the device performance [6–9]. Classical design approaches based on analytical theory and intuition, however, are limited in small design space and relatively simple parameter tuning by hand. Capitalizing on the increased degrees of freedom in design space, nanophotonic devices have been designed with novel functionalities, high performance, efficiencies, and robustness that have been proven difficult to implement in traditional intuition-based methods [10].

There recently have been significant interests in using *computational inverse design* approaches to explore the full design space of novel photonic devices with a broad variety of applications [11–21]. Much of this progress is made by the gradient-based algorithm, which is a promising method to efficiently search the enormous degrees of freedom in high-dimensional design space. The gradient-based optimization typically relies on adjoint method [22,23], which is a technique that enables the local gradient of an objective function and constraints to be calculated with respect to arbitrarily large design variables using forward physical simulation such as electromagnetic simulations. To this end, several recent studies further use automatic differentiation and backpropagation tools that are beneficial from machine learning research, to efficiently evaluate the local gradient by reducing the number of simulations [11,17,24,25]. These approaches make a feasible gradient-based design of photonic structures, particularly nanophotonic devices, with better efficiency and smaller footprints than traditional devices. However, these approaches basically depend on an efficient estimation or analytical derivation of the gradient (or called sensitivity analysis). Typically, several assumptions are often made for simplicity to utilize gradient-based optimization in the context of electromagnetism. For example, electromagnetism is modeled using Maxwell's equations assuming statics, linear, homogeneous, and isotropic materials as well as time-harmonic behavior of the field and transverse electric and magnetic problems with material invariance in the polarization direction [26,27]. All of these assumptions lead to a large challenge in real-world photonic design with dynamic, nonlinear, and dispersive material properties in complex multiphysics conditions. In addition, for complicated non-convex objective functions and constraints, gradient estimation relies on the adjoint method may be either not easily accessible or unreliable. Sometimes, additional efforts are required to derive the sensitivity analysis if unusual objectives or constraints are incorporated into the optimization formulation even though the existing gradient-based scheme has been used in device design.

Another important challenge is that, up to now, most of the studies use local gradient-based approaches for inverse design so that the optimized devices converge to a *local minimum* with respect to the design parameters. In many electromagnetic design problems, their landscapes have been proven to be highly nonlinear and non-convex such that many possible local minima exist [7,11]. These local minima depend on the initialization and vary largely as the initial guess change. These challenges in gradient-based approaches have attracted much attention [11,28–30]. In practice, one common way is to run an optimization several times with different initial guesses that provide a rough estimate of the device performance. However, this approach has limitations in maximizing device performance and computational efficiency, and meanwhile, it possibly gives rise to a large variation of design performance. Alternatively, several evolutionary algorithms, including Generic Algorithm (GA), Particle Swarm Optimization (PSO) and Simulated Annealing (SA) are used to explore the global minima but finding the optimal solution to complex high-dimensional, multimodal problems often converges very slowly and requires very expensive fitness function evaluations [31]. Some recent studies optimize the photonic device performance using derivative-free methods, including Bayesian optimization (BO) [32] and differential evolution (DE) algorithms [31], but these methods have limitations in scaling to high-dimensional problems [33,34] in photonic device design. Therefore, it is necessary to develop an efficient optimization algorithm that can also escape from local minima in non-convex, high-dimensional landscape, to overcome the challenges in the local gradient-based approaches, evolutionary algorithms and derivative-free global optimization methods.

In principle, it is feasible to optimize the photonic devices directly by changing the value of permittivity distribution at every point. However, it is more critical to impose fabrication constraints into optimization workflow because a fundamental challenge in nanophotonic device design is that arbitrary permittivity distribution, such as very tiny feature and grey-scale value, can not be fabricated in practice [23,35,36]. The difficulty is often addressed by choosing an appropriate parameterization via a series of transformations that are simply an operation that affects the state of optimization. The use of transformation allows different parameterization and optimization stages to be easily swapped in and out for one another. This requires the optimizer has the capability of naturally integrating transformation into the design process. Another common constraint in optimizing material layout is the material usage (or volume fraction) in device design. In other words, it is desirable to use fewer materials but able to achieve performance that is as good as the target. To the author's knowledge, relatively few studies have accounted for the problem of inverse design with volume constraint in nanophotonic design. One possible solution is to add a penalty term into objective function and thus convert the unconstrained optimization to constrained optimization through the augmented Lagrangian formalism. However, the penalty coefficient is very sensitive and typically difficult to control in practical implementation. Although recent advances in stochastic optimization algorithms, e.g., SGD, Adam, and RMSProp, have attracted much attention and widely used in machine learning training, it is a non-trivial task to incorporate multiple equality or inequality constraints into these stochastic methods that mainly focus on unconstrained optimization problems.

To address these challenges, we propose a nonlocal inverse design workflow by incorporating the nonlocal gradient that was recently developed in [47]. The nonlocal gradient was defined by directional Gaussian smoothing, thus it is referred to as the DGS gradient hereinafter. The DGS gradient conducts 1D nonlocal explorations along with d orthogonal directions and each of which defines a nonlocal directional derivative as a 1D integral. The d directional derivatives are assembled to the DGS gradient. The Gauss-Hermite (GH) quadrature is used to approximate the 1D integrals (i.e., the directional derivatives) to achieve higher accuracy than Monte Carlo (MC) sampling. We improved the existing DGS approach from two perspectives in the context of inverse design. First, we established a workflow in which the DGS gradient can be combined with a variety of constraints, e.g., fabrication constraints and materials usage constraint in the practical material design. Second, we developed a series of adaptive strategies for the smoothing radius and the learning rate in order to improve computational efficiency and robustness. Compared to the local gradient, the directional smoothing allows for a large smoothing radius to capture the global structure of loss landscapes and thus provide a strong nonlocal exploration capability for escaping from local minima in non-convex landscapes. Furthermore, our workflow does not rely on the sensitivity analysis with multiple

assumptions, so that it has wider feasibility to nonlinear, dynamic, and non-isotropic materials under complex physical conditions. In the meantime, our method having the benefits of gradient-based optimization can be easily scaled to high-dimensional design spaces, which alleviates the challenges in derivative-free global optimization, such as Bayesian optimization.

The paper is structured as follows. Section 2 provides a brief mathematical formulation and overview of inverse design. Section 3 presents the DGS gradient operator in principle and explain how DGS gradient optimization can overcome the challenges and difficulties in the local gradient optimization approaches. In Section 4, we show an example of designing wavelength demultiplexers, explain the implementation of the nonlocal optimization method using the DGS gradient in detail, and demonstrate the strength and advantages via the discussion and comparison. After that, we provide a discussion section to address the implementation concern and current limitations of the proposed method. Finally, we provide a brief conclusion and discussion of future work.

2. Mathematical formulation of inverse design

This section provides a brief overview of the mathematical foundations behind the inverse design in photonic devices. Although the exact optimization problem may vary from case to case, the photonic design generally shares a similar set of features and steps, which include formulating an optimization problem, incorporating fabrication constraints and parameterization, and solving the inverse problem by optimization.

2.1. Optimization problem formulation

A general electromagnetic design problem can be cast into the following optimization formulation:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{E}_1, \ldots, \mathbf{E}_n, \boldsymbol{\varepsilon}_1, \ldots, \boldsymbol{\varepsilon}_n, \mathbf{x}) \\ \text{subject to} \quad & g_j(\mathbf{x}) = 0, \quad j = 1, \ldots, m \\ & h_k(\mathbf{x}) \le 0, \quad k = 1, \ldots, l \end{aligned} \tag{1}$$

where $\mathbf{E}_i$ is the electric field corresponding to the permittivity distribution $\boldsymbol{\varepsilon}_i$, which depends on a parameterization vector $\mathbf{x} \in \mathbb{R}$ which is the computational design domain, and f is the objective function that defines the target of the optimization. A typical objective is to maximize the transmission, which is equivalent to minimize the negative

$$f_{obj}(\mathbf{x}) = -\left|\mathbf{c}^{\dagger}\mathbf{E}(\boldsymbol{\varepsilon}(\mathbf{x}))\right|^2 \tag{2}$$

where $\mathbf{c}^{\dagger}\mathbf{E}$ means the overlap integrals to compute the model coupling efficiency of the electric field $\mathbf{E}$ with the target mode at the output. $h_k(\mathbf{x})$ and $g_j(\mathbf{x})$ in Eq. (1) are inequality and equality constraints on $\mathbf{x}$, particularly fabrication and volume of materials constraints, and the index k and j mean the number of inequality and equality constraints respectively. For the optimization problem in Eq. (1), the electric fields $\mathbf{E}_i$ generated by the input permittivity distribution $\boldsymbol{\varepsilon}(\mathbf{x})$ should satisfy the Maxwell's equations in the frequency domain,

$$\nabla \times \frac{1}{\mu}\nabla \times \mathbf{E}_i - \omega_i^2 \boldsymbol{\varepsilon}(\mathbf{x})\mathbf{E}_i = -i\omega_i \mathbf{J}_i \tag{3}$$

where $i = 1, \ldots, n$ is the input modes, ω_i is the angular frequency, μ is the magnetic permeability of free space and $\mathbf{J}_i$ is the input source which injects the current mode into the input waveguide. Eq. (3) is often solved by electromagnetic simulation using the finite-difference frequency-domain (FDFD) method [37] or finite-difference time-domain (FDTD) method [38]. Typically, the perfectly matched layer (PML) boundary condition as an artificial absorbing layer for wave equations, is used to truncate computational regions in numerical methods to simulate problems with open boundaries [39]. In short, the computational inverse design problem can be addressed by solving the optimization problem in Eq. (1), which is to find optimal $\mathbf{x}$ to minimize the objective function, defined by Eq. (2), subject to Maxwell's equations in Eq. (3), and fabrication constraints $g_j(\mathbf{x})$ and $h_k(\mathbf{x})$.

2.2. Parameterization and constraints in optimization

Solving the optimization problem defined in Eq. (1) led to continuously varying features of $\boldsymbol{\varepsilon}(\mathbf{x})$, which is difficult for fabricating devices in practice. This is because the fabricated devices are typically composed of distinct materials so the permittivity can only take on certain discrete values and must keep the same along the vertical direction in fabrication with top-down lithography. Minimum feature size is another essential fabrication constraint. It is therefore critical to describe the permittivity distribution through a *parameterization* that addresses the fabrication challenges in device design [11].

Parameterization basically consists of two key components: *projection* operator and *filtering* operator. Projection operator aims to convert the continuous features to a binary feature that better captures a clear "0–1" design, where "0" represents a *background* material and "1" represents a *foreground* material in permittivity distribution. This can be achieved by defining an operator through the equation

$$\varepsilon(\mathbf{x}) = \varepsilon_b(\mathbf{x}) + \mathrm{H}(\varphi(\mathbf{x})) \tag{4}$$

where $\boldsymbol{\varepsilon}_b(\mathbf{x})$ is a permittivity background (constant) and $\varphi(\mathbf{x})$ is a 2D slice of the permittivity distribution and ranges from 0 to 1. A possible projection operator $\mathbb{H}$ is using nonlinear penalty methods [13,22]. Filtering operator is often used to eliminate very tiny features and avoid to the formation of checker-board pattern in material layout [40]. For example, level set methods [41,42] construct a fabrication constraint penalty function for geometry representation of the devices. Using an appropriate parameterization, the fabrication constraints $h_k(\mathbf{x})$ can be imposed and naturally perform a binary device design.

Another common constraint in optimizing material layout is the volume fraction of material usage, which is defined by

$$h_1(\mathbf{x}) = V(\mathbf{x})/V_0 - \gamma \le 0 \tag{5}$$

where V and V_0 are the expected material volume and design domain volume respectively, and γ is the specific volume fraction. A simple solution of incorporating volume constraints into optimization is to add penalty terms into objective function and thus convert to unconstrained optimization so that several algorithms, for example, gradient descent, Adam, etc. can be used. However, the penalty coefficient, in fact, is very sensitive and difficult to determine in practical implementation.

2.3. Solving the inverse design problem

The inverse design problem can be defined to find the best permittivity distribution $\boldsymbol{\varepsilon}$ and the corresponding electric field $\mathbf{E}_i$ to maximize the device performance described by the objective function in Eq. (1) and simultaneously satisfy the physics constraints in Eq. (3), fabrication constraints in Eq. (4) and material volume constraints in Eq. (5). It is a challenging task to solve this kind of constrained optimization problem that involves large-scale, high-dimensional design degrees of freedom and a highly non-convex and non-linear landscape [11,43]. Many recent efforts have been made to develop gradient-based optimization techniques for addressing the challenges [11,44].

