RESEARCH ARTICLE

LASER & PHOTONICS REVIEWS

www.lpr-journal.org

Inverse Design of Photonic Systems

Benjamin MacLellan, Piotr Roztocki, Julie Belleville, Luis Romero Cortés, Kaleb Ruscitti, Bennet Fischer, José Azaña, and Roberto Morandotti**

Inverse design methods use optimization and learning algorithms to pair desired functionalities with the corresponding high-performing systems. Such methods have significant potential for discovering novel photonics solutions, with inverse design techniques already mediating significant milestones in nanophotonics, quantum optics, and lens systems. However, while computational tools for identifying optimal system parameters (i.e., component settings) have reached significant maturity, the identification of suitable system topologies (i.e., component choice and arrangement) has remained challenging, especially for the design of complex photonic schemes. Here, a framework for the inverse design of practical photonic systems is presented, capable of efficiently and automatically searching for high-performance topologies and their associated operational parameters. It is demonstrated that the approach can aid in the discovery of practical photonic systems, that are both physically feasible and non-trivial, by leveraging system-level automatic differentiation and discrete topological changes. The versatility of the platform is supported with example designs for waveform generation, noise suppression, and sensing, among others.

1. Introduction

In recent years, inverse design has emerged as a promising and effective approach for advancing and discovering new photonic schemes. In this paradigm, form follows function: the desired system functionality is first defined via an application-dependent performance metric (i.e., loss/cost/fitness/figure-of-merit function,[1] and search algorithms are then used to optimize this metric and identify high-performing systems.[2] Inverse design methods have been instrumental to recent performance records for nanophotonic devices,[2–4] meta-optics,[5,6] lens systems,[7–9] and quantum optical setups[10] – milestones enabled by concurrent advancements in fabrication capabilities and design techniques. Similar approaches have also been adopted more broadly across scientific disciplines, e.g., for the discovery of novel electronic circuits,[11,12] materials,[13,14] artificial neural network architectures,[15,16] quantum circuits,[17] biosystems,[18] and pharmaceutical compounds.[19]

The design of photonic schemes requires consideration of the system topology and its associated operational parameters. Topology corresponds to the choice of components comprising the system (e.g., sources, modulators, detectors) and how they are arranged, while operational parameters correspond to the settings of these components (e.g., source optical power, modulation patterns, detector gain). The design of such systems can be challenging (**Figure 1**): setups must often bridge the electrical and optical domains, achieve multi-parameter performance targets, and rely on the complex interplay of system processes, e.g., gain, loss, modulation, dispersion, interference, nonlinearity, etc.[20–24] As early as the 1950s, numerical approaches have been used to support photonics design,[24] ultimately evolving into today's rich ecosystem of optics simulation and optimization techniques. Nevertheless, the state-of-the-art in photonic systems often remains hampered by time-consuming design-test cycles and by the limits of human intuition,[25] especially for complex, high-dimensional design spaces.[26]

Methods to identify optimal system parameters are well-developed in photonic inverse design. Early demonstrations in optics made common use of black-box/derivative-free optimization techniques, such as genetic algorithms[27] and simplex methods[28] to identify optimal parameters for fixed system topologies. More recently, gradient-based methods (relying on computing derivatives of the performance metric with respect to system parameters) have enabled a dramatic expansion of the size and complexity achievable with inversely-designed optical systems. For example, the use of adjoint methods[29,30] has allowed impressive realizations of high-performance

B. MacLellan, P. Roztocki, J. Belleville, L. Romero Cortés, K. Ruscitti, B. Fischer, J. Azaña, R. Morandotti
Institut National de la Recherche Scientifique (INRS-EMT)
1650 Lionel-Boulet Blvd., Varennes, QC J3X 1P7, Canada
E-mail: piotr.roztocki@inrs.ca; roberto.morandotti@inrs.ca

B. MacLellan, P. Roztocki
Ki3 Photonics Technologies
2547 Rue Sicard, Montreal, QC H1V 2Y8, Canada

L. Romero Cortés
Photonics Research Labs
Universitat Politècnica de València
Camino de Vera, S/N, Valencia 46022, Spain

The ORCID identification number(s) for the author(s) of this article can be found under https://doi.org/10.1002/lpor.202300500

