

GTXXXXX

AN AUTOMATED GAUSSIAN PROCESS INTEGRATED WITH A BAYESIAN OPTIMIZATION APPROACH TO DESIGN PIN-FINS

Susheel Dharmadhikari, Reid A. Berdanier, Karen A. Thole, and Amrita Basak*

Department of Mechanical Engineering, Pennsylvania State University, University Park, PA, USA
Email: aub1526@psu.edu

ABSTRACT

Pin fins are imperative in the cooling of turbine blades. The designs of pin fins, therefore, have seen significant research in the past. With the developments in metal additive manufacturing, novel design approaches towards complex geometries are now feasible. To that end, this article presents a Bayesian optimization approach for designing inline pin-fin arrays that can achieve low pressure loss. The pin-fin shape is defined using featurized (parametrized) piecewise cubic splines in 2D. The complexity of the shape is dependent on the number of splines used for the analysis. From a method development perspective, the study is performed using three splines. Owing to this piece-wise modeling, a unique pin fin design is defined using five features. After specifying the design, a computational fluid dynamics-based model is developed that computes the pressure drop during the flow. Bayesian optimization is carried out on a Gaussian processes-based surrogate model to obtain an optimal combination of pin-fin features to minimize the pressure drop. The results showed that the optimization tends to approach an aerodynamic design leading to low pressure drop corroborating with the existing knowledge. Furthermore, multiple iterations of optimizations were conducted with varying degree of input data. The results revealed that a convergence to similar optimal design is achieved with a minimum of 25 initial datapoints for the surrogate. A sensitivity analysis showed that the distance between the rows of the pin fins is the most dominant feature influencing the pressure drop.

NOMENCLATURE

Design and Optimization:

- 2D Two Dimensional
- BO Bayesian Optimization
- ΔP Pressure drop
- D_x Projection of the pin fin on the x-axis
- D_y Projection of the pin fin on the y-axis
- $f(\theta), g(\theta)$ Functions of θ
- GP Gaussian Process
- K GP-based covariance/variance of a design Ω
- \underline{K} Covariance matrix of n designs
- K^{n+1} Covariance of the $n + 1^{th}$ design
- m GP-based mean of a design Ω
- \underline{m} A vector of n means
- m^{n+1} Mean of the $n + 1^{th}$ design
- Ω A pin fin design vector
- Ω^* Design Space
- $\underline{\Omega}$ A collection on n pin fin designs
- ϕ Objective function
- $\underline{\phi}$ A vector of n objective functions
- $r_i \quad \forall \quad i = 1, 2, 3, i^{th}$ radial distance of a pin fin shape
- S Distance between the rows of two fin
- θ^* Orientation of the pin fin from the x-axis
- x x-direction
- X Distance between the columns of two fins
- y y-direction

CFD Model and Equations:

- δ_{ij} Kronecker delta
- E Energy per unit mass

* Address all correspondence to this author.

\bar{f}_i External force in the i^{th} direction
 k_{eff} Effective conductivity
 μ Viscosity
 P_{gauge} Gauge Pressure
 \bar{p} Mean pressure
 ρ Density of the fluid
 T Temperature
 τ_{eff} Effective shear stress
 T_{fin} Temperature of pin fins
 \bar{u}_i Flow velocity in the i^{th} direction
 u'_i Fluctuating component of velocity
 \mathbf{V} Velocity vector
 v_{in} Inlet velocity

1 INTRODUCTION

The hot-section components, e.g., turbine rotor blades of gas turbines operate at upwards of 1500 K creating a harsh environment. For this reason, innovative technologies are used to facilitate cooling of these components. A rotor blade typically has a complex cooling configuration. The blade can be divided into three primary sections, each of which has a dedicated coolant supply plenum. These sections include: (i) the leading edge region, (ii) the mid-chord region, and (iii) the trailing edge region. The pressure side of a turbine blade has a relatively higher temperature than the suction side and is, therefore, provided with additional cooling. This is accomplished by internal and external cooling that favors the pressure side of the blade, particularly in the trailing edge region [1].

Internal cooling of the trailing edge has been investigated extensively. Heat transfer enhancement designs have been developed to fully utilize the cooling capacity of the coolant. Unique to this blade region is the pressure drop between the internal plenum and the external mainstream conditions. This region allows for greater heat transfer enhancement at the cost of increased frictional loss. Fully-bridged pin-fin arrays result in large pressure losses but also excel at heat transfer enhancement. As a result, pin-fins represent an ideal candidate for trailing edge cooling. As an additional benefit, the fully-bridged pin-fin design also increases the structural rigidity of the blade. For these reasons, a vast amount of literature supports the implementation of impingement arrays to enhance heat transfer and structurally support the trailing edge section of airfoils [2].

One of the primary objectives of the pin-fin design is, therefore, to decrease the pressure drop while increasing the heat transfer. Both experimental and computational investigations have been carried out in the past to achieve this objective. Otto et al. [3] performed a particle image velocimetry study to understand the developing flow characteristics of a staggered pin-fin array. Horseshoe type vortices and Karman instabilities were identified as the key contributors to turbulent mixing. Chyu et al. [4] investigated the dependence of pin-fin cross-sections on

the thermal performance. Square, circular, and diamond-shaped pins were studied and circular pin-fins were found to have the best trade-off between pressure drop and heat transfer.

With the advancements in metal additive manufacturing, the feasibility of making complex pin fin designs that are not limited to the traditional manufacturing constraints has increased multi-fold. This is evident from the recent research efforts towards the testing of such unique designs [5], and their corresponding parameters [6]. The introduction of this manufacturing technology has led to the initiation of research towards more innovative design strategies, particularly with the use of data-driven tools [7]. These strategies are built upon the parametrization of the pin fin designs followed by an optimization that leads to the desired pin fin shape.

Accordingly, there are several parameters pertaining to the pin shape that could be optimized to minimize pressure drop and maximize heat transfer. Existing literature has shown experimental results with unique geometries for pins such as a star or a dimpled sphere [5]. However, experimental optimization of pin-fins is expensive and time-consuming. For this reason, computational investigations have played a crucial role in the development of novel designs and creative solutions. Eyi et al. [8,9] used parameterized Bezier curves to define and, then, optimize the leading edge of a fin in a parametric form. Wileke et al. [10] used adjoint optimization for a U-shaped channel to reduce the total pressure loss. On a similar note, Ghosh et al. and Dilgen et al. implemented a topology optimization technique to explore manufacturing constraints [11, 12]. Fabbri [13] applied a genetic algorithm to optimize pin fin designs. Hamadneh et al. used particle swarm optimization (PSO) to evaluate several pin fin geometries for enhanced thermal performance [14].

