

RESEARCH ARTICLE | JULY 22 2025

Bayesian optimization of poloidal field coil positions in tokamaks

Special Collection: [Papers from the 5th International Conference on Data-Driven Plasma Science](#)

Timothy Nunn ; Kamran Pentland ; Vignesh Gopakumar ; James Buchanan 



Phys. Plasmas 32, 072507 (2025)

<https://doi.org/10.1063/5.0272085>



Articles You May Be Interested In

FreeGSNKE: A Python-based dynamic free-boundary toroidal plasma equilibrium solver

Phys. Plasmas (April 2024)

CRATOS-GS: A free-boundary, hierarchical adaptive mesh refinement Grad–Shafranov solver

AIP Advances (September 2025)

Emulation techniques for scenario and classical control design of tokamak plasmas

Phys. Plasmas (April 2024)



Physics of Plasmas

Special Topics Open
for Submissions

[Learn More](#)

Bayesian optimization of poloidal field coil positions in tokamaks

Cite as: Phys. Plasmas **32**, 072507 (2025); doi: [10.1063/5.0272085](https://doi.org/10.1063/5.0272085)

Submitted: 21 March 2025 · Accepted: 3 July 2025 ·

Published Online: 22 July 2025



View Online



Export Citation



CrossMark

Timothy Nunn,^{a)} Kamran Pentland, Vignesh Copakumar, and James Buchanan

AFFILIATIONS

UKAEA (United Kingdom Atomic Energy Authority), Culham Campus, Abingdon, Oxfordshire OX14 3DB, United Kingdom

Note: This paper is part of the Special Topic on Papers from the 5th International Conference on Data-Driven Plasma Science.

^{a)} Author to whom correspondence should be addressed: timothy.nunn@ukaea.uk

ABSTRACT

The tokamak is a world-leading concept for producing sustainable energy via magnetically confined nuclear fusion. Identifying where to position the magnets within a tokamak, specifically the poloidal field (PF) coils, is a design problem which requires balancing a number of competing economical, physical, and engineering objectives and constraints. In this paper, we show that multi-objective Bayesian optimization (BO), an iterative optimization technique utilizing probabilistic machine learning models, can effectively explore this complex design space and return several optimal PF coilsets. These solutions span the Pareto front, a subset of the objective space that optimally satisfies the specified objective functions. We outline an easy-to-use BO framework and demonstrate that it outperforms alternative optimization techniques while using significantly fewer computational resources. Our results show that BO is a promising technique for fusion design problems that rely on computationally demanding high-fidelity simulations.

© 2025 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/5.0272085>

I. INTRODUCTION

A. Motivation and aims

A spherical tokamak is a torus-shaped device with a low aspect ratio that uses strong magnetic fields to confine and control a thermonuclear fusion plasma, with the goal of producing fusion energy.¹ The Spherical Tokamak for Energy Production (STEP), currently in the design phase and targeting completion in 2040,² is one of a few ongoing fusion research and development projects based on the spherical tokamak concept. To deliver fusion power to the grid on such a short timescale, researchers are increasingly designing next-generation tokamaks *in silico* with multi-physics simulations, many of which require high-performance computing (HPC) resources.

STEP is no exception, with initial concept designs for the tokamak (and the associated plasma) being generated by low-fidelity integrated modeling codes³ such as PROCESS^{4,5} and Bluemira.^{6–8} These codes use simplified physics and engineering models to produce designs within seconds or minutes. In contrast, more complex medium- to high-fidelity codes, such as JINTRAC,⁹ incorporate more detailed physics models but can require days or weeks to complete a single simulation. These higher-fidelity simulations play a crucial role in refining, integrating, and validating the initial concept design across the entire fusion power plant.¹⁰ Making the most efficient use of these

computationally expensive simulations is critical if we wish to accelerate the design of future fusion power plants like STEP.

Our focus here will be on the design of the poloidal field (PF) coil system, which plays a critical role in controlling the position and shape of the plasma in both the core and divertor regions of the tokamak.¹¹ In particular, some coils are crucial for managing the vertical stability of elongated plasmas, such as those in spherical tokamaks, where the higher elongation can lead to larger vertical instability, risking disruption without appropriate control.¹² By generating poloidal magnetic fields, the PF coil system ensures the plasma remains in equilibrium, balancing the inward-facing magnetic forces produced by the coils against the outward-facing pressure-driven forces generated by the plasma.¹³ The design of the system—in terms of the coil positions, sizes, and shapes—will have a significant impact on plasma performance and stability and will therefore need to satisfy a number of competing (and often conflicting) constraints. In the plasma, for example, constraints are required to ensure X-points form in specific locations (for stability), strikepoints hit the correct divertor plates (for heat management), and total current density limits on the PF coils are not exceeded. In terms of the tokamak itself, the locations/sizes of the coils will inevitably be constrained by the vacuum vessel, diagnostic systems, and maintenance ports (to name but a few).

In addition to constraints, there will be a number of objectives related to the desired operational plasma conditions that we wish the chosen coilset to minimize or maximize (depending on the objective). This could include minimizing the coil size to reduce fabrication, construction, and installation costs or could include minimizing current flows to reduce power consumption and structural stresses from forces produced by the coils.¹⁴ Moreover, we may wish to optimize certain properties of the plasma in the divertor chambers in order to minimize heat loads on plasma facing components and improve exhaust performance.^{15,16} Simultaneously satisfying both the objectives and constraints will require the solution of a complex optimization problem that needs to be tackled in a systematic, computationally efficient manner.

In this paper, we will perform multi-objective Bayesian optimization (BO) on an earlier baseline design of the STEP PF coil system.¹⁷ Our aims are to:

- (i) design and outline an easy-to-use BO framework, which is flexible, data efficient (reducing the computational cost of design), and can yield more optimal designs than obtained through other exhaustive optimization schemes.
- (ii) identify a *Pareto front*, i.e., a set of optimal PF coil locations, that outperform the baseline for some given objectives and constraints.
- (iii) motivate more widespread adoption of BO for the in silico design of interlinked components on future tokamak devices to save time, minimize financial costs, and improve plasma performance.

We should stress that this work has not had a direct impact on the current design of the STEP PF coil system¹⁸ and is instead a demonstration of a generalizable BO framework for PF coil system design. We do wish to highlight, however, that the framework is completely machine agnostic and can be used with different objectives and constraints to the ones we use here. It is the hope that frameworks such as this will be adopted more regularly within the integrated modeling codes currently used for tokamak design.

B. Related work

PF coilsets are typically optimized using integrated modeling codes for tokamak power plant design. A common approach is to force the PF coils to lie on a contour “rail” that surrounds the core plasma, reducing the number of degrees of freedom in the optimization problem.^{14,19} Exclusion zones along the rails enforce engineering constraints, before nonlinear (non-Bayesian) optimization is performed with respect to some pre-specified objectives and constraints on the plasma boundary shape.

While well-established, rail-based methods can restrict the PF coil design space, often rely on estimated objective function gradients, and can struggle with multiple competing objectives. They are primarily suited to conventional aspect ratio tokamaks, where PF coil rails are placed outside (and close to) the toroidal field (TF) coils, sometimes leading to intersection issues. BO, on the other hand, performs gradient-free global optimization, can handle diverse constraints, and uses a surrogate model of the multi-output objective function to intelligently guide function evaluations. This helps balance exploration of new designs and exploitation of known optimal designs, leading to high levels of data efficiency.

Despite these advantages, the adoption and application of BO in fusion engineering and design has, so far, remained relatively limited. Brown *et al.*^{20,21} aimed to improve six key properties of the safety factor profile by using BO on the current profiles in STEP. They also demonstrate that BO performs better than a genetic algorithm with the same number of black-box function evaluations (as we will do later on). Mehta *et al.*²² use BO to find the parameters such as neutral beam injection power, plasma current, and plasma elongation in the DIII-D tokamak that safeguard against disruption during the ramp-down phase. Similarly, Pusztai *et al.*²³ use BO to mitigate the impact of disruptions in ITER by exploring how injected deuterium and neon can minimize runaway electron currents, transported heat, and quench time post-disruption. Järvinen *et al.*²⁴ also investigate runaway electron currents using BO as an advanced sampling method to help calibrate uncertainty and minimize the discrepancy between simulations and experimental data. For fusion component design, Humphrey *et al.*²⁵ demonstrate the use of BO to minimize stresses in parametrized divertor monoblocks under fusion conditions. The most relevant work to ours is that of Nunn *et al.*,²⁶ who use multi-objective BO to optimize TF coil shapes to reduce both financial costs and magnetic ripples (which affect plasma stability and performance). In contrast, our approach deals with more computationally expensive, failure-prone plasma equilibrium simulations without analytic objective/constraint functions, necessitating the use of a classifier alongside the surrogate model.

The work here is inspired by that of Hudoba *et al.*,¹⁷ in which the authors seek to optimize the STEP PF coil system by minimizing deviations of key plasma parameters from a baseline scenario (which we adopt) and coil currents, while maximizing divertor performance metrics. Using a free-boundary equilibrium solver, thousands of potential PF coilsets are sampled and evaluated (in a Monte Carlo-type approach) before optimal solution sets are identified heuristically. We aim to provide and fully outline an alternative, much more data efficient, framework for carrying out similar multi-objective optimization that can return a STEP equilibrium similar to the baseline.

There are also a number of areas in fusion design where BO has yet to be applied but could potentially offer significant benefits. For example, parameter scans for optimal magnetic sensor placement, as explored for TCV²⁷ and SPARC,²⁸ could benefit from BO's sample efficiency, saving computational resources and time. Similarly, these benefits could transfer to existing frameworks for stellarator coil design.^{29–31}

C. Outline

In Sec. II, we describe the multi-objective BO problem, the Gaussian process (GP) surrogate model, the classifier scheme, and the acquisition function required in the BO loop. We follow this in Sec. III by defining the PF coil design problem in terms of the input space, the objectives we seek to optimize, and the constraints on the plasma and the machine. In addition, we describe the simulator used to generate the plasma equilibria for each PF coilset and define cases in which the simulator may fail to produce a valid equilibrium (requiring the classifier). The numerical experiments are detailed and presented in Sec. IV. To highlight the data efficiency of the BO scheme, we carry out a number of experiments with a fixed computational budget and assess performance against alternative optimization methods. In Sec. V, we discuss our findings, highlight any major advantages and

disadvantages of the BO framework applied to this problem, and propose avenues for future work.