DOI: 10.1002/lpor.202300500

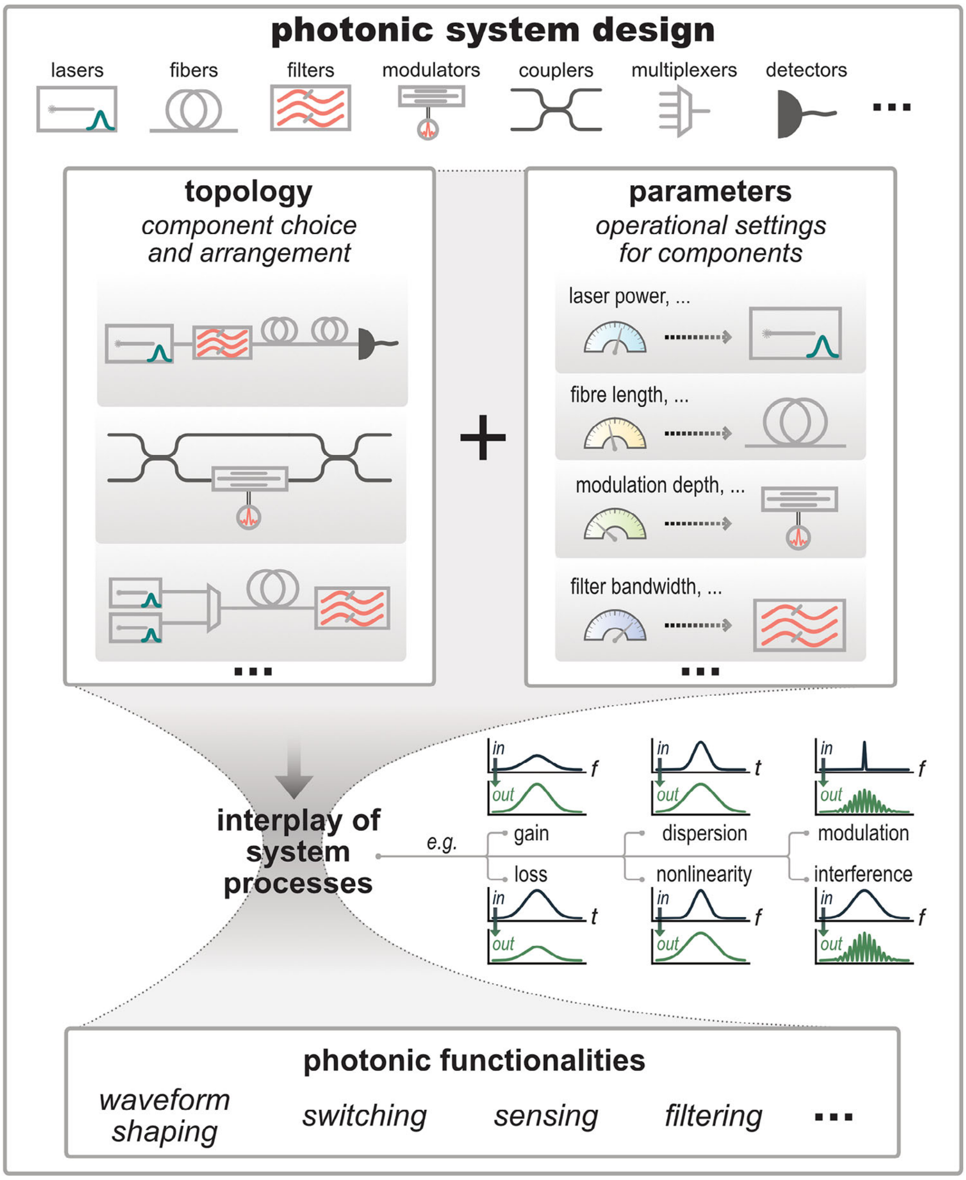

Figure 1. Design of photonic systems. Photonic systems are generally comprised of standardized components (e.g., lasers, filters, modulators – top panel) that are arranged into unique setups. Together, the system topology (left) and the set of component parameters (right) define a system, the operation of which is then mediated by an interplay of photonics processes (e.g., loss, gain, dispersion) that are unique to the setup. While high-quality designs are the basis of the many photonics technologies in use today, finding the specific system topologies and parameters that are appropriate towards a defined functionality remains difficult, especially for complex use cases. Inverse design methods aim to invert the flow illustrated in the figure, beginning with the definition of a target functionality and leveraging search/optimization algorithms to identify promising system topologies and parameters.

nanophotonic splitters/couplers[31,32] and multiplexers,[33,34] among others. More generally, automatic differentiation tools[35,36] (AD, i.e., algorithmic differentiation) compute derivatives with minimal programming and analytical overhead, even for complex models. Spurred by the growth of machine and deep learning, numerous mature and easy-to-use AD tools[37] are now available, enabling the rapid development of differentiable photonic models and associated design tools, such as gradient-based optimization techniques (e.g., gradient descent variants,[38] quasi-Newton methods[39] and gradient-based sensitivity analysis (e.g., local perturbation analysis,[40] Hessian analysis.[41] The efficient identification of optimal photonic system parameters via AD has been demonstrated for, e.g., lens systems,[42] meta-optics,[5] and nanophotonic devices.[3,4]

However, inverse design techniques that optimize system topology alongside component parameters are, by contrast, underdeveloped. The possible topologies of a photonic system correspond to a discrete combinatorial space (Figure 1a). Methods for exploring such a design space tend to be demanding, generally requiring: (i) a mechanism to sample distinct, yet

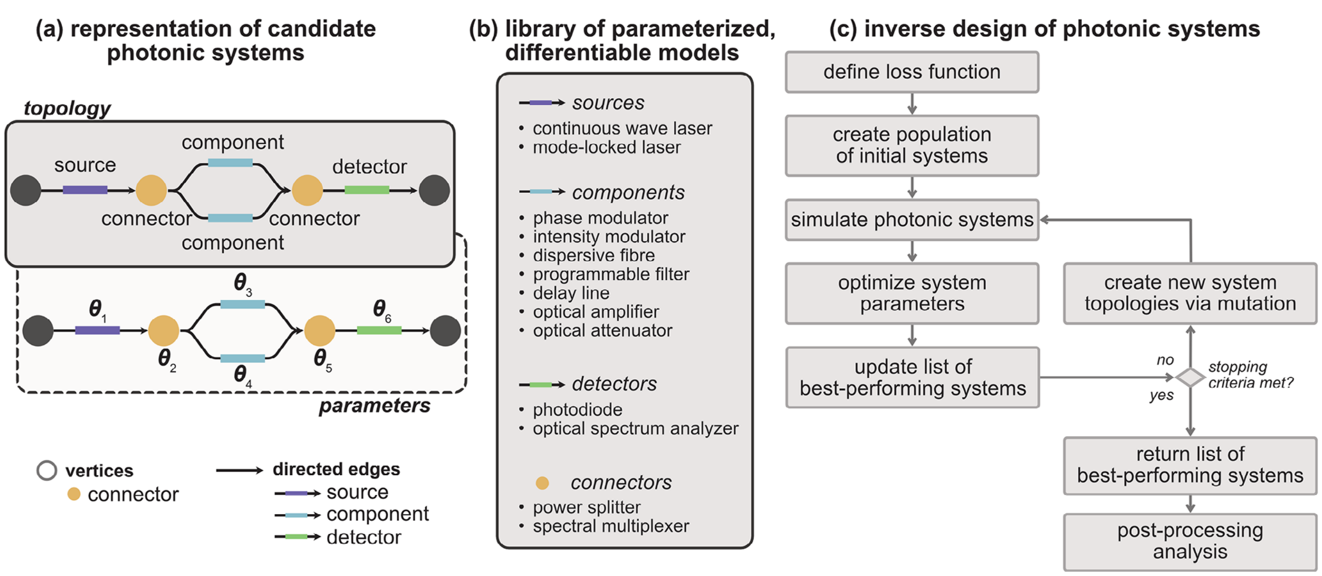

Figure 2. Inverse design framework for functional photonic systems: a) Representation of systems as directed, acyclic graphs (DAGs), with edges and nodes corresponding to different photonic elements from the b) library of photonic element models. The library is comprised of four classes: sources, detectors, components, and connectors. c) Flowchart of the automated search for new photonic signal processing systems. The design space of photonic systems is explored via iterative mutations of the DAGs, with system parameters optimized at each iteration. A list of the best-performing systems (i.e., Hall of Fame) is updated throughout the search and finally output to the user for further analysis and experimental implementation.

valid, designs; and (ii) a way to evaluate and select system topologies. Recent demonstrations of topology search include the design of, for example, quantum optics setups that produce complex entangled states,[10,43] and optical lens systems with optimal arrangements of component lenses.[7,8,44] However, such demonstrations in photonics continue to be limited to specific applications and geometries (with the majority of results also focusing on non-parametrized components), meaning that the accessible range of photonics designs remains constrained by human intuition and bias.

Here we introduce an inverse design framework for the exploration of both the topology and parameter design spaces associated with photonic systems. The guiding principles of our approach are the physical realizability of designed systems, versatility in the photonic applications targeted, and a balance between exploring and converging on designs. Towards these ends, we use a compact, general, and evolvable graph representation of photonic setups, alongside system-level differentiation capabilities for gradient-based optimization and sensitivity analysis. Within our framework, we also promote different solutions in the final systems, while also aiming to minimize the number of system components. We illustrate the effectiveness of our approach by designing example photonic systems in fiber and integrated optics. Our methodology is highly flexible, and can be adapted towards realizing free-space, fiber, integrated, and hybrid photonic platforms.

