

On the Role of Bayesian Learning for Electronic Design Automation: A Survey

Federico Garbuglia, Dirk Deschrijver, *Senior Member, IEEE*, Tom Dhaene, *Senior Member, IEEE*,

Abstract—Machine learning algorithms and artificial intelligence (AI) have driven a revolution in many scientific and engineering fields. Now, they are slowly making their way into the field of electronic design automation (EDA). Bayesian learning (BL) has emerged as a promising approach to modeling and optimization for systems when the data is scarce and computational resources are strictly limited. Using BL, it is possible to build more accurate models by collecting data samples adaptively. This paper presents an overview of popular machine learning strategies that have been developed to automate different steps of the electronic design process. Particular attention is paid to Bayesian learning and how it is applied to the modeling and optimization of analog devices. In addition, this paper analyses some recent BL variants proposed to tackle specific machine learning issues and solve application-specific tasks.

Index Terms—Electronic design automation (EDA), machine learning (ML), Bayesian learning (BL), Gaussian processes (GP).

I. INTRODUCTION

Designing complex systems is a resource-intensive process that crosses many disciplines and many levels of abstraction. This is definitely the case for electronic design: an electronic product has to undergo numerous design iterations before all functional, manufacturing, and legal requirements are met. Meeting these stringent requirements is only possible by continuously increasing the technological complexity, which is itself a source of ever-evolving challenges. For example, the high integration of modern microwave systems has caused critical issues in power and thermal management. Moreover, the coexistence of numerous devices in modern user environments imposes electromagnetic compatibility (EMC), power integrity, and signal integrity (SI) constraints early on in the design process. Due to the design complexity, a high number of design variables, specifications, and performance metrics have to be considered at any stage of the design process. Thus, for given technological capabilities, the optimal design has to be identified within an extremely large and high-dimensional space of possible designs [1]. Let us consider a simple microwave filter (Fig.1). First, an initial topology can be selected and parametrized with respect to one or many geometrical variables (L in this example) or material variables such as substrate permittivity and metal conductivity (Fig.1a). Next, it is interesting to find the variable values for which the

design satisfies the specifications (Fig.1b), such as minimum attenuation, bandwidth, central frequency, or losses.

This search problem is typically more demanding for analog devices than for digital ones. The size and the topological configuration of analog devices may require encodings in larger design spaces, which are constituted by tens or hundreds of design variables. In addition, the specifications of analog devices are highly diverse, depending on the application. This prevents the standardization of design variables and requirements and it constitutes an additional obstacle to design automation.

The typical analog design workflow follows a hierarchical approach, as widely discussed in [4]: the full system is decomposed into component blocks defined at different levels of abstraction. The components are organized according to a chosen topology that is expected to meet the system specifications. The abstraction levels can be descended using a top-down path: after forwarding the specifications to single components, each one can be realized at a lower abstraction level. On the other hand, a bottom-up path consists of creating a low-level component layout and extracting an equivalent higher-level representation. Next, it is possible to generate and verify a topology that uses the equivalent high-level components. Figure 2 illustrates the main design steps that constitute the top-down and bottom-up path between the abstraction levels. The full design workflow requires a sequence of top-down and bottom-up steps until a layout of manufacturable components is obtained that satisfies all the specifications. To date, no known technique can dive into all the abstraction levels and automate the full analog design workflow. Human expertise appears still indispensable for this purpose. Nonetheless, the research effort in the last years has been directed to the automation of many steps of the workflow. In fact, even partial automation strategies can significantly speed up the design by reducing the number of design steps or by reducing the cost of each step. For this purpose, several Machine Learning (ML) techniques have been applied to EDA problems. Section II introduces the main approaches followed by ML algorithms, including an overview of recent ML techniques that have been applied to different steps of the analog design flow, such as data-efficient modeling and optimization techniques based on Bayesian learning (BL). Then, Section III explains the working principles of Bayesian learning by focusing on its usage for analog device sizing. In addition, Section IV discusses recent variants of BL, that have been proposed to mitigate the main issues of BL and to solve more specific modeling and optimization tasks. Finally, Section V suggests promising research directions toward the development of more

F. Garbuglia, D. Deschrijver, and T. Dhaene are with the Department of Information Technology, Ghent University-imec, Technologiepark-Zwijnaarde 126, 9052 Ghent, Belgium e-mail: federico.garbuglia@ugent.be.

This research received funding from the Flemish Government via the AI Research Programme and the Fonds Wetenschappelijk Onderzoek (FWO) programme.