To perform gradient-based optimization, the gradient $df_{obj}/d\mathbf{x}$ is required. Note that $\mathbf{E}_i$ and $\boldsymbol{\varepsilon}_i$ can be complex-valued, where i means the i-th mode. Suppose the objective function f is real-valued, the gradient can be computed by

$$\frac{df_{obj}}{d\mathbf{x}} = \frac{\partial f_{obj}}{\partial \mathbf{x}} + 2\mathrm{R}\left[\sum_i \left(\frac{\partial f_{obj}}{\partial \mathbf{E}_i}\frac{d\mathbf{E}_i}{d\mathbf{x}} + \frac{\partial f_{obj}}{\partial \boldsymbol{\varepsilon}_i}\frac{d\boldsymbol{\varepsilon}_i}{d\mathbf{x}}\right)\right] \tag{6}$$

where $\mathscr{R}[\cdot]$ denotes taking the real part. The derivative terms $\partial f_{obj}/\partial \mathbf{E}_i$, $\partial f_{obj}/\partial \boldsymbol{\varepsilon}_i$ and $d\boldsymbol{\varepsilon}_i/d\mathbf{x}$ depend on the form of the objective function but $d\mathbf{E}_i/d\mathbf{x}$ is always required in electromagnetic simulation. It is therefore necessary to derive the gradient for FDFD or other simulation methods. Given the FDFD equation in Eq. (3), we can rewrite it by

$$(\Omega - \omega^2 \mathrm{diag}(\boldsymbol{\varepsilon}))\mathbf{E} = -i\omega\mathbf{J} \tag{7}$$

where Ω is the *discretized* version of the $\nabla \times \mu^{-1} \nabla \times$ operator. Differentiating by through by ε, we have

$$\Omega\frac{d\mathbf{E}}{d\boldsymbol{\varepsilon}} - \left[\frac{d\mathbf{E}}{d\boldsymbol{\varepsilon}}\omega^2\mathrm{diag}(\boldsymbol{\varepsilon}) + \omega^2\mathrm{diag}(\mathbf{E})\right] = 0. \tag{8}$$

If we rearrange the Eq. (8), we have

$$(\Omega - \omega^2\mathrm{diag}(\boldsymbol{\varepsilon}))\frac{d\mathbf{E}}{d\varepsilon} = \omega^2\mathrm{diag}(\mathbf{E}) \tag{9}$$

and the simulation gradient $d\mathbf{E}_i/d\mathbf{x}$ is therefore derived by

$$\frac{d\mathbf{E}_i}{d\mathbf{x}} = \frac{d\mathbf{E}_i}{d\boldsymbol{\varepsilon}_i}\frac{d\boldsymbol{\varepsilon}_i}{d\mathbf{x}} = (\Omega - \omega_i^2\mathrm{diag}(\boldsymbol{\varepsilon}_i))^{-1}\omega_i^2\mathrm{diag}(\mathbf{E}_i)\frac{d\boldsymbol{\varepsilon}_i}{d\mathbf{x}}. \tag{10}$$

Note that, the computing in Eq. (10) is often computationally intensive because it requires the same number of electromagnetic simulations as the number of design degrees of freedom. Thanks to the development of automatic differentiation techniques in machine learning [45,46], efficient implementation relies on automatic differentiation and backpropagation are introduced to reduce the computational cost [11,24,44].

3. The DGS optimization method

In this section, we describe the DGS gradient operator that was recently developed in our previous work [47]. To better explain the direction Gaussian smoothing strategy, we briefly recall the standard Gaussian smoothing [48] for estimating local gradients. Specifically, it starts by defining a smoothed loss function

$$F_\sigma(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{N}(0,\mathbf{I}_d)}[F(\mathbf{x}+\sigma\mathbf{u})],$$

where $\mathcal{N}(0,\mathbf{I}_d)$ is the d-dimensional standard Gaussian distribution, and $\sigma > 0$ is the smoothing radius. $F_\sigma(\mathbf{x})$ inherits many characteristics from $F(\mathbf{x})$, e.g., convexity, the Lipschitz constant. Then, the gradient $\nabla F_\sigma(\mathbf{x})$ can be represented as an expectation and estimated by drawing M random samples $\{\boldsymbol{u}_m\}_{m=1}^M$ from $\mathcal{N}(0,\mathbf{I}_d)$, i.e.,

$$\nabla F_\sigma(\mathbf{x}) = \frac{1}{\sigma}\mathbb{E}_{\mathbf{u}\sim\mathcal{N}(0,\mathbf{I}_d)}[F(\mathbf{x}+\sigma\boldsymbol{u})\boldsymbol{u}] \approx \frac{1}{M\sigma}\sum_{m=1}^{M} F(\mathbf{x}+\sigma\boldsymbol{u}_m)\boldsymbol{u}_m. \tag{11}$$

The Monte Carlo (MC) estimator in Eq. (11) is substituted into any gradient-based algorithm to update the state $\mathbf{x}$. *The major drawback is that the error of the MC estimator in Eq. (11) is on the order of* $\varepsilon\tilde{\mathcal{O}}\left(d\sigma/\sqrt{M}\right)$. When the dimension d is large (e.g., on the order of thousands) and the computing budget (the upper bound of M) is given, practitioners often have to sacrifice a nonlocal smoothing effect (with a relatively big σ) that helps skipping local mimina to achieve a required accuracy. In other words, Eq. (11) is mostly used in the local regime with a small value for σ.

3.1. The nonlocal DGS gradient operator

The DGS gradient was developed to alleviate the above challenge with the standard Gaussian smoothing. The key idea behind the DGS gradient is to conduct 1D nonlocal explorations along d orthogonal directions in $\mathbb{R}^d$, each of which defines a nonlocal directional derivative as a 1D integral. The Gauss-Hermite quadrature, instead of MC sampling, is used to estimate the d 1D integrals to achieve high accuracy.

Specifically, we first define a 1D cross section of $F(\mathbf{x})$ as

$$G(y\,|\,\mathbf{x},\boldsymbol{\xi}) = F(\mathbf{x}+y\boldsymbol{\xi}), \quad y\in\mathrm{R},$$

where $\mathbf{x}$ is the current state of $F(\mathbf{x})$ and $\boldsymbol{\xi}$ is a unit vector in $\mathbb{R}^d$. The Gaussian smoothing of $G(y)$, denoted by $G_\sigma(y)$, is defined by

$$\begin{aligned} G_\sigma(y\,|\,\mathbf{x},\boldsymbol{\xi}) &:= \frac{1}{\sqrt{2\pi}}\int_{\mathrm{R}} G(y+\sigma v\,|\,\mathbf{x},\boldsymbol{\xi})\,\mathrm{e}^{-\frac{v^2}{2}}dv \\ &= \mathbb{E}_{v,\tilde{\mathcal{N}}(0,1)}[G(y+\sigma v\,|\,\mathbf{x},\boldsymbol{\xi})]. \end{aligned} \tag{12}$$

This is also the Gaussian smoothing of $F(\mathbf{x})$ along the direction $\boldsymbol{\xi}$ in the neighbourhood of $\mathbf{x}$. The derivative of $G_\sigma(y|\mathbf{x},\boldsymbol{\xi})$ at $y = 0$ can be represented by a 1D integral

$$\mathcal{D}[G_\sigma(0\,|\,\mathbf{x},\xi)] = \frac{1}{\sigma}\mathbb{E}_{v\sim\mathcal{N}(0,1)}[G(\sigma v\,|\,\mathbf{x},\xi)\,v], \tag{13}$$

where $\mathcal{D}[\cdot]$ denotes the differential operator. We emphasize that Eq. (13) is fundamentally different from the directional derivative of $F_\sigma(\mathbf{x})$, because $G_\sigma(0|\,\mathbf{x},\boldsymbol{\xi})$ only conducts the directional smoothing along $\boldsymbol{\xi}$. For a matrix $\boldsymbol{\Xi} := (\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_d)$ consisting of d orthonormal vectors, we can define d directional derivatives like those in Eq. (13) and assemble our DGS gradient as

$$\nabla_{\sigma,\Xi}[F](\mathbf{x}) := [\mathcal{D}[G_\sigma(0\,|\,\mathbf{x},\boldsymbol{\xi}_1)],\cdots,\mathcal{D}[G_\sigma(0\,|\,\mathbf{x},\boldsymbol{\xi}_d)]]\,\Xi, \tag{14}$$

where the orthogonal system Ξ and the smoothing radius σ can be adjusted during an optimization process.

The next step is to develop an accurate DGS estimator. We exploit that each component of $\nabla_{\sigma,\Xi}[F](\mathbf{x})$ only involves a 1D integral, such that the Gauss-Hermite quadrature rule [1,46] can be used to approximate the integrals with high accuracy (shown in Eq. (16)). By doing a simple change of variable in Eq. (13), the GH rule can be directly used to obtain the following estimator for each directional derivative $\mathcal{D}[G_\sigma(0\,|\,\mathbf{x},\boldsymbol{\xi})]$ in Eq. (13)

$$\widetilde{\mathcal{D}}^M[G_\sigma(0\,|\,\mathbf{x},\boldsymbol{\xi})] = \frac{1}{\sqrt{\pi}\sigma}\sum_{m=1}^{M} w_m F\left(\mathbf{x}+\sqrt{2}\sigma v_m\boldsymbol{\xi}\right)\sqrt{2}v_m, \tag{15}$$

where $\{v_m\}_{m=1}^M$ are the roots of the M-th order Hermite polynomial and $\{w_m\}_{m=1}^M$ are quadrature weights. Both v_m and w_m can be found online[1] or in [50]. Compared with MC sampling, the error of Eq. (15) can be bounded by

$$\left|\left(\widetilde{\mathcal{D}}^M - \mathcal{D}\right)[G_\sigma]\right| \le C\frac{M!\sqrt{\pi}}{2^M(2M)!}\sigma^{2M-1}, \tag{16}$$

where $M!$ is the factorial of M and the constant $C > 0$ is independent of M and σ. Applying the GH quadrature rule $\widetilde{\mathcal{D}}^M$ to each component of $\nabla_{\sigma,\Xi}[F](\mathbf{x})$ in Eq. (14), we define the following estimator:

$$\widetilde{\nabla}^M_{\sigma,\Xi}[F](\mathbf{x}) = \left[\widetilde{\mathcal{D}}^M[G_\sigma(0\,|\,\mathbf{x},\boldsymbol{\xi}_1)],\cdots,\widetilde{\mathcal{D}}^M[G_\sigma(0\,|\,\mathbf{x},\boldsymbol{\xi}_d)]\right]\Xi. \tag{17}$$

The DGS estimator has the following features:

[1] Nodes and weights for GH quadrature: https://keisan.casio.com/exec/system/1281195844

- *Nonlocality*: The directional smoothing allows for a large radius σ to capture global structures of loss landscapes and help escape from local minima.
- *Accuracy*: The GH quadrature with the error bounded in Eq. (16) provides an estimator having much higher accuracy than MC, even when a large smoothing radius σ is used.
- *Portability*: The DGS gradient can be integrated into a majority of gradient-based algorithms, e.g., gradient descent, Adam, and those with constraints.
- *Scalability*: The DGS estimator in Eq. (17) requires $M \times d$ evaluations of $F(\mathbf{x})$, and these evaluations are completely parallelizable as those in random sampling.

3.2. A mathematical example

To illustrate the performance of the DGS gradient, we combine the DGS gradient with the standard gradient descent algorithm to optimize the 1000D Ackley function, which is one of the benchmark functions used to test non-convex optimization algorithms [51,52]. The Ackley function is defined by

$$F(\mathbf{x}) = -a\exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d}x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cx_i)\right) + a + \exp(1), \tag{18}$$

where d is the dimension and $a = 20$, $b = 0.2$, $c = 2\pi$ are used in our experiments. The input domain $\mathbf{x} \in [-32.768, 32.768]$. The global minimum is $f(\mathbf{x}^*) = 0$, at $\mathbf{x}^* = (0, \ldots, 0)$. The Ackley function represents non-convex landscapes with nearly flat outer region. The function poses a risk for optimization algorithms, particularly hill-climbing algorithms, to be trapped in one of its many local minima. At each iteration, we update the state $\mathbf{x}_t$ to $\mathbf{x}_{t+1}$ by

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \lambda_t \widetilde{\nabla}^M_{\sigma_t,\Xi}[F](\mathbf{x}_t), \tag{19}$$

where $\Xi = \mathbf{I}_d$. The learning rate λ_t follows a polynomial decay schedule $\lambda_t = (\lambda_0 - \lambda_T)(1 - \frac{t}{T})^{\tau} + \lambda_T$ with $\lambda_0 = 4000$, $\lambda_T = 0.001$, $\tau = 4$, $T = 60$. The smoothing radius also follows a polynomial decay schedule $\sigma_t = (\sigma_0 - \sigma_T)(1 - \frac{t}{T})^{\nu} + \sigma_T$ with $\sigma_0 = 2.0$, $\sigma_T = 0.001$, $\nu = 2.0$. We compare our method with the standard Gaussian smoothing method (i.e., replacing $\widetilde{\nabla}^M_{\sigma_t,\Xi}[F](\mathbf{x}_t)$ with Eq. (11)), the BFGS method, and the finite difference method for estimating local gradients. The result is shown in Fig. 1. As shown in Fig. 1(Left), the Ackley function has many local minima which pose significant challenges for optimization. Fig. 1 (Right) shows that the DGS gradient exploited its nonlocal exploration ability to skip the local minima and converge to the global minimum. The other baseline methods do not converge because they are trapped in some local minima.

3.3. High-dimensional benchmark function demonstration

We further test the performance of DGS method on three 2000 dimensional benchmark functions for global optimization, i.e. rotated Ellipsoidal, rotated Ackely and rotated Schaffer. Their definitions and properties can be found in [53]. These rotated functions that are more challenging can be used to verify the performance and rotation-invariant property of the DGS method. To illustrate the merts of the DGS method, we provide a comparison between the DGS method and the state-of-the-art approaches on computational efficiency and accuracy through these benchmark high-dimensional functions. The compared optimization algorithms are listed as follows: (a) **ES-Bpop**: the standard OpenAI evolution strategy (ES) in [54] with a big population (i.e., using the same number of samples as DGS method), (b) **ASEBO**: Adaptive ES-Active Subspaces for Blackbox Optimization [55] with a populatioin of size $4 + 3\log(d)$ where d is the dimensionality, (c) **IPop-CMA**: the restart covariance matrix adaptation evolution strategy (CMA-ES [56]) with increased population size [57], (d) **Nesterov**: the random search method in [58], (e) **FD**: the classical central difference scheme, (f) **Cobyla**: the Constrained Optimization BY Linear Approximation algorithm [59], (g) **Powell**: the Powell's conjugate direction method [59], (h) **DE**: the Differential Evolution algorithm [34] and (i) **PSO**: the Particle Swarm Optimization algorithm [60].