II. MULTI-OBJECTIVE BAYESIAN OPTIMIZATION

BO is a method for performing gradient-free global optimization of black-box functions, typically utilized when the function is expensive-to-evaluate.³² Practitioners will often want to identify (feasible) optimal points of the function's input/output spaces with as few function evaluations as possible—especially if there is a limited computational budget.

Here, we are interested in optimizing the nonlinear function $f: \mathcal{U} \subseteq \mathbb{R}^d \rightarrow \mathbb{R}^{l+m}$ that takes in a d -dimensional input and returns l objectives and m constraints. More formally, the aim of *multi-objective BO* ($l > 1$) is to solve

$$\begin{aligned} \operatorname{argmin}_{\mathbf{x} \in \mathcal{U},} & f_{[1..l]}(\mathbf{x}), \\ & f_{[l+1..l+m]} \leq 0, \end{aligned} \quad (1)$$

where $f_{1..k}$ denotes the first k components of f and $\leq$ denotes a component-wise less than or equal to comparison.

Given we need to optimize over multiple competing objectives, problems such as (1) will often involve tradeoffs where improving one objective may come at the expense of another. The aim is therefore to seek the set of *Pareto optimal* solutions $\mathcal{P}$ that are not *dominated* by any other solutions. A solution $\mathbf{x}$ *dominates* another $\mathbf{x}'$, denoted $\mathbf{x} \prec \mathbf{x}'$, if and only if $f(\mathbf{x}) \leq f(\mathbf{x}')$ and $\exists j \in \{1, \dots, l\}$ such that $f_j(\mathbf{x}) < f_j(\mathbf{x}')$. In short, a solution $\mathbf{x}$ dominates $\mathbf{x}'$ if it is at least as good in all objectives and strictly better in at least one. Given a dataset

$$\mathcal{D} = \{(\mathbf{x}_i, f(\mathbf{x}_i))\}_{i=1}^N,$$

consisting of N evaluations of f , the *Pareto set* for (1) is defined as

$$\mathcal{P}(\mathcal{D}) := \{\mathbf{x} \in \mathcal{D} \mid \nexists \mathbf{x}' \in \mathcal{D} \text{ s.t. } \mathbf{x}' \prec \mathbf{x}\}. \quad (2)$$

The *Pareto front*, denoted $\mathcal{P}_f$, is defined as the image of the Pareto set, i.e., $\mathcal{P}_f := \{f(\mathbf{x}) \mid \mathbf{x} \in \mathcal{P}(\mathcal{D})\}$. See Ref. 32, Chap. 11.7 for an illustration of the Pareto front.

A. The Bayesian optimization loop

The key component in BO for identifying feasible and optimal trade-offs between the objectives is a *probabilistic surrogate model*, capable of performing uncertainty-based exploration. This model is typically trained on some initial dataset by maximizing its marginal likelihood—more details on this surrogate model are given in Sec. II B.

The first stage in BO (refer to Fig. 1) is to construct this initial dataset (which we will call $\mathcal{D}$) by taking N samples $\mathbf{x} \in \mathcal{U}$ and evaluating them all using f . One popular method used is Sobol sampling,³³ whereby samples are chosen quasi-randomly with low discrepancy to achieve approximately uniform coverage of the input space. The number of samples N chosen/required may depend on the size of d , the computational budget available, and if parallel processing is available (for the f evaluations). Note that at this point, while we could use $\mathcal{D}$ to immediately generate a Pareto set $\mathcal{P}(\mathcal{D})$, this would almost certainly be a poor estimate given a lack of data points and that most would reside in nonoptimal regions of the objective space.

It is worth noting that for many black-box functions such as f , there will be *failure regions* of the input space that cannot be evaluated.

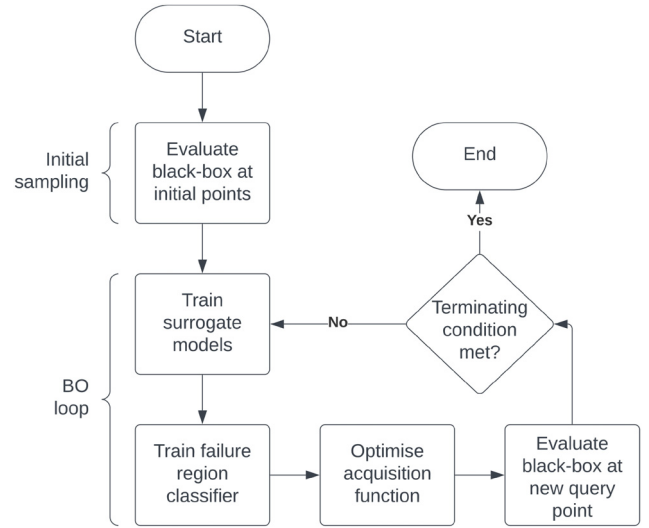


FIG. 1. A flow chart illustrating the standard “BO loop” along with an additional failure region classifier step—see text for more details.

The reasons for failure in our particular setting are discussed further in Sec. III B 5. During construction of the initial dataset, samples that lie in failure regions may be encountered and so we do not wish to include these in the dataset. We do, however, wish to learn from these samples so that we do not encounter similar samples again and therefore we train a *classifier* to predict when this may happen—a similar approach was taken by Hornsby *et al.*³⁴ when generating gyrokinetic simulation datasets for spherical tokamaks. This classifier will be used within the BO loop, which can be seen in Fig. 1 and is now outlined:

Stage 1: Generate the initial dataset $\mathcal{D}$ using the Sobol sampling scheme.

Stage 2: Train the surrogate model using the dataset $\mathcal{D}$ (see Sec. II B) to obtain a probabilistic model: $f(\mathbf{x})|\mathcal{D}$. Note that we exclude *failure* samples during training as they do not provide any valid information about the objective or constraint values.

Stage 3: Train the failure region classifier using the generated data (see Sec. II C).

Stage 4: Maximize the *acquisition function* over the input space to identify the most “informative” point $\mathbf{x}^*$ to observe next, given the current surrogate model (and classifier) $f(\mathbf{x}^*)|\mathcal{D}$ (see Sec. II D).

Stage 5: Evaluate $f(\mathbf{x}^*)$ and add it to the dataset: $\mathcal{D} := \mathcal{D} \cup \{(\mathbf{x}^*, f(\mathbf{x}^*))\}$.

Stage 6: Check whether the terminating condition is met and if not, return to Stage 2. In our case, we check whether or not the maximum number of iterations has been exceeded (to remain within our computational budget). Other terminating conditions include stopping once improvements in the acquisition function are below some threshold or when the objectives are deemed to be sufficiently optimal.³⁵

B. The Gaussian process surrogate

In BO, the most common type of surrogate used is a Gaussian process (GP), which is a probabilistic machine learning model for

performing inference on the value of a function $f: \mathcal{U} \rightarrow \mathbb{R}$ given some training data—see Ref. 32, Chap. 2. It is characterized by a mean function $m: \mathcal{U} \rightarrow \mathbb{R}$ and a positive semi-definite covariance function $k: \mathcal{U} \times \mathcal{U} \rightarrow \mathbb{R}$ (with $k(\cdot, \cdot) \geq 0$) such that the prior can be defined as

$$f \sim \mathcal{GP}(m, k). \quad (3)$$

The distribution of this prior is the joint distribution of (infinitely) many Gaussian random variables and can be thought of as a distribution over functions. Therefore, at a finite set of evaluation points $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots\} \subset \mathcal{U}$, we have that

$$f(\mathbf{X}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad (4)$$

where $\boldsymbol{\mu} = [m(\mathbf{x}_1), m(\mathbf{x}_2), \dots]^T$ is the mean vector and $[\boldsymbol{\Sigma}]_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) \forall i, j \in \{1, 2, \dots\}$ is the covariance matrix.

Training a GP requires conditioning the prior (3) on the dataset of known function evaluations $\mathcal{D}$ (with outputs standardized to mean 0, standard deviation 1 and the inputs transformed to the unit hypercube) such that we obtain the following posterior distribution:

$$f(\mathbf{X})|\mathcal{D} \sim \mathcal{N}(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}). \quad (5)$$

This conditioning can be done analytically (see Ref. 32, Chap. 2.2 for formulas for $\hat{\boldsymbol{\mu}}$ and $\hat{\boldsymbol{\Sigma}}$) and effectively tells the model to assign higher probability to functions that fit the training data well.

The quality of this posterior distribution (in terms of the mean accuracy and variance calibration), however, is highly dependent on the choices made for the functions m and k . A typical choice for the mean function is $m \equiv 0$, which assumes no prior knowledge of the function being modeled and ensures model predictions from (5) are heavily influenced by the training data. The choice of covariance kernel is formed via our prior belief in the expected behavior of the true function being modeled (e.g., nonperiodicity and smoothness). The covariance function used here is the Matérn-(1/2) (or exponential) kernel

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2}{\ell}\right),$$

where $\|\cdot\|_2$ denotes the Euclidean distance.³⁶ In addition, the parameters ℓ and σ define the input length scale (smaller values produce more “wiggly” functions) and the function noise (smaller values lead to lower predictive uncertainty in the function). The covariance kernel encodes the relationship between input points and the resulting covariance matrix quantifies how a change in one point influences changes in another across the domain. The *hyperparameters* ℓ and σ are tuned (for example, using traditional non-Bayesian optimization algorithms) to produce the best fit to the training data such that the marginal log-likelihood of the posterior (5) is maximized.

It should be noted that while we have described *scalar* output GPs here, in practice we model each output dimension of $\mathbf{f}$ using its own scalar GP. This assumes each output of $\mathbf{f}$ is uncorrelated (i.e., independent) of one another and means that we require $l + m$ “stacked” GPs to model the joint distribution over $\mathbf{f}$. More importantly for BO, it is crucial that the surrogate model is relatively cheap to train and evaluate compared to the cost of evaluating $\mathbf{f}$.

C. The classifier

The aim of binary classification is to label each data point in the input space as either a failure (0) or a nonfailure (1). The probability

that a point $\mathbf{x} \in \mathcal{U}$ is a nonfailure is modeled using a GP over a latent function, which is then transformed via the logistic function:³⁷

$$p(\text{non-failure} | \mathbf{x}) = \frac{1}{1 + e^{-f(\mathbf{x})}} \quad \mathbf{x} \in \mathcal{U}. \quad (6)$$

Using the logistic function transforms the GP from a regression model (as it is used in Sec. II B) to a classifier by mapping its prediction into a probability that a point belongs to the nonfailure class.