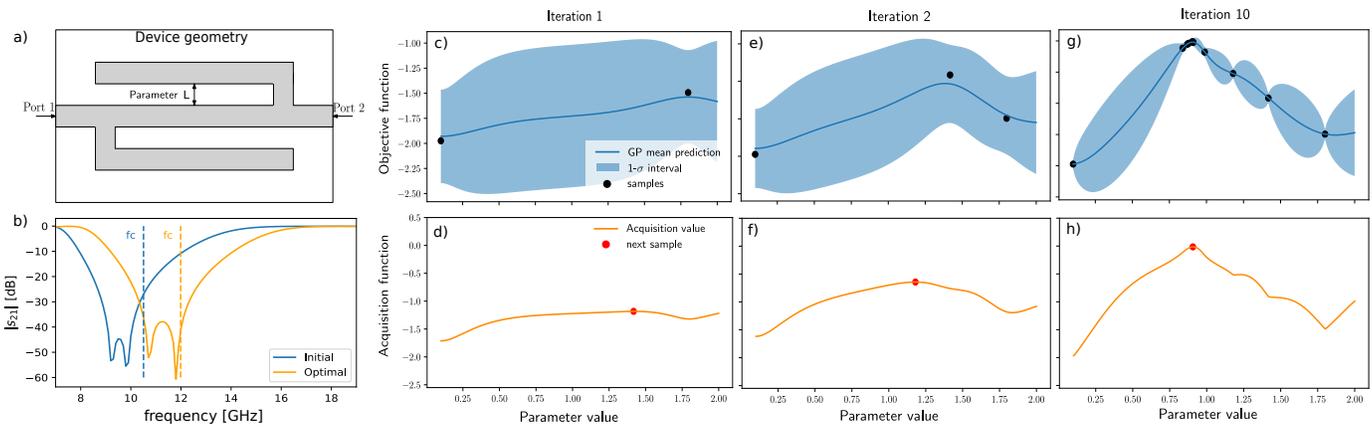


Fig. 1: Example of Bayesian optimization applied to a folded stub filter [2]. The objective is to find the values of the single variable L for which the $|S_{21}|$ response presents a -3 dB stop-band centered at exactly 12 GHz. The central frequency is indicated as f_c . At each BO iteration, a Gaussian process is built to represent the objective function with respect to the design parameter L (Fig.4c, 4e, 4g). Second, the chosen acquisition function (here the upper confidence bound [3]) selects the next L value to evaluate (Fig.4d, 4f, 4h). Next, a new $|S_{21}|$ response is simulated for the next L value and added to the initial data samples. The optimization continues by updating the Gaussian process and selecting new samples until $f_c = 12$ GHz (Fig.4b).

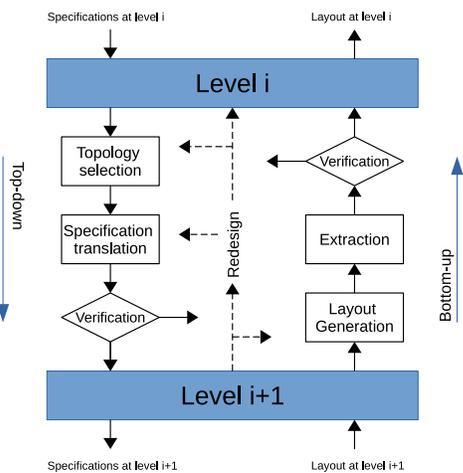


Fig. 2: Typical steps in the hierarchical analog design flow, between two levels of abstraction, as described in [4].

powerful Machine learning and Bayesian learning techniques for EDA.

II. MACHINE LEARNING FOR ELECTRONIC DESIGN AUTOMATION

In the last decade, many impactful results in AI, such as text-to-image generation [32], generative language modeling [33], and self-taught gaming [34], have been achieved by models with millions - or billions - of internal trainable parameters. Such models can only be built using expensive computing clusters and mass data collection. For the time being, similar resources are hard to allocate for electronic design automation (EDA). As a consequence, the research in EDA has been mostly restricted to specific and often resource- and data-demanding ML models.

There seem to be some scenarios in which the resource limitation is more severe. The first is the design at low abstraction levels, which necessitates accurate and time-consuming

numerical simulations or measurements. The second is the automation of selection tasks, which requires the harvesting and labeling of pre-existing design examples from the available literature. Finally, data scarcity is to be expected in physical experimentation, in which human intervention constitutes the main speed bottleneck.

Data scarcity in all these scenarios is exacerbated by the reticence of integrated circuit and semiconductor companies and research institutions to disclose IC intellectual property and adhere to more open data policies.¹ Furthermore, in analog design, expensive electromagnetic (EM) simulations are crucial for the characterization of second-order effects that cannot be assessed with faster circuit simulations. Consequently, a dual connection appears between the available AI techniques and the need for simulation data. On the one hand, the cost of running simulations restricts the applicability of AI to the more data-efficient techniques, which limits the generalization possibilities. On the other hand, there is a relentless interest in developing data-efficient ML techniques specifically designed to lower the overall cost of the simulations required. In particular, the most promising automation strategies have come from the field of machine learning. As a branch of artificial intelligence, ML encompasses algorithms and models that are able to solve tasks without explicit programming. Rather, ML models can be trained to recognize patterns and make predictions from a dataset of examples. The training consists of refining the output of ML models until they are able to perform the desired task on new, unseen data. Depending on the interaction between the model and the data collection process, three well-known ML paradigms can be identified: supervised learning, unsupervised learning, and reinforcement learning. The different ML paradigms can be separately applied or combined into more elaborated ML pipelines to automate single operational blocks or full paths in the analog design workflow (Fig.2). This Section briefly

¹Instead, open source code and data policies appear to be sustainable for AI-focused software companies and software research institutions.