Fig. 2 shows the performance comparison for 2000D rotated functions given a same number of function evaluations. For each function, we tested 5 random rotations and we run 5 trials with different random initial states, i.e., 25 trials in total. In Fig. 2, the solid lines represent the mean loss decay and the shadow areas cover the interval between the maximum and minimum loss values. The statistical confidence bounds reflect the effect of random initial states on the optimization performance. It is observed that the DGS method outperform all other baseline methods. In particular, the DGS method demonstrates significantly superior performance in optimizing the high non-convex and multi-modal functions (i.e., Ackley and Schaffer). This is because the two advantages of DGS: strong nonlocal exploration and smaller variation of gradient estimators.

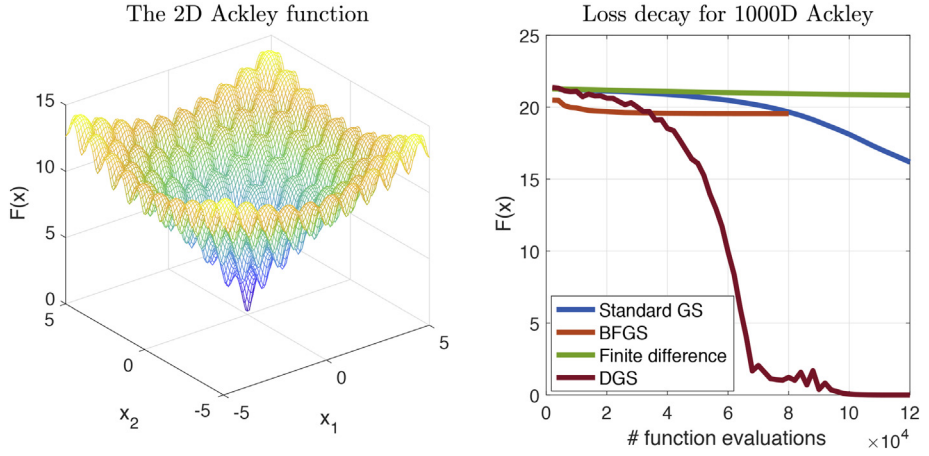


Fig. 1. (Left) The landscape of the 2D Ackley function that possesses many local minima. (Right) Comparison of the loss decay w.r.t. # function evaluations for the 1000D Ackley function. Each curve was generated by averaging 20 independent trials with random initial states. The global minimum is $F(\mathbf{x}) = 0$. DGS gradient successfully found the global minimum while the other baselines are trapped in local minima.

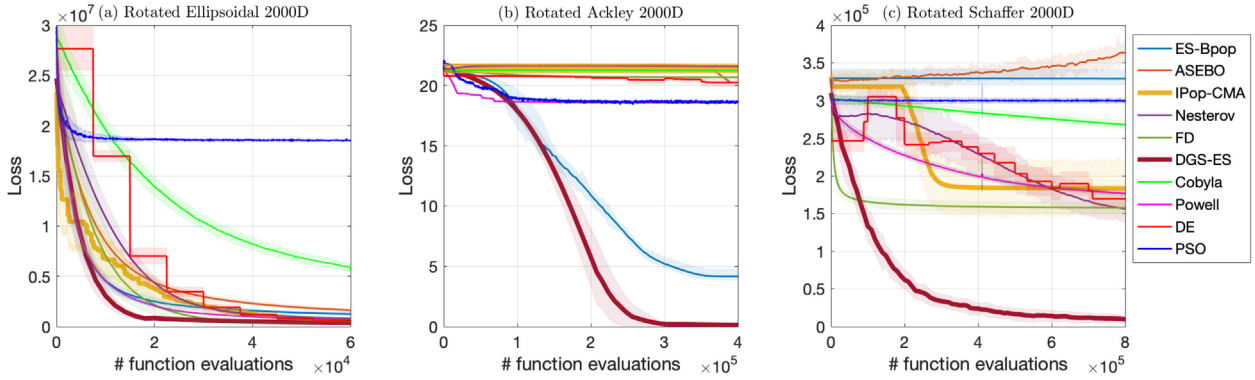


Fig. 2. Performance comparison of the loss decay with respect to the number of functional evaluations for the three benchmark functions in 2000-dimensional spaces. The global minimum is 0 for all the three functions.

4. Inverse design of wavelength demultiplexer

In this section, we use the DGS method to design a optimal wavelength demultiplexer in 3D. This example is a canonical benchmarking demonstration for inverse design in nanophotonic devices. We first provide a problem description with an objective function definition and parameterization scheme. Then a methodology workflow illustrates how to incorporate DGS optimization method for inverse design. Finally, we demonstrate the superior performance of the optimized device using the proposed method, investigate the effect of random initialization on robustness, and conduct constrained optimization with a limited amount of materials.

4.1. Problem description

As shown in Fig. 3, we choose a three-port structure with 500 nm input waveguide and output waveguides and a square 2.5 $\mu m \times 2.5$ μm design region. We design a device for the 220 nm silicon-on-insulator (SOI) platform where the structure is constrained to a single fully etched Si layer on a SiO_2 substrate with air cladding. For illustration, the refractive indices of $n_{air} = 1$, $n_{SiO2} = 1.45$ and $n_{Si} = 3.5$ are used. The purpose of inverse design is to separate 1300 nm signal to the upper waveguide and 1500 nm signal to the bottom waveguide.

In this example, the fundamental first-order mode of the input waveguide is used as the input mode for the inverse design, and the fundamental first-order modes of the two output waveguides are used as the output modes. Initially, the permittivity in design region is homogeneously distributed as shown in Fig. 4 (c) and the resulting electric field intensity E_{z1} at 1500 nm and E_{z2} at 1300 nm are calculated by FDFD simulations, as shown in Fig. 4 (a) and (b) respectively. To conduct the FDFD simulation, the computational domain of entire structure $\mathbf{x}$, as shown in Eq. (21) is discretized by 120 × 120 pixels and the design region is parametrized by 60 × 60 pixels, leading to the pixelated design. For ease of inverse design process, we define a relative permittivity $\boldsymbol{\varepsilon}_i$ with a minimum value $\boldsymbol{\varepsilon}_{\min} = 1.0$ (white color in Fig. 4 (c)) and maximum value $\boldsymbol{\varepsilon}_{\max} = 12.0$ (black color in Fig. 4 (c)). The initial permittivity distribution with $\boldsymbol{\varepsilon}_{\text{ini}} = 6.5$ are set up for the design region.

4.2. Objective function

We define the output modes of interest as ζ_1 and ζ_2 over output surface S. The device performance is then specified by the overlap integral, which is given by

$$\mathbf{c}^{\dagger}\mathbf{E}_i(\boldsymbol{\varepsilon}(\mathbf{x})) = \iint_S \zeta_i \cdot \mathbf{E}_i(\boldsymbol{\varepsilon}(\mathbf{x}))dS \tag{20}$$

where we use it to compute the mode coupling efficiency into each output mode. To achieve the goal of maximum overlap integral, the optimization problem is formulated as follows:

$$\begin{aligned} \min_{\mathbf{x}} \quad & -\exp\left[\log\left(\mathbf{c}^{\dagger}\mathbf{E}_1\right) - \log\left(\mathbf{c}^{\dagger}\mathbf{E}_{z_1}\right) + \log\left(\mathbf{c}^{\dagger}\mathbf{E}_2\right) - \log\left(\mathbf{c}^{\dagger}\mathbf{E}_{z_2}\right)\right] \\ \text{subject to} \quad & \nabla \times \frac{1}{\mu}\nabla \times \mathbf{E}_1 - \omega_1^2 \boldsymbol{\varepsilon}(\mathbf{x})\mathbf{E}_1 = -i\omega_1 \mathbf{J}_1 \\ & \nabla \times \frac{1}{\mu}\nabla \times \mathbf{E}_2 - \omega_2^2 \boldsymbol{\varepsilon}(\mathbf{x})\mathbf{E}_2 = -i\omega_2 \mathbf{J}_2 \end{aligned} \tag{21}$$

where ω_1 and ω_2 are the angular frequencies at 1300 and 1500 nm, $\mathbf{E}_1$ and $\mathbf{E}_2$ are the electric field, and $\mathbf{J}_1$ and $\mathbf{J}_1$ inject input sources into the waveguide for frequency ω_1 and ω_2. The objective is a sum with four terms using a negative log-sum-exp smooth approximation of the maximum function, and each term corresponds to a sub-objective. As shown in Eq. (21), two terms $\log(\mathbf{c}^{\dagger}\mathbf{E}_1)$ and $\log(\mathbf{c}^{\dagger}\mathbf{E}_2)$ correspond to maximizing transmission efficiency through the top waveguide at 1300 nm and bottom waveguide at 1500 nm, given the specific permittivity distribution $\boldsymbol{\varepsilon}(\mathbf{x})$, and two terms $\log(\mathbf{c}^{\dagger}\mathbf{E}_{z1})$ and $\log(\mathbf{c}^{\dagger}\mathbf{E}_{z2})$ correspond to the initial overlap integral given the homogeneous permittivity distribution $\boldsymbol{\varepsilon}_{\text{ini}}(\mathbf{x}_{\text{ini}})$. Note that, the $\mathbf{E}_{z1}$ and $\mathbf{E}_{z2}$ are constant during the design process, which are mainly used to normalize the objective, but the $\mathbf{E}_1$ and $\mathbf{E}_2$ vary along with the update of $\mathbf{x}$ to minimize the objective function. One may consider to include other sub-objectives using alternative mathematical

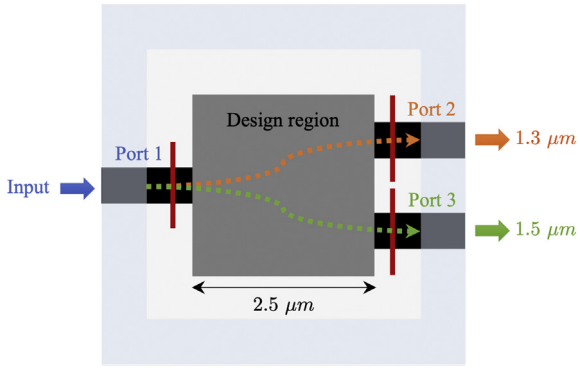


Fig. 3. Illustration of wavelength demultiplexer design. The structure consists of one input waveguide (port 1), two output waveguides (port 2 and port 3), and a 2.5 $\mu m \times 2.5$ μm design region. All three waveguides are the same, with a width of 500 nm. The outer hatched light blue frame represents the simulation domain, specifically, the perfectly matched layer (PML) boundaries. The goal of inverse design is route to 1.3 μm through the top waveguide and 1.5 μm through the bottom waveguide.

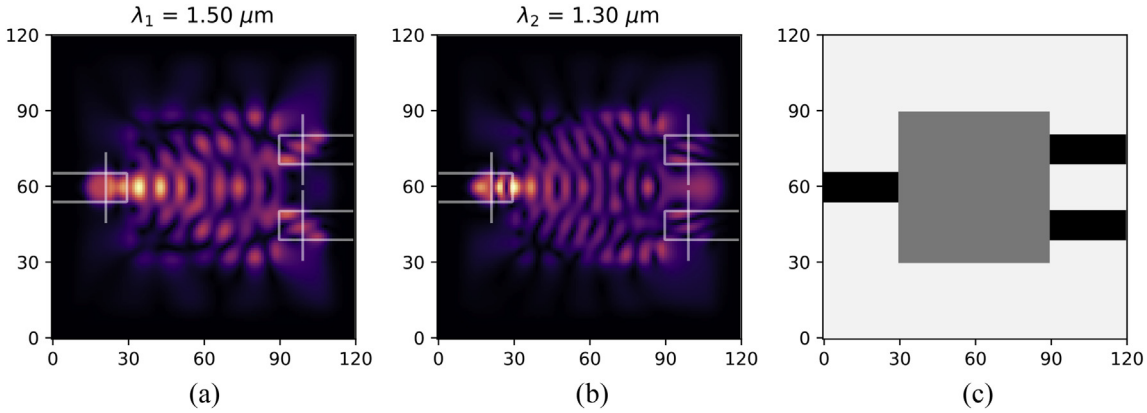


Fig. 4. Initialization of inverse design. (a) Electric field intensity at 1500 nm, (b) electric field intensity at 1300 nm, and (c) initial permittvity distribution.

formulation for improving the optimization performance. Interested reader may find more discussion in [11].

4.3. Parameterization scheme

By selecting an appropriate parameterization, the fabrication constraints in optimization can be naturally imposed. The parameterization scheme here consists of two crucial operators: nonlinear projection and convolution filtering. Nonlinear projection aims to binarize the permittivity distribution by

$$\begin{aligned}\boldsymbol{\varepsilon}(\mathbf{x}) &= \boldsymbol{\varepsilon}_{\min} + (\boldsymbol{\varepsilon}_{\max} - \boldsymbol{\varepsilon}_{\min})\varphi(\mathbf{x}), \quad \varphi(\mathbf{x}) \\ &= \frac{\tanh(\beta\cdot\eta) + \tanh(\beta\cdot(\mathbf{x}-\eta))}{\tanh(\beta\cdot\eta) + \tanh(\beta\cdot(1-\eta))}\end{aligned} \tag{22}$$

where β is the coefficient of projection strength, and η is the center of the projection. Fig. 5 (a) shows the nonlinear projected mapping between original input $\mathbf{x}$ and projected $\hat{\mathbf{x}} = \varphi(\mathbf{x})$ given different projection strength and fixed $\eta = 0.5$. As the increasing of projection strength β, the projected $\hat{\mathbf{x}}$ shows a clear trend to binary value 0 or 1.