The GP model with classification now has a Bernoulli likelihood $p(\mathcal{D}|\mathbf{f})$, making the calculation of the posterior distribution $p(\mathbf{f}|\mathcal{D})$ analytically intractable—unlike in (5) where the prior, likelihood, and, therefore, posterior were all Gaussian [see Ref. 38, Chap. 3.4]. To address this, we approximate the posterior using a variational distribution $q(\mathbf{f}|\mathcal{D}; \boldsymbol{\lambda})$, chosen such that its likelihood $q(\mathcal{D}|\mathbf{f}; \boldsymbol{\lambda})$ is Gaussian—with $\boldsymbol{\lambda}$ parameterizing the new distribution.³⁹ The parameters $\boldsymbol{\lambda}$ are found by maximizing the evidence lower bound

$$\mathcal{L}(\boldsymbol{\lambda}) := \mathbb{E}_{q(\mathcal{D}|\mathbf{f}; \boldsymbol{\lambda})}[\log p(\mathcal{D}|\mathbf{f})] - \text{KL}[q(\mathbf{f}; \boldsymbol{\lambda}) || p(\mathbf{f})].$$

The first term represents the expected log likelihood (observing the training data given the probability distribution over functions), while the second term denotes the (non-negative) Kullback–Leibler divergence between the two distributions. Clearly if the KL divergence was zero (the distributions were identical), we would be maximizing over the original (log) Bernoulli likelihood. Once the $\boldsymbol{\lambda}$ are found, the GP can be conditioned on the data (as was shown in Sec. II B) using the new Gaussian likelihood $q(\mathcal{D}|\mathbf{f}; \boldsymbol{\lambda})$.

In classification, imbalance in the dataset—where one label is significantly more prevalent—can create a poor quality classifier. This results in the classifier being accurate by simply predicting the majority class, rendering it useless for identifying failure regions. To combat this, we employ *oversampling*, which randomly duplicates samples in the minority class such that both labels are equally represented in the training dataset.⁴⁰ As a result, the classifier cannot achieve a high accuracy by simply predicting one class and a higher quality model is produced.

D. The acquisition function

Based on knowledge from the trained GP and classifier, the acquisition function provides us with a way to estimate how informative evaluating $\mathbf{f}$ at a previously unseen point $\mathbf{x} \in \mathcal{U}$ will be. Depending on the task at hand, there are many possible choices of acquisition functions, each tailored to specific objectives. As mentioned before, the key factor in selecting an appropriate one is that it should be computationally cheap (compared to $\mathbf{f}$) to evaluate given the surrogate model.

Here, we use the *expected hypervolume improvement* (EHVI) function, which seeks to quantify the expected increase in the hypervolume of $\mathcal{P}_{\mathbf{f}}$ when adding a new point to the dataset $\mathcal{D}$.⁴¹ The *hypervolume* HV of $\mathcal{P}_{\mathbf{f}}$ is defined as the l -dimensional integral of the subspace

$$\{\mathbf{y} \in \mathbb{R}^l | \exists \mathbf{p} \in \mathcal{P}_{\mathbf{f}} \text{ s.t. } \mathbf{p} \prec \mathbf{y}\},$$

dominated by $\mathcal{P}_{\mathbf{f}}$.⁴²

EHVI is particularly suited to multi-objective optimization, as it effectively balances exploration and exploitation by focusing on

regions of the search space that are both uncertain and potentially optimal. The EHVI function $\alpha_{\text{EHVI}} : \mathcal{U} \rightarrow \mathbb{R}$ is given by

$$\alpha_{\text{EHVI}}(\mathbf{x}) = \mathbb{E}_f [\text{HV}(\mathcal{P}_f \cup \{\mathbf{f}(\mathbf{x})\}) - \text{HV}(\mathcal{P}_f)],$$

where $\mathbb{E}_f$ is the expectation operator of (5) (with respect to the l objectives, not the constraints). Recalling that $\mathbf{f}(\mathbf{x})$ is a random variable, this function describes how much additional volume in objective space we expect to gain by sampling at a new point $\mathbf{x}$, relative to the current Pareto front. Please refer to Yang *et al.*⁴² for a more rigorous treatment of this material.

As mentioned before, we have both constraints on the function $\mathbf{f}$ and failure regions in the input space. To this end, we define the *probability of feasibility* as

$$g(\mathbf{x}) = p(\text{non-failure}|\mathbf{x}) \prod_{i=1,\dots,m} p(f_{i+1}(\mathbf{x}) \leq 0),$$

where f_{i+1} to f_{i+m} are the GP models of the m constraint functions and $p(\text{non-failure}|\mathbf{x})$ is the classification model (6). This measures the joint probability that a given point in the input space is feasible (respects all of the constraints) and is not a failure.

Using this probability, we can then define a *constrained acquisition function* (ECHVI)

$$\alpha_{\text{ECHVI}}(\mathbf{x}) = \begin{cases} \alpha_{\text{EHVI}}(\mathbf{x}) \cdot g(\mathbf{x}) & \text{if } g(\mathbf{x}) \geq \lambda, \\ 0, & \text{otherwise,} \end{cases} \quad (7)$$

which weights α_{EHVI} by g and enforces a cutoff threshold^{43,44} (here we use $\lambda = 0.5$). This ensures that samples with a probability of feasibility less than λ are excluded from consideration, while those with probabilities higher are more likely to be selected during the optimization than those with lower probability (but still above λ). We also note that α_{EHVI} must be calculated with respect to the hypervolume of the *feasible* Pareto frontier by excluding infeasible points from the dataset.

To find the next most informative sample, we use single-objective (non-Bayesian) optimization to find the point that maximizes α_{ECHVI}

$$\mathbf{x}^* = \underset{\mathbf{x} \in \mathcal{U}}{\text{argmax}} \alpha_{\text{ECHVI}}(\mathbf{x}).$$

This is done using the L-BFGS-B⁴⁵ algorithm, which makes use of multiple restarts to avoid local maxima and avoid the discontinuity in α_{ECHVI} . Once found, $\mathbf{x}^*$ is evaluated using $\mathbf{f}$ and added to the dataset $\mathcal{D}$.

III. THE POLOIDAL FIELD COIL DESIGN PROBLEM

The PF coil design problem described here is concerned with identifying the set of PF coil positions that will optimize some aspects of both cost and performance of a STEP-like tokamak, subject to strict design and engineering requirements. In this section, we will describe the inputs, objectives, and constraints required to formulate the optimization problem as well as the underlying STEP baseline design and the simulator required to calculate plasma equilibria. Throughout, we will be working within a cylindrical coordinate system (R, ϕ, Z) which denotes the major radius, the toroidal direction (into the page), and the height, respectively.

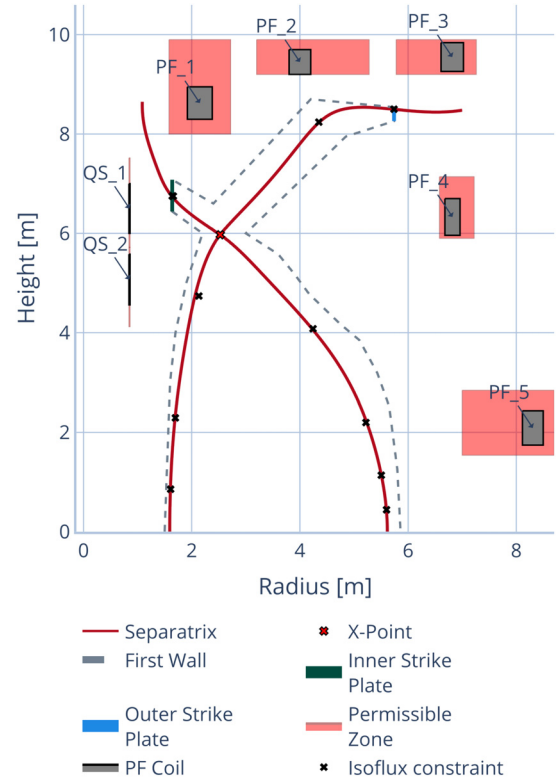


FIG. 2. The baseline setup from Hudoba *et al.*¹⁷ in the RZ plane (due to vertical symmetry about $Z = 0$ only the top half of the tokamak is shown). The separatrix (red) outlines the shape of the plasma core and the divertor legs which hit the inner (green) and outer (blue) strike plates. The initial location of the seven PF coils (gray rectangles) are displayed within their respective permissible zones (red rectangles). Note the absence of the central solenoid, which is not used in the flat-top phase of operation shown here. The isoflux constraints (black crosses) define locations at which the separatrix should pass through.

A. The STEP baseline design

We will be working with the initial PF coil setup and limiter geometry from the baseline STEP design presented by Hudoba *et al.*¹⁷ The design is shown in Fig. 2, and the information available to us from the baseline dataset is as follows:

- Names, centroid positions (R^c, Z^c) , and half width/heights (dR, dZ) of the PF coils.
- Permissible zones for each PF coil, i.e., the region of the RZ-plane in which each coil can be placed without intersecting the TF coils, diagnostic systems, or other parts of the tokamak.
- Limiter contour that confines the plasma equilibrium. This was constructed using the strike plate locations and made to match the geometry illustrated in Tholerus *et al.*⁴⁶
- Strike plate locations, i.e., segments of the limiter in the inner and outer divertor where the legs of the plasma separatrix will strike.
- Separatrix of the plasma equilibrium, the X-points, and the strikepoints.
- Plasma pressure and toroidal magnetic field profiles required to solve for the equilibrium.

B. Optimization problem

The mathematical formulation of the PF coil optimization problem requires stacked scalar inputs, objectives, and constraints so that we can map a vector of PF coil positions to a vector of objective/constraint values.

1. Input space

As can be seen in Fig. 2, there are seven up-down symmetric (around $Z = 0$) PF coil circuits each with their own (R^c, Z^c) centroid coordinate that is allowed to move freely such that no part of the coil leaves the permissible zone. The exceptions are the two quasi-solenoid (QS) coils, positioned above (and below) the central solenoid for magnetic shaping in the inner divertor, which are only able to move vertically. This results in a twelve dimensional input space for the optimization problem: five pairs of (R^c, Z^c) coordinates for the PF coils and one Z^c coordinate for each of the two QS coils. We normalize each of the coordinates with respect to their own permissible zones so that we can work with the unit hypercube $[0, 1]^{12}$ as our input space. A more detailed explanation of the normalization process can be found in Appendix.