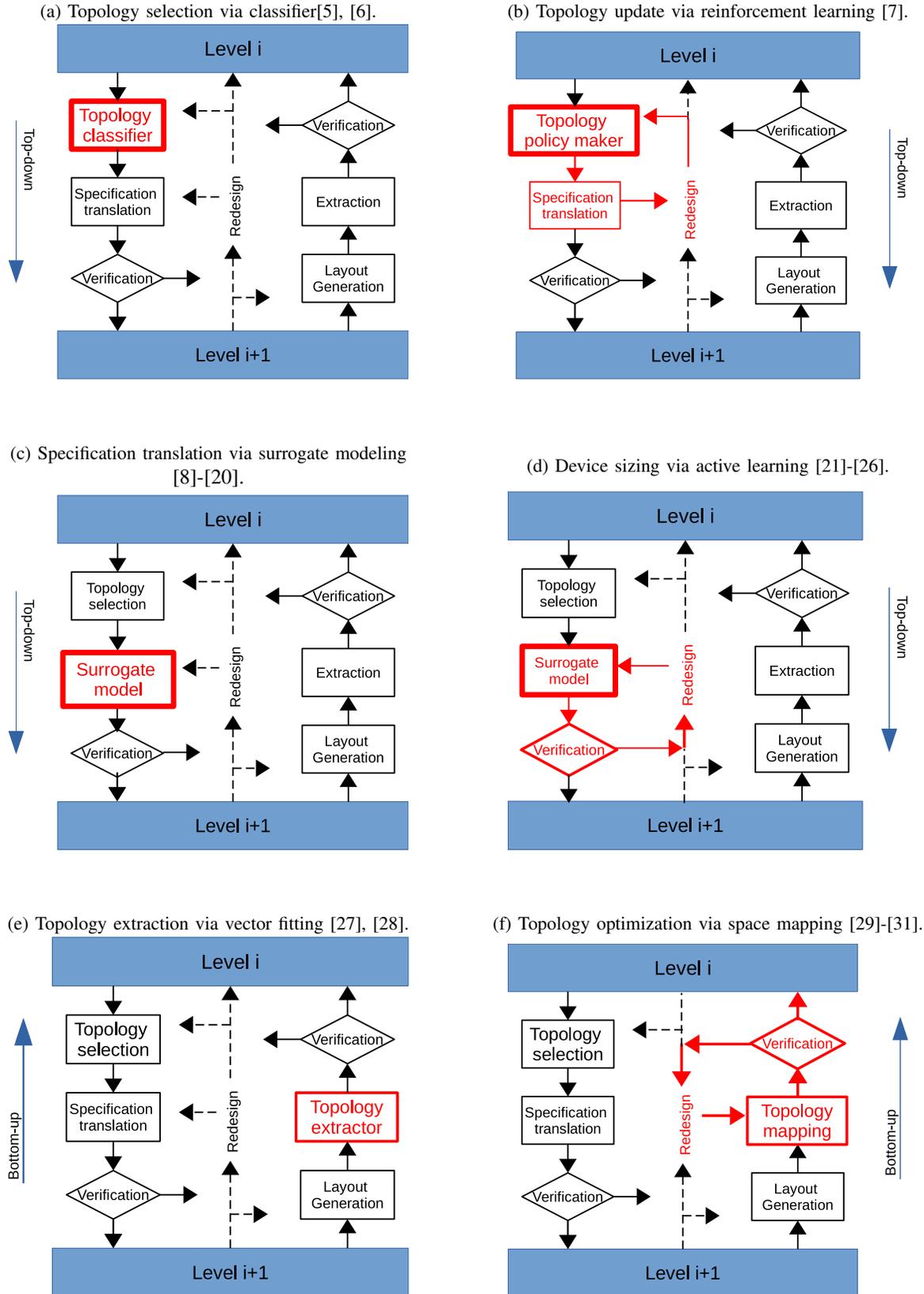


Fig. 3: Examples of tasks automated with machine learning in the analog design hierarchy. The automated steps and operations are depicted in red.

overviews some common design tasks that can be automated using recently developed ML techniques. Here, the common ML approaches are associated with the steps defined in [4], to better identify their role in the complete design workflow. A summary of the described tasks and available techniques is shown in Fig. 3. **Depending on the application, ML techniques can be used on different device parametrizations. For example, the topology can be obtained as a combination of circuit blocks [5], as a template parametric geometry [35], or as free-shape mesh encoding [36].**

A. Topology design

The topology selection is the first step that can be automated (Fig.3a). It can be framed as a supervised classification problem and solved with ML models such as graph neural networks [5] or recurrent neural networks [6]. Instead, other ML techniques based on reinforcement learning (RL) have been proposed to modify selected topologies iteratively. In the RL approach, the ML model learns a mapping, called policy, between the design topology and a set of actions. The learned policy determines how to update the current topology variables, until the design specifications are met (Fig.3b). For example, the method in [7] can construct complex operational amplifier topologies by assigning specific building blocks to different actions.

B. Specification translation via surrogate modeling

The translation of specifications to lower abstraction levels can also be automated using ML. In recent years, various techniques have been developed to predict figures of merit for a pre-selected topology with respect to lower-level variables [37]. In other words, ML models can be trained as *surrogate models*, and replace simulators to estimate the figures of merit (FoM) for a given parametric design (Fig.3c). For example, let us consider a microstrip antenna topology.

Surrogate modeling is usually tackled as a supervised learning regression task, in which the data labels come from costly numerical simulations or measurements. The key advantage of a surrogate model is that it can be agnostic to the device physics and its validity can extend to a wide range of design variables, from circuit schematic variables to geometrical layout or substrate electrical features [21]. Particularly interesting is modeling the figures of merit based on the frequency response of an analog device, that needs to be simulated at the EM level to characterize signal propagation, cross-talk, or EM emissions. A popular technique for this task is Vector Fitting (VF) [8]. In its base form, VF represents a device transfer function in terms of a complex rational function defined by parameters (poles and residues) over the frequency. Later on, several capabilities have been added to VF techniques, such as the parametric modeling over multiple design variables [11] and the inclusion of signal propagation delays [10], [12].

In order to model figures of merit over many design variables, a wide variety of black-box ML models have been employed. Indeed, black-box models can represent more general relationships among design variables that are unfeasible to obtain in an analytical form, based on theoretical

derivation.² Popular black-box models include support vector machines (SVMs), Gaussian Processes (GPs), and artificial neural networks (ANNs). Least-squares SVM (LS-SVM) have been employed to build data-efficient surrogate models over tens of design variables, in training times of seconds [13]. Instead, various ANN architectures have been developed as high-dimensional surrogates, such as [35]. At the expense of bigger datasets, ANNs are more versatile and they can be trained on hundreds or thousands of design variables. An intriguing advancement in ANN surrogates is integrating physical knowledge to enhance accuracy. For example, in [17], physical properties are enforced on ANNs, such as frequency response's causality and passivity. Moreover, ANNs can be combined with VF to produce physics-informed models over extended parameter spaces with improved data efficiency [35], [18]. Instead, Gaussian process regression, also known as Kriging, can be used to build stochastic surrogate models [19], [20]: by representing figures of merit as stochastic processes, Gaussian processes (GPs) offer a confidence estimation on their predictions. This property makes GPs better suited to optimization tasks and uncertainty quantification, while they retain similar data efficiency to SVMs. However, the computational complexity of the GP limits its usage to a lower number of design variables than ANNs. Table I summarizes the main ML techniques that have been employed as surrogate models for EDA.