The convolution operator is used as a blurring filter that results in smooth features of the permittivity distribution and avoids the tiny features that are less than the minimum feature size of fabrication. Integrating nonlinear projection ($\beta = 50$ and $\eta = 0.5$) and convolution filtering, we visualize the parameterized distribution with varying circle radius based on a specific permittivity distribution, as shown in Fig. 5 (b). It is clear to see that the parameterized distribution shows a clear black-white pixel (material layout) without intermediate grey pixels. The feature size can be controlled by determining a specific convolution radius.

4.4. Methodology workflow

We illustrate a workflow to implement DGSoptimization method for inverse design problems. Fig. 6 shows the four core components, that are parameterization, physics simulation, objective formulation and DGS optimization. The detailed procedure is summarized as follows:

- Step 0: Initialization. An initial design variable $\mathbf{x}_0$ are set up through a homogeneously distribution or a random distribution with noise.
- Step 1: Parameterization. For design variable $\mathbf{x}_k$ at the k-th iteration, convolution filtering with a specific radius is imposed to eliminate the small features, followed by the nonlinear projection in Eq. (22) that binarizes the design variables. Parameterization builds up a transformation between design variable $\mathbf{x}_k$ and the corresponding permittivity distribution $\boldsymbol{\varepsilon}(\mathbf{x}_k)$.
- Step 2: Physics simulation. The permittivity distribution $\boldsymbol{\varepsilon}(\mathbf{x}_k)$ are taken as input to physics simulation, for example, electromagnetic simulation. The electric field intensity $\mathbf{E}(\boldsymbol{\varepsilon}(\mathbf{x}_k))$ are obtained by solving the Maxwell's equation in Eq. (3) using FDFD method.
- Step 3: Objective formulation. The objective function in Eq. (21) is formulated to conduct the optimization for inverse design. The resulting $\mathbf{E}(\boldsymbol{\varepsilon}(\mathbf{x}_k))$ is used to calculate the overlap integral and then yield a scale value $\mathscr{S}_k$ as the loss. The constraints on fabrication and volume fraction are also defined in this step.

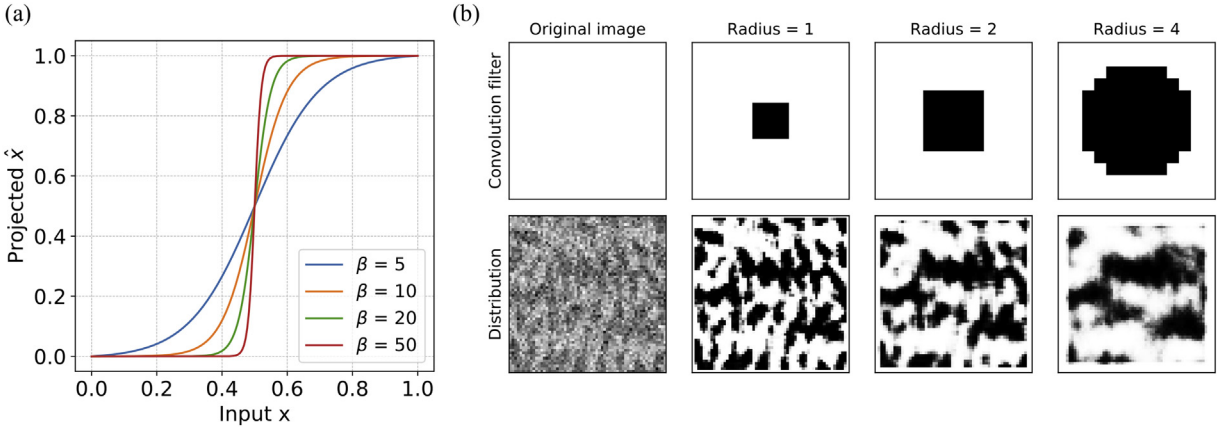


Fig. 5. Parameterization scheme for inverse design. (a) Nonlinear projection function for binarizing the input design variables and (b) visualization of convolution filtering with different filter radius given a specific projection strength.

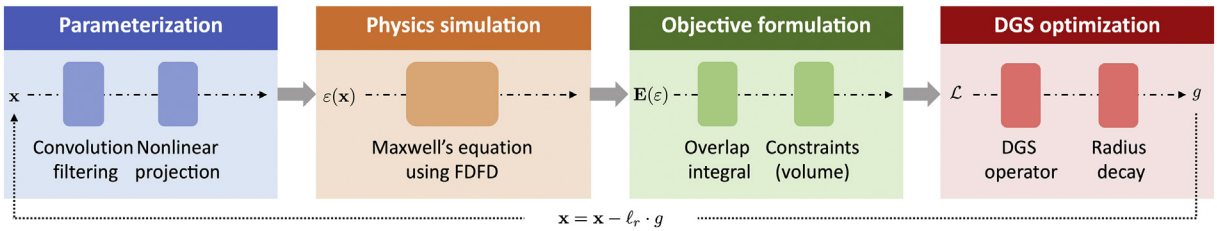


Fig. 6. Methodology workflow of inverse design using the optimization method.

- Step 4: DGS optimization. The Step 1–3 can be considered as a forward evaluation where a set of input-output paired samples $\mathbb{S} = \left\{\left(\mathbf{x}_k^{(1)}, \mathrm{L}_k^{(1)}\right), \ldots, \left(\mathbf{x}_k^{(l)}, \mathrm{L}_k^{(l)}\right)\right\}$ can be drawn for DGS gradient operator. These samples drawn by Gauss-Hermite quadrature rule are used to estimate each directional derivative in Eq. (15) and are then assembled to accurately approximate the full d-dimensional gradient $\mathbf{g}_k$ in Eq. (17). The DGS gradient is well-suited to update the design variable via gradient descent algorithm:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \ell_k \cdot \mathbf{g}_k \tag{23}$$

The new design variable $\mathbf{x}_{k+1}$ goes back to Step 1 for iterative updating until the convergence criteria is satisfied.

To improve the optimization performance on accuracy, convergence and robustness, we implement an adaptive decay scheme for updating hyperparameters, including a large DGS radius σ_r in Eq. (15), learning rate ℓ_r in Eq. (23) and projection strength β in Eq. (5). Specifically, the quadratic decay is used

$$z_l = \left(z^{\mathrm{ini}} - z^{\mathrm{end}}\right) \times \left(1.0 - \frac{k}{k_{\max}}\right)^{\alpha} + z^{\mathrm{end}}, \tag{24}$$

where z_l represents the hyperparameters that can be σ_r, ℓ_r or β at the k-th iteration, $k_{\max}$ is the maximum of iteration and $\alpha = 2$ is the coefficient of decay rate. z^{ini} and z^{end} are the initial value and end value of hyperparameters respectively. In this study, we use $\sigma_r^{\mathrm{ini}} = 0.25$ and $\sigma_r^{\mathrm{end}} = 0.05$ for DGS radius, $\ell_r^{\mathrm{ini}} = 1.0$ and $\ell_r^{\mathrm{end}} = 0.01$ for learning rate, and $\beta^{\mathrm{ini}} = 0.2$ and $\beta^{\mathrm{end}} = 0.05$ for the reciprocal of projection strength. In addition, the radius $r = 2$ in convolution filtering is used for parameterization. The physics simulation is achieved by *ceviche* (https://github.com/fancompute/ceviche) that is an electromagnetic simulation tool for solving Maxwell's equations. In DGS gradient operator, five GH quadrature points are used and the nodes of points in practical computation can be reduced to three due to symmetric property [49]. All numerical experiments (physics simulation and optimization) are implemented in Python 3.6 and conducted on a cluster with 44 Intel Xeon E5–2699 v4 CPUs at 2.20 GHz. Each iteration in optimization takes around 1.2 min using a parallel implementation of DGS gradient operator.

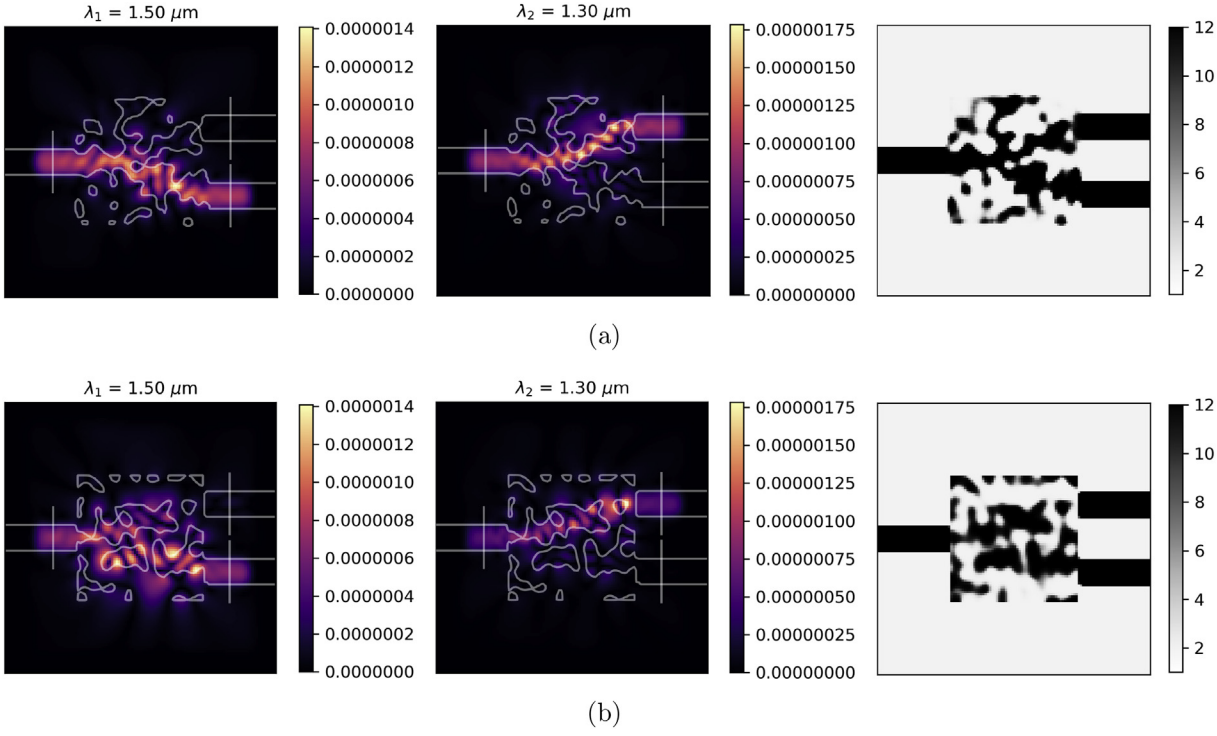


Fig. 7. Electric field intensity of the optimized device at 1500 nm (left column) and 1300 nm (middle column), as well as the optimized permittivity distribution (right column) by optimization using (a) the nonlocal DGS gradient and (b) the local gradient method]

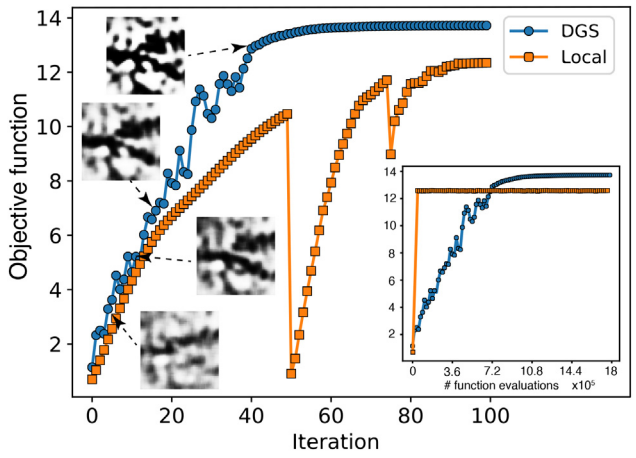


Fig. 8. The optimization iteration history. The blue circles represent the optimization using DGS gradient operator and the orange squares represent the optimization using local gradient algorithm.

4.5. Result and discussion

The methodology workflow of inverse design is applied here to maximize the performance of nanophotonic devices described in Section 4.1. The final optimized design and the corresponding electric field intensity are diagrammed in Fig. 7. Note that the device designed by the DGS optimization method, shown in Fig. 7 (a), displays a nonintuitive geometry while retaining relatively large features and a clear permittivity distribution with ideal binarization. The light takes a relatively confined path through the structure at both wavelengths. The optimization history, shown in Fig. 8, provides iterative changes of the permittivity distribution during the optimization process. At each iteration, the DGS method updates the smoothing gradient estimator that requires $5 \times 60 \times 60$ evaluations of objective functions, which requires significantly larger computational cost than the local gradient algorithm. We realize this is the computational challenge of the DGS method and we provide a detailed discussion of computational cost and several strategies to mitigate this burden, particularly given a limited computational budget. In fact, the iteration results shown here aim to illustrate the detailed optimization process using *gradient information*. This is different from the derivative-free global optimization algorithms, such as GA, PSO, and Bayesian optimization. It can be seen that the local gradient method shows significant oscillations at 50 and 75 iterations, shown in Fig. 8. This is mainly due to the fact that the projection strength is increased by a discrete step function in the classical local gradient method. The DGS optimization method shows a relatively smoothing iteration curve since a dynamic decay mechanism is implemented to adaptively update the projection strength. From the computational cost perspective, we also show a fair comparison using the same number of evaluations of the objective function (see the subfigure in Fig. 8 (right corner)). It is clear to note that the local gradient shows a fast convergence but it quickly traps into a local minimum and difficult to escape even though a large number of evaluations are performed. On the contrary, the DGS method increases slowly initially but gradually converges to a better solution, $f^*_{\text{DGS}} = 13.58$, that performs ~10% improvement compared with the local gradient method that is $f^*_{\text{local}} = 12.40$. The gap at the final objective values between the DGS optimization method and the local gradient method is also demonstrated by the optimized design and the resulting electric field intensity in Fig. 7. It is easy to observe that the light path in local gradient design, specifically at 1500 nm wavelength (left in Fig. 7(b)) is relatively diffused.