2. Objectives

In this problem, we consider two scalar objective functions that we wish to optimize with multi-objective BO—though we should note that nothing prevents us from adding more objectives.

The first objective is to *minimize* the volumetric sum of the PF coils. This is important as smaller coils require less physical material and therefore weigh less, making the fabrication, transportation, and installation process less arduous and costly. Recall, each PF coil is modeled as a rectangle in the RZ -plane and as an annulus in the $R\phi$ -plane. The total volume of the seven PF coils (upper and lower components inclusive) can therefore be defined as

$$V = 8\pi \sum_{i=1}^{14} R_i^c dR_i dZ_i, \quad (8)$$

where the coil index $i = 1, \dots, 7$ for upper and $i = 8, \dots, 14$ for lower coils. It is important to note that the PF coils do not change shape in the poloidal plane (i.e., cross-sectional areas are fixed, equal to $4dRdZ$) and so the volume of each coil will change only when its central radial position R^c changes.

The second objective is to *maximize* the average of the inner connection length (ICL) and the outer connection length (OCL). The connection length is the distance traced out by a helical (i.e., moving both poloidally and toroidally) magnetic field line that starts at the inner (outer) edge of the last closed flux surface (LCFS) at the midplane and ends at the inner (outer) strikepoint.^{47,48} We refer to these inner and outer midplane points as the IMP and OMP, respectively. Larger connection lengths ensure that hot plasma leaving the core edge region will travel a farther distance and therefore cool to more acceptable temperatures before hitting the strike plate. This is an important aspect of managing heat loads in the divertor region.

The ICL and OCL are calculated by carrying out an integral over the length of the magnetic field line of interest.⁴⁹ Tracing the magnetic field line requires the solution of a system of ordinary differential

equations (ODEs) for the position vector of a point along the field line trajectory $\mathbf{r}(\ell)$,

$$\frac{d\mathbf{r}(\ell)}{d\ell} = \mathbf{b}(\mathbf{r}(\ell)), \quad \ell \in [0, L], \quad (9)$$

where $\mathbf{b}$ is a unit vector along the direction of the magnetic field $\mathbf{B}$ in cylindrical coordinates

$$\mathbf{B}(\mathbf{r}(\ell)) = \begin{pmatrix} -\frac{1}{R} \frac{\partial \psi(R, Z)}{\partial Z} \\ \frac{F(\psi(R, Z))}{R} \\ \frac{1}{R} \frac{\partial \psi(R, Z)}{\partial R} \end{pmatrix}.$$

Here, $\psi(R, Z)$ denotes the scalar poloidal magnetic flux and F the toroidal magnetic field profile. To solve this problem, we use a fourth-order Runge–Kutta method (with appropriately chosen step size $\Delta\ell$) and integrate until a terminating condition is met (see next paragraph), recording the value of L obtained (i.e., the total number of steps $\Delta\ell$).

In practice, however, the calculation is split into two stages. The first stage involves integrating from an initial position $\mathbf{r}(0)$, which is selected to be 3 mm radially outside the IMP/OMP and ending at some very small distance away from the upper X-point. The second stage integrates between the inner/outer strikepoint and the point close to the upper X-point. The length of these individual sections is then combined to return the final connection length. The two flux surfaces traced out when calculating the ICL and OCL are visualized in Fig. 3. These techniques reduce the likelihood of the integrator getting stuck at the exact X-point, traveling around the LCFS (instead of going up into a divertor), and from traveling into the wrong divertor.

3. Constraints

In addition to the objectives, we also have a number of design and engineering constraints that need to be satisfied so that each PF coilset considered in the BO loop produces an equilibrium with key targets that are similar to the baseline equilibrium and does not violate coil current limits. The bounds of the constraints are summarized in Table I.

The first constraint is on the shape of the LCFS, which is defined as the contour of (R, Z) points that pass through the X-point closest to the magnetic axis—recall Fig. 2. We denote this region of the RZ -plane as Ω_p and quantify the difference between two different regions using

$$\eta(\Omega_p^1, \Omega_p^2) := \frac{|\Omega_p^1 \cup \Omega_p^2| - |\Omega_p^1 \cap \Omega_p^2|}{|\Omega_p^1| + |\Omega_p^2|} \in [0, 1],$$

where $|\cdot|$ denotes the cross-sectional area of a region in the poloidal plane.⁵⁰ This parameter quantifies the ratio of the total nonoverlapping areas and the sum of the two areas. Placing an upper limit on this ratio enables us to constrain the LCFS shape of the new equilibrium (Ω_p^2) to be similar to that of the baseline (Ω_p^1). This helps to ensure the new equilibrium has similar core performance to the baseline.

The second and third constraints place an upper limit on the distance between the strikepoints (i.e., where the separatrix first intersects the limiter geometry at some location) and the center of the strike

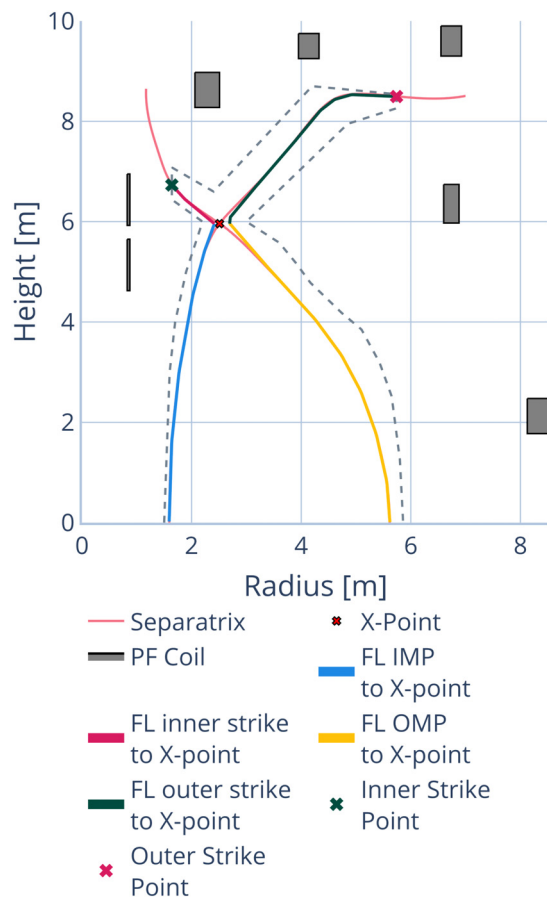


FIG. 3. The baseline setup from Fig. 2, this time marked with objective and constraint quantities. Shown are the separatrix (light pink), the upper X-point (red cross), and the inner (green star) and outer (red star) strikepoints. Also shown are the flux surfaces traced out to calculate the ICL (blue plus red lines) and OCL (yellow plus green lines).

plates. In rare cases, an equilibrium may have a separatrix that intersects the limiter multiple times on the same plate; hence, we need to account for that. The bound is half of the length of the strike plate, with one constraint on each of the inner and outer strike plates.

The fourth constraint will place an upper limit on the distance between the two X-point locations when mirrored about $Z = 0$. This

TABLE I. Constraint bounds enforced on the equilibria generated by FreeGS for a particular PF coilset.

Constraint	Bound (unit)
LCFS area ratio	≤ 0.012
Outer strike distance	≤ 0.14 (m)
Inner strike distance	≤ 0.32 (m)
X-point distance	≤ 0.01 (m)
Maximum current density	≤ 100 (MA/m ²)

distance should be minimal in a double-null plasma scenario as considered here. See Fig. 3 for the strikepoint and X-point locations.

The final constraint ensures the maximum current density

$$J_{\max} = \frac{1}{4} \max_{i \in [1 \dots 7]} \frac{I_i}{dR_i dZ_i},$$

in the PF coilset remains below the engineering limit defined in Nasr *et al.*¹⁸ Here, I_i denotes the coil current and the denominator is the coil area. This limits stresses in the PF coil structures and helps avoid quench events—a sudden loss of superconductivity, which can damage the coils.⁵¹

4. Equilibrium simulator

In order to calculate the aforementioned objective functions and evaluate whether or not the constraints have been met, we need a simulator that is able to generate a plasma equilibrium using the STEP baseline design and a given PF coilset. For this we use *FreeGS*, a free-boundary static inverse equilibrium solver.⁵² *FreeGS* will return a plasma equilibrium (in terms of the poloidal flux) and the PF coil currents required to generate it. It uses an optimization routine to identify the coil currents with respect to some constraints on the chosen plasma shape and a Picard iteration scheme to solve the free-boundary Grad-Shafranov problem.^{53,54} The required inputs to solve the equilibrium problem are as follows:

- The STEP baseline parameters and a PF coilset (permissible zones not required).
- Two X-point locations, one at (R^X, Z^X) and the other mirrored at $(R^X, -Z^X)$ (as we required an up-down symmetric double-null configuration like the baseline equilibrium).
- 23 isoflux constraints that link poloidal flux values on the core plasma boundary to the X-points and the divertor regions (i.e., constraints that ensure the poloidal flux $\psi(R, Z)$ at two different locations (R_1, Z_1) and (R_2, Z_2) are the same).

Given these inputs, *FreeGS* will return the coil currents in the PF coils required to generate an equilibrium that (closely) matches the one provided in the baseline. From this equilibrium, we can then calculate the values of the objectives and the constraints. From time to time, however, the simulator may fail to converge on a physically “valid” equilibrium, returning spurious objective and constraint values. This could be for a number of reasons such as solver instability or a physically incompatible PF coilset. This requires care and will be discussed in Sec. III B 5.