2. Methods

In our method, system topologies are represented as directed acyclic graphs (DAGs, **Figure** 2a). Graph vertices, V_i, represent components which separate and/or combine propagation paths (e.g., splitters, couplers), while graph edges, E_i, represent components that operate within a single propagation path (e.g., sources, detectors, and control elements). The direction of light propagation is represented by the edge direction (see Experimental Section). In the simulation, each vertex or edge corresponds to a particular photonics component, c_i (Figure 2b), which is modelled with a component-specific transfer function that is dependent on a vector of experimentally-tunable values, $\boldsymbol{\theta}_{c_i}$. In the present work, our component library comprises standard fiber and integrated optics components for time-frequency signal operations (see Supplementary Materials), yet the library can easily be amended, e.g., to reflect available infrastructure. A complete system description is thus provided by the graph topology $G(V, E)$ alongside the parameter vector $\boldsymbol{\theta}_G \in \mathbb{R}^{N_G}$, where N_G is the total number of system parameters. Once G and $\boldsymbol{\theta}_G$ are set, the system behaviour is simulated by composing the component transfer functions in the order defined by the graph topology. Light propagation is simulated in a feedforward manner from the source terminal vertex(s) to the sink terminal vertex(s) (solid grey circles in Figure 2a). While this representation does not encompass, e.g., loop-based structures and does not account for backwards-propagating light and reflections, it is closely tied to the physical setup, experimentally intuitive, and computationally straightforward.

Additionally, we translate the design objective or desired system functionality into a loss function $\mathcal{L}(G, \boldsymbol{\theta}_G)$, which quantifies the relative performance of candidate systems. This could be, for example, the mean-squared error (MSE) calculated between a targeted and simulated optical output, or the amount of power in a specified optical wavelength or photodiode radiofrequency band (see Experimental Section). Leveraging automatic differentiation of the optical system propagation, first- and higher-order derivatives of the loss function(s) with respect to system parameters, $\boldsymbol{\theta}_G$, can be efficiently computed. We subsequently use a

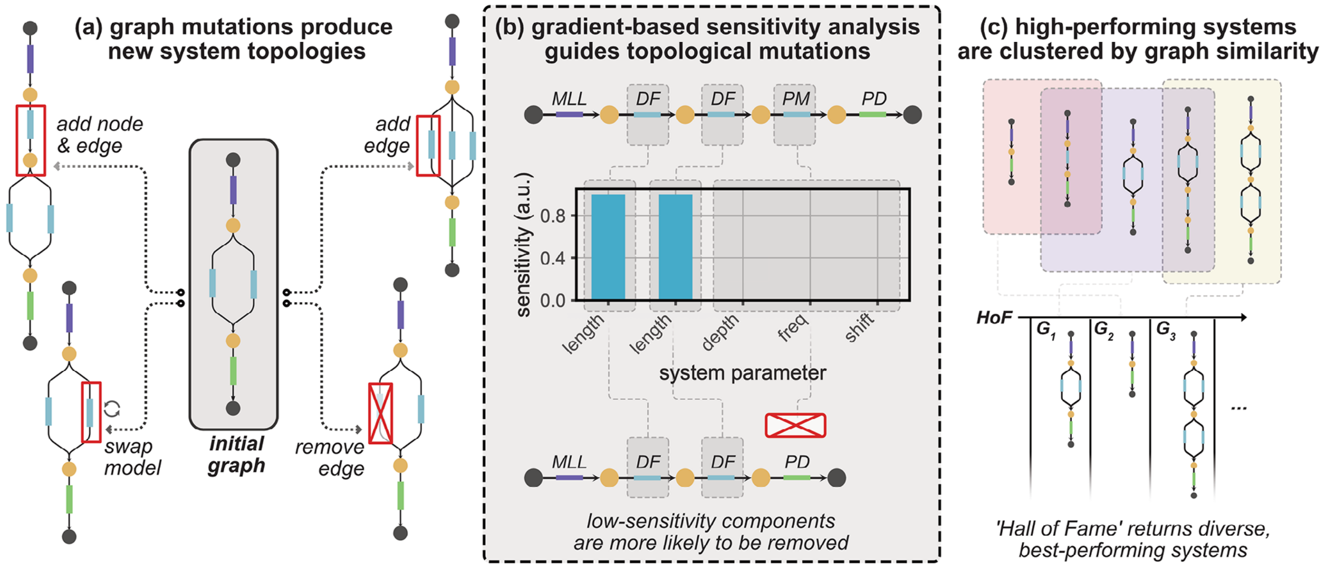

Figure 3. Methods for photonic system topology search. a) Graph mutations produce new system topologies by enacting discrete, elementary changes. These operators are stochastically applied at each generation of the evolutionary algorithm, adding, swapping, and removing components to produce the next generation of topologies. b) Automatic, derivative-based sensitivity analysis guides topology evolution by removing (with a higher probability) components that demonstrate little influence over the system. c) To promote diverse systems, candidate setups are clustered by graph similarity using the Graph Edit Distance metric. High-performing, unique systems are recorded in the Hall of Fame, which is returned to the user at the end of the search. MLL: mode-locked laser, PD: photodiode, PM: phase modulator, DF: dispersive fiber.

gradient-based sensitivity analysis (see Experimental Section) to efficiently estimate how changes in component parameters effect the overall system performance,[41] both individually and in combination, which can be used to guide experimental implementations of designs.

Our approach has two interlinked parts: parameter optimization and topology search (Figure 2c). The parameter optimization, for a fixed system topology, is a multivariate, bounded optimization problem which can be approached using many standard routines. We employ a hybrid technique comprising a genetic algorithm for a coarse-grained search of the parameter space, followed by gradient-based optimization for fine-grained convergence towards the closest local minimum, $\boldsymbol{\theta}^*_G$. We find this method to be robust in targeting a variety of functional characteristics and loss functions without requiring the tuning of hyperparameters or making application-specific changes to the algorithm (see Experimental Section).

In turn, our topology search is based on an evolutionary algorithm. We employ a set of fundamental graph mutation rules to iteratively produce new system topologies. For every iteration, each candidate DAG can undergo one of four different mutation operators (**Figure** 3a). Most importantly, these operators have been constructed to preserve the physical realizability of the produced systems, such that any system realized through iterative mutations can be mapped into an experimental setup. The probability of applying each of these mutation operators is adaptive (i.e., iteration-dependent), providing a mechanism to modify the trade-off between exploration and exploitation as the search progresses. To further guide the topology search and manage system complexity, we introduce a heuristic method based on gradient-based sensitivity analysis to remove photonic components that have little effect on the overall system performance (Figure 3b, see Supplemental Materials). Specifically, this mechanism aims to remove extraneous components that are unnecessary to realize the target functionality and which neither increase nor decrease the loss function value (thus having no selective pressure to be removed). Finally, our method prioritizes the return of diverse designs through a 'Hall of Fame' (HoF),[45] which contains the highest-performing systems found during the search. The HoF is updated during each iteration of the topology search, based on the performance of the systems after parameter optimization (see Figure 2c). To enforce diversity among entries and ensure similar designs do not overtake the HoF, a minimum Graph Edit Distance (GED)[46,47] is used as a threshold between designs (Figure 3c and Supplemental Materials).

3. Results

Bringing these aspects together, our method can autonomously explore the photonic system design space for a wide variety of target functionalities. Using the Python programming language to implement our framework (see Data Availability), we demonstrate the autonomous design of function-defined optical systems.