4.5.1. Effect of random initialization with Gaussian noise

Here we investigate the effect of random initialization on the optimized permittivity distribution and device performance. In this case, we repeatably run 100 times optimization with three different levels of randomness described by Gaussian noise: $N(0,0.1)$, $N(0,0.05)$ and $N(0,0.01)$. The iteration history and histogram of final objective values are shown in Fig. 9 and Fig. 10. The solid curves in Fig. 9 represent the mean value of 100 trials and the dash area represents the confidence intervals with $[-\sigma, +\sigma]$, where σ is the standard deviation of the objective values at a specific iteration. It can be seen that the variation of objective value is relatively large but it is quickly narrowed and converged to the final value, which is pretty close in all three levels of randomness. The histogram in Fig. 10 shows the distribution of the final objective values using the DGS gradient (dark color) and local gradient (light color). The proposed method outperforms the local gradient in terms of the objective values and shows a smaller variation in all three levels of randomness. Table 1 provides a statistical comparison of objective values between the DGS gradient method and local gradient methods. Compared with the local gradient method, the DGS optimization method shows superior performance with an improvement of 9.37%, 9.42%, and 9.22% in the mean value respectively. For the standard deviation, two methods show almost the same variation level if the noise is tiny, specifically at $\sigma_N = 0.01$ but the DGS optimization method achieves a significant reduction of 21.1% and 30.6% when the noise level is relatively large, typically at $\sigma_N = 0.1$ and $\sigma_N = 0.05$. Through a statistical analysis of the objective values, the DGS optimization method shows a higher objective value and much smaller uncertainty on the final performance, particularly given relatively large randomness

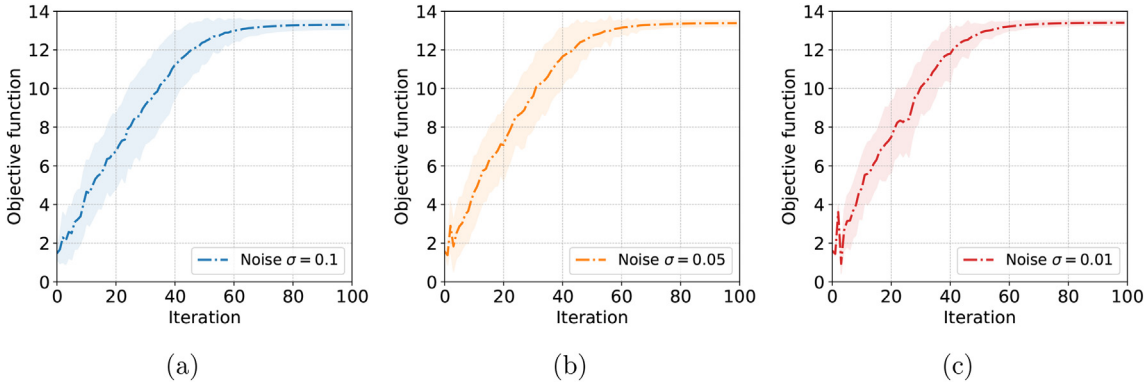


Fig. 9. Study of the local minima with three initialization with random Gaussian noise for the inverse design performance. The optimization is run 100 times with random initial guess (different random seeds). (a) Gaussian noise $N(0,0.1)$, (b) Gaussian noise $N(0,0.05)$ and (c) Gaussian noise $N(0,0.01)$.

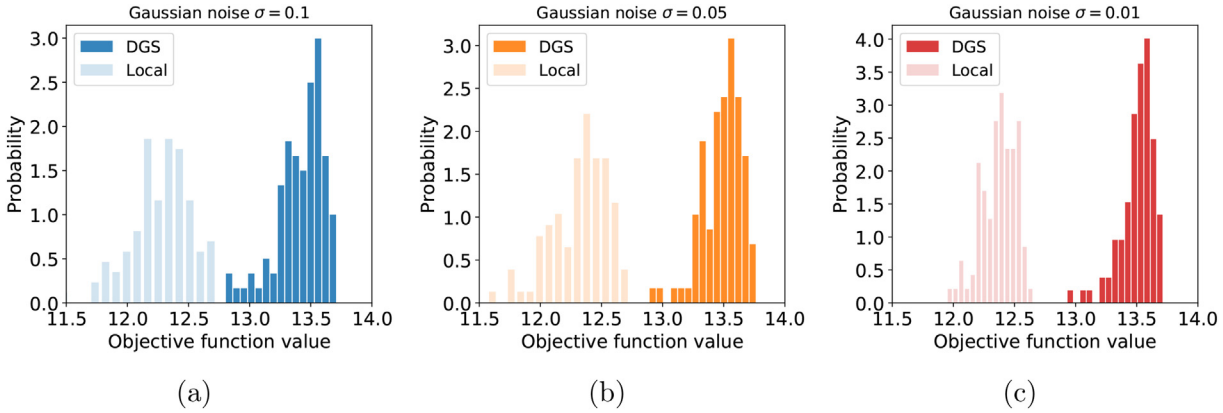


Fig. 10. Histogram of the objective function values of 100 local minmia given different noise levels. Dark color represents the distribution using the DGS gradient and light color represents the distribution using local gradient algorithm. (a) Gaussian noise $N(0,0.1)$, (b) Gaussian noise $N(0,0.05)$ and (c) Gaussian noise $N(0,0.01)$.

Table 1
Statistical comparison of optimized performance between the DGS gradient and local gradient method.

	DGS gradient		Local gradient	
Gaussian noise	Mean	Standard deviation	Mean	Standard deviation
$\sigma_N = 0.1$	13.42	0.191	12.29	0.242
$\sigma_N = 0.05$	13.48	0.159	12.33	0.229
$\sigma_N = 0.01$	13.51	0.136	12.37	0.141

associated with the initial guess. In other words, the DGS optimization method demonstrates stronger robustness and reliability to resist the local minima caused by random initialization.

Figs. 11-13 show a sample collection of optimized design with the electric field intensity given Gaussian noise $N(0,0.1)$, $N(0,0.05)$ and $N(0,0.01)$ respectively. For a specific noise level, three selected samples are provided, and each of samples corresponds to a random initial guess (first column), DGS-based optimized design (column 4) with the electric field intensity at 1500 nm wavelength (column 2) and 1300 nm wavelength (column 3), as well as the local gradient-based design (column 7) with the electric field intensity at 1500 nm wavelength (column 5) and 1300 nm wavelength (column 6). In particular, when the noise level is relatively high, as shown in Fig. 11, the optimized structure using the DGS gradient shows a clear binary distribution with an overall consistent pattern, while some small features and differences exist. The corresponding electric field shows a similar confined and clear transmission path. However, the optimized design using the local gradient method greatly varies with the random initialization. It is difficult to see a clear path as several noise features are embedded into the structure. This results in the fact that the electric field spreads across the entire device, suggesting that multi-path interference contributes to unexpected device performance. This is because the local gradient method is limited to escape the local minima that strongly depend on the initial guess. Although noise level is decreased from $N(0,0.1)$ to $N(0,0.05)$, as shown in Fig. 12, the local gradient method is still affected by the initial random noise. As a result, the optimized structure includes a few small features with noise and the overall structure is unstable and unclear. On the contrary, the DGS optimization method is well-suited to handle the noise using a Gaussian smoothing operator with a large radius and thus achieves a robust and binarized design. When a small noise is imposed into the initialization, three

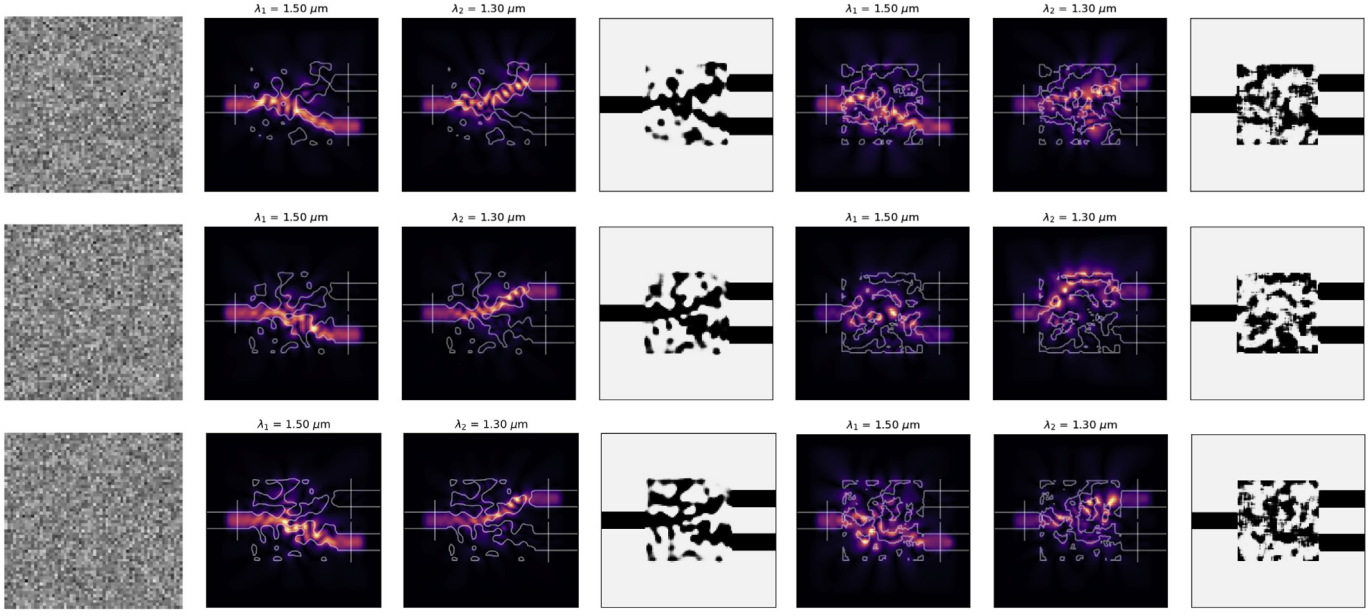


Fig. 11. A sample collection of optimized permittivity distribution and the corresponding electric field intensity given initial Gaussian noise $N(0,0.1)$.

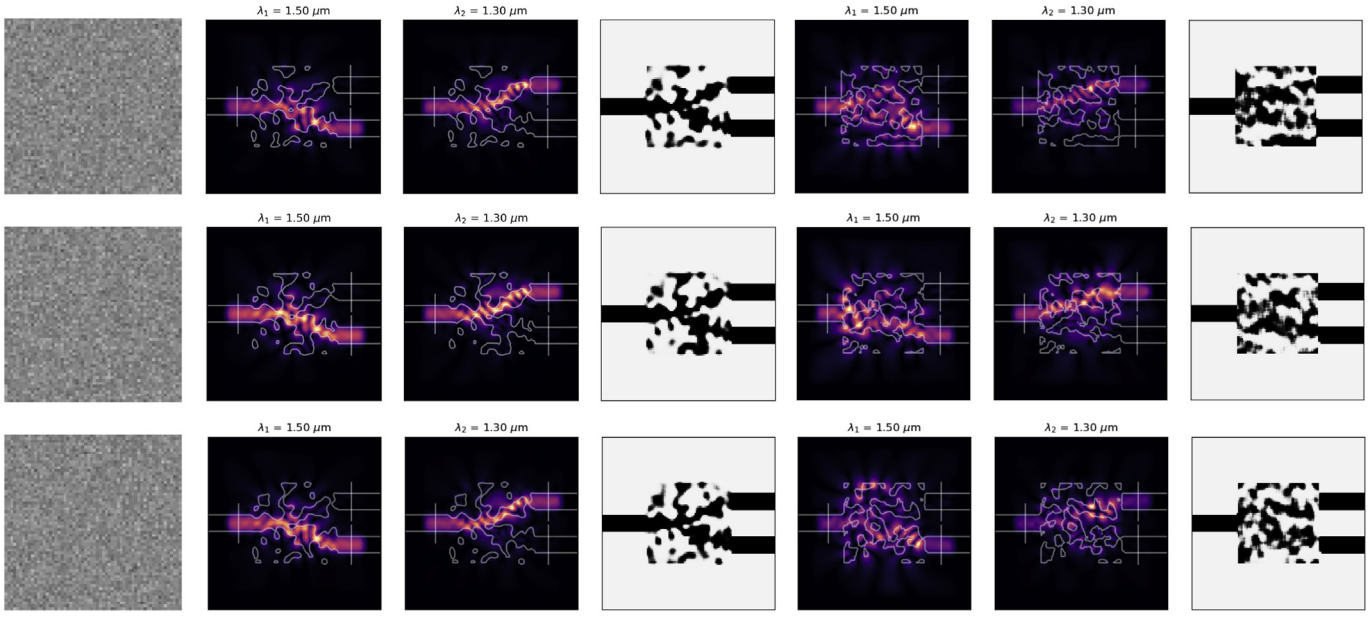


Fig. 12. A sample collection of optimized permittivity distribution and the corresponding electric field intensity given initial Gaussian noise $N(0,0.05)$.

samples from the DGS optimization method in Fig. 13 are almost identical and converged to the optimized design as similar as the result using homogeneous initialization without any randomness. For the local gradient method, the noised feature is mitigated from the optimized structure but the resulting electric field intensity still underperforms the DGS method, specifically at the 1500 nm.