Recently, Ghosh et al. used Gaussian process (GP) surrogates with constrained Bayesian Optimization (BO) for optimizing the thermal performance of the pin-fin arrays [7]. Due to the black-box nature of CFD simulations for complex geometries, the use of GP and BO has shown promising results, particularly while working with limited data. However, due to the relatively nascent percolation of such techniques for pin fin optimization, more complex formulations in the design space are not yet fully explored. In addition to that, such studies have been hampered by the lack of automated simulations using established computational tools such as ANSYS Fluent, thereby, restricting the optimization to a few manually conducted iterations. This paper addresses these existing shortcomings by applying a novel spline-based pin-fin definition that offers a potential to explore complex geometries. In addition to that, the simulations conducted in this analysis are completely automated leading to a relatively high number of design iterations.

The optimization process in the present study was performed to minimize the pressure drop. The results revealed that the framework can learn the design principles with limited training data and converged to a desired solution with < 50 iterations.

A study on the data requirement of the algorithm was also conducted to quantify the need of initial data for the algorithm to perform adequately. Furthermore, a sensitivity analysis was presented to understand the impact of the features on the pressure drop. Although the analysis presented in the article was studied for minimizing pressure drop, it can be easily modified to address other desired objectives.

2 METHODOLOGY

2.1 FEATURIZATION OF PIN FINS

Features of the general pin fin shape

A general closed shape in 2D is a locus of points, $x = f(\theta)$, and $y = g(\theta)$, parameterized over θ . The most common form of this parametrization is visualized with a circle of radius r , an outcome of $x = r \cos \theta$ and $y = r \sin \theta$. By employing more complex functional representations in the construction of $f(\theta)$ and $g(\theta)$, a range of variations in the shapes can be generated. Among several such strategies, this paper uses piecewise-cubic splines to generate parametric shapes in 2D. The complexity of these shapes, owing to their construction, further depends on the number of splines used in the process.

With this in mind, the analysis in this paper was performed by using three splines. As an example, Fig. 1(a) depicts the two curves ($x = f(\theta)$ and $y = g(\theta)$), both constructed using three cubic splines. The resultant shape indicating the contribution of the individual splines is shown in Fig. 1(b). The shape indicates three radial distances (r_1 , r_2 , and r_3) that provide the necessary coordinates for spline interpolation.

To be able to optimize this shape, features that impact the geometry need to be chosen. Three such features, viz. r_2 , r_3 , and θ^* were identified to be the defining elements for any shape generated using the aforementioned procedure. The feature, r_1 , was maintained at a constant magnitude of 1 mm to ensure a reference dimension to prevent the optimization algorithm from choosing extreme (either too small or too large) geometries. The angle of r_1 from the X-axis is denoted by θ^* , as shown in Fig. 1(a) and (b). It controls the orientation of the pin fin. Fig. 1(b) also depicts D_x and D_y which denote the projection length of the pin on X and Y axis, respectively.

Features of Pin-Fin Arrays

In addition to the three features of a single fin, the setup of the array is controlled with two additional features that account for the distance between the rows and columns of the fins. In literature, the distance between the rows is denoted by S , whereas, for columns, it is denoted by X as shown in Fig. 1(c). Accordingly, S/D and X/D are the two ratios that are commonly discussed in pin fin literature where D is the projected length of the fin perpendicular to the flow. In this formulation, a minor variation of these ratios is used to avoid numerical inconsistencies in

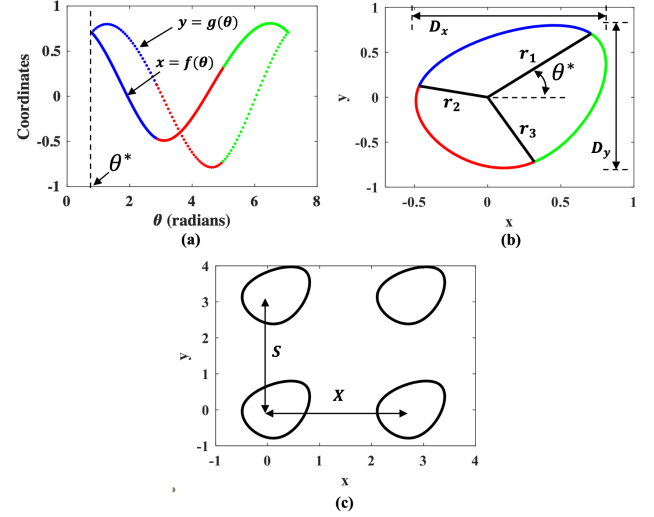


FIGURE 1. (a) Two curves ($x = f(\theta)$ and $y = g(\theta)$) constructed with three piece-wise cubic splines. (b) The pin-fin shape resulting from three splines with the defining parameters. (c) A pin fin array constructed with two rows and two columns.

design. Instead of using S/D and X/D , the authors define S/D_y and X/D_x as the two additional features based on their projection lengths from Fig. 1(b). The inclusion of this modification avoids intersection of two fin shapes on the grid, which was observed to be happening in cases where the optimization converged towards fins with a high D_x/D_y ratio.

Therefore, in total, the design (Ω) of a pin-fin array can be uniquely defined by five features, viz. r_2 , r_3 , θ^* , S/D_y , and X/D_x :

$$\Omega = [r_2, r_3, \theta^*, S/D_y, X/D_x] \quad (1)$$

Design Space

To proceed with the optimization, some constraints on the range of variation for all the five features is decided. Because a reference radial distance (r_1) was maintained at 1 mm, the other two radii r_2 , and r_3 were varied from a lower bound of 0.1 mm to an upper bound of 1 mm. The lower bound was chosen to avoid numerical complications in spline computation which were observed for radii approaching zero. The impact of variation of these parameters on the fin shape is shown in Fig. 2.