First, we target setups that output specific high-rate radiofrequency waveforms (**Figure** 4a, so-called photonic arbitrary waveform generation – an application case gaining traction as the bandwidth limitations of existing electronic technologies are reached.[48–52] Here, we target a 12 GHz electronic sawtooth waveform, representing our loss function in the electrical domain as the MSE between the system's photodiode output and the target waveform (see Supplemental Materials). To illustrate the evolution of topologies during the course of the search, we visualize the intermediate setups, i.e., the ancestors of the highest-

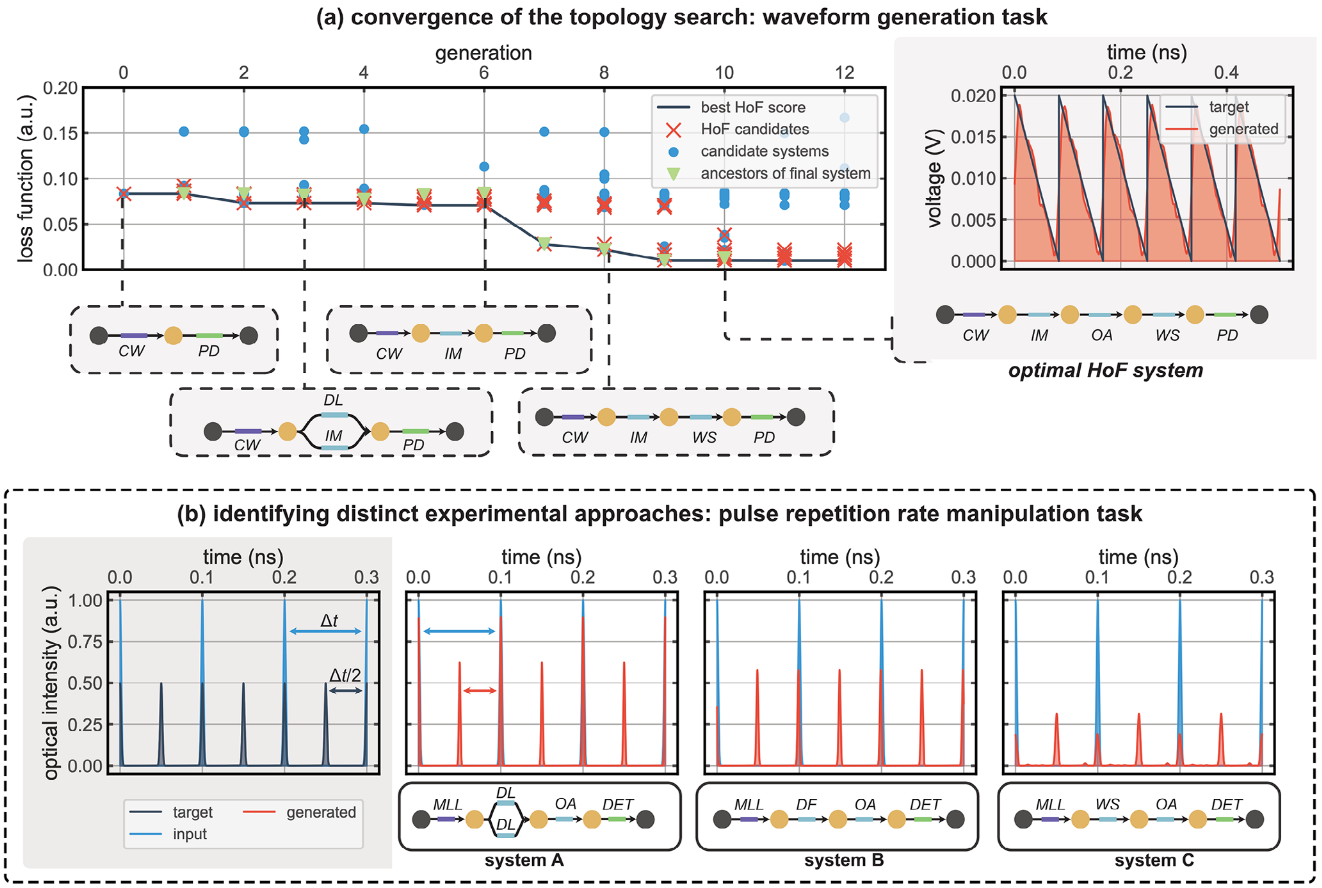

Figure 4. Topological search for photonics systems. a) Convergence of system performance through evolutionary topology search, targeting setups capable of producing a specific RF waveform. The optimization begins with a population of identical, minimal optical systems (top left), and as new topologies are generated via the mutation operators, components which enable the desired optical effect are added iteratively. Here we show the loss value of all system topologies in each generation (solid blue circles, total of 12 generations), the current entries to the HoF (red crosses), as well as the ancestors of the final, highest-performing system (green triangles). b) Selected systems for pulse repetition rate manipulation, suggested by our framework. These reproduce a variety of well-known experimental techniques, including (from left-to-right), split-and-delaying of a pulse train, controlled pulse dispersion via the temporal Talbot effect, and line-by-line amplitude and phase filtering. CW: continuous wave laser, MLL: mode-locked laser, PD: photodiode, DL: delay line, IM: intensity modulator, WS: waveshaper, OA: optical amplifier, DF: dispersive fiber.

performing system (Figure 4a). We note that the designs suggested by the framework make use of operating principles that are well-established in experimental literature,[49] without any explicit introduction of such concepts to the algorithm. The total runtime for the search procedure is 16 minutes (all reported runtime results are for a standard desktop workstation, see Supplemental Materials).

Next, we target the design of photonic systems that manipulate the repetition rate of a pulsed laser – a common signal processing task required for applications such as pump-probe measurements,[53] arbitrary waveform generation,[54] and optical signal denoising.[55] We present three high-performing system designs from the output HoFs of two search runs (Figure 4b, 63 minutes total search time), aiming for setups that halve the pulse period and peak power (see Supplemental Materials). The systems found are reflective of common and niche designs found in the literature, relying on both more and less intuitive techniques: A) a split-and-delay method using a two-arm Mach-Zehnder interferometer;[55] B) the temporal Talbot effect enacted via dispersive propagation of light;[53] and C) the line-by-line spectral amplitude and phase shaping of the pulse train via a programmable filter.[56] The system parameters found for these designs match well-known, analytically-optimal values: the optical delay in system A is almost exactly equivalent to half the input pulse period, and the fiber length parameter in system B satisfies the temporal Talbot condition for doubling the pulse repetition rate.