4.5.2. Inverse design optimization with volume constraint

The optimized design discussed above is achieved by optimization without any constraint of materials usage. To investigate the effect of material usage on device performance, we conduct a study to investigate the relationship between volume fraction of material usage and objective function values. The general volume constraint in Eq. (5) can be added to Eq. (21), which is given by

$$h(\mathbf{x}) = \frac{V(\mathbf{x})}{V_0} - \gamma = \frac{\sum_{i=1}^{N_p} \mathbf{x}}{N_p} - \gamma \le 0, \quad \mathbf{x} \in [0,1] \tag{25}$$

where N_p is the total number of pixels, V is the volume of material usage, V_0 is the original homogeneous distribution where all the computational design variables $\mathbf{x} = 1$ and γ is the specific constant that is the volume constraint fraction. Adding Eq. (25) to Eq. (21), the design problem is transferred from unconstrained optimization to constrained optimization problem and the materials usage can be controlled by assigning a specific value of γ. As shown in Fig. 14, a total of 300 samples of three levels of Gaussian noise are used in this case but unfortunately, we didn't observe a strong correlation between the volume fraction and final objective values. Most cases of volume fraction γ in Fig. 14

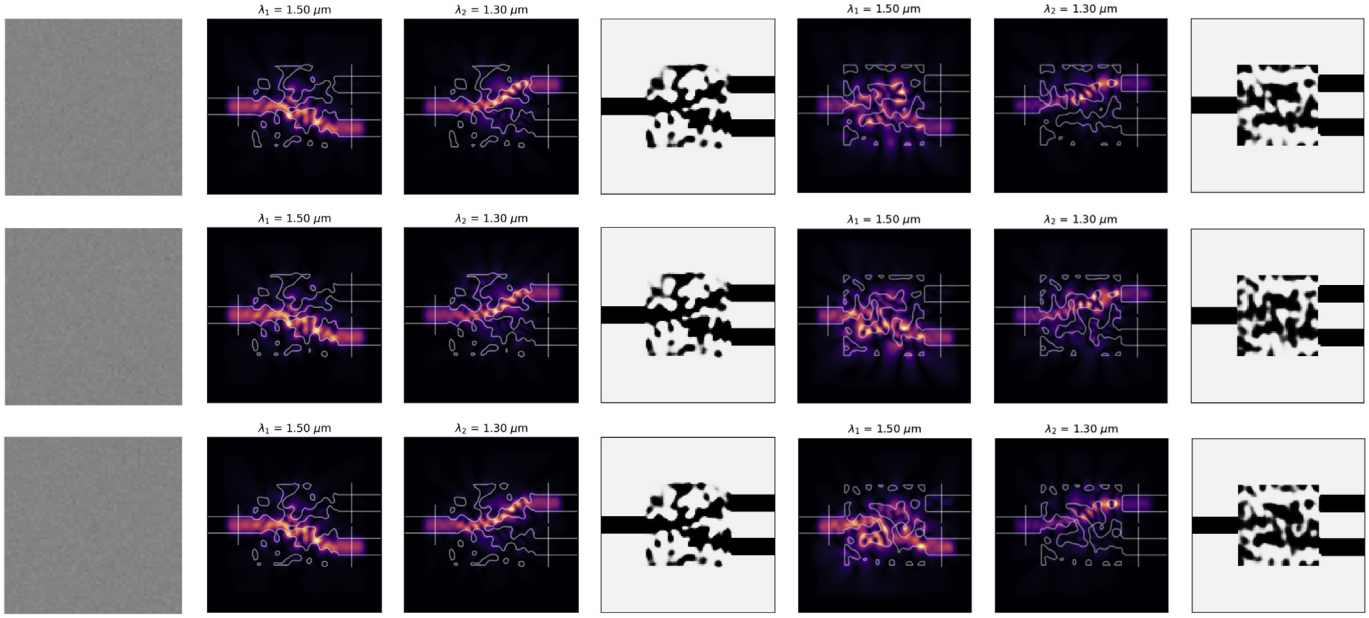


Fig. 13. A sample collection of optimized permittivity distribution and the corresponding electric field intensity given initial Gaussian noise $N(0,0.01)$.

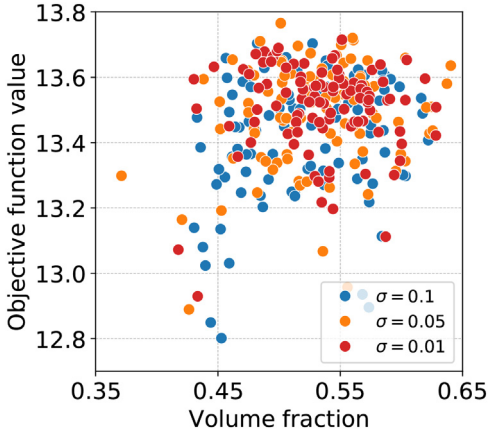


Fig. 14. Relationship between volume fraction and optimized objective value under random initialization. There is no clear correlation between the used amount of materials and the optimized performance.

concentrates in 0.45−0.65 and no cases are lower than 0.35. It naturally gives rise to an interesting question: is that possible to use fewer materials but achieve equivalently good performance?

To answer this question, we reformulate the optimization by adding a volume constraint and solve this constrained optimization problem using the Method of Moving Asymptotes (MMA) [61]. MMA is the state-of-the-art optimizer, which has been demonstrated to be versatile and well suited for wide range engineering design problems, particular in topology optimization. The basic of MMA aims at solving general nonlinear constrained optimization problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & : \quad f_0(\mathbf{x}) + a_0 z + \sum_{i=1}^{m}\left(c_i y_i + \frac{1}{2} d_i y_i^2\right) \\ s.t. \quad & : \quad f_i(\mathbf{x}) - a_i z - y_i \le 0, \quad i = 1, \ldots, m \\ & : \quad \mathbf{x} \in X, \mathbf{y} \ge 0, z \ge 0 \end{aligned} \tag{26}$$

Here, $X = \{x \in \mathbb{R}^n | x_j^{\min} \le x_j \le x_j^{\max}, j = 1, \ldots, n\}$, where $x_j^{\min}$ and $x_j^{\max}$ are given real numbers which satisfy $x_j^{\min} < x_j^{\max}$ for all j, $f_0, f_1, \ldots, f_m$ are given, continuously differentiable, real-valued functions on X, a_0, a_i, c_i and d_i are given real numbers which satisfy $a_0 > 0$, $a_i \ge 0$, $c_i \ge 0$ and $d_i \ge 0$ and $c_i + d_i > 0$ for all i and also $a_i c_i > a_0$ for all i with $a_i > 0$.

MMA is a gradient-based method for solving Eq. (26) using the following steps. In each iteration, given the current point $(\mathbf{x}^{(k)}, \mathbf{y}^{(k)}, z^{(k)})$, MMA generates an approximating subproblem, where the functions $f_i(\mathbf{x})$ are replaced by convex functions $\widetilde{f}_i^{(k)}(\mathbf{x})$. The approximating functions are determined by the *gradient information* at the current iteration point and moving asymptotes parameters which are updated in each iteration based on information from previous iteration points. The next iteration point $(\mathbf{x}^{(k+1)}, \mathbf{y}^{(k+1)}, z^{(k+1)})$ is obtained by solving the subproblem, which is defined in [61]. However, MMA is limited to seek optima using local gradients information, either via adjoint method or finite difference. We address this challenge by inserting the DGS gradient into the MMA optimizer so that we can exploit the nonlocal exploration of the DGS operator to search for a better design. Fig. 15 shows the optimized design with volume constraint $\gamma = 0.3$ that means only 30% of materials in the design domain are used. We easily observe a clear and confined light transmission path at both 1500 nm and 1300 nm. The optimized permittivity distribution shows the fundamental splitter-like feature and eliminates a lot of unnecessary material in the device compared with the design without volume constraint.

Fig. 16 shows the iteration history of constrained optimization with $\gamma = 0.3$. To illustrate the optimized process in detail, Fig. 16 (a) shows the iterative change of electric field intensity at 1500 nm and the objective value that corresponds the term $\exp[\log(\mathbf{c}^\dagger \mathbf{E}_1) - \log(\mathbf{c}^\dagger \mathbf{E}_{z1})]$ in objective function (see Eq. (21)). Similarly, the iteration history of objective term $\exp[\log(\mathbf{c}^\dagger \mathbf{E}_2) - \log(\mathbf{c}^\dagger \mathbf{E}_{z2})]$ at 1300 nm is shown in Fig. 16 (b). Note that, under the volume fraction constraint, the iteration curves show some oscillations initially but quickly converge at the 60th iteration. The designed device successfully separate both signals at the assigned port. Fig. 16 (c) shows the iteration history of overall objective (red color), the decay of volume fraction (black color) and the changes of permittivity distribution. The constrained optimization using DGS gradient integrating with MMA optimizer achieves a nearly same high performance, $f_{\mathrm{DGS}}^{0.3} = 13.09$ as the optimization without volume constraint. But the amount of material usage is significantly reduced from 0.474 (the case in Fig. 7(a)) to 0.3 and we therefore save 36.7% material usage.

We further reduce the volume fraction from 0.3 to 0.2 to exploit the maximizing capability of the nonlocal exploration using the DGS gradient integrating with the MMA optimizer. Fig. 17 shows the electric field intensity based on the optimized permittivity distribution given $\gamma = 0.2$. The optimized design still retains the principle splitter-like features that connect the input port and two output ports, even though a very limited amount of material is used. As similar to the optimized design with $\gamma = 0.3$, few small spots are disconnected from the main structure and probably play a limited role in the effective transmission of the input source. This special feature, observed by several previous studies [10,11,20,44] is probably due to the issue of local minima. Although the DGS operator enables nonlocal exploration to facilitate the global search, the optimized devices may be trapped into one of the local minima.

Fig. 18 shows the iterative process of constrained optimization. It is noted that the iteration history at 1500 nm (Fig. 18 (a)) and 1300 nm (Fig. 18 (b)) shows a relatively large oscillation, specifically in the initial stage. This probably results from the fast decay of volume fraction in the initial period, as shown in Fig. 18 (c). After 60 iterations, the objective values tend to converge and the volume fraction also approaches to

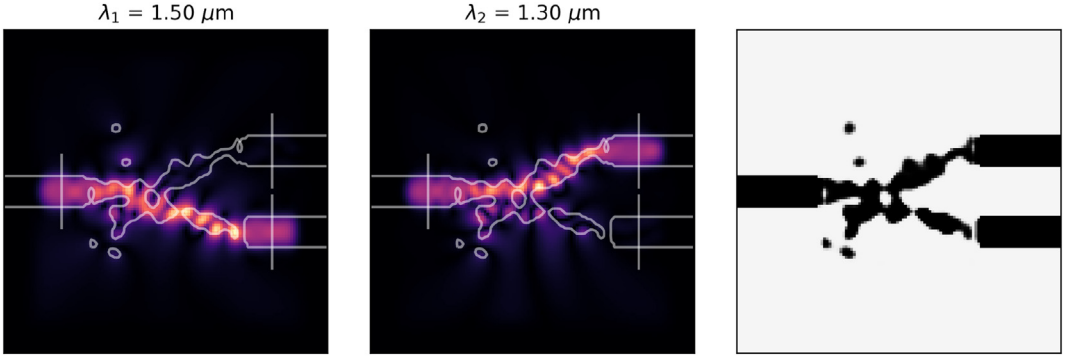


Fig. 15. Optimized device with volume constraint $\gamma = 0.3$. The electric field intensity of the optimized device at 1500 nm (left) and at 1300 nm (midde), as well as the optimized permittivity distribution (right) using DGS gradient with MMA optimizer.

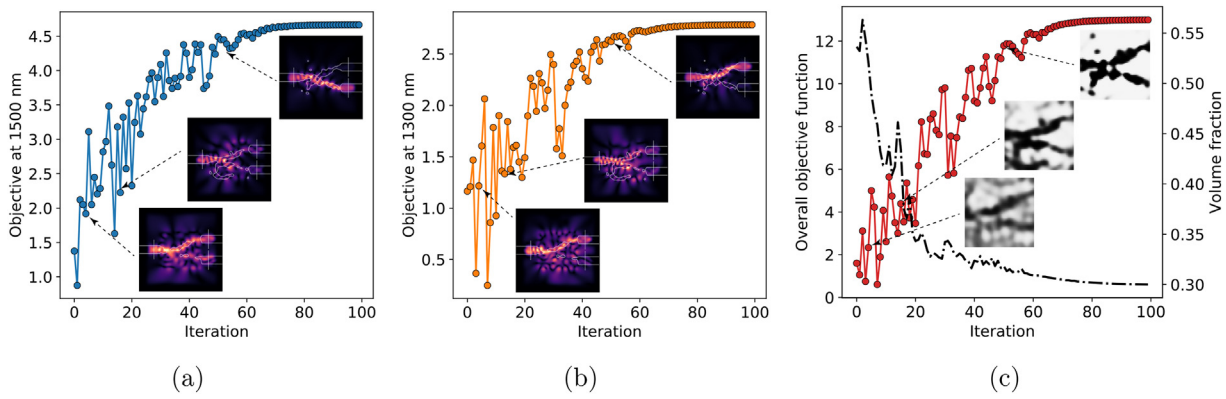


Fig. 16. Iteration history of constrained optimization with $\gamma = 0.3$ and the corresponding electrical field intensity and permittivity distribution. (a) Objective at 1500 nm, (b) objective at 1300 nm and (c) overall objective and volume fraction iteration.

the constrained value, $\gamma = 0.2$. This case with a smaller volume fraction achieves a final objective value $f_{\mathrm{DGS}}^{0.2} = 12.70$ that is slightly lower (~3.71%) than the case of $\gamma = 0.3$% but saves 33.3% material usage.