As depicted in Fig. 2, the orientation parameter (θ^*) was varied from 0 to π radians. Due to the nature of the flow, any pin fin design with θ^* has the same characteristics as $2\pi - \theta^*$. Therefore, the range 0 to π ensured all distinct orientations were taken into account. Every single variation between these three parameters θ^* , r_2 , and r_3 lead to a unique shape and thereby provided a multitude of design combinations. The variation in

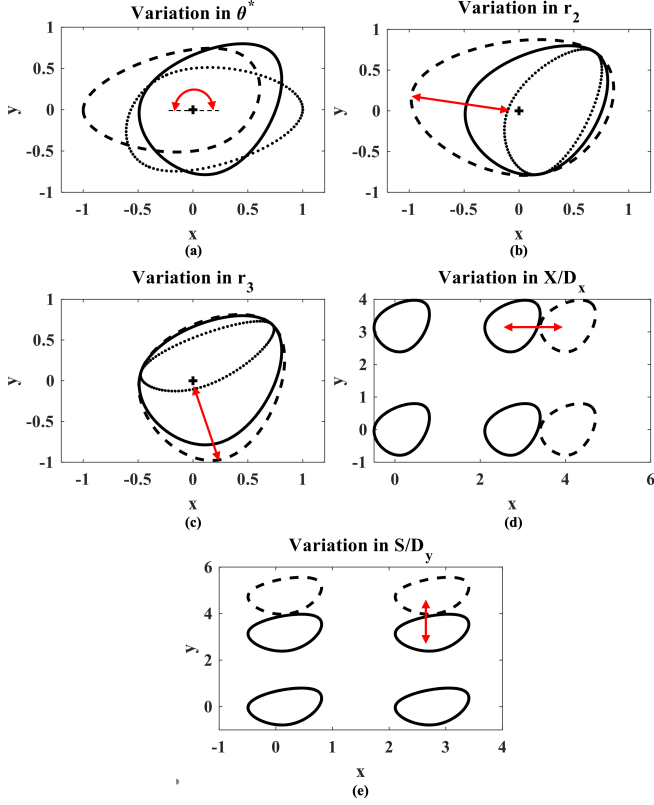


FIGURE 2. Feature space variation for (a) θ^* , (b) r_2 , (c) r_3 , (d) X/D_x , and (e) S/D_y . The (red) arrows indicate the extent and the direction of variation of the parameter.

array parameters was relatively straightforward. Based on literature [5], the range of these parameters was chosen to vary from 2 to 3 and its impact on the array is shown in Fig. 2(d) and (e). The design space is represented through a vector Ω^* as follows:

$$\Omega^* = [r_2 \in [0.1, 1], r_3 \in [0.1, 1], \theta^* \in [0, \pi], S/D_y \in [2, 3], X/D_x \in [2, 3]] \quad (2)$$

2.2 Development of the CFD Model

Simulation Domain and Boundary Conditions

The geometry of the 2D pin-fin array and the corresponding flow domain is shown in Fig. 3. The inlet passage of the domain was extended to $10D_x$ upstream to allow the flow to fully develop before interacting with the fins. An inlet velocity (v_{in}) of 100 m/s was maintained with a temperature of 300 K. The fins were

modelled as a stationary wall with a no slip condition and are maintained at a constant temperature (T_{fin}) of 350 K. The domain outlet was defined at atmospheric pressure ($P_{gauge} = 0$), with a temperature of 300 K, and the outlet passage was also extended to $5D_x$ to capture the flow behavior in the wake region. The top and bottom regions have a symmetry boundary condition. This ensures that any variation in the design space (for ex. due to S/D_y) would not have any impact on the simulations. The region of interest encapsulating the pins, shown in the inset in Fig. 3, will be shown in the results going further.

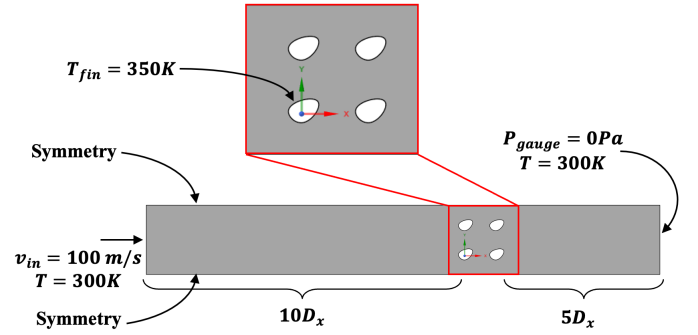


FIGURE 3. Simulation domain and boundary conditions implemented in the CFD model.

Flow-Thermal Governing Equations

The dynamics of the fluid flow is governed by the conservation of mass and the Cauchy momentum equation. In addition, the following assumptions about the physics of the fluid flow are made:

1. The fluid (i.e., air) is considered incompressible (with the constant density), thus the mass conservation translates into the volume conservation.
2. The material properties of air are modeled with a linear constitutive behavior, i.e. a Newtonian fluid, where the internal shear stress is proportional to the shear rate.
3. The fluid adheres to the surfaces of the fins and the walls; there is a no slip condition.
4. The turbulence is specified using a 5% turbulent intensity and a viscosity ratio of 10.
5. In specifying the wall properties for the pin fin, a standard roughness model with a roughness constant of 0.5.

Using the above assumptions, the time-averaged Reynolds-averaged Navier–Stokes (RANS) equations are used to describe the flow through pin fin arrays. The equations in Einstein notation for an incompressible Newtonian fluid and a stationary flow are written as:

$$\rho \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = \rho \bar{f}_i + \frac{\partial}{\partial x_j} \left[-\bar{p} \delta_{ij} + \mu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \rho \overline{u'_i u'_j} \right] \quad (3)$$

Here, ρ is the density of the fluid, \bar{u}_j is the mean velocity in j^{th} direction, and $\frac{\partial \bar{u}_i}{\partial x_j}$ is the velocity gradient with respect to the j^{th} direction. Therefore, $\rho \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j}$ is the change in the mean momentum of a fluid element owing to the convection in the mean flow. This change is balanced by the mean body force $\rho \bar{f}_i$, the isotropic stress due to the mean pressure field $-\bar{p} \delta_{ij}$, the viscous stresses $\mu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right)$, and apparent stress $\rho \overline{u'_i u'_j}$ owing to the fluctuating velocity field, generally referred to as the Reynolds stress. Here f_i represents the external force, \bar{p} is the average pressure, \bar{u}'_i denotes the mean of the fluctuating component of the velocity. For modelling turbulence, the SST $k - \omega$ formulation is used [15].

One of the future objectives of this research is to perform optimization studies that simultaneously minimize pressure drop and maximize heat transfer in a pin-fin array. Hence, in the present research the energy conservation equations are also solved. The transport of thermal energy is computed using the following equation:

$$\frac{\partial(\rho E)}{\partial t} + \nabla \cdot [\mathbf{V}(\rho E + p)] = \nabla \cdot [k_{eff} \nabla T + \boldsymbol{\tau}_{eff} \cdot \mathbf{V}] \quad (4)$$

Here, E is the energy per unit mass, \mathbf{V} is the velocity vector, p is the pressure, k_{eff} is the effective thermal conductivity, and $\boldsymbol{\tau}_{eff}$ is the effective shear stress.

Model Implementation

The fluid flow and thermal evolution are simulated with the software ANSYS® Fluent 2020 R2, which is based on the finite-volume method. The governing equations are integrated over a finite set of quadrilateral control volumes that meshes the simulation domain. Details of the mesh are represented in Fig. 4. Following a cell-centered discretization, the numerical solver computes discrete values of the continuous velocity and pressure fields at the center of the control volumes. The values at any other locations are interpolated from the discrete values, whenever required. The numerical scheme evaluates the advection and diffusion fluxes of momentum through all the faces of the control volumes. Then, the accumulated quantity of momentum inside each control volume is updated according to its net fluxes. As the fluid is incompressible, the pressure field is a result of the continuity constraint. A pressure equation is derived from the law of mass conservation.