Further, by making the photonic simulations fully differentiable, our framework can target loss functions made uniquely accessible via automatic differentiation. By using AD, loss functions can incorporate arbitrary derivative terms corresponding to rates of change for system observables with respect to system parameters. As an example, we target the design of an 'optical sensor', by minimizing the loss function $\mathcal{L}(G,\theta_G) = -\left|\frac{\partial P_{\text{avg}}}{\partial \theta_\varphi}\right|_{\theta_G}$ (equivalent to maximizing the system's phase sensitivity), where P_{avg} is the average optical power measured at the system output and θ_φ is the optical phase-shift parameter imparted by a phase-

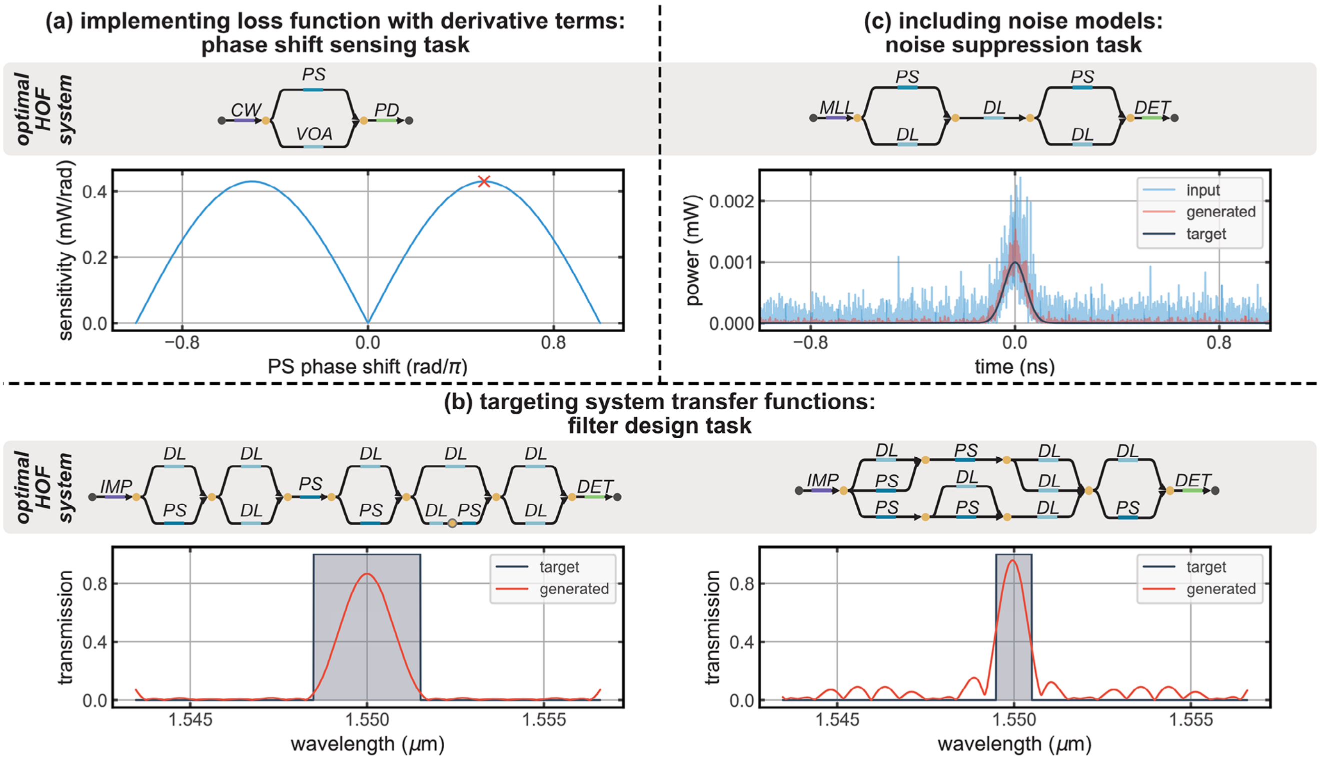

Figure 5. Targeting extended photonics functionalities. a) With automatic differentiation, system functionalities based on derivatives, such as sensitivity, can be incorporated into the loss function. b) The impulse response of candidate systems is used to converge on the design of the targeted rectangular spectral filters. The optimal systems suggested by our framework make use of multiple interference effects, proposing a FIR filter of concatenated Mach-Zehnder interferometers (*58*). c) Random additive noise models are used to simulate corrupted signals. Here, a system capable of recovering a true signal under noise is targeted. All systems shown are the highest scoring graphs in the respective HoF. CW: continuous wave laser, MLL: mode-locked laser, PD: photodiode, PS: phase shifter, DL: delay line, VOA: variable optical attenuator, IMP: impulse response.

shifting (PS) element (constrained to remain in all candidate designs in this search). The best-performing system returned by our framework corresponds to a Mach-Zehnder interferometric setup (**Figure** 5a, 23 minutes total search time) that obeys well-known experimental criteria for maximizing phase sensitivity: losses between the arms are balanced, the central phase parameter is set to $\theta_\varphi = \pi/2$ (a maximum sensitivity point), and the input laser's optical power is maximized to its upper bound. This simple demonstration suggests that our method can aid in the design of systems that have, for example, high sensitivities and can act as sensors, or, on the other hand, have low sensitivities and are stable against imperfections and perturbations.

Our inverse design framework can also target systems that enact specific optical transfer functions. We illustrate this with the design of an optical filter, obtained by optimizing the impulse response of candidate systems in the spectral domain (Figure 5b, 199 minutes total search time). Note that in Figure 5b,c, we constrain the components accessible to our software implementation to those that are well-established in the integrated photonics platform, as we found that the path coherence demanded by the suggested designs could prove experimentally challenging in fiber[57] (see Experimental Section). In this example, the suggested filter designs are comprised of concatenated Mach-Zehnder interferometer structures (Figure 5b left). We note that using more search iterations yields yet more exotic topologies (Figure 5b right).

Finally, while automatic sensitivity analysis techniques (see Supplemental Materials) are beneficial in estimating how parameters influence the loss function, they do not encompass all non-idealities encountered in real photonic systems, e.g., stochastic thermal, mechanical, and electronic noise processes. Towards considering such factors, for active photonic component models (e.g., laser sources, photodiodes, and amplifiers) we incorporate an additive Gaussian noise model. As an example of its use (and its compatibility with AD, see Experimental Section), we target the recovery of an input noisy signal (Figure 5c). The highest-scoring HoF systems here (and in Figure 5b, 11 minutes total search time) are seen to reflect the operational principles of lattice Mach-Zehnder interferometer filters.[58]

4. Conclusion

In conclusion, we have demonstrated a novel framework for the inverse design of photonic systems, which is based on an evolvable graph representation of system topologies alongside system-level automatic differentiation. The method is general and flexible – a broad range of photonic functionalities can be targeted with minimal changes to the underlying system representations or algorithms. Here, we demonstrate its use in identifying diverse photonic systems for optical waveform generation, filtering, and sensing tasks. Furthermore, we compare our model with spectral shaping measurements, finding very good agreement with the related experimental results (see Supplemental Materials). The identified systems match with both well-known and niche experimental techniques in the literature, illustrating the ability of our approach to discover design classes without explicit programming. To further increase access to computer-inspired operating principles (i.e., systems that use optical processes in a novel or surprising way), future work should reduce the domain expertise injected into the algorithm,[59] e.g., through more general model representations/parameterizations and new topology sampling methods.