C. Device sizing via active learning

The use of surrogate models allows us to solve device sizing problems. Device sizing, sometimes simply referred to as *device optimization*, consists of identifying the values of design variables that allow the device to satisfy all the specifications on some given figures of merit. A possible approach to speed up the device sizing using ML is the following. First, the figures of merit can be translated into one of multiple user-defined *objective functions*, that assign a score to any realization of design variables. Next, the search for the right parameter values \mathbf{p}^* is framed as an optimization problem:

$$\mathbf{p}^* = \arg \max_{\mathbf{p}} f(\mathbf{p}) \quad (1)$$

where \mathbf{p} is a vector of design variables, and f is the objective function for some chosen figures of merit. Note that if multiple objectives are defined, the optimal solution is not unique but constitutes a Pareto set of possible design solutions.

A viable solution to the optimization problem is given by iterative techniques like evolutionary (or genetic) algorithms [38], particle swarm optimization [39], and other meta-heuristic search algorithms. However, they typically require the collection of large datasets, even thousands of samples for less than 10 design variables, while they do not allow easy integration of prior knowledge. For faster optimization, it is beneficial to use a predictive ML model M for the objective(s), such that $f_M \sim f(\mathbf{p})$. The predictive model can be tested to

²Even if possible to derive complicated analytical forms that are sufficiently general, it may still be hard to develop a proper training method to fit the data in a high-dimensional design parameter space.

ML models	Data efficiency	Max. Input dimensions	Trainable internal parameters	Stochastic	Physical properties
ANN [35]	Low	>100	yes	no	no
GP [3]	High	<20	no	yes	no
SVM [13]	High	<100	no	no	no
VF [8]	High	<5	yes	no	yes
ANN + VF [17]	Low	>100	yes	no	yes

TABLE I: Main ML models used as surrogate models for EDA. Here the properties of the most common formulations are reported. Different properties result from the numerous versions and combinations of these models.

gain insight into the relationship among the design variables and how they concur with the final result.³ Furthermore, the predicted model is reusable and can be used to automate subsequent sizing steps.

Assuming a sufficiently accurate ML model, the device sizing can be immediately performed by querying the model exhaustively, until suitable parameter values are found. Since the surrogate model is typically fast to evaluate by construction, the querying can also be implemented as a random sampling strategy, such as Monte Carlo. **Alternatively, the surrogate can also be trained as an inverse model to return the parameter values that produce the objective or figure of merit provided as input [40]. The exhaustive search** only if the amount of pre-collected data samples is sufficient to train a surrogate model that is globally accurate, for any objective function or figure of merit. Unfortunately, such an amount is hard to estimate in advance. Therefore, it is preferable to collect data by iteratively executing only the most informative simulations that improve the accuracy of a model (Fig.3d). This iterative data collection process is known as adaptive sampling or sequential design of experiments (seqDoE), or *active learning* [21], [41] when it involves a model querying stage. More specifically, active learning is referred to as Bayesian learning (BL) or Bayesian optimization (BO) when it is applied to stochastic models built via Bayesian inference, such as Gaussian processes. Unlike other ML models, Gaussian processes are particularly suitable for active learning since they provide a more consistent and statistically interpretable variance estimation on their prediction. The use of Bayesian learning in analog design is thoroughly discussed in III.

An alternative ML-based approach for the device sizing is to use ANNs with reinforcement learning [42], [43]. Different from active learning, the goal of RL is to train a model (usually a neural network) to predict the optimal sequence of actions that maximize a reward based on the design specifications. Therefore, RL does not necessarily produce a data-efficient surrogate model. Instead, it aims at a general solution for the device sizing for a wide range of specification values, without specifically minimizing the amount of data required.

D. Topology extraction

Further automation of the EDA design workflow can be achieved for bottom-up design tasks. For example, vector fitting has also become a popular method to extract a circuit-level representation from simulated frequency- or transient-responses [27], [28]. The circuit-level representation provided

³These problems are part of the *interpretability* and *explainability* study, in both AI and machine learning.

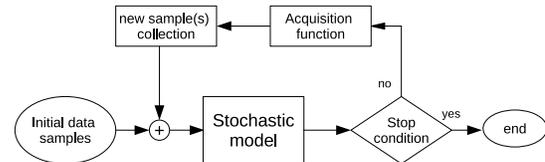


Fig. 4: Simplified Bayesian learning scheme.

by VF can replace the ideal component used in the initial topology (Fig.3e).

Subsequently, once the initial topology is selected, it can be adjusted with space mapping (SM) techniques [29]. Space mapping automatically adjusts the variables of a low-fidelity representation using limited queries on a high-fidelity (expensive) representation until all the specifications are satisfied. In analog design, space mapping techniques allow one to optimize a circuit level using limited, expensive EM simulations (Fig.3f). In more recent developments, the mapping between the low- and high-level representation is provided by neural networks [30], [31], which guarantee high accuracy on complicated designs and mappings.