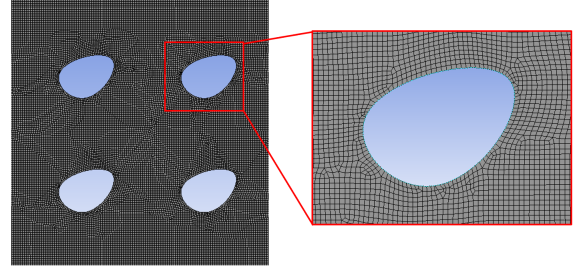


FIGURE 4. Mesh distribution near the fin boundaries.

The evolution of the system was solved incrementally with an automatic time stepping method using pseudo transient settings. The simulation was run for 200 iterations which was observed to be a sufficient duration for model convergence. In the simulations, the maximum element size was maintained at 0.05 mm. The convergence conditions are set at $1e - 6$ for all residuals. The discretized momentum equations and pressure equations relative to the set of control volumes were solved with an implicit solver that ensured stability of the numerical scheme. At each incremental time step, the numerical algorithm computes the new values of the primary discrete variables that are the local velocities, the pressure and temperature. Secondary results, such as the streamline of the flow, the shear rate, the viscous stress, or fluxes, are computed from the primary variables. The pressure drop (ΔP), which represents the critical objective for optimization in this study, was calculated by simply recording the inlet pressure at the end of the simulation (since the outlet is maintained at $p_{gauge} = 0$ Pa).

2.3 Optimization Framework

Gaussian Process-Based Surrogate

The surrogate development strategy is based on a class of stochastic processes called Gaussian Processes (GPs) that assume any finite collection of random variables to follow a multivariate jointly Gaussian distribution. For a finite collection of n designs, $\underline{\Omega}$, the corresponding function outputs, $\underline{\phi}$ are assumed to have a multivariate jointly Gaussian distribution,

$$\underline{\phi} \sim \mathcal{N}(\underline{m}(\underline{\Omega}), \underline{K}(\underline{\Omega}, \underline{\Omega}')) \quad (5)$$

Here, \mathcal{N} implies a Gaussian distribution. The underlying GP is completely characterized by a mean function: $\underline{m}(\underline{\Omega}) = E[\underline{\phi}]$, and a covariance function: $\underline{K}(\underline{\Omega}, \underline{\Omega}') = E[(\underline{\phi} - \underline{m}(\underline{\Omega}))(\underline{\phi}' - \underline{m}(\underline{\Omega}'))]$ [16]. Here, $E[\star]$ denotes the expectation of \star . $\underline{\Omega}'$ and $\underline{\phi}'$ denote a set of finite designs other than $\underline{\Omega}$ and the corresponding functional output of it, respectively. In the context of this problem, the output ϕ corresponds to ΔP .

In order to understand the application of surrogate modeling, consider the situation where n designs denoted by the $\underline{\Omega}$ have been computationally evaluated to generate the outputs $\underline{\phi}$. Using this data, a surrogate model can be established with the multivariate Gaussian formulation. The surrogate model, can now be used to estimate the output of a new design Ω^{n+1} using the following formulation for a conditional distribution:

$$\phi^{n+1} | \underline{\phi}, \phi^{n+1}, \Omega \sim \mathcal{N}(m^{n+1}, K^{n+1}) \quad (6)$$

Here,

$$m^{n+1} = \underline{K}(\Omega^{n+1}, \underline{\Omega}) \underline{K}(\underline{\Omega}, \underline{\Omega})^{-1} \underline{\phi} \quad (7)$$

$$K^{n+1} = \underline{K}(\Omega^{n+1}, \Omega^{n+1}) - \underline{K}(\Omega^{n+1}, \underline{\Omega}) \underline{K}(\underline{\Omega}, \underline{\Omega})^{-1} \underline{K}(\underline{\Omega}, \Omega^{n+1}) \quad (8)$$

Here, \underline{K} is the covariance matrix. Thus, the predicted posterior distribution of the outputs at every test data point is also a Gaussian distribution, characterized by the mean, m^{n+1} and covariance, K^{n+1} . A detailed mathematical account of GPs can be found in [16].

Bayesian Optimization

The term *optimization* is used to denote minimization of an objective function. A maximization problem can be posed similarly by taking the negative of the objective function. In a single-fidelity setting, there is a single objective function ϕ . To minimize ϕ over its domain, the solver needs to find:

$$\hat{\Omega} = \underset{\Omega \in \Omega^*}{\operatorname{argmin}} \phi(\Omega) \quad (9)$$

Here, ‘argmin’ finds the argument that gives the minimum value from an objective function, ϕ . The functional form of ϕ is typically unknown and, hence, a gradient-free or *black-box* optimization is often utilized. BO is one such *black-box* optimization technique [17] that leverages the predictions through a surrogate for sequential *active learning* to find the global optima of the objective function. The active learning strategies find a trade-off between *exploration* and *exploitation* in possibly noisy settings [17], which facilitates a balance between the global search and local optimization through *acquisition functions*. One commonly used acquisition function in BO is Expected Improvement (*EI*).

The objective function, ϕ , expressed as a GP, yields a posterior predictive Gaussian distribution characterized by the mean

$m(\Omega)$ and standard deviation $K(\Omega)$ for $\Omega \in \Omega^*$, where Ω^* is the search space of the optimization challenge. The optimization algorithm proceeds sequentially by sampling $\hat{\Omega} = \operatorname{argmax}_{\Omega} EI(\Omega)$ at every step of the iteration process to add on to the dataset, after which the GP surrogate is retrained with the new data set to predict the acquisition potential for the next iterative step. This process continues until an optimum is reached, or the computational budget is extinguished. Since the acquisition potential is predicted over the entire search space by the surrogate, BO can achieve fast predictions without a lot of function calls in the search space (i.e., without having to run the simulations to obtain the objectives at all the search locations). Without the benefits of BO, optimizations might be computationally infeasible when the search space is high-dimensional and the simulations are expensive.