Our framework can be readily extended to the design of systems based on other optical degrees-of-freedom (e.g., polarization, orbital angular momentum) or component platforms. In particular, the approach can be adapted towards photonic systems in free-space, fiber, integrated, and hybrid optics by tailoring the component library and objective metrics. As well, we foresee that it can be extended to account for nonlinear optical phenomena and cavity structures (for example, by using more sophisticated numerical approaches like the split-step Fourier method or Lugiato–Lefever equation[60,61]). We note that the software can be easily adapted to take into account hardware-specific considerations or available infrastructures by, e.g., constraining the model library, bounding the component parameters, or implementing additional rule-sets into the topology evolution. Such extensions would be particularly suited to the design of photonic integrated circuits,[62] where fabricable components are often limited to well-defined building block elements (for example, modulators, delays, couplers). Future work should also aim to merge component-level and system-level inverse design methods through an end-to-end differentiable design flow – which would provide an invaluable tool in the search for novel computer-inspired photonic solutions.

5. Experimental Section

Software Implementation: The inverse design framework was realized in the Python programming language (see Data Availability). All results presented here were based on designs suggested autonomously by this implementation. The topology search leverages parallel processing, such that the parameter optimization and sensitivity analysis of each system were computed in parallel on separate cores. All numerical runs were performed on a 16 CPU core Intel workstation with 16 GB RAM.

Modelling of Photonic Systems: As part of the approach, photonic signal processing systems were represented as graphs which were directed (specifying the direction of optical propagation), acyclic (constraining the photonic circuits to be feed-forward, i.e., disallowing optical cavities, which would require more sophisticated simulations), and may contain multiple edges between the same nodes (i.e., a multi-graph, which represent interferometric paths). Component models can be defined in both the temporal and spectral domains directly, as derivative terms can be tracked through complex-valued functions such as the Fast-Fourier and Inverse Fast-Fourier Transforms, providing efficient system-level differentiation of the optical propagation. Simulations consider single-mode, axial propagation of the time-dependent electro-magnetic field, $E(t)$, and assume that polarization and nonlinear optical effects can be ignored. The components were all modelled with appropriate per-element losses. The photonic models were grouped into four classes: sources, detectors, connectors, and components (see Supplementary Text). This allows for optical elements of the same class to be exchanged by the graph mutation operators, while maintaining the physical realizability of the system. Terminal nodes (dark grey circles) simply define the start and end of propagation. The order in which components were simulated was determined by graph topology. Stochastic noise can be included on active photonic elements (e.g., lasers, amplifiers) via additive Gaussian noise models, sampled as a pseudo-random vector and added to the electro-magnetic field. During automatic differentiation, these noise vectors were treated as constants, yet resampled periodically during the optimization procedure.

Loss Functions and Parameter Optimization: A loss function summarizes the targeted objective of the design. In the first examples presented (Figure 4), the Euclidean distance was minimized (i.e., the mean-squared error, MSE) between a targeted, time-dependent optical intensity waveform and the waveform generated by the design. For a photonic system graph G with parameters θ_G, the loss function takes the form,

$$\mathcal{L}\left(G, \theta_G\right) = \frac{1}{T} \int_T \left(I_{\text{target}}\left(t\right) - I_{\text{generated}}\left(t,\ G,\ \theta_G\right)\right)^2 \mathrm{d}t \tag{1}$$

where I_{target} ($I_{\text{generated}}$) was the target (generated) optical intensity, $I(t) = |E(t)|^2$, and T was the temporal simulation window. Prior to computing this distance function, any global phase shift between the target and generated waveforms was compensated for by matching the optical phases at the fundamental harmonic frequency of the target waveform. This loss function was also used to optimize electronic waveforms by replacing the optical intensity terms with target and generated time-domain radio-frequency signals measured at the terminating photodiode (as in Figure 4a). The complexity of computing the gradient of this loss function $\frac{\partial \mathcal{L}(G,\theta_G)}{\partial \theta_G}\Big|_{\theta_G^*}$ via reverse-mode automatic differentiation scales approximately linearly with the number of parameters.[35] For coarse optimization of the parameters, a standard genetic algorithm was used,[63] which was then followed by gradient-based optimization via the limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm.[39]

Gradient-Based Sensitivity Analysis: In real systems, parameters often change over time and vary between implementations due to, e.g., finite fabrication tolerances and component aging. Local sensitivity analysis was used to estimate the impact of parameter perturbations on the system performance, providing insight into the operation and the limitations of designs. Using the gradient vector for sensitivity analysis has limitations in that it does not characterize sensitivities at stationary points (e.g., loss function minima) and only quantifies the effect of each parameter in isolation. The second-order derivative Hessian matrix, $H_{\boldsymbol{G}}$, a real, symmetric matrix with elements $H_{ij} = \partial^2 \mathcal{L}/\partial\theta_i\partial\theta_j$, was further used to estimate how combinations of parameters influence system performance.[41] Eigen-decomposition of $H_{\boldsymbol{G}}$ can be used to extract interpretable, experimentally-relevant information about the underlying operation of the system. An example of such a Hessian-based analysis is reported in the Supplementary Text (see Figures S1 and S2, Supporting Information) and show that, besides providing useful characterization, the method can also uncover system design redundancies, which heuristically was used when iteratively mutating the topologies.

Topology Search: New system topologies were generated by iteratively applying graph mutations. At each generation of the evolutionary algorithm, graphs which seed the next population were selected via a tournament selection process.[63] This method was beneficial over, e.g., fitness proportional selection, as it was invariant to the scale of the loss function. The top 5% of graphs were copied, unchanged, to the next population, which ensures that these graphs have further opportunities to improve and that their genetic information was not lost (a mechanism known as elitism).

System topologies were altered via the following graph mutations: add a node and an edge ($M_{\text{addNodeEdge}}$), add an edge between two existing nodes (M_{addEdge}), remove an edge ($M_{\text{removeEdge}}$), or swap a photonic element model on a given node or edge for another of the same device class ($M_{\text{swapModel}}$). The first two mutations, $M_{\text{addNodeEdge}}$ and M_{addEdge}, are equivalent to adding a new optical element in series and in parallel (i.e., a cascaded and an interferometric-type setup), respectively. The topology search was initialized with minimal systems, for example those composed only of a single optical source and detector. As the optimization progresses, new photonic elements were automatically added, changed, and removed from the system(s) by stochastically selecting and applying these mutations. The nodes and/or edges in the graph on which the mutations were enacted, as well as which model was added, were selected with uniform probability – with the exception of $M_{\mathrm{removeEdge}}$, which is discussed below.

Both mutations $M_{\mathrm{addNodeEdge}}$ and M_{addEdge} expand the size of the graph (increase the cardinality of the edge and/or node set), while $M_{\mathrm{removeEdge}}$ contracts it (decreases the cardinality of the edge and/or node set). Thus, by balancing the relative probability of selecting these mutations, one can, to some extent, control the trade-off between exploration and exploitation – i.e., whether the algorithm tries new designs or aims to improve already discovered ones. An adaptive mutation probability was used, which trades off between exploration and exploitation during the topology search process. Initially, the probabilities of adding components, $\mathrm{Pr}_0(M_{\mathrm{addNode}})$ and $\mathrm{Pr}_0(M_{\mathrm{addNodeEdge}})$, were high while the probability of removing components, $\mathrm{Pr}_0(M_{\mathrm{removeEdge}})$, was zero. As the optimization progresses, these probabilities trade off linearly with the number of generations: the probability of expanding the graph size decreases while the probability of contracting the graph size increases. Specifically, $\mathrm{Pr}_i\ (M_{\mathrm{addNode}}) = \mathrm{Pr}_i\ (M_{\mathrm{addNodeEdge}}) = 0.25(1 - i/N)$, $\mathrm{Pr}_i\ (M_{\mathrm{removeEdge}}) = 0.5(i/N)$ where i is the iteration.