5. Failure regions

The equilibrium simulator will either (in rare cases) fail to converge or stop once the relative difference between the poloidal flux at successive iterations is below some tolerance threshold (returning an equilibrium solution). This stopping criteria does not, however, consider the physical validity of the equilibrium identified. In some cases, nonphysical equilibria that do not satisfy the X-point and isoflux constraints may be returned. In other cases, we may have an equilibrium for which we either cannot calculate the objectives/constraints or which return spurious objective/constraint values. The regions of the

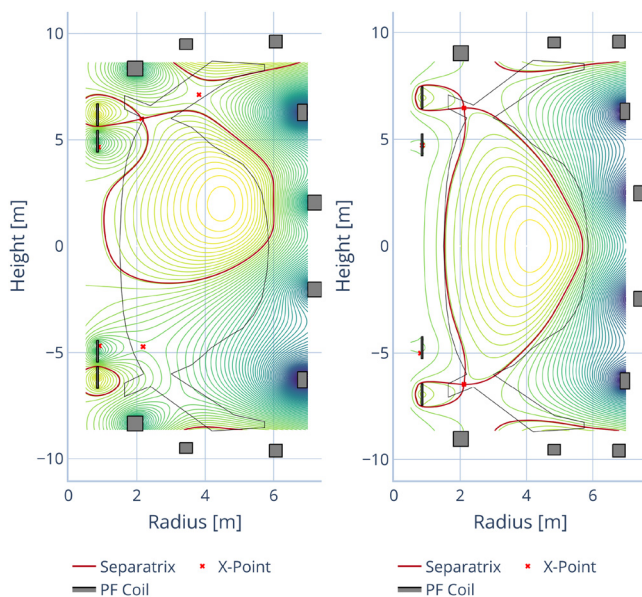


FIG. 4. Two examples of invalid equilibria returned by “converged” FreeGS simulations with different PF coilsets.

input space for which nonphysical (*invalid*) equilibria are returned (or if the simulator outright fails) will be referred to as *failure regions*.

In Fig. 4, we illustrate two cases of invalid equilibria returned by FreeGS. The left panel shows a single-null equilibrium with the LCFS intersecting the limiter in the core region. The right panel depicts an equilibrium in which both X-points have formed inside the divertor regions, far from the desired locations in the core, resulting in the LCFS again intersecting the limiter. This is problematic when calculating the ICL, as this calculation assumes the LCFS does not intersect the limiter geometry until hitting the strikeplate, resulting in an unfeasibly small ICL value. Calculations for both the ICL/OCL and the strike distances are spurious in this case.

To mitigate these issues, we can classify (recall Sec. II C) whether an equilibrium is valid by checking the following conditions:

1. the X-points (R^X, Z^X) and ($R^X, -Z^X$) must be to within 10 cm of the limiter boundary with $R^X \in [2.2, 3]$.
2. the LCFS does not intersect the limiter.⁵⁵

By actively avoiding sampling the PF coilsets where the simulator fails or produces such invalid equilibria (via the classifier), we can avoid wasting computational resources on solutions that do not provide any useful information to the BO loop.

IV. NUMERICAL EXPERIMENTS

In this section, we will perform the design optimization of the PF coil set problem. The first experiment will use multi-objective BO to find several Pareto optimal PF coilsets that respect the engineering and design constraints in Sec. III B. We analyze two of the Pareto optimal solutions in more detail, highlighting how the BO explores the solution space while respecting the trade-off between the objective functions. To further illustrate the data efficiency of the BO, we compare its performance against two other optimization methods (simple Sobol

sampling and a genetic algorithm) when using both identical and larger computational budgets.

To generate these results, we use the Trieste^{56,57} package, which provides the software implementations for Sobol sampling, acquisition functions, and Gaussian processes (via GPflow⁵⁸) Pygmo2⁵⁹ provides the genetic algorithm which we will use for benchmarking. When evaluating the Sobol samples with FreeGS,⁵² we make use of the CSD3 HPC cluster (see Acknowledgements) and the Simvue platform⁶⁰ to monitor simulation progress and store the objective/constraint data.

A. Stand-alone BO

In this experiment, we will limit ourselves to 128 evaluations of f : 64 Sobol samples to build the initial dataset and 64 sequential BO samples to intelligently explore the objective space and identify feasible optimal points.

In Table II, we display the proportions of each sampling set that result in feasible, infeasible (violating one or more constraints), and failed (invalid) PF coilsets. We can see that only 10% of the Sobol samples provide feasible designs and that once the BO loop begins running, we accumulate a much larger proportion of feasible designs with fewer failures. This shows that the GPs can accurately model the constraint responses and the acquisition function uses this to propose feasible samples.

The drop in failure region samples likely results from a combination of explicit classifier intervention and the scarcity of optimal samples near these regions, making them less likely to be chosen by the acquisition function. The classifier has a precision of 0.82, meaning 82% of the area included in the acquisition maximization (by zeroing the ECHVI in these regions) is indeed nonfailing, reducing waste of computational resources by potentially sampling failing points. A recall of 0.86 shows that only 14% of the nonfailure region is incorrectly avoided by the classifier; it is more important that this percentage is low because Pareto optimal solutions could exist here but would be missed.

The BO loop successfully identifies the Pareto front (see Fig. 5), capturing the trade-off between the volumetric sum (8) and the average connection length [recall (9)]. Generally, the latter BO iterations produce samples that dominate earlier samples, highlighting how BO learns from new data and exploits its new understanding of the functions to produce higher-quality samples.

During the initial BO iterations, the data are sparse, resulting in highly uncertain and inaccurate GPs. As a result, the exact location of the feasible regions is unclear and, as it explores for the first 20 iterations, the BO produces few feasible samples—see Fig. 6. Following this initial exploration, 30 iterations of exploitation takes place, where BO reliably produces feasible samples (those seen in Fig. 5). The final 20 iterations yield no feasible samples, indicating a return to exploration and suggesting there is little scope to find feasible optimal solutions around the current Pareto optimal points.

TABLE II. The number (and percentage) of samples from each sampling method that lie in failure, infeasible, or feasible regions.

Method	Failure	Infeasible	Feasible	Total
Sobol	23 (36%)	31 (48%)	10 (16%)	64
BO	4 (6%)	37 (58%)	23 (36%)	64

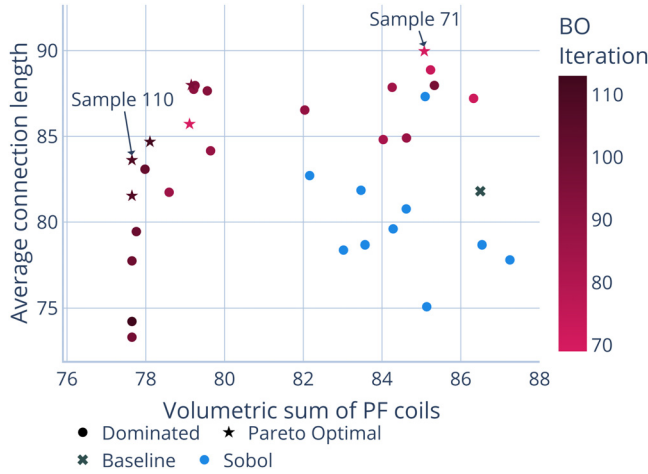


FIG. 5. Feasible solutions in the objective space for the 64 Sobol + 64 BO experiment. Shown are the baseline (gray cross), Sobol solutions (blue), and the BO solutions (light pink to red). Pareto optimal solutions are denoted with a star and dominated solutions with a circle, with light pink to red indicating successive BO iterations. Two of the BO samples, 71 and 110, are highlighted for further analysis.

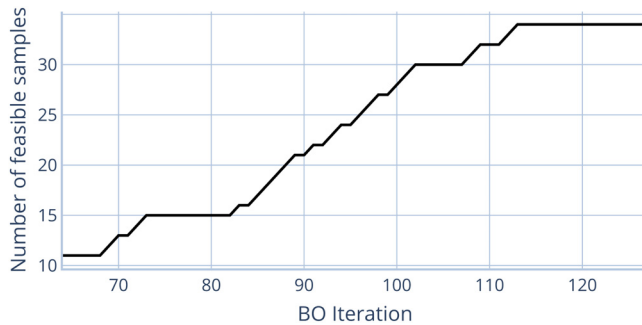


FIG. 6. The total number of feasible BO samples at the end of each BO iteration.

TABLE III. The objectives and constraint values (to three significant figures) for the baseline and two of the Pareto optimal solutions shown in Fig. 7. Also shown are the inner and outer line-averaged poloidal magnetic field readings.

Objective/constraint	Baseline	Sample 110	Sample 71
V (m ³)	86.5	77.6	85.1
ICL (m)	96.5	96.9	108
OCL (m)	67.1	70.3	71.8
LCFS area ratio	0.0108	0.0117	0.0103
Outer strike distance (m)	0.0934	0.0957	0.0962
Inner strike distance (m)	0.0297	0.0233	0.0305
X-point distance (m)	0.000374	0.00394	0.000919
$J_{\max}$ (MA/m ²)	71.4	82.4	98.8
Inner $\langle B_p \rangle$ (T)	0.414	0.413	0.373
Outer $\langle B_p \rangle$ (T)	0.552	0.537	0.530

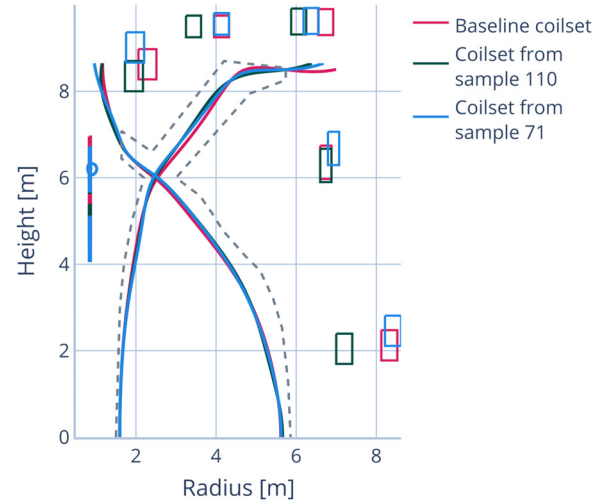


FIG. 7. The PF coilsets and corresponding separatrices for the baseline (pink) and two of the Pareto optimal equilibria: sample 71 (blue) and 110 (green) of the simulator (corresponding to samples from the 7th and 46th BO iteration, respectively).

In Table III, we display the objective/constraint values obtained from the baseline and two of the Pareto optimal PF coil sets shown in Fig. 7, with both samples obtained during the BO iterations. The 71st sample yields the highest average connection length while the 110th sample has the joint lowest volumetric sum (of these tied samples, it has the higher connection length). The first three rows of the table show the objective quantities for both samples while the intermediate five rows show the constraint values.