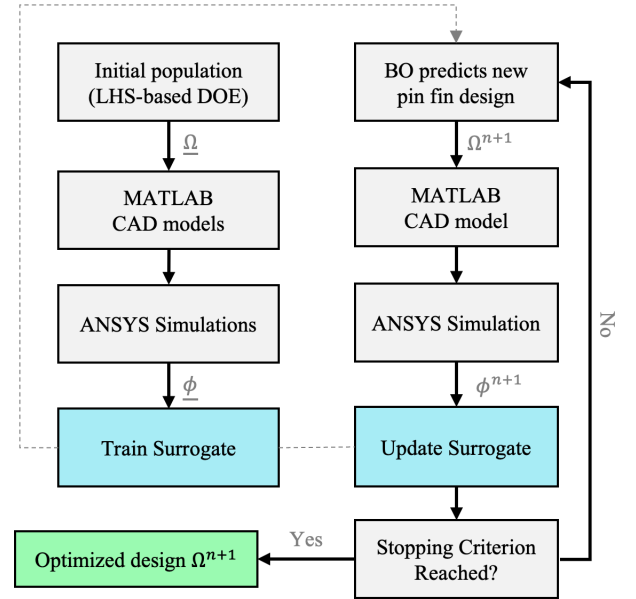


FIGURE 5. Flow chart of the optimization framework.

2.4 Implementation of Iterative and Automated GP and BO

The implementation of the proposed optimization framework is hinged on training and updating a surrogate model. Fig. 5 depicts the workflow of the algorithm during the training and updating phase. In the training phase, a surrogate model is trained using an initial design population that is generated through a Latin hypercube sampling (LHS)-based design of experiment (DOE). For every design in the population a CAD model is generated in MATLAB, followed by an ANSYS simulation. Based on these outputs, a GP-based surrogate is trained.

This surrogate forms the basis of the BO framework that again consists of three main computation aspects that need to be operated in sync iteratively. These aspects are (i) numerical pin fin shape generation and translation to a CAD geometry, (ii) CFD model setup and simulation, and (iii) Iterative BO using steps (i)-(ii). To achieve these steps, multiple softwares are operated through a master script in Python. The pin fin shapes are generated in MATLAB using the spline toolbox. The shape is then converted to an AutoCad file (.dxf) from MATLAB using the open source library `DXFLib`. The geometry is imported to ANSYS and the simulation is setup using pre-recorded journals. The output of the simulations is read from the simulations backup files by the python masterscript and is used to update the Bayesian model until convergence or maximum specified iterations. The BO algorithm is employed using the `GPyOpt` toolbox. The entire framework is run without any manual intervention.

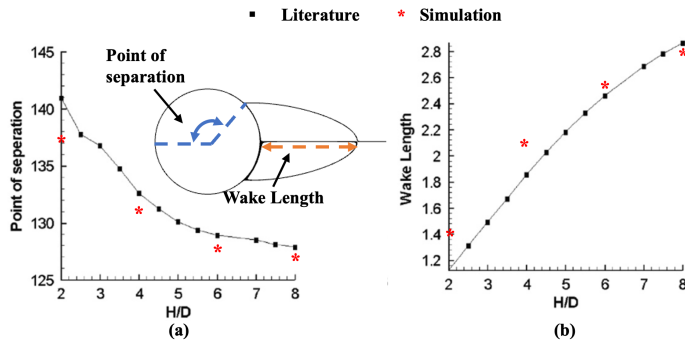


FIGURE 6. Comparison of the (a) point of separation and (b) wake length with published literature [18]

3 RESULTS AND DISCUSSION

3.1 Validation of the CFD Model

Grid convergence study is a necessary test in CFD simulations. In this study, the variation in the performance parameters e.g., ΔP is studied by altering the element size from 0.5 mm to 0.03 mm. This leads to a variation of 3471 to 0.8 million nodes. All parameters, except the element size, are kept identical for all simulations. The variation in pressure drop is not significant (2%) beyond 0.05 mm element size. Therefore, the element size of 0.05 mm is chosen for all analyses. To verify the setup of the Fluent module, a study is performed to compare the wake length and the point of separation to similar published research [18]. The comparison is shown in Fig. 6 indicating that the Fluent module is adequately set.

3.2 Performance of the Gaussian Process (GP) Surrogate

Before moving to optimization, a Latin hypercube sampling (LHS)-based design of experiment (DOE) is conducted for 100

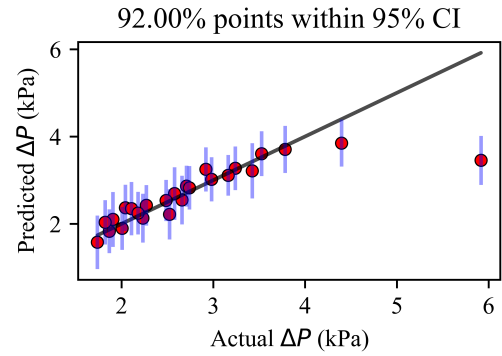


FIGURE 7. Comparison of ΔP between the prediction of the GP-based model against actual data.

designs to develop a surrogate model. This surrogate model forms the basis of the Bayesian optimization framework, and an efficient search of the optimal design depends on the construction of this model. Therefore, before diving into optimization, it is often advisable to check the accuracy of the surrogate using some regression metrics. In this study, to test the surrogate model, the available data is randomly split using a 75%-25% ratio into a training and testing set. The training set is used to build the surrogate and the testing set is used to assess it. The result of the predictions against the actual data is shown in Fig. 7. The model predicts 92% of the testing data within the 95% confidence interval indicating that the surrogate indeed has emulated the actual physics. The Pearson correlation (r-square), however, is low at 0.67. There is one conspicuous outlier in the data that shows a ΔP of 6 kPa. By removing that outlier, the model is capable of achieving an almost perfect accuracy. However, since the reasons for the outlier are enmeshed in the physics of the system, it is not removed during the optimization computations. It is also important for the GP to have some data for worst designs which helps in avoiding those design combinations later during optimization. The choice of 100 for the initial designs to build the surrogate is random. With optimization problems that are computationally expensive, 100 initial simulations already pose a challenge, and therefore, it is also essential to address these data requirements for the proposed algorithm to be successful. To address this issue, a discussion is followed in Section 3.4.