III. BAYESIAN LEARNING FOR DEVICE SIZING

As mentioned in Section II-C, Bayesian learning (BL) has become a common technique for the automated sizing of analog devices. The key idea behind BL is to build a stochastic model of the uncertainty about the target function - either a figure of merit or an objective function defined on top of that- and make decisions about which additional data points to collect based on this model uncertainty. As discussed in Section II-C, the purpose of automating the device sizing is ultimately to reduce the need for expensive simulators, which constitute the accurate data collector from an ML perspective. In its base form, BL consists of several steps that follow the simplified scheme shown in Fig 4:

- 1) **Initial data samples:** The first step is collecting a small initial set of data samples, e.g. 10 samples per design variable according to a Latin HyperCube (LHC) sampling [44]. In analog design applications, this is typically constituted by design parameter values (inputs) and their corresponding objective function evaluation (output(s)). This provides a starting point for building a surrogate model that can match the underlying data distribution.
- 2) **Stochastic model:** The stochastic ML model (often based on Gaussian processes) creates a computationally efficient representation of the data distribution. It is able to provide an expected value for unobserved samples and a variance, that indicates the estimated uncertainty.

- 3) **Stop condition:** If verified, it halts the learning loop. It is usually defined based as a threshold on an accuracy metric, or on a fixed computational budget.
- 4) **Acquisition function:** It guides the selection of the next data sample to be collected by balancing between the exploration of designs with high uncertainty on the objective and the exploitation of design space regions with high predicted objective value, in order to efficiently improve the ML model. The next data samples are usually selected as the global maximum - or the local maxima - of the acquisition function over the input data space. Since the acquisition function is fast to evaluate on the GP model, its maxima are usually identified using a Monte Carlo sampling or a gradient-based optimizer [45].
- 5) **New sample(s) collection:** The next data sample(s) selected by the acquisition function is evaluated, by means of costly simulations or measurements and added to the initial dataset. The collection provides the objective function values for the selected sample.
- 6) **Model update:** The ML model is re-trained iteratively on the extended dataset, including the newly collected samples.

If BL is successful, the model's accuracy in identifying the maximum objective value progressively improves. As an alternative, it is possible to improve the global accuracy of the model, rather than maximizing the objective, by choosing an acquisition function that only selects the data samples of higher model uncertainty, also known as *maximum variance*. A simple example of Bayesian Optimization on a microstrip filter [2] is illustrated in Fig. 1. The objective is to minimize the distance between the filter's central frequency (f_c) and 12 GHz. It is possible to observe how the acquisition function gradually adds samples that converge to the highest objective while exploring the range of L values. Moreover, the uncertainty of the model represented by the shaded areas ($1 - \sigma$ standard deviation interval) decreases at each iteration in the whole domain.

A. Gaussian process models

In the past decade, the Gaussian process (GP) has been successfully used as a surrogate model in Bayesian learning, for analog design automation. The GP defines a prior Gaussian distribution and updates it with observed data samples to obtain a posterior probability distribution. Then, an expected value and a variance can be extracted from the posterior, for each observed or unobserved sample. Necessary to compute the posterior is a user-defined kernel function, referred to as kernel, that represents the correlation among any pair of data samples. A detailed description of the functioning of Gaussian processes can be found in [3]. Gaussian processes have become a popular choice for BL, thanks to their several advantages:

- **Variable complexity and prior knowledge integration:** The kernel can encode assumptions on the stochastic process underlying the data. Therefore, the user can select or combine kernels to build a model of different

complexity. Moreover, the kernel can be chosen according to prior knowledge about the function to be modeled, such as linearity, stationarity, periodicity, and discontinuities.

- **Few hyperparameters:** GPs present only a few significant hyperparameters, like the kernel hyperparameters, that can easily be optimized using maximum marginal likelihood estimation [3]. This method does not require a separate validation set for model selection or hyperparameter tuning.
- **Uncertainty estimation:** GPs provide a principled way to estimate prediction uncertainty, that is represented by the posterior variance.
- **Data efficiency:** GPs can provide meaningful predictions even with a smaller amount of data points by leveraging the prior and the kernel assumptions.

IV. VARIANTS OF BAYESIAN LEARNING

In analog design applications, there are still challenges that limit the usage of GPs and Bayesian learning strategies. Thus, several BL variants have been proposed to tackle the specific issues.

A. Mitigation of the curse of dimensionality

One of the most common issues is the curse of dimensionality, which affects GPs and stochastic models more severely than other ML models [46]: in high dimensions, where the data become sparse, the kernel assigns low similarity among data samples. Thus, a more substantial amount of data is needed to recognize patterns and achieve sufficient modeling accuracy. In surrogate modeling applications, the curse of dimensionality becomes significant when more than 10 real-valued design variables are considered, which may require 10^4 or more data samples to train an accurate Gaussian Process, depending on the complexity of the modeled function. Several approaches have been presented to mitigate this issue in analog design applications. For example in [47], sensitivity data and principal component analysis are used to restrict the GP modeling to a progressively smaller area of interest in the design space. Alternatively, the technique suggested in [26] enables BL for up to 25 design variables, by decomposing a high-dimensional GP in a sum of lower-dimensional GPs according to a partition tree structure. A side effect of the curse of dimensionality is that it may deteriorate the maximization of the acquisition function, leading to a sub-optimal choice of data samples in the BL loop. For this reason [24] presents a modified acquisition function that is only maximized in a one-dimensional subspace of design variables, leading to improved optimization performance up to around 30 design variables.

B. Objective function definition in EDA

When performing tasks like device sizing using BL, it is crucial to formulate the design specifications as suitable objective functions. The performance of the optimization is heavily affected by the objective definition. In fact, the objective should incorporate as many specifications as possible, to identify the correct optimal design. However, this may produce

complicated objectives that are more difficult to model. As an alternative, a multi-objective formulation can be employed to obtain a Pareto set of possible designs with contrasting objective values. For this purpose, different acquisition functions for BL have been proposed to rapidly identify the Pareto set [48], [49]. The objective definition is particularly challenging when the device under test has to meet the specifications over a whole range of operating conditions. For example, this is the case in the optimization of frequency responses. In order to improve the objective definition for frequency response optimization, a particular formulation of BO has been developed in [50].