The objective values of the baseline (and its location in Fig. 5) relative to the Pareto optimal samples demonstrate that BO is able to improve the PF coilset design significantly. At a minimum, BO has yielded a reduction of V by 1.4 m³ and an increase in average connection length of 1.8 m over the baseline; this is not insignificant considering the low computational budget to achieve these gains. The constraints show that samples 110 and 71 are close to the constraint bounds for the LCFS shape and maximum coil current density, respectively. This could indicate that further optimization of these samples (and other Pareto optimal samples) is not possible without violating the constraints, hence the lack of feasible samples in the final BO iterations.

It is clear from Fig. 7 that the 110th sample has a smaller volumetric sum because PF coils 2, 3, 4, and 5 are closer to the device centerline ($R = 0$). However, the difference in average connection length is less obvious because the separatrices look (qualitatively, at least) very similar. The lower connection length in sample 110 results from a higher poloidal field, causing particles traveling from the midplane into the divertors to move faster, decreasing the number of times (and thus the distance) they orbit the tokamak toroidally. This can be seen in the final two rows of Fig. 7, which shows the line-averaged poloidal magnetic field $\langle B_p \rangle$ along the inner and outer connection length field lines, respectively.

B. Comparison between BO, Sobol sampling, and a genetic algorithm

Next, we compare the BO scheme against two other multi-objective optimization methods. The first method we compare against

uses quasi-Monte Carlo sampling via the Sobol method, essentially relying on random chance to sample feasible and optimal coilsets. The second method will use a genetic algorithm, specifically the “Non-Dominated Sorting Genetic Algorithm 2” (NSGA-II).⁶¹ NSGA-II, like all genetic algorithms, operates on the principle that combining the inputs of well-performing individuals within a population can produce offspring (new samples) that inherit characteristics from its parents and therefore may perform similarly or better. Starting with an initial population (in our case, Sobol samples), the algorithm generates additional samples through iterative recombination and mutation (applying slight random changes to the inputs), therefore introducing variation to explore the solution space.⁶² Constraints are handled by penalizing the objectives according to the number of violated constraints.⁶³ Similarly, the failure region is handled by returning large constants for the objectives, artificially making the sample appear very nonoptimal.

We run six additional experiments with each of these methods, the results of which are presented in Table IV. The first three (II, III, and IV) use the same computational budget as the BO experiment (I) from Sec. IV A, while the final three (V, VI, and VII) have a budget that is 8× larger.

Four of the experiments contain results for the NSGA-II genetic algorithm. For each budget, we include an NSGA-II experiment with the same initial population (64 samples) as experiment I (experiments III and VI) and another with a lower initial population (experiments IV and VII) but with the ability to run over more generations. In both cases, the experiments using a lower initial population size (IV and VII) outperform their counterparts with larger initial population sizes. This is because they are able to explore the solution space more widely (using more iterations), thus generating more feasible solutions and making incremental progress each generation toward the Pareto frontier. This can be seen by the higher percentage of feasible solutions sampled by these experiments compared to the others. For the remaining analysis, we will compare only these best performing NSGA-II experiments (IV and VII) against the Sobol sampling and BO.

From the results, we can see that experiment I produces better samples than both II and IV with a hypervolume at least 20% larger. Recall that a larger hypervolume indicates a feasible objective space with better trade-offs that are further from the (anti-optimal) reference point—the feasible solutions are shown in the objective space in Fig. 8.

TABLE IV. The hypervolume of the feasible solution set and the percentage of total samples taken that were feasible for each of the experiments run. Here, we compare the 64 Sobol + 64 BO experiment from Sec. IV A with pure Sobol sampling and the NSGA-II algorithm, each with the same number of samples (128). We also display Sobol and NSGA-II experiments that use 8× the number of samples (1024). All hypervolumes are calculated with respect to the same reference point.

	Experiment	<i>HV</i>	Feasible
I	64 Sobol + 64 BO	1990.9	26%
II	128 Sobol	1626.6	18%
III	64 Sobol + 64 NSGA-II	1544.2	16%
IV	8 Sobol + 120 NSGA-II	1659.2	49%
V	1024 Sobol	1711.5	13%
VI	64 Sobol + 960 NSGA-II	1713.6	23%
VII	32 Sobol + 992 NSGA-II	1820.3	64%

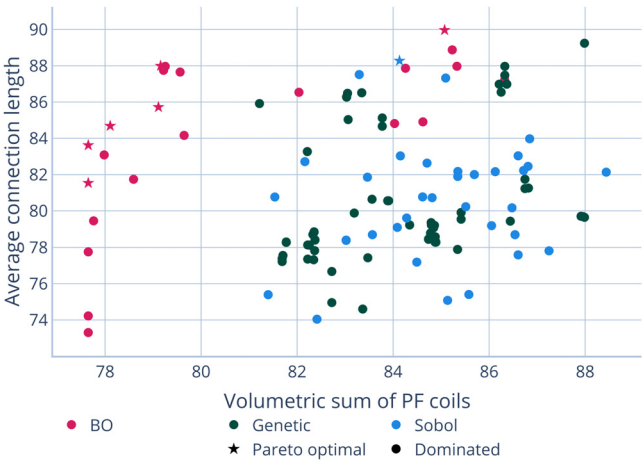


FIG. 8. Feasible solutions in the objective space for experiments I (red), II (blue), and IV (green) in Table IV. Pareto optimal solutions are denoted with a star and dominated solutions with a circle.

The genetic algorithm finds the most feasible samples, outperforming BO by nearly 2×. However, the hypervolume of experiment IV indicates few of these feasible samples offer any improvement over even quasi-random samples. This illustrates how the BO performs significantly better than the Sobol sampling and the genetic algorithm at finding Pareto optimal PF coilsets. Genetic algorithms find the most feasible PF coilsets; however, all of the samples are of significantly lower quality than those from BO.

BO continues to outperform Sobol sampling and the genetic algorithm even when we increase their computational budgets to 1024 samples. While the hypervolume returned in experiments V and VII are larger compared to those in II and IV (as expected), they still cannot reach the level achieved by the BO (with 1/8 th of the data). In Fig. 9, we again see the majority of Pareto optimal samples coming from the BO with a few being found by the genetic algorithm, with BO

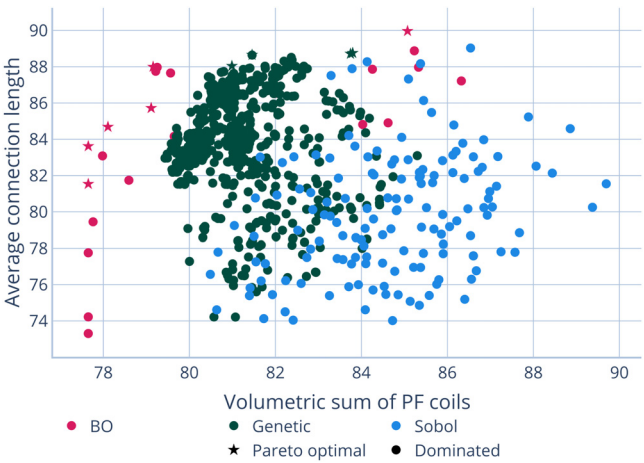


FIG. 9. Feasible solutions in the objective space for experiments I (red), V (blue), and VII (green) in Table IV. Pareto optimal solutions are denoted with a star and dominated solutions with a circle.

finding the best samples in each objective (the samples that optimize the marginals of the objective space). The vast majority of samples taken by the alternative methods are, however, dominated by others from the BO. Again, genetic algorithms find the most feasible samples; however, they form a front that underperforms that of BO, particularly in the volumetric sum.

V. DISCUSSION AND OUTLOOK

In this paper, we have demonstrated that BO can successfully identify a set of Pareto optimal PF coilsets in a spherical tokamak. Using underlying probabilistic models, it learns the trade-off between the volume of the PF coilset (i.e., the financial cost) and the average connection length produced by the corresponding equilibrium state, simultaneously respecting several physical plasma and engineering constraints. Compared to some existing optimization methods, quasi-Monte Carlo (Sobol) sampling, and a genetic algorithm (NSGA-II), BO identifies better solutions while using a significantly smaller computational budget, highlighting its effectiveness and data efficiency. Overall, the successful application of BO to a complex tokamak design problem should reinforce its suitability for future fusion power plant design challenges, particularly given the increasing reliance on high-fidelity, high-runtime HPC codes where data efficiency is critical.

The relatively poor performance of the Sobol sampling is expected and can likely be attributed to its sparse quasi-uniform coverage of the sample space. While uniform coverage is good for exploring high-dimensional spaces and training emulators (such as the one in our BO loop), the Sobol scheme lacks the ability to hone in on more desirable regions given it is forced to sample inputs within uniformly spaced partitions of the space. NSGA-II outperforms Sobol sampling, especially when both are afforded moderately high computational budgets, however, has an underwhelming performance against BO. While it excels at finding feasible samples, NSGA-II fails to find samples dominant over BO, even with a significantly higher computational budget. This is likely because the genetic algorithm favors sampling feasible points instead of exploring toward the feasible boundary and potentially finding a more optimal sample—the cost of infeasibility does not outweigh the reward of slight improvements in the objectives. It is possible more advanced treatments of the constraints⁶⁴ would improve the genetic algorithm's performance and allow it to explore closer to the feasible boundary, however, that is beyond the scope of this work.

To increase the applicability and extend this BO framework to ongoing and future PF coil design projects, a number of avenues of future work can be considered. For example, incorporating additional objectives and constraints should be a trivial task and could be used to help find coilsets that further improve performance. For example, one could try to maximize flux expansion to improve divertor performance or include PF coil shaping (in the input space) to try to extract further financial cost savings. While this would increase the dimensionality of the BO problem, the current framework can be readily adapted to support this via dimensionality reduction techniques.^{65–67}

ACKNOWLEDGMENTS

The authors would like to thank Agnieszka Hudoba for providing the baseline STEP data files and Theodore Brown for discussions around BO.

This work was funded by the EPSRC Energy Programme (Grant No. EP/W006839/1). To obtain further information, please

contact PublicationsManager@ukaea.uk. For the purpose of open access, the author(s) have applied a Creative Commons Attribution (CC BY) license to any Author Accepted Manuscript version arising.

This work was performed using resources provided by the Cambridge Service for Data Driven Discovery (CSD3), operated by the University of Cambridge Research Computing Service (www.csd3.cam.ac.uk). These resources were provided by Dell EMC and Intel using tier-2 funding from the Engineering and Physical Sciences Research Council (capital Grant No. EP/T022159/1) and DiRAC funding from the Science and Technology Facilities Council (www.dirac.ac.uk).