3.3 Performance of Bayesian Optimization (BO)

The convergence of the BO algorithm against the iterations and initial DOE information is shown in Fig. 8(a). The algorithm, that uses a surrogate built with 100 initial designs, finds an optimum within a few iterations, followed by an occasional (unsuccessful) exploitation of the design space indicated by the peaks in the convergence curve. The best design (Fig. 8(b)) with

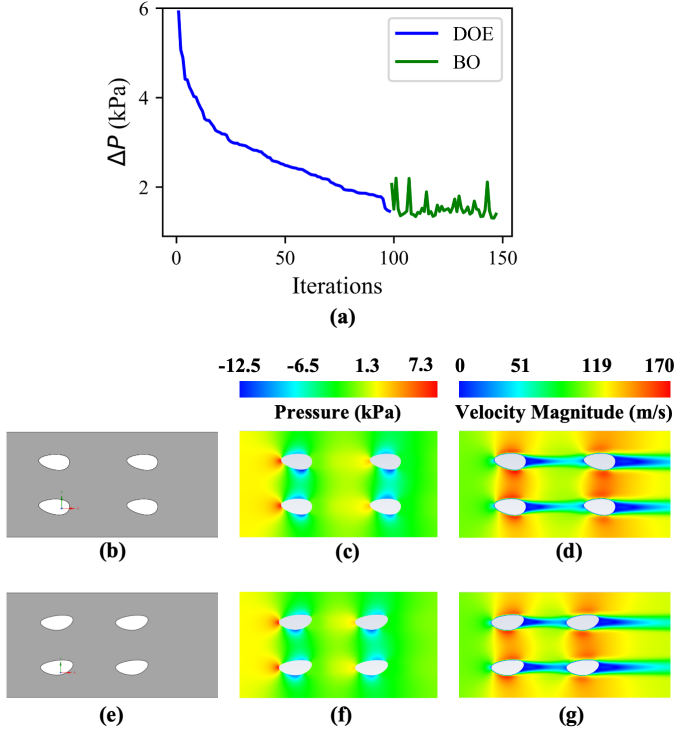


FIGURE 8. (a) Convergence of the BO algorithm. (b), (c), and (d) Best design, the corresponding pressure and velocity field, respectively, provided through the DOE. (d), (e), and (f) Optimized design, the corresponding pressures and velocity field, respectively.

the corresponding pressure and velocity fields (Figs. 8(c) and (d)) provided by the initial DOE exhibits a ΔP of 1.46 kPa. The optimization algorithm is able to reduce it further to 1.3 kPa with a design (and pressure, velocity fields) shown in Figs. 8(d), (e), and (f). A comparison between the two designs in Figs. 8(b) and (e) reveal the impact of BO on making a more aerodynamic design resulting in an improved ΔP .

3.4 Evaluation of the Optimization Algorithm with Minimal Data

The optimization result in the preceding section could reduce the pressure drop by 0.1 kPa using the information from 100 initial designs. In some practical scenarios, evaluating 100 simulations may not be feasible. Therefore, the capability of the BO algorithm to work with less information needs to be studied. In order to do that, the optimization is now carried out using four instances of less initial data by testing the functionality of the BO algorithm with 75, 50, 25, and 0 initial designs. The omission of the designs in each instance is such that the best designs are removed, thereby providing incrementally low information about the optimal solution to the BO algorithm. The best designs obtained through these simulations are shown in Fig. 9 and the

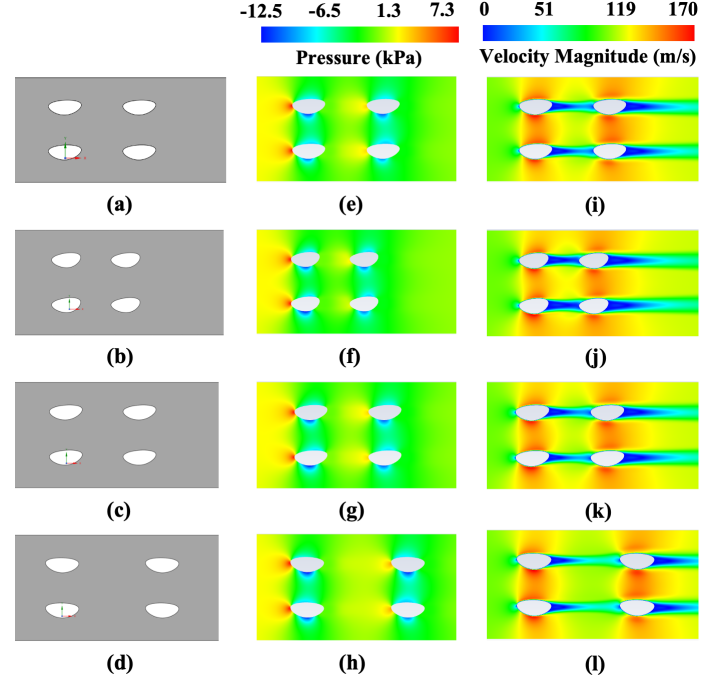


FIGURE 9. Optimized designs for cases with (a) 75, (b) 50, (c) 25, and (d) 0 initial designs prior to optimization, with their respective pressure fields in (e), (f), (g), and (h), and velocity fields in (i), (j), (k), and (l).

convergence rates are shown in Fig. 10. The features for all the optimized designs are tabulated in Table 1.

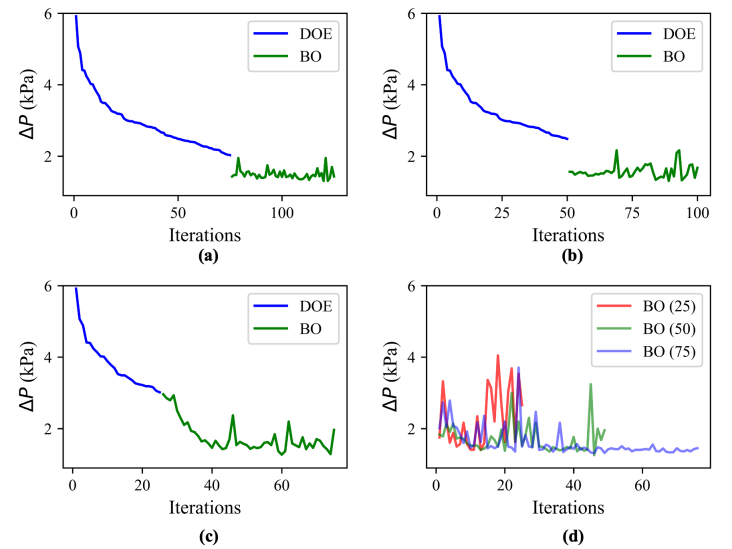


FIGURE 10. Convergence plot for BO with (a) 75, (b) 50, (c) 25, and (d) 0 initial designs.

The optimized designs (Figs. 9 (a)-(d)) tend to approach a similar shape for all the instances indicating the presence of a global optima for this particular problem. The similarity in the performance of these designs can be further compared with the pressure and velocity fields in Figs. 9(e)-(l). The subtle differences in them can be studied through the numerical values in the table. The designs, however, do not reveal the intricacies in which the algorithm approached the optima. That behavior is better exemplified by the convergence rates in Fig. 10. With 75 initial designs (Fig. 10(a)), the behavior of BO is almost similar to the previous case with all information. With 50 designs (Fig. 10(b)), more understanding of BO can be inferred. By comparing Figs. 10(a) and (b), one can notice that the first prediction of the BO algorithm is almost similar in both cases. This implies that the underlying GP learnt by the model with 75 and 50 designs is similar in its functional form.