During the topology selection stage, each element within a DAG was assigned a 'total sensitivity score' based on the Hessian matrix. Elements with a lower sensitivity score were removed with higher probability (Figure 3b). The sensitivity score was defined as

$$S_C = \frac{\sum_{i=1}^{N_c} \sum_{j=1}^{N_c} \sigma_i \sigma_j H_{ij}}{N_C^2} \tag{2}$$

where i, j are indices corresponding to the N_c real and bounded parameters of the component c. This total sensitivity score was used to guide which element was removed, but does not alter the overall probability $\mathrm{Pr}_i(M_{removeEdge})$. Scaling factors, σ_i, were included to account for different parameter units/scales and reflect parameter imprecisions. This removal was stochastically chosen and uses a tournament-style selection, such that the probability of an element being removed increases with a lower sensitivity score.

When updating the HoF, which tracks the best-performing systems so far, entries were clustered by their Graph Edit Distance (GED),[46] which was determined via a weighted sum of the number of graph edit operations required for one graph to transform into another. A threshold GED value was defined such that there was a minimum difference between all pairs of systems in the HoF (a threshold of 1.5 was used in the results shown here).

Targeted Photonic Applications: The following search settings were used for the example tasks demonstrated in this work. The waveform generation and pulse repetition rate manipulation examples (Figure 4a,b) use the MSE between the target and generated waveforms as their loss functions; the former was calculated in the electronic domain (i.e., after optical-to-electronic conversion, accounting for the photodetector bandwidth), while the latter was calculated directly in the optical intensity domain. It was assumed that the input pulse train was unchirped, Gaussian, and did not produce nonlinear optical interactions during propagation. The pulsed laser was constrained to remain fixed in the setup through the entirety of the topology search. Optimal systems in Figure 4b were selected from the HoFs of 2 runs of the algorithm with the same settings (8 generations, population size of 8, HoF size of 8).

The example in Figure 5a uses the detection sensitivity as its loss function, i.e. the derivative of the average optical power measured at the terminating graph node with respect to the phase shifting (PS) element in the setup. The PS element was constrained to remain within the setup, but the phase shift that it imparts was adjusted during parameter optimization. Finally, both the filter design task (Figure 5b) and the noise suppression task (Figure 5c) use the MSEs between the target and output profiles as loss functions; the former was defined in the spectral domain and the latter in the temporal domain. In these examples (Figure 5b,c), the photonic components accessible during the topology search were constrained to a limited set, comprising delay lines, phase shifting elements, and variable couplers (all well-established in the integrated photonics platform). The use of this constraint illustrates how a designer could inject domain knowledge of available infrastructure into the framework.

Supporting Information

Supporting Information is available from the Wiley Online Library or from the author.

Acknowledgements

The authors thank James van Howe for fruitful discussions. B.M. acknowledges funding through the NSERC CGS-M program. P.R. acknowledges funding from the NSERC Vanier CGS and Novascience programs. J.B. and K.R. acknowledge funding from the NSERC USRA program. R.M. acknowledges support from the NSERC Strategic, Alliance and Canada Research Chair Grant Programs.

Conflict of Interest

B.M., P.R., L.R.C., B.F., J.A., and R.M. were inventors on a patent application related to the presented work filed by Institut National de la Recherche Scientifique (US17/303,705). The authors declare no other competing interests.

Author Contributions

B.M. and P.R. contributed equally to this work. B.M., P.R. performed conceptualization. B.M., P.R., J.B., K.R. performed implementation. B.M., P.R., L.R.C., B.F. performed analysis. J.A., R.M performed supervision. B.M., P.R. wrote the original draft. B.M., P.R., J.B., L.R.C., K.R., B.F., J.A., R.M. reviewed and edited the original draft.

Data Availability Statement

The software implementation of our framework, which supports the findings of this study, is openly available the Github repository https://github.com/benjimaclellan/aesop, reference number [64].

Received: June 1, 2023
Revised: January 4, 2024
Published online: February 20, 2024

[1] I. Griva, S. G. Nash, A. Sofer, in *Linear and Nonlinear Optimization*, 2nd ed., SIAM, Philadelphia **2009**.
[2] S. Molesky, Z. Lin, A. Y. Piggott, W. Jin, J. Vucković, A. W. Rodriguez, *Nat. Photonics* **2018**, *12*, 659.
[3] Z. Lin, C. Roques-Carmes, R. Pestourie, M. Soljačić, A. Majumdar, S. G. Johnson, *Nanophotonics* **2021**, *10*, 1177.
[4] T. W. Hughes, M. Minkov, I. A. D. Williamson, S. Fan, *ACS Photonics* **2018**, *5*, 4781.
[5] S. Colburn, A. Majumdar, *Commun. Phys.* **2021**, *4*, 65.
[6] W. Hadibrata, H. Wei, S. Krishnaswamy, K. Aydin, *Nano Lett.* **2021**, *21*, 2422.
[7] G. König, G. König, C. Fu, C. Fu, J. Stollenwerk, J. Stollenwerk, C. Holly, P. Loosen, P. Loosen, *Opt. Express* **2021**, *29*, 39027.
[8] X. Chen, K. Yamamoto, *J. Mod. Opt.* **1997**, *44*, 1693.
[9] K. Höschel, V. Lakshminarayanan, *J. Opt.* **2019**, *48*, 134.