C. Multi-fidelity optimization

Another challenge in BL is the integration of data samples that are collected from information sources with different fidelity, such as EM simulations with coarse and fine meshes. This *multi-fidelity* setting requires a strategy to select the information source for new sample evaluation, by balancing the information gain and the computational cost. In EDA, the multi-fidelity problem has been tackled using heuristic approaches such as [51], [52]. More recently, for synthetic problems, the multi-fidelity optimization has been framed as one additional objective function [53] or using a modified acquisition function for continuous fidelity levels [54].

D. Robustness

In addition, robust Bayesian optimization techniques have been introduced to account for the uncertainty about the design variables when their values are affected by stochastic variabilities [23], [55]. In this case, the BL prioritizes design solutions that are less sensitive to the stochastic variability of the design variables, rather than searching for the absolute optimal solution.

E. Feasibility region identification

In the context of device sizing, it is sometimes requested that the chosen figures of merit fall within a certain range of feasibility. In this case, it is not necessary to find a solution that maximizes an objective function. Rather, BL can be used to identify a subspace of design variables that correspond to acceptable designs according to the feasibility ranges. This task, known as feasibility region identification, can be performed with an appropriate choice of acquisition function for the BL algorithm [2].

The aforementioned techniques are only a few of the available BL variants. Many more have been introduced in different fields [56] and still need to be tested and eventually adapted to electronic design applications.

V. CONCLUSION

Machine learning is a highly advantageous tool within the realm of electrical design automation. In the last decades, the continuous development of machine learning techniques has contributed significantly to the automation of numerous tasks

within the analog design workflow. In particular, Bayesian learning (BL) has emerged as a powerful set of data-efficient techniques applicable to surrogate modeling and device sizing. The main scope of BL is the minimization of the number of simulations or experiments required in analog design.

Aside from the specific issues, several challenges remain common to all Bayesian learning techniques. Data efficiency is still a primary one due to the curse of dimensionality (see Section IV). To date, applying Bayesian learning for high-dimensional optimization tasks, like device sizing over hundreds of design variables, is nearly unfeasible.

A possible research direction to alleviate the curse of dimensionality in BL may be a deeper integration of pre-existing physical knowledge in the ML models. For example, some available modeling techniques combine a simple physics-based model with a general-purpose machine-learning one (Section II-B). Different solutions may come from other stochastic models that are yet to be experimented for device sizing or surrogate modeling, such as Bayesian neural networks (BNNs) [57], variational auto-encoders (VAEs) [58], or neural processes (NP) [59]. In the near future, it is easy to imagine the usage of these models in a Bayesian learning framework even if the actual implementation would not be trivial. In the short term, Bayesian learning will likely remain a highly accessible and cost-effective technique to solve electronic design automation problems.

REFERENCES

- [1] G. Huang, J. Hu, Y. He, J. Liu, M. Ma, Z. Shen, J. Wu, Y. Xu, H. Zhang, K. Zhong, X. Ning, Y. Ma, H. Yang, B. Yu, H. Yang, and Y. Wang, "Machine learning for electronic design automation: A survey," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 26, no. 5, jun 2021. [Online]. Available: <https://doi.org/10.1145/3451179>
- [2] F. Garbuglia, J. Qing, N. Knudde, D. Spina, I. Couckuyt, D. Deschrijver, and T. Dhaene, "Bayesian active learning for multi-objective feasible region identification in microwave devices," *Electronics Letters*, vol. 57, no. 10, pp. 400–403, 2021. [Online]. Available: <http://dx.doi.org/10.1049/ell2.12022>
- [3] C. E. Rasmussen and C. K. Williams, *Gaussian processes for machine learning*, 1st ed. Cambridge: MIT Press, 2008.
- [4] G. Gielen and R. Rutenbar, "Computer-aided design of analog and mixed-signal integrated circuits," *Proceedings of the IEEE*, vol. 88, no. 12, pp. 1825–1854, 2000.
- [5] A. Deeb, A. Ibrahim, M. Salem, J. Pichler, S. Tkachov, A. Karaj, F. Al Machot, and K. Kyandoghere, "A robust automated analog circuits classification involving a graph neural network and a novel data augmentation strategy," *Sensors*, vol. 23, no. 6, 2023. [Online]. Available: <https://www.mdpi.com/1424-8220/23/6/2989>
- [6] M. Rotman and L. Wolf, "Electric analog circuit design with hyper-networks and a differential simulator," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 4157–4161.
- [7] Z. Zhao and L. Zhang, "Analog integrated circuit topology synthesis with deep reinforcement learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 12, pp. 5138–5151, 2022.
- [8] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Transactions on Power Delivery*, vol. 14, no. 3, pp. 1052–1061, 1999.
- [9] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. D. Zutter, "Macro-modeling of multiport systems using a fast implementation of the vector fitting method," *IEEE Microwave and Wireless Components Letters*, vol. 18, no. 6, pp. 383–385, 2008.
- [10] A. Chinea, P. Triverio, and S. Grivet-Talocia, "Delay-based macromodeling of long interconnects from frequency-domain terminal responses," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 1, pp. 246–256, 2010.