This interpretation is further emphasized by comparing with Fig. 10(c) where BO gradually moves towards an optimal design until 40 iterations, thereby indicating that the GP needed more than 25 designs to make a better informed decision. With the final case of 0 initial designs (Fig. 10(d)), the convergence is not as steady as the previous cases. Since there is no initial data for this case, multiple instances with different limitations on maximum allowable iterations are conducted. The results for BO (25), BO (50), and BO (75) in Fig. 10(d) exemplify the random nature of convergence for these simulations. Moreover, the intermittent peaks for ΔP that correspond to the exploitation phase in optimization have larger variance than the previous cases due to the unavailability of data. The predicted optima, however, is still close to the previous cases indicating the intelligent sampling procedure of BO. However, the predictions from such optimizations have a high probability of exploring local optima and are therefore unreliable. On an average, the BO algorithm is able to improve ΔP by more than 1 kPa as compared to the best design provided by the DOEs in all the cases.

3.5 Sensitivity Analysis

From a design and manufacturing point of view, it is essential to understand the relative impact of the features on the performance. Moreover, exploring the functional forms learnt by the GP can further help in understanding the system behavior. To address these points, a global and local sensitivity analysis is now performed. The global analysis is essential to understand the impact of the features, whereas the functional forms from the GP can only be understood in a local context due to the multi-parametric nature of the problem.

A SHAP (SHapley Additive exPlanations) analysis is performed to understand the global sensitivity of the features. SHAP is a method from coalitional game theory, developed to understand the individual impact of all the features in a prediction [19]. Visually, the interpretation from this analysis can be presented in

TABLE 1. Features of optimized designs

# designs	r_2 (mm)	r_3 (mm)	θ^* (rad.)	S/D_y	X/D_x	ΔP (kPa)
100	0.1	0.74	2.8	3	2.3	1.3
75	0.1	1.0	2.6	3	2.26	1.3
50	0.17	0.75	2.8	3	2.04	1.3
25	0.1	1.0	2.7	3	2.24	1.27
0 (50)	1.0	0.1	0.49	3	3	1.25
0 (25)	0.62	1.0	2.86	3	2	1.4

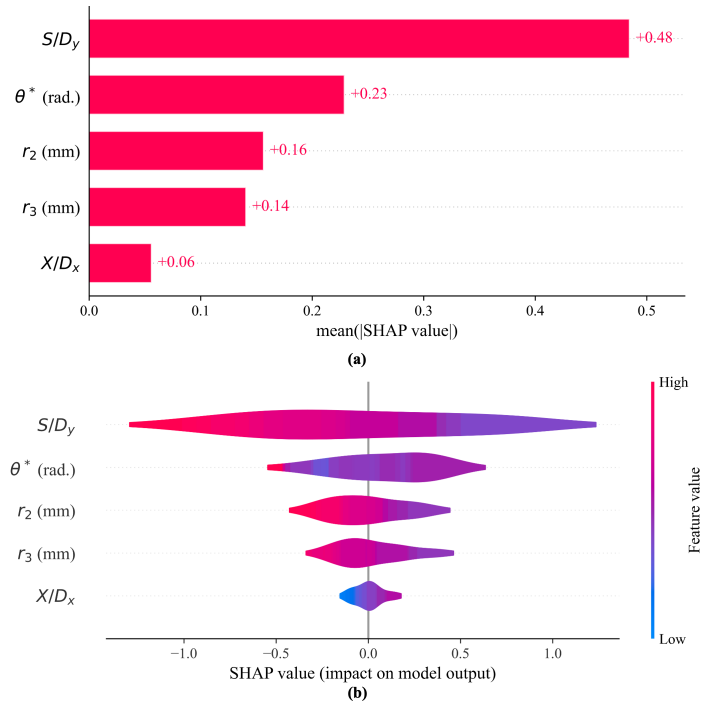


FIGURE 11. (a) A bar plot showing the relative absolute impact of the features. (b) A summary plot revealing the impact of the features on the output with respect to the changes in feature magnitudes.

two forms, viz. (i) Using a bar chart as shown in Fig. 11(a), and (ii) Using a summary plot as shown in Fig. 11(b). Both the figures reveal important information about the feature behavior.

The bar graph indicates the relative impact of the features on ΔP . The X-axis of the graph shows the mean SHAP values that denote the average contribution of the features towards ΔP . For example, for S/D_y , the mean SHAP value of 0.48 indicates that

S/D_y contributes to 0.48 kPa of the total ΔP predicted by the GP model. Fig. 11(a) shows S/D_y to be the most dominant factor influencing ΔP , whereas X/D_x has the least impact. Among r_2 , r_3 , and θ^* (the three features that create a shape), θ^* has the largest influence on ΔP .

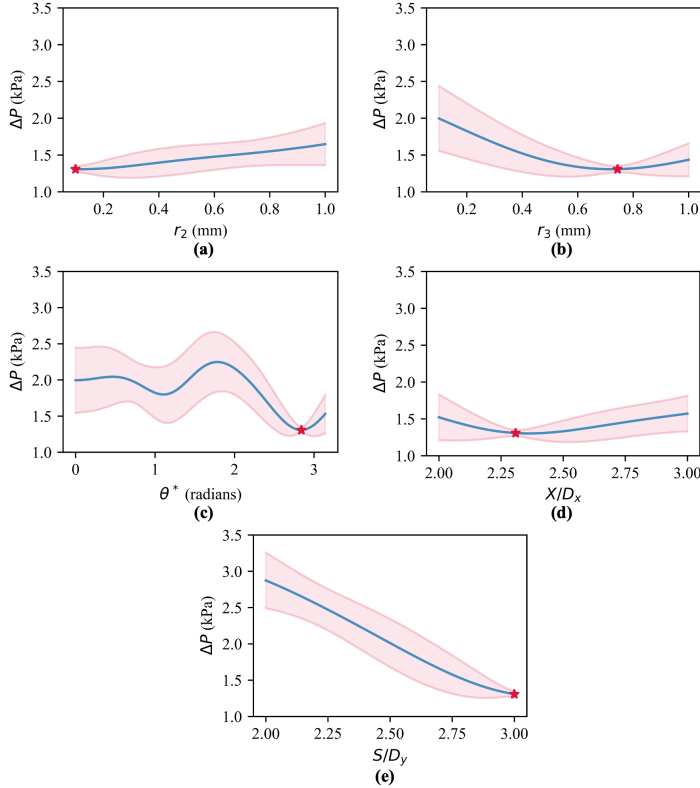


FIGURE 12. Local feature sensitivity on ΔP for (a) r_2 , (b) r_3 , (c) θ^* , (d) X/D_x , and (e) S/D_y .