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Timothy Nunn: Data curation (lead); Formal analysis (lead); Investigation (lead); Methodology (lead); Resources (lead); Software (lead); Validation (lead); Visualization (lead); Writing – original draft (equal). **Kamran Pentland:** Conceptualization (supporting); Methodology (supporting); Writing – original draft (equal). **Vignesh Gopakumar:** Conceptualization (supporting); Methodology (supporting); Project administration (supporting); Writing – review & editing (equal). **James Christopher Buchanan:** Conceptualization (lead); Project administration (lead); Supervision (lead); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

APPENDIX: COIL LOCATION NORMALIZATION

Here, we outline how to normalize the centroid coordinates of each PF coil with respect to its permissible zone. First, define the lower left and upper right corners of each permissible zone as $V_1 = (R_{\min}, Z_{\min})$ and $V_3 = (R_{\max}, Z_{\max})$, respectively. Given each coil must entirely reside within its permissible zone, we know that the centroid must remain within a half-thickness of the permissible zone

$$(R^c, Z^c) \in [R_{\min} + dR, R_{\max} - dR] \times [Z_{\min} + dZ, Z_{\max} - dZ].$$

We can then obtain the normalized centroid coordinates (with respect to the permissible zone) by defining

$$\tilde{R}^c = \frac{R^c - (R_{\min} + dR)}{(R_{\max} - dR) - (R_{\min} + dR)} \in [0, 1],$$

$$\tilde{Z}^c = \frac{Z^c - (Z_{\min} + dZ)}{(Z_{\max} - dZ) - (Z_{\min} + dZ)} \in [0, 1].$$

An illustration of a PF coil and its permissible zone are shown in Fig. 10.

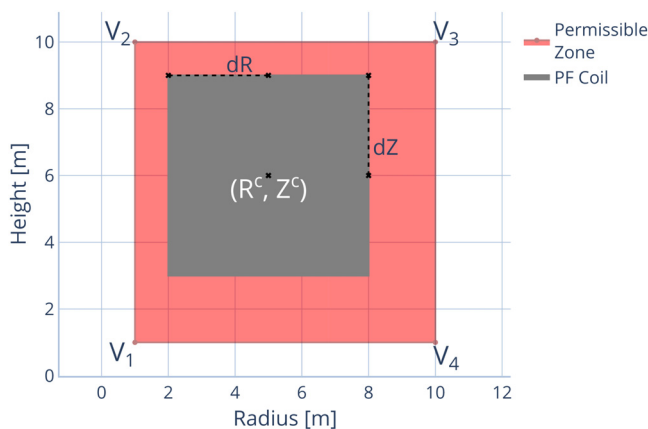


FIG. 10. Illustration of a rectangular PF (dark gray), with its centroid and half-width/height marked, and its permissible zone (red), with corner vertices marked.

REFERENCES

- ¹J. P. Freidberg, *Plasma Physics and Fusion Energy* (Cambridge University Press, 2008).
- ²I. Chapman, S. Cowley, and H. Wilson, "The spherical tokamak for energy production: Theme issue introduction," *Philos. Trans. R. Soc. A* **382**, 20230416 (2024).
- ³S. I. Muldrew, C. Harrington, J. Keep, C. Waldon, C. Ashe, R. Chapman, C. Griesel, A. J. Pearce, F. Casson, S. P. Marsden *et al.*, "Conceptual design workflow for the STEP prototype powerplant," *Fusion Eng. Des.* **201**, 114238 (2024).
- ⁴J. Morris, S. Muldrew, P. Knight, M. Kovari, A. Pearce, S. Kahn, J. Maddock, T. Nunn, R. Chapman, C. Swanson, H. Lux, J. Foster, K. Ellis, M. Kumar, G. Turkington, J. Lion, S. Chislett-McDonald, A. Brown, J. Edwards, C. Ashe, E. Fable, K. Zarebski, C. Griesel, R. Kemp, S. Gubbins, S. Pickering, J. Matthews, G. Graham, C. Mould, T. Miller, S. Gadgil, D. Short, J. Cook, and P. Lloyd, "PROCESS," 2024.
- ⁵S. I. Muldrew, H. Lux, G. Cunningham, T. C. Hender, S. Kahn, P. J. Knight, B. Patel, G. M. Voss, and H. R. Wilson, "PROCESS": Systems studies of spherical tokamaks," *Fusion Eng. Des.* **154**, 111530 (2020).
- ⁶M. Coleman, J. E. Cook, F. Franza, I. A. Maione, S. McIntosh, J. Morris, D. Short, A. I. Blair, M. Bluteau, H. Brooks, I. Chiang, S. Desai, M. Foord, O. Funk, G. A. Graham, J. Hagues, L. Humphrey, M. Johnson, S. Kahn, C. MacMackin, S. Mason, H. Saunders, D. Vaccaro, and O. Wong, "BLUEMIRA," 2025.
- ⁷M. Coleman and S. McIntosh, "BLUEPRINT: A novel approach to fusion reactor design," *Fusion Eng. Des.* **139**, 26–38 (2019).
- ⁸F. Franza, L. Boccaccini, E. Fable, I. Landman, I. Maione, S. Petschanyi, R. Stieglitz, and H. Zohm, "Mira: A multi-physics approach to designing a fusion power plant," *Nucl. Fusion* **62**, 076042 (2022).
- ⁹M. Romanelli, G. Corrigan, V. Parail, S. Wiesen, R. Ambrosino, P. D. S. A. Belo, L. Garzotti, D. Harting, F. Koechl, T. Koskela *et al.*, "JINTRAC: A system of codes for integrated simulation of tokamak scenarios," *Plasma Fusion Res.* **9**, 3403023 (2014).
- ¹⁰A. Davis, C. Waldon, S. I. Muldrew, B. S. Patel, P. Verrier, T. R. Barrett, and G. A. Politis, "Digital: Accelerating the pathway," *Philos. Trans. R. Soc. A* **382**, 20230411 (2024).
- ¹¹B. Lim, F. Simon, Y. Ilyin, C. Gung, J. Smith, Y. Hsu, C. Luongo, C. Jong, and N. Mitchell, "Design of the ITER PF coils," *IEEE Trans. Appl. Supercond.* **21**, 1918–1921 (2011).
- ¹²H. Anand, O. Bardsley, D. Humphreys, M. Lennholm, A. Welander, Z. Xing, J. Barr, M. Walker, J. Mitchell, and H. Meyer, "Modelling, design and simulation of plasma magnetic control for the spherical tokamak for energy production (step)," *Fusion Eng. Des.* **194**, 113724 (2023).
- ¹³J. Wesson and D. J. Campbell, *Tokamaks*, Vol. 149 (Oxford University Press, 2011).
- ¹⁴M. Coleman and S. McIntosh, "The design and optimisation of tokamak poloidal field systems in the BLUEPRINT framework," *Fusion Eng. Des.* **154**, 111544 (2020).
- ¹⁵A. Hudoba, S. Newton, G. Voss, G. Cunningham, and S. Henderson, "Divertor optimisation and power handling in spherical tokamak reactors," *Nucl. Mater. Energy* **35**, 101410 (2023).
- ¹⁶A. Hudoba, S. Bakes, G. Cunningham, S. Henderson, F. Eriksson, S. Marsden, T. Wilson, and STEP Team, "Optimisation of the poloidal field system for advanced divertor configurations in STEP," *Nucl. Fusion* **64**, 086055 (2024).
- ¹⁷A. Hudoba, G. Cunningham, and S. Bakes, and STEP Team and others, "Magnetic equilibrium optimisation and divertor integration in spherical tokamak reactors," *Fusion Eng. Des.* **191**, 113704 (2023).
- ¹⁸E. Nasr, S. C. Wimbush, P. Noonan, P. Harris, R. Gowland, and A. Petrov, "The magnetic cage," *Philos. Trans. R. Soc. A* **382**, 20230407 (2024).
- ¹⁹O. Meneghini, T. Slendebröck, B. C. Lyons, K. McLaughlin, J. McClenaghan, L. Stagner, J. Harvey, T. F. Neiser, A. Ghiozzi, G. Dose, J. Guterl, A. Zalzal, T. Cote, N. Shi, D. Weisberg, S. P. Smith, B. A. Grierson, and J. Candy, "FUSE (Fusion Synthesis Engine): A next generation framework for integrated design of fusion pilot plants," *arXiv:2409.05894* (2024).
- ²⁰T. Brown, S. Marsden, V. Gopakumar, A. Terenin, H. Ge, and F. Casson, "Multi-objective Bayesian optimization for design of Pareto-optimal current drive profiles in STEP," *IEEE Trans. Plasma Sci.* **52**, 3904–3906 (2024).
- ²¹T. Brown, A. Cioba, and I. Bogunovic, "Sample-efficient Bayesian optimisation using known invariances," *arXiv:2410.16972* (2024).
- ²²V. Mehta, J. Barr, J. Abbate, M. D. Boyer, I. Char, W. Neiswanger, E. Kolenen, and J. Schneider, "Automated experimental design of safe rampdowns via probabilistic machine learning," *Nucl. Fusion* **64**, 046014 (2024).
- ²³I. Pusztai, I. Ekmark, H. Bergström, P. Haldestam, P. Jansson, M. Hoppe, O. Vallhagen, and T. Fülöp, "Bayesian optimization of massive material injection for disruption mitigation in tokamaks," *J. Plasma Phys.* **89**, 905890204 (2023).
- ²⁴A. E. Järvinen, T. Fülöp, E. Hirvijoki, M. Hoppe, A. Kit, J. Åström, and J. Contributors, "Bayesian approach for validation of runaway electron simulations," *J. Plasma Phys.* **88**, 905880612 (2022).
- ²⁵L. R. Humphrey, A. J. Dubas, L. C. Fletcher, and A. Davis, "Machine learning techniques for sequential learning engineering design optimisation," *Plasma Phys. Controlled Fusion* **66**, 025002 (2024).
- ²⁶T. Nunn, V. Gopakumar, and S. Kahn, "Shaping of magnetic field coils in fusion reactors using Bayesian optimisation," *arXiv:2310.01455* (2023).
- ²⁷J. A. Romero and J. Svensson, "Optimization of out-vessel magnetic diagnostics for plasma boundary reconstruction in tokamaks," *Nucl. Fusion* **53**, 033009 (2013).
- ²⁸I. G. Stewart, R. S. Granetz, C. E. Myers, C. Paz-Soldan, R. Sweeney, C. J. Hansen, D. T. Garnier, D. J. Battaglia, A. J. Creely, and M. L. Reinke, "Optimization of the equilibrium magnetic sensor set for the SPARC tokamak," *Nucl. Fusion* **63**, 126014 (2023).
- ²⁹R. Jorge, A. Giuliani, and J. Loizu, "Simplified and flexible coils for stellarators using single-stage optimization," *Phys. Plasmas* **31**, 112501 (2024).
- ³⁰A. Giuliani, "Direct stellarator coil design using global optimization: Application to a comprehensive exploration of quasi-axisymmetric devices," *J. Plasma Phys.* **90**, 905900303 (2024).
- ³¹A. A. Kaptanoglu, M. Landreman, and M. C. Zarnstorff, "Optimization of passive superconductors for shaping stellarator magnetic fields," *arXiv:2501.12468* (2025).
- ³²R. Garnett, *Bayesian Optimization* (Cambridge University Press, 2023).
- ³³I. Sobol, "The distribution of points in a cube and the accurate evaluation of integrals (in Russian)," *Zh. Vychisl. Mat. Mater. Phys.* **7**, 784–802 (1967).
- ³⁴W. Hornsby, A. Gray, J. Buchanan, B. Patel, D. Kennedy, F. Casson, C. Roach, M. Lykkegaard, H. Nguyen, N. Papadimas *et al.*, "Gaussian process regression models for the properties of micro-tearing modes in spherical tokamaks," *Phys. Plasmas* **31**, 012303 (2024).
- ³⁵H. Ishibashi, M. Karasuyama, I. Takeuchi, and H. Hino, "A stopping criterion for Bayesian optimization by the gap of expected minimum simple regrets," in *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, edited by F. Ruiz, J. Dy, and J.-W. van de Meent (PMLR, 2023), Vol. 206, pp. 6463–6497.