- [11] D. Deschrijver and T. Dhaene, "Stability and passivity enforcement of parametric macromodels in time and frequency domain," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 11, pp. 2435–2441, 2008.
- [12] M. Sgueglia, A. Sorrentino, M. de Magistris, D. Spina, D. Deschrijver, and T. Dhaene, "A novel parametric macromodeling technique for electromagnetic structures with propagation delays," in *2017 IEEE 21st Workshop on Signal and Power Integrity (SPI)*, Lake Maggiore, Italy, 2017, pp. 1–4.
- [13] R. Trincherio, M. Larbi, H. M. Torun, F. G. Canavero, and M. Swaminathan, "Machine learning and uncertainty quantification for surrogate models of integrated devices with a large number of parameters," *IEEE Access*, vol. 7, pp. 4056–4066, 2019.
- [14] R. Trincherio, P. Manfredi, I. S. Stievano, and F. G. Canavero, "Machine learning for the performance assessment of high-speed links," *IEEE Transactions on Electromagnetic Compatibility*, vol. 60, no. 6, pp. 1627–1634, 2018.
- [15] Q.-J. Zhang, K. C. Gupta, and V. K. Devabhaktuni, "Artificial neural networks for rf and microwave design - from theory to practice," *IEEE Transactions on Microwave Theory and Techniques*, vol. 51, no. 4, pp. 1339–1350, 2003.
- [16] H. Kabir, Y. Wang, M. Yu, and Q.-J. Zhang, "Neural network inverse modeling and applications to microwave filter design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 4, pp. 867–879, 2008.
- [17] H. M. Torun, A. C. Durgun, K. Aygün, and M. Swaminathan, "Causal and passive parameterization of s-parameters using neural networks," *IEEE Transactions on Microwave Theory and Techniques*, vol. 68, no. 10, pp. 4290–4304, 2020.
- [18] Y. Cao, G. Wang, and Q.-J. Zhang, "A new training approach for parametric modeling of microwave passive components using combined neural networks and transfer functions," *IEEE Transactions on Microwave Theory and Techniques*, vol. 57, no. 11, pp. 2727–2742, 2009.
- [19] S. Koziel, A. Pietrenko-Dabrowska, and U. Ullah, "Low-cost modeling of microwave components by means of two-stage inverse/forward surrogates and domain confinement," *IEEE Transactions on Microwave Theory and Techniques*, vol. 69, no. 12, pp. 5189–5202, Dec. 2021.
- [20] S. Koziel and A. Pietrenko-Dabrowska, "Design-oriented computationally-efficient feature-based surrogate modelling of multi-band antennas with nested kriging," *International Journal of Electronics and Communications*, vol. 120, no. 12, p. 153202, Jun. 2020.
- [21] D. Gorissen, D. Deschrijver, T. Dhaene, and D. D. Zutter, "A software framework for automated behavioral modeling of electronic devices [application notes]," *IEEE Microwave Magazine*, vol. 13, no. 6, pp. 102–118, 2012.
- [22] W. Lyu, P. Xue, F. Yang, C. Yan, Z. Hong, X. Zeng, and D. Zhou, "An efficient bayesian optimization approach for automated optimization of analog circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 6, pp. 1954–1967, 2018.
- [23] D. De Witte, J. Qing, I. Couckuyt, T. Dhaene, D. Vande Ginste, and D. Spina, "A robust bayesian optimization framework for microwave circuit design under uncertainty," *ELECTRONICS*, vol. 11, no. 14, p. 14, 2022. [Online]. Available: <http://dx.doi.org/10.3390/electronics11142267>
- [24] S. Zhang, F. Yang, C. Yan, D. Zhou, and X. Zeng, "Lineasybo: Scalable bayesian optimization approach for analog circuit synthesis via one-dimensional subspaces," in *Proceedings of the 2022 ACM/IEEE Workshop on Machine Learning for CAD*, ser. MLCAD '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 27–34. [Online]. Available: <https://doi.org/10.1145/3551901.3556496>
- [25] X. Yang, H. M. Torun, J. Tang, P. R. Paladhi, Y. Zhang, W. D. Becker, J. A. Hejase, and M. Swaminathan, "Parallel bayesian active learning using dropout for optimizing high-speed channel equalization," in *2021 IEEE 30th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS)*, 2021, pp. 1–3.
- [26] H. M. Torun and M. Swaminathan, "High-dimensional global optimization method for high-frequency electronic design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 67, no. 6, pp. 2128–2142, 2019.
- [27] A. Ramirez, "Vector fitting-based calculation of frequency-dependent network equivalents by frequency partitioning and model-order reduction," *IEEE Transactions on Power Delivery*, vol. 24, no. 1, pp. 410–415, 2009.
- [28] S. Grivet-Talocia, F. Canavero, I. Stievano, and I. Maio, "Circuit extraction via time-domain vector fitting," in *2004 International Symposium on Electromagnetic Compatibility (IEEE Cat. No.04CH37559)*, vol. 3, 2004, pp. 1005–1010 vol.3.
- [29] S. Koziel, Q. S. Cheng, and J. W. Bandler, "Space mapping," *IEEE Microwave Magazine*, vol. 9, no. 6, pp. 105–122, 2008.
- [30] W. Liu, L. Zhu, W. Na, and Q.-J. Zhang, "An overview of neuro-space mapping techniques for microwave device modeling," in *2016 IEEE MTT-S Latin America Microwave Conference (LAMC)*, 2016, pp. 1–3.
- [31] D. Gorissen, L. Zhang, Q.-J. Zhang, and T. Dhaene, "Evolutionary neuro-space mapping technique for modeling of nonlinear microwave devices," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 2, pp. 213–229, 2011.
- [32] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," 2022.
- [33] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.
- [34] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016. [Online]. Available: <https://doi.org/10.1038/nature16961>
- [35] F. Feng, W. Na, J. Jin, J. Zhang, W. Zhang, and Q.-J. Zhang, "Artificial neural networks for microwave computer-aided design: The state of the art," *IEEE Transactions on Microwave Theory and Techniques*, vol. 70, no. 11, pp. 4597–4619, 2022.
- [36] A. Gupta, E. A. Karahan, C. Bhat, K. Sengupta, and U. K. Khankhoje, "Tandem neural network based design of multiband antennas," *IEEE Transactions on Antennas and Propagation*, vol. 71, no. 8, pp. 6308–6317, 2023.
- [37] M. Fayazi, Z. Colter, E. Afshari, and R. Dreslinski, "Applications of artificial intelligence on the modeling and optimization for analog and mixed-signal circuits: A review," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 6, pp. 2418–2431, 2021.
- [38] Q. S. Cheng, J. W. Bandler, S. Koziel, M. H. Bakr, and S. Ogurtsov, "The state of the art of microwave cad: Em-based optimization and modeling," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 20, no. 5, pp. 475–491, 2010.
- [39] S. Liang, Z. Fang, G. Sun, Y. Liu, G. Qu, and Y. Zhang, "Sidelobe reductions of antenna arrays via an improved chicken swarm optimization approach," *IEEE Access*, vol. 8, pp. 37 664–37 683, 2020.
- [40] L.-Y. Xiao, W. Shao, F.-L. Jin, B.-Z. Wang, and Q. H. Liu, "Inverse artificial neural network for multiobjective antenna design," *IEEE Transactions on Antennas and Propagation*, vol. 69, no. 10, pp. 6651–6659, 2021.
- [41] J. Qing, N. Knudde, I. Couckuyt, D. Spina, and T. Dhaene, "Bayesian active learning for electromagnetic structure design," in *2020 14TH European Conference on Antennas and Propagation (EUCAP 2020)*, Copenhagen, Denmark, 2020.
- [42] Y. Uhlmann, M. Essich, L. Bramlage, J. Scheible, and C. Curio, "Deep reinforcement learning for analog circuit sizing with an electrical design space and sparse rewards," in *2022 ACM/IEEE 4th Workshop on Machine Learning for CAD (MLCAD)*, 2022, pp. 21–26.
- [43] Z. Wei, Z. Zhou, P. Wang, J. Ren, Y. Yin, G. F. Pedersen, and M. Shen, "Automated antenna design via domain knowledge-informed reinforcement learning and imitation learning," *IEEE Transactions on Antennas and Propagation*, vol. 71, no. 7, pp. 5549–5557, 2023.
- [44] F. A. Viana, G. Venter, and V. Balabanov, "An algorithm for fast optimal latin hypercube design of experiments," *International Journal of Numerical Methods in Engineering*, vol. 82, no. 2, pp. 135–156, Oct. 2010.
- [45] J. Wilson, F. Hutter, and M. Deisenroth, "Maximizing acquisition functions for bayesian optimization," *Advances in neural information processing systems*, vol. 31, 2018.
- [46] Y. Bengio, O. Delalleau, and N. Roux, "The curse of highly variable functions for local kernel machines," in *Advances in Neural Information Processing Systems*, Y. Weiss, B. Schölkopf, and J. Platt, Eds., vol. 18. MIT Press, 2005.
- [47] S. Koziel and A. Pietrenko-Dabrowska, "Recent advances in high frequency modeling by means of domain confinement and nested kriging," *IEEE Access*, vol. 8, pp. 189 326–189 342, 2020.
- [48] N. R. B. Satrio, I. Couckuyt, F. Garbuglia, D. Spina, I. Van Nieuwenhuysse, and T. Dhaene, "Bi-objective bayesian optimization of engineering problems with cheap and expensive cost functions," *Engineering with Computers*, p. 11, 2023. [Online]. Available: <http://dx.doi.org/10.1007/s00366-021-01573-7>
- [49] S. Koziel and S. Ogurtsov, "Multi-objective design of antennas using variable-fidelity simulations and surrogate models," *IEEE Transactions*