[10] M. Krenn, M. Malik, R. Fickler, R. Lapkiewicz, A. Zeilinger, *Phys. Rev. Lett.* **2016**, *116*, 090405.
[11] C.-Y. Chen, R.-C. Hwang, *Exp. Syst. Appl.* **2009**, *36*, 634.
[12] A. Thompson, in *Hardware Evolution: Automatic Design of Electronic Circuits in Reconfigurable Hardware by Artificial Evolution*, Springer, Berlin, Germany **2012**.
[13] B. Sanchez-Lengeling, A. Aspuru-Guzik, *Science* **2018**, *361*, 360.
[14] J. G. Freeze, H. R. Kelly, V. S. Batista, *Chem. Rev.* **2019**, *119*, 6595.
[15] K. O. Stanley, R. Miikkulainen, *Evol. Comput.* **2002**, *10*, 99.
[16] E. Real, C. Liang, D. R. So, Q. V. Le, *Proceedings of the 37th International Conference on Machine Learning*, JMLR, **2020**, pp. 8007–8019.
[17] L. Mortimer, M. P. Estarellas, T. P. Spiller, I. D'Amico, *Adv. Quantum Technol.* **2021**, *4*, 2100013.
[18] M. HamediRad, R. Chao, S. Weisberg, J. Lian, S. Sinha, H. Zhao, *Nat. Commun.* **2019**, *10*, 5150.
[19] K. Kim, S. Kang, J. Yoo, Y. Kwon, Y. Nam, D. Lee, I. Kim, Y.-S. Choi, Y. Jung, S. Kim, W.-J. Son, J. Son, H. S. Lee, S. Kim, J. Shin, S. Hwang, *npj Comput. Mater.* **2018**, *4*, 67.
[20] A. Guo, G. J. Salamo, D. Duchesne, R. Morandotti, M. Volatier-Ravat, V. Aimez, G. A. Siviloglou, D. N. Christodoulides, *Phys. Rev. Lett.* **2009**, *103*, 093902.
[21] B. Crockett, L. Romero Cortés, S. R. Konatham, J. Azaña, *Nat Commun.* **2021**, *12*, 2402.
[22] A. Ruehl, M. J. Martin, K. C. Cossel, L. Chen, H. McKay, B. Thomas, C. Benko, L. Dong, J. M. Dudley, M. E. Fermann, I. Hartl, J. Ye, *Phys. Rev. A* **2011**, *84*, 011806.
[23] T. Baba, *Nat. Photon.* **2008**, *2*, 465.
[24] D. P. Feder, *Appl. Opt.* **1963**, *2*, 1209.
[25] G. S. Adams, B. A. Converse, A. H. Hales, L. E. Klotz, *Nature* **2021**, *592*, 258.
[26] M. Christopher, *Pattern Recognition and Machine Learning*, Springer, New York, NY **2006**, https://link.springer.com/book/9780387310732.
[27] M. M. Spuhler, B. J. Offrein, G.-L. Bona, R. Germann, I. Massarek, D. Erni, *J. Lightwave Technol.* **1998**, *16*, 1680.
[28] R. J. Koshel, *Opt. Lett.* **2005**, *30*, 649.
[29] L. Su, D. Vercruysse, J. Skarda, N. V. Sapra, J. A. Petykiewicz, J. Vučković, *Appl. Phys. Rev.* **2020**, *7*, 011407.
[30] C. M. Lalau-Keraly, S. Bhargava, O. D. Miller, E. Yablonovitch, *Opt. Express* **2013**, *21*, 21693.
[31] A. Michaels, E. Yablonovitch, *Opt. Express* **2018**, *26*, 4766.
[32] 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, J. Vučković, *Nat. Commun.* **2019**, *10*, 3309.
[33] A. Y. Piggott, J. Lu, K. G. Lagoudakis, J. Petykiewicz, T. M. Babinec, J. Vučković, *Nat. Photonics* **2015**, *9*, 374.
[34] L. Su, A. Y. Piggott, N. V. Sapra, J. Petykiewicz, J. Vučković, *ACS Photonics* **2018**, *5*, 301.
[35] A. Griewank, A. Walther, in *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*, 2nd Ed., SIAM, Philadelphia **2008**.
[36] R. E. Wengert, *Commun. ACM.* **1964**, *7*, 463.
[37] C. C. Margossian, *WIREs Data Min. Knowl. Discov.* **2019**, *9*, e1305.
[38] S. Ruder, An overview of gradient descent optimization algorithms **2017**.
[39] D. C. Liu, J. Nocedal, *Math. Program.* **1989**, *45*, 503.
[40] J. P. C. Kleijnen, *Eur. J. Operat. Res.* **2005**, *164*, 287.
[41] X. Xing, R. Rey-de-Castro, H. Rabitz, *New J. Phys.* **2014**, *16*, 125004.
[42] J.-B. Volatier, Á. Menduiña-Fernández, M. Erhard, *J. Opt. Soc. Am. A* **2017**, *34*, 1146.
[43] X. Gao, M. Erhard, A. Zeilinger, M. Krenn, *Phys. Rev. Lett.* **2020**, *125*, 050501.
[44] L. Sun, B. Guenter, N. Joshi, P. Therien, J. Hays, Lens Factory: automatic lens generation using off-the-shelf components **2015**.
[45] F. A. Fortin, F. M. De Rainville, M. A. Gardner, M. Parizeau, C. Gagné, *J. Mach. Learn. Res.* **2012**, *13*, 2171.
[46] X. Gao, B. Xiao, D. Tao, X. Li, *Pattern Anal. Appl.* **2010**, *13*, 113.
[47] Z. Abu-Aisheh, R. Raveaux, J.-Y. Ramel, P. Martineau, presented at *4th International Conference on Pattern Recognition Applications and Methods* Lisbon, Portugal **2015**.
[48] J. Hu, J. He, J. Liu, A. S. Raja, M. Karpov, A. Lukashchuk, T. J. Kippenberg, C.-S. Brès, *Nat. Commun.* **2020**, *11*, 4377.
[49] S. T. Cundiff, A. M. Weiner, *Nat. Photon* **2010**, *4*, 760.
[50] A. Rashidinejad, A. M. Weiner, *J. Lightwave Technol.* **2014**, *32*, 3383.
[51] S. R. Konatham, R. Maram, L. Romero Cortés, J. H. Chang, L. Rusch, S. LaRochelle, H. Guillet de Chatellus, J. Azaña, *Nat. Commun.* **2020**, *11*, 3309.
[52] W. Li, W. T. Wang, N. H. Zhu, *IEEE Photonics J.* **2014**, *6*, 1.
[53] R. Maram, L. R. Cortés, J. V. Howe, J. Azaña, *J. Lightwave Technol.* **2017**, *35*, 658.
[54] Q. Xie, C. Shu, *IEEE Photonics Technol. Lett.* **2018**, *30*, 242.
[55] C. W. Siders, J. L. W. Siders, A. J. Taylor, S.-G. Park, A. M. Weiner, *Appl. Opt.* **1998**, *37*, 5302.
[56] J. Caraquitena, Z. Jiang, D. E. Leaird, A. M. Weiner, *Opt Lett* **2007**, *32*, 716.
[57] P. Roztocki, B. MacLellan, M. Islam, C. Reimer, B. Fischer, S. Sciara, R. Helsten, Y. Jestin, A. Cino, S. T. Chu, B. Little, D. J. Moss, M. Kues, R. Morandotti, *Laser Photonics Rev.* **2021**, *15*, 2000524.
[58] K. Jinguji, M. Kawachi, *J. Lightwave Technol.* **1995**, *13*, 73.
[59] M. Krenn, J. S. Kottmann, N. Tischler, A. Aspuru-Guzik, *Phys Rev X* **2021**, *11*, 031044.
[60] J. M. Dudley, G. Genty, S. Coen, *Rev. Mod. Phys.* **2006**, *78*, 1135.
[61] L. A. Lugiato, R. Lefever, *Phys. Rev. Lett.* **1987**, *58*, 2209.
[62] W. Bogaerts, D. Pérez, J. Capmany, D. A. B. Miller, J. Poon, D. Englund, F. Morichetti, A. Melloni, *Nature* **2020**, *586*, 207.
[63] D. Simon, in *Evolutionary Optimization Algorithms*, Wiley, Berlin, Germany **2013**.
[64] AESOP Software, https://github.com/benjimaclellan/aesop (accessed: February 2204).