The final designed devices with three different amounts of material usage are shown in Fig. 19. From the angled view, the vertical sidewalls are clearly visible and the permittivity distribution along the vertical direction are all same. This is easy for fabricating using electron-beam (top-down) lithography followed by etching the 220-nm-thick layer of an SOI substrate, leaving the structure with an air cladding. Compared with the classical design approaches, the proposed inverse design framework integrating the DGS gradient with MMA optimizer provides higher final performance with less material usage for real fabrication in practice.

5. Discussion and limitation

This section will address several concerns in the practical implementation of the proposed method and the existing limitations with the current version of the DGS algorithm in computational inverse design.

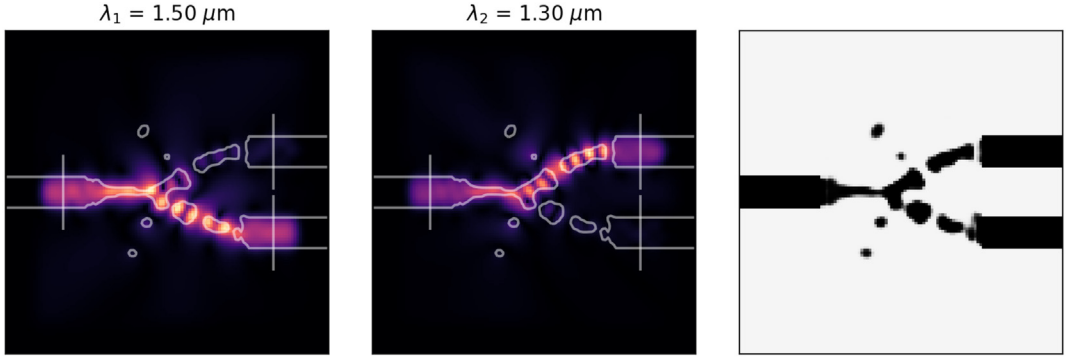


Fig. 17. Optimized device with volume constraint $\gamma = 0.2$. The electric field intensity of the optimized device at 1500 nm (left) and at 1300 nm (midde), as well as the optimized permittivity distribution (right) using DGS gradient with MMA optimizer.

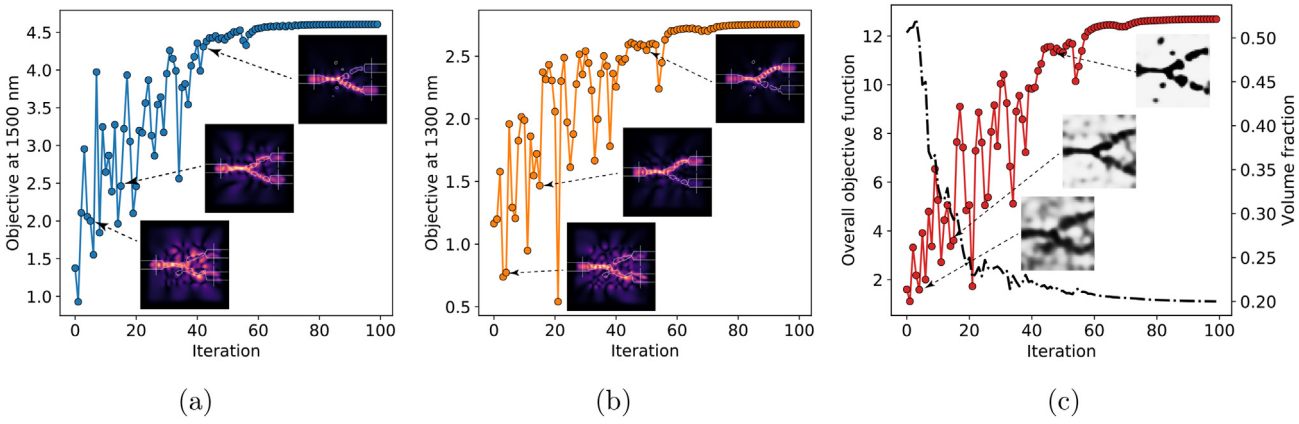


Fig. 18. Iteration history of constrained optimization with $\gamma = 0.2$ and the corresponding electrical field intensity and permittivity distribution. (a) Objective at 1500 nm, (b) objective at 1300 nm and (c) overall objective and volume fraction iteration.

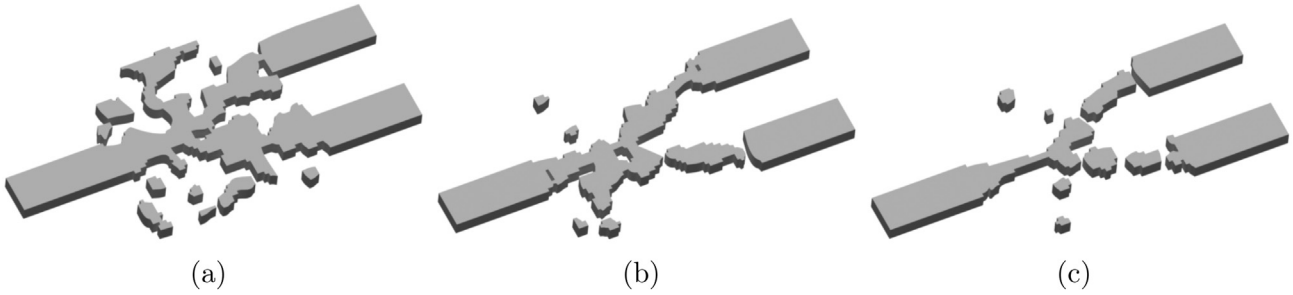


Fig. 19. A three-dimensional rendering of the optimized design. Silicon is shown in grey and light enters the optimized device from the input waveguide on the left-hand side (port 1) and exits via one of the two output waveguides (port 2 and port 3) on the right side. There are three optimized devices with different volume fraction (a) $\gamma = 0.474$, (b) $\gamma = 0.3$ and (c) $\gamma = 0.2$.

5.1. Hyperparameters choice

Hyperparameters used in the proposed method play important roles in the final optimized performance. In the DGS gradient operator, the most sensitive hyperparameter is the smoothing radius of σ_r. If σ_r is too small, the loss function will be insufficiently smoothed, such that the optimizer may be trapped in a local minimum. In contrast, if σ_r is too big, the loss function is overly smoothed, such that the convergence will become much slower. Based on our hyperparameter tuning experience, our suggestion is to set up the smoothing radius σ_r according to the search domain. In the photonic design problem, the computational domain is $\mathbf{x}$ and its design domain is $[0,1]$. Specifically, we adopted a quadratic decay schedule ($\alpha = 2$) for smoothing radius σ_r and learning rate ℓ_r. The initial value of smoothing radius σ_r is set to 20 % − 25% of the search domain width, and the ending value of smoothing radius σ_r is set to 1 % − 5% of the search domain width. The learning rate of ℓ_r is the second important parameter that affects the convergence performance. Similarly, we determine the learning rate ℓ_r according to the search domain and set the initial value to be 10% and ending value to be 1% of the search domain width. Although the Gauss-Hermite quadrature is the key to the overall superior performance, we consider that 5 GH points already have a high estimate accuracy and can be used for most of the problems.

The hyperparameters in the parameterization scheme will also affect the final performance in terms of the fabrication constraints. Typically, the projection strength and convolution filtering radius depend on the computational design domain $\mathbf{x}$. A general guideline to choose the projection strength is that the initial value β^{ini} can be 20 % − 25% and the ending value β^{end} can be 1 % − 5% of the design domain width. The filtering radius $\alpha = 2$ is typically used and the effect of the different radius is discussed, as shown in Fig. 4. We realize that the hyperparameter tuning is a non-trivial task and how to adaptively adjust these parameters is still an open question.

5.2. Computational cost challenge

Differing from the local gradient method, the computational cost of the DGS method linearly depends on the dimensionality of the design space. At each iteration, a relatively large number of function evaluations are required to estimate the nonlocal gradient using a smoothing strategy. Using a big learning rate, DGS can potentially achieve a faster convergence with fewer iterations but the total computational cost is still large. We realize this is a critical challenge, specifically given a small computational budget. This computational challenge can be mitigated by using parallel computing because all evaluations at each iteration are completely parallelizable as those in random sampling with very small communication costs. In this work, we implement the DGS method by using 44 core CPUs but it can be easily scaled to thousands of CPUs on Supercomputers, like OLCF Summit, which has 9216 POWER9 22-core CPUs. Ideally, each function evaluation can be deployed to one CPU so that DGS achieves an equivalent cost to the local gradient at each iteration. Due to its ideal scalability, DGS is well-suite to the applications where the simulation or modeling can make use of high performance computing (HPC) if large computational resources are available. In the next version of the DGS method, we will incorporate dimension reduction techniques, e.g., active subspace [62], and nonlinear level-set method [43], to reduce the dimensionality requiring directional smoothing, which can alleviate the dependence on computing resources and finally overcome the limitation of the relatively large computational cost.

5.3. Optimization performance and sub-optimal solution

Although the computational cost is relatively large, the DGS shows superior capabilities to explore the better solution, which is more critical to address the design challenges by optimization. The local gradient method performs efficiently and converges fast, but it often traps to a local minimum that underperforms the DGS method. In addition, DGS shows better robustness than the local gradient method, which is sensitive to the choice of initial guess and the final optimized design varies largely. For the derivative-free global optimization algorithms, e.g., GA or PSO, they are also very computationally intensive and frequently fail to find an optimal solution in such high dimensional space. To some extent, these evolutionary algorithms show a similar performance as the "random search" method. Like Bayesian optimization (BO), it is intractable to handle more than thousands of dimensional problems due to the limitation in Gaussian process modeling. The constraints in the BO framework and stochastic optimization such as SGD and Adam is still a critical issue. We realize that there is a trade-off between computational cost and final optimized performance, but we consider it is more important to first search a better solution rather than a worse solution but saving time.

The proposed DGS method has advantages to capture the global structure in the loss landscape. However, if the loss function without global structures, for example, the Schwefel function, as shown in [47], the DGS method cannot find the global minimum. This could happen in real-world inverse design problems and other engineering applications. Even though our method outperforms the local gradient method in solving the nanophotonic design problem, we cannot verify that the final design optimized by our method is globally optimal.

6. Conclusion

This work focuses on the development of a DGS optimization method for computational inverse design in nanophotonics. A novel DGS gradient operator is introduced to improve the nonlocal exploration capability required for escaping from local minima in the high-dimensional non-convex landscapes. The DGS gradient operator is achieved by conducting 1D nonlocal explorations along with d orthogonal directions in $\mathbb{R}^d$, each of which defines a nonlocal directional derivative as a 1D integral. Instead of Monte Carlo (MC) sampling, a deterministic Gauss-Hermite (GH) quadrature is used to estimate each of 1D integrals in d-

dimension to achieve high accuracy. Compared with the local gradient method, the directional smoothing allows for a large smoothing radius to capture the global structure of loss landscapes. GH quadrature with error bound provides guarantees higher accuracy than random sampling, even though a large smoothing radius is used.

The DGS optimization method has advantages in portability and flexibility so that it is naturally incorporated with parameterization, physics simulation, and objective formulation to build up an effective optimization workflow for inverse design. Within the DGS optimization scheme, an adaptive smoothing radius and learning rate with quadratic decay is proposed to accelerate the convergence and improve the robustness by reducing the dependence of optimized design on random initialization. To make the optimized design easy to fabrication, a dynamic growth mechanism is imposed on the projection strength in parameterization to achieve a clear material layout. Moreover, we investigate the effect of material usage on optimized performance by integrating the DGS gradient with MMA optimizer to conduct a constrained optimization by adding volume constraint into the optimization formulation.

The proposed method is demonstrated on a wavelength demultiplexer design problem that aims to split 1500 nm and 1300 nm signals from an input waveguide into two output waveguides. The results show that the proposed inverse design framework using the DGS optimization method achieves an improvement of approximate 10% performance with faster convergence compared with the classical local gradient-based approaches. Given different levels of the random initial guess, the DGS \ optimization method presents a smaller variation and higher robustness on the final optimized design. Optimization with volume constraint demonstrates the final optimized device can maintain the high performance as similar to the optimized design without volume concern but achieves a significant reduction (36.7% for $\gamma = 0.3$ and 57.8% for $\gamma = 0.2$) of material usage.

The future work may potentially explore the effect of different types of random initial guess on the local minima. The current study only addresses the Gaussian noise with different standard deviation to the initial guess. It would be interesting to study the other types of random noise, for example, Perlin noise and Gabor noise as suggested in [11]. Besides, the optimized devices are composed of a small number of distinct spot materials, which may play an uncertain role in light transmission and separation. We plan to make a further investigation on the reason for these small features and explore their effect on the performance of electric field intensity, specifically when volume fraction is relatively low.