- on *Antennas and Propagation*, vol. 61, no. 12, pp. 5931–5939, Sep. 2013.
- [50] F. Garbuglia, D. Spina, D. Deschrijver, I. Couckuyt, and T. Dhaene, “Bayesian optimization for microwave devices using deep gp spectral surrogate models,” *IEEE Transactions on Microwave Theory and Techniques*, p. 8, 2023. [Online]. Available: <http://dx.doi.org/10.1109/TMTT.2022.3228951>
- [51] J. P. Jacobs and S. Koziel, “Two-stage gaussian process modeling of microwave structures for design optimization,” in *Simulation-Driven Modeling and Optimization: ASDOM, Reykjavik, August 2014*. Springer, 2016, pp. 161–184.
- [52] S. Koziel, S. Ogurtsov, I. Couckuyt, and T. Dhaene, “Multi-objective design of antenna structures using variable-fidelity em simulations and co-kriging,” in *Proceedings of the European Conference on Antennas and Propagation*. IEEE, 2014, pp. 2884–2886.
- [53] F. Irshad, S. Karsch, and A. Döpp, “Expected hypervolume improvement for simultaneous multi-objective and multi-fidelity optimization,” *arXiv preprint arXiv:2112.13901*, 2021.
- [54] J. Wu, S. Toscano-Palmerin, P. I. Frazier, and A. G. Wilson, “Practical multi-fidelity bayesian optimization for hyperparameter tuning,” in *Uncertainty in Artificial Intelligence*. PMLR, 2020, pp. 788–798.
- [55] J. Qing, I. Couckuyt, and T. Dhaene, “A robust multi-objective bayesian optimization framework considering input uncertainty,” *Journal of Global Optimization*, p. 19, 2023. [Online]. Available: <http://dx.doi.org/10.1007/s10898-022-01262-9>
- [56] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, “Taking the human out of the loop: A review of bayesian optimization,” *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2016.
- [57] J. T. Springenberg, A. Klein, S. Falkner, and F. Hutter, “Bayesian optimization with robust bayesian neural networks,” in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016.
- [58] A. Grosnit, R. Tutunov, A. M. Maraval, R.-R. Griffiths, A. I. Cowen-Rivers, L. Yang, L. Zhu, W. Lyu, Z. Chen, J. Wang *et al.*, “High-dimensional bayesian optimisation with variational autoencoders and deep metric learning,” *arXiv preprint arXiv:2106.03609*, 2021.
- [59] M. Garnelo, J. Schwarz, D. Rosenbaum, F. Viola, D. J. Rezende, S. M. A. Eslami, and Y. W. Teh, “Neural processes,” 2018.