- ³⁶R. K. Pandit and D. Infield, "Comparative analysis of Gaussian process power curve models based on different stationary covariance functions for the purpose of improving model accuracy," *Renewable Energy* **140**, 190–202 (2019).
- ³⁷C. K. Williams and D. Barber, "Bayesian classification with gaussian processes," *IEEE Trans. Pattern Anal. Mach. Intell.* **20**, 1342–1351 (1998).
- ³⁸C. K. Williams and C. E. Rasmussen, *Gaussian Processes for Machine Learning* (MIT Press Cambridge, MA, 2006), Vol. 2.
- ³⁹D. Tran, R. Ranganath, and D. Blei, "The variational gaussian process," *arXiv:1511.06499* (2015).
- ⁴⁰T. R. Hoens and N. V. Chawla, "Imbalanced datasets: From sampling to classifiers," in *Imbalanced Learning* (John Wiley & Sons, Ltd., 2013), Chap. 3, pp. 43–59, <https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781118646106.ch3>.
- ⁴¹S. Daulton, M. Balandat, and E. Bakshy, "Differentiable expected hypervolume improvement for parallel multi-objective Bayesian optimization," *Adv. Neural Inf. Process. Syst.* **33**, 9851–9864 (2020).
- ⁴²K. Yang, M. Emmerich, A. Deutz, and T. Bäck, "Multi-objective Bayesian global optimization using expected hypervolume improvement gradient," *Swarm Evol. Comput.* **44**, 945–956 (2019).
- ⁴³J. R. Gardner, M. J. Kusner, Z. E. Xu, K. Q. Weinberger, and J. P. Cunningham, "Bayesian optimization with inequality constraints," in *International Conference on Machine Learning, ICML'14 (JMLR, 2014)*, pp. 937–945.
- ⁴⁴M. Abdolshah, A. Shilton, S. Rana, S. Gupta, and S. Venkatesh, "Expected hypervolume improvement with constraints," in *2018 24th International Conference on Pattern Recognition (ICPR)* (IEEE, 2018), pp. 3238–3243.
- ⁴⁵R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM J. Sci. Comput.* **16**, 1190–1208 (1995).
- ⁴⁶E. Tholurus, F. Casson, S. Marsden, T. Wilson, D. Brunetti, P. Fox, S. Freethy, T. Hender, S. Henderson, A. Hudoba, K. Kirov, F. Koechl, H. Meyer, S. Muldrew, C. Olde, B. Patel, C. Roach, S. Saarelma, G. Xia, and the STEP Team, "Flat-top plasma operational space of the STEP power plant," *Nucl. Fusion* **64**, 106030 (2024).
- ⁴⁷S. J. Doyle, D. Lopez-Aires, A. Mancini, M. Agredano-Torres, J. Garcia-Sanchez, J. Segado-Fernández, J. Ayllon-Guerola, M. Garcia-Muñoz, E. Viezzer, C. Soria-Hoyo *et al.*, "Magnetic equilibrium design for the SMART tokamak," *Fusion Eng. Des.* **171**, 112706 (2021).
- ⁴⁸R. Albanese, A. G. Chiariello, L. E. Di Grazia, A. Iaiunese, R. Martone, M. Mattei, F. Villone, and P. Zumbolo, "Three-dimensional evaluation of the connection lengths in a tokamak," *Fusion Eng. Des.* **192**, 113622 (2023).
- ⁴⁹L. Kos, R. Pitts, G. Simić, M. Brank, H. Anand, and W. Arter, "SMITER: A field-line tracing environment for ITER," *Fusion Eng. Des.* **146**, 1796–1800 (2019).
- ⁵⁰O. Bardsley, J. Baker, and C. Vincent, "Decoupled magnetic control of spherical tokamak divertors via vacuum harmonic constraints," *Plasma Phys. Controlled Fusion* **66**, 055006 (2024).
- ⁵¹M. Coatanea-Gouachet, D. Carrillo, S. Lee, and F. Rodríguez-Mateos, "Electromagnetic quench detection in ITER superconducting magnet systems," *IEEE Trans. Appl. Supercond.* **25**, 1–7 (2015).
- ⁵²B. Dudson, "FreeGS: Free-boundary Grad-Shafranov solver (0.8.0)," 2023.
- ⁵³X. Song, B. Leard, Z. Wang, S. T. Paruchuri, T. Rafiq, and E. Schuster, "Control-oriented free-boundary equilibrium solver for tokamaks," *Plasma* **7**, 842–857 (2024).
- ⁵⁴K. Pentland, N. C. Amorisco, O. El-Zobaidi, S. Etches, A. Agnello, G. K. Holt, A. Ross, C. Vincent, J. Buchanan, S. Pamela, G. McArdle, L. Kogan, and G. Cunningham, "Validation of the static forward Grad-Shafranov equilibrium solvers in FreeGSNKE and Fiesta using EFIT++ reconstructions from MAST-U," *Phys. Scr.* **100**, 025608 (2025).
- ⁵⁵There is now an extension to FreeGS—FreeGSNKE⁶⁸—which ensures the core remains within the limiter. Use of this equilibrium solver would remove the need for this constraint, however, it was released following the completion of this work.
- ⁵⁶V. Picheny, J. Berkeley, H. B. Moss, H. Stojic, U. Granta, S. W. Ober, A. Artemev, K. Ghani, A. Goodall, A. Paleyes, S. Vakili, S. Pascual-Diaz, S. Markou, J. Qing, N. R. B. S. Loka, and I. Couckuyt, "Trieste: Efficiently exploring the depths of black-box functions with TensorFlow," *arXiv:2302.08436* (2023).
- ⁵⁷J. Berkeley, H. B. Moss, A. Artemev, S. Pascual-Diaz, U. Granta, H. Stojic, I. Couckuyt, J. Qing, N. Loka, A. Paleyes, S. W. Ober, A. Goodall, K. Ghani, and V. Picheny, "Trieste (4.2.2)," 2024.
- ⁵⁸A. G. D. G. Matthews, M. van der Wilk, T. Nickson, K. Fujii, A. Boukouvalas, P. León-Villagrà, Z. Ghahramani, and J. Hensman, "GPflow: A Gaussian process library using TensorFlow," *J. Mach. Learn. Res.* **18**, 1–6 (2017).
- ⁵⁹F. Biscani and D. Izzo, "A parallel global multiobjective framework for optimization: Pagmo," *J. Open Source Software* **5**, 2338 (2020).
- ⁶⁰A. Lahiff, A. Abraham, V. Gopakumar, K. Zarebski, M. Field, and J. Panayis, Simvue (1.1.4), 2024.
- ⁶¹K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.* **6**, 182–197 (2002).
- ⁶²S. Verma, M. Pant, and V. Snasel, "A comprehensive review on NSGA-II for multi-objective combinatorial optimization problems," *IEEE Access* **9**, 57757–57791 (2021).
- ⁶³A. F. Kuri-Morales and J. Gutiérrez-García, "Penalty function methods for constrained optimization with genetic algorithms: A statistical analysis," in *Mexican international conference on artificial intelligence* (Springer, 2002), pp. 108–117.
- ⁶⁴Q. Long, "A constraint handling technique for constrained multi-objective genetic algorithm," *Swarm Evol. Comput.* **15**, 66–79 (2014).
- ⁶⁵M. Malu, G. Dasarathy, and A. Spanias, "Bayesian optimization in high-dimensional spaces: A brief survey," in *2021 12th International Conference on Information, Intelligence, Systems & Applications (IISA)* (IEEE, 2021), pp. 1–8.
- ⁶⁶P. G. Constantine, E. Dow, and Q. Wang, "Active subspace methods in theory and practice: Applications to kriging surfaces," *SIAM J. Sci. Comput.* **36**, A1500–A1524 (2014).
- ⁶⁷A. Grosnit, R. Tutunov, A. M. Maraval, R.-R. Griffiths, A. I. Cowen-Rivers, L. Yang, L. Zhu, W. Lyu, Z. Chen, J. Wang, J. Peters, and H. Bou-Ammar, "High-dimensional Bayesian optimisation with variational autoencoders and deep metric learning," *arXiv:2106.03609* (2021).
- ⁶⁸N. C. Amorisco, A. Agnello, G. Holt, M. Mars, J. Buchanan, and S. Pamela, "FreeGSNKE: A python-based dynamic free-boundary toroidal plasma equilibrium solver," *Phys. Plasmas* **31**, 042517 (2024).