7. Data availability statement

The data of this study will be made available on request.

8. Acknowledgements

This work was supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Applied Mathematics program under contract ERKJ352 and ERKJ369; and by the Artificial Intelligence Initiative at the Oak Ridge National Laboratory (ORNL). ORNL is operated by UT-Battelle, LLC., for the U.S. Department of Energy under Contract DEAC05-00OR22725.

Credit Author Statement.

Jiaxin Zhang: conceptualization, methodology, software, validation, investigation, writing - original draft, writing- review & editing, visualization.

Sirui Bi: methodology, software, validation, investigation, writing - original draft, writing- review & editing, visualization.

Guannan Zhang: methodology, software, validation, investigation, writing - original draft, writing- review & editing, visualization, supervision, funding acquisition.t

Declaration of Competing Interest

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

References

[1] L.A. Coldren, S.W. Corzine, M.L. Mashanovitch, Diode Lasers and Photonic Integrated Circuits, Volume 218, John Wiley & Sons, 2012.
[2] D.A. Miller, Optical interconnects to electronic chips, Appl. Opt. 49 (2010) F59–F70.
[3] V.S.-Y. Lin, K. Motesharei, K.-P.S. Dancil, M.J. Sailor, M.R. Ghadiri, A porous silicon-based optical interferometric biosensor, Science 278 (1997) 840–843.
[4] B. Kress, T. Starner, A review of head-mounted displays (hmd) technologies and applications for consumer electronics, in: Photonic Applications for Aerospace, Commercial, and Harsh Environments IV, volume 8720, International Society for Optics and Photonics, p. 87200A.
[5] P. Kok, W.J. Munro, K. Nemoto, T.C. Ralph, J.P. Dowling, G.J. Milburn, Linear optical quantum computing with photonic qubits, Rev. Mod. Phys. 79 (2007) 135.
[6] A. Polman, H.A. Atwater, Photonic design principles for ultrahigh-efficiency photovoltaics, Nat. Mater. 11 (2012) 174–177.
[7] S. Molesky, Z. Lin, A.Y. Piggott, W. Jin, J. Vuckovic, A.W. Rodriguez, Inverse design in nanophotonics, Nat. Photonics 12 (2018) 659–670.
[8] D. Wu, C. Liu, Z. Xu, Y. Liu, Z. Yu, L. Yu, L. Chen, R. Li, R. Ma, H. Ye, The design of ultra-broadband selective near-perfect absorber based on photonic structures to achieve near-ideal daytime radiative cooling, Mater. Des. 139 (2018) 104–111.
[9] K.Y. Yang, J. Skarda, M. Cotrufo, A. Dutt, G.H. Ahn, M. Sawaby, D. Vercruysse, A. Arbabian, S. Fan, A. Alù, et al., Inverse-designed non-reciprocal pulse router for chip-based lidar, Nat. Photonics 14 (2020) 369–374.
[10] A.Y. Piggott, J. Lu, K.G. Lagoudakis, J. Petykiewicz, T.M. Babinec, J. Vuković, Inverse design and demonstration of a compact and broadband on-chip wavelength demultiplexer, Nat. Photonics 9 (2015) 374–377.
[11] L. Su, D. Vercruysse, J. Skarda, N.V. Sapra, J.A. Petykiewicz, J. Vuckovic, Nanophotonic Inverse Design with SPINS: Software Architecture and Practical Considerations, Applied Physics Reviews 7, 2020.
[12] J. Peurifoy, Y. Shen, L. Jing, Y. Yang, F. Cano-Renteria, B.G. DeLacy, J.D. Joannopoulos, M. Tegmark, M. Soljačić, Nanophotonic particle simulation and inverse design using artificial neural networks, Sci. Adv. 4 (2018) 1–8.
[13] J.S. Jensen, O. Sigmund, Topology optimization for nano-photonics, Laser Photonics Rev. 5 (2011) 308–321.
[14] G. Angeris, J. Vuckovic, S.P. Boyd, Computational bounds for photonic design, ACS Photonics 6 (2019) 1232–1239.
[15] T. Phan, D. Sell, E.W. Wang, S. Doshay, K. Edee, J. Yang, J.A. Fan, High-efficiency, large-area, topology-optimized metasurfaces, Light: Science & Applications 8 (2019) 1–9.
[16] L.H. Frandsen, Y. Elesin, L.F. Frellsen, M. Mitrovic, Y. Ding, O. Sigmund, K. Yvind, Topology optimized mode conversion in a photonic crystal waveguide fabricated in silicon-on-insulator material, Opt. Express 22 (2014) 8525–8532.
[17] T.W. Hughes, M. Minkov, I.A. Williamson, S. Fan, Adjoint method and inverse design for nonlinear Nanophotonic devices, ACS Photonics 5 (2018) 4781–4787.
[18] D. Yan, L. Qiu, M. Xue, Z. Meng, Y. Wang, A flexible surface-enhanced raman substrates based on cellulose photonic crystal/ag-nanoparticles composite, Mater. Des. 165 (2019) 107601.
[19] L.F. Frellsen, Y. Ding, O. Sigmund, L.H. Frandsen, Topology optimized mode multiplexing in silicon-on-insulator photonic wire waveguides, Opt. Express 24 (2016) 16866–16873.
[20] L. Su, A.Y. Piggott, N.V. Sapra, J. Petykiewicz, J. Vučković, Inverse design and demonstration of a compact on-Chip narrowband Three-Channel wavelength Demultiplexer, ACS Photonics 5 (2018) 301–305.
[21] J. Jiang, J.A. Fan, Simulator-Based Training of Generative Neural Networks for the Inverse Design of Metasurfaces, Nanophotonics 1, 2019.
[22] J.S. Jensen, O. Sigmund, Systematic design of photonic crystal structures using topology optimization: low-loss waveguide bends, Appl. Phys. Lett. 84 (2004) 2022–2024.
[23] P.I. Borel, A. Harpøth, L.H. Frandsen, M. Kristensen, P. Shi, J.S. Jensen, O. Sigmund, Topology optimization and fabrication of photonic crystal structures, Opt. Express 12 (2004) 1996–2001.
[24] T.W. Hughes, I.A. Williamson, M. Minkov, S. Fan, Forward-mode differentiation of maxwell's equations, ACS Photonics 6 (2019) 3010–3016.
[25] M. Minkov, I.A. Williamson, L.C. Andreani, D. Gerace, B. Lou, A.Y. Song, T.W. Hughes, S. Fan, Inverse design of photonic crystals through automatic differentiation, ACS Photonics 7 (7) (2020) 1729–1741, https://doi.org/10.1021/acsphotonics.0c00327.
[26] R. E. Christiansen, O. Sigmund, A tutorial for inverse design in photonics by topology optimization, arXiv preprint arXiv:2008.11816 (2020).
[27] R. E. Christiansen, O. Sigmund, "a 200 line matlab code for inverse design in photonics by topology optimization, in review (2020).
[28] Y. Jiao, S. Fan, D.A. Miller, Systematic photonic crystal device design: global and local optimization and sensitivity analysis, IEEE J. Quantum Electron. 42 (2006) 266–279.
[29] Y. Elesin, B.S. Lazarov, J.S. Jensen, O. Sigmund, Design of robust and efficient photonic switches using topology optimization, Photonics and nanostructures-Fundamentals and Applications 10 (2012) 153–165.

[30] S.D. Campbell, D. Sell, R.P. Jenkins, E.B. Whiting, J.A. Fan, D.H. Werner, Review of numerical optimization techniques for meta-device design, Optical Materials Express 9 (2019) 1842–1863.
[31] P.-I. Schneider, X. Garcia Santiago, V. Soltwisch, M. Hammerschmidt, S. Burger, C. Rockstuhl, Benchmarking five global optimization approaches for nano-optical shape optimization and parameter reconstruction, ACS Photonics 6 (2019) 2726–2733.
[32] A. Sakurai, K. Yada, T. Simomura, S. Ju, M. Kashiwagi, H. Okada, T. Nagao, K. Tsuda, J. Shiomi, Ultranarrow-band wavelength-selective thermal emission with aperiodic multilayered metamaterials designed by bayesian optimization, ACS central science 5 (2019) 319–326.
[33] J. Snoek, H. Larochelle, R. P. Adams, Practical bayesian optimization of machine learning algorithms, in: Advances in neural information processing systems, pp. 2951–2959.
[34] K. Price, R.M. Storn, J.A. Lampinen, Differential Evolution: A Practical Approach to Global Optimization, Springer Science & Business Media 2006.
[35] D. Vercruysse, N.V. Sapra, L. Su, R. Trivedi, J. Vuckovic, Analytical level set fabrication constraints for inverse design, Sci. Rep. 9 (2019) 1–7.
[36] A.Y. Piggott, J. Petykiewicz, L. Su, J. Vuckovic, Fabrication-constrained nanophotonic inverse design, Sci. Rep. 7 (2017) 1–7.
[37] A. Christ, H.L. Hartnagel, Three-dimensional finite-difference method for the analysis of microwave-device embedding, IEEE Transactions on Microwave Theory and Techniques 35 (1987) 688–696.
[38] D.M. Sullivan, Electromagnetic Simulation Using the FDTD Method, John Wiley & Sons, 2013.
[39] J.-P. Berenger, et al., A perfectly matched layer for the absorption of electromagnetic waves, J. Comput. Phys. 114 (1994) 185–200.
[40] O. Sigmund, J. Petersson, Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima, Structural optimization 16 (1998) 68–75.
[41] A.Y. Piggott, J. Petykiewicz, L. Su, J. Vuckovic, Fabrication-constrained nanophotonic inverse design, Sci. Rep. 7 (2017) 1–7.
[42] D. Vercruysse, N.V. Sapra, L. Su, R. Trivedi, J. Vuckovic, Analytical level set fabrication constraints for inverse design, Sci. Rep. 9 (2019) 1–7.
[43] G. Zhang, J. Zhang, J. Hinkle, Learning nonlinear level sets for dimensionality reduction in function approximation, in: Advances in Neural Information Processing Systems, pp. 13220–13229.
[44] T.W. Hughes, M. Minkov, I.A. Williamson, S. Fan, Adjoint method and inverse design for nonlinear nanophotonic devices, ACS Photonics 5 (2018) 4781–4787.
[45] A.G. Baydin, B.A. Pearlmutter, A.A. Radul, J.M. Siskind, Automatic differentiation in machine learning: a survey, The Journal of Machine Learning Research 18 (2017) 5595–5637.
[46] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch (2017) https://openreview.net/pdf/25b8eee6c373d48b84e5e9c6e10e7cbbbce4ac73.pdf (accessed on 12 December 2017).
[47] J. Zhang, H. Tran, D. Lu, G. Zhang, A Novel Evolution Strategy with Directional Gaussian Smoothing for Blackbox Optimization, arXiv e-prints (2020) arXiv:2002.03001.
[48] Y. Nesterov, V. Spokoiny, Random gradient-free minimization of convex functions, Found. Comput. Math. 17 (2017) 527–566.
[49] A. Quarteroni, R. Sacco, F. Saleri, Numerical Mathematics, Volume 332, Springer Science Business Media &, 2007.
[50] M. Abramowitz, I. Stegun (Eds.), Handbook of Mathematical Functions, Dover, New York, 1972.
[51] M. El-Abd, Black-Box Optimization Benchmarking for Noiseless Function Testbed Using Artificial Bee Colony Algorithm, in: Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation, GECCO '10, Association for Computing Machinery, New York, NY, USA, 2010 1719–1724.
[52] M. Jamil, X.-S. Yang, A literature survey of benchmark functions for global optimisation problems, IJMNO 4 (2013) 150–194.
[53] S. Finck, N. Hansen, R. Ros, A. Auger, Real-Parameter Black-Box Optimization Benchmarking 2009: Presentation of the Noiseless Functions, Technical Report, Citeseer, 2010.
[54] T. Salimans, J. Ho, X. Chen, I. Sutskever, Evolution strategies as a scalable alternative to reinforcement learning, arXiv preprint arXiv:1703.03864 (2017).
[55] K. M. Choromanski, A. Pacchiano, J. Parker-Holder, Y. Tang, V. Sindhwani, From complexity to simplicity: Adaptive es-active subspaces for blackbox optimization, in: Advances in Neural Information Processing Systems, pp. 10299–10309.
[56] N. Hansen, A. Ostermeier, Completely derandomized self-adaptation in evolution strategies, Evol. Comput. 9 (2001) 159–195.
[57] A. Auger, N. Hansen, A restart cma evolution strategy with increasing population size, in: 2005 IEEE congress on evolutionary computation, volume 2, IEEE, pp. 1769–1776.
[58] Y. Nesterov, V. Spokoiny, Random gradient-free minimization of convex functions, Found. Comput. Math. 17 (2017) 527–566.
[59] M.J. Powell, A view of algorithms for optimization without derivatives, Mathematics Today-Bulletin of the Institute of Mathematics and its Applications 43 (2007) 170–174.
[60] J. Kennedy, R. Eberhart, Particle Swarm Optimization, in: Proceedings of ICNN'95-International Conference on Neural Networks, Volume 4, IEEE, Pp, 1942–1948.
[61] K. Svanberg, The method of moving asymptotes—a new method for structural optimization, Int. J. Numer. Methods Eng. 24 (1987) 359–373.
[62] P.G. Constantine, Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies, SIAM, 2015.