Although this information is useful, it is impossible from the bars in Fig. 11(a) to interpret *how* these features impact the outcome. For example, the bar chart does not tell whether increasing or decreasing S/D_y is beneficial. This shortcoming is addressed through the summary plot in Fig. 11(b). The summary plot shows a scatter of color-coded plots across the X-axis for different features. The colors represent the relative magnitude of the features and the X-axis is the SHAP value. The length of the scatter indicates the relative influence. For example, for S/D_y , the scatter is the largest, indicating that it has the most influence on the model output. And the higher magnitudes of S/D_y (red color) are towards the left end of the spectrum indicating that a higher S/D_y would reduce ΔP . This interpretation is also aligned with the optimized features (Table 1) where all designs have converged to the maximum possible S/D_y to reduce ΔP .

To understand the local sensitivity, the behavior of the surrogate models is studied for the optimized design with 100 initial points. Fig. 12 shows the variation of each feature with ΔP as learned by the GP. To compute the variation for each feature, all other features are held constant at the optimized value indicated by BO. Therefore, the functional forms are heavily influenced by the constant feature values and the analysis is thereby termed as local. Even so, the variations are useful in understanding the impact on the optimized design. All the features, except θ^* show monotonic variation with ΔP . The periodic variation in θ^* alludes to a symmetry that may be embedded in the model. Among all the features, the relative total variation in ΔP indicates the impact of the feature on the outcome. As identified from the SHAP analysis, S/D_y again has the maximum variation in ΔP indicating its dominant impact. The star indicates the feature value in the optimized design. The sensitivity analysis therefore provides a comprehensive relationship between the objective and the features which ultimately aids in the design and manufacturing phases. The SHAP analysis provides a toolkit for varying features to satisfy the objective, i.e. setting a high value of S/D_y in this case. Once an optimal design has been chosen, the local sensitivity analysis helps in identifying the features that need to be monitored (or controlled) more strictly than others depending on their impact on the outcome.

4 CONCLUSION AND FUTURE WORK

The article presents a unique piece-wise cubic spline based framework for featurizing pin fin arrays. An optimization problem for computing the pin fin arrays with minimum pressure drop is setup using a CFD framework coupled with a surrogate-based Bayesian optimization approach. The optimized designs are observed to follow an aerodynamic shape leading to a reduction in the pressure drop. The capability of the BO framework is further tested with low initial information. The optimization is observed to efficiently find an optimum design with 25-50 initial data points. Furthermore, a sensitivity analysis is performed to reveal S/D_y to be the most dominant feature to influence the pressure drop. Knowledge of the minimum number of designs needed for optimization coupled with the sensitivity analysis provide valuable information to design engineers.

The convergence to an aerodynamic shape with piece-wise cubic splines shows promise and will be explored further to test the capabilities of the method. With higher number of splines, more complex shapes emulating some of the tested prototypes [5] can be generated. The mathematical setup of the pin fin designs also provides opportunities to include shape distortion that has been observed in additively manufactured specimens [6]. The studies with the modelling and impacts of such shape distortions will also be conducted for optimization. Geometrical constraints to compensate for these effects will make this approach more impactful and application-oriented. In addition to that, an

extension of the method to three dimensions will also be pursued in the future. An imperative part of the current approach is the symmetry condition in the CFD model which in-theory implies infinite arrays of pins and an unbounded domain. To improve the predictions further, a bounded simulation emulating the actual testing environment will be conducted after finding the optimal pin fin shape. Moreover, the current method only tackles the pressure drop minimization problem. In the future, studies will also be conducted to perform a multi-objective optimization targeted towards enhancing heat transfer while reducing pressure drop.

ACKNOWLEDGEMENT

This material is based upon work supported by the NASA Aeronautics Research Mission Directorate (ARMD) University Leadership Initiative (ULI) under cooperative agreement number 80NSSC21M0068. The authors would like to thank Ritam Pal and Nandana Menon for their help with the CFD modelling and Bayesian Optimization, respectively. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Aeronautics and Space Administration.

REFERENCES

- [1] Town, J., Straub, D., Black, J., Thole, K. A., and Shih, T. I., 2018. "State-of-the-art cooling technology for a turbine rotor blade". *Journal of Turbomachinery*, **140**(7), p. 071007.
- [2] Taslim, M., Setayeshgar, L., and Spring, S., 2001. "An experimental evaluation of advanced leading edge impingement cooling concepts". *J. Turbomach.*, **123**(1), pp. 147–153.
- [3] Otto, M., Hodges, J., Gupta, G., and Kapat, J. S., 2019. "Vortical structures in pin fin arrays for turbine cooling applications". In Turbo Expo: Power for Land, Sea, and Air, Vol. 58646, American Society of Mechanical Engineers, p. V05AT16A003.
- [4] Chyu, M., Yen, C., and Siw, S., 2007. "Comparison of heat transfer from staggered pin fin arrays with circular, cubic and diamond shaped elements". In Turbo Expo: Power for Land, Sea, and Air, Vol. 47934, pp. 991–999.
- [5] Ferster, K. K., Kirsch, K. L., and Thole, K. A., 2018. "Effects of geometry, spacing, and number of pin fins in additively manufactured microchannel pin fin arrays". *Journal of Turbomachinery*, **140**(1).
- [6] Corbett, T. M., Thole, K. A., and Bollapragada, S., 2023. "Impacts of pin fin shape and spacing on heat transfer and pressure losses". *Journal of Turbomachinery*, **145**(5), p. 051014.
- [7] Ghosh, S., Mondal, S., Kapat, J. S., and Ray, A., 2020. "Shape optimization of pin fin arrays using gaussian process surrogate models under design constraints". In Turbo Expo: Power for Land, Sea, and Air, Vol. 84164, American Society of Mechanical Engineers, p. V07AT15A021.
- [8] Eyi, S., Hanquist, K. M., and Boyd, I. D., 2019. "Aerothermodynamic design optimization of hypersonic vehicles". *Journal of Thermophysics and Heat Transfer*, **33**(2), pp. 392–406.
- [9] Eyi, S., Hanquist, K. M., and Boyd, I. D., 2019. "Shape optimization of reentry vehicles to minimize heat loading". *Journal of Thermophysics and Heat Transfer*, **33**(3), pp. 785–796.
- [10] Willeke, S., and Verstraete, T., 2015. "Adjoint optimization of an internal cooling channel u-bend". In Turbo Expo: Power for Land, Sea, and Air, Vol. 56710, American Society of Mechanical Engineers, p. V05AT11A029.
- [11] Ghosh, S., and Kapat, J. S., 2019. "Topology optimization of serpentine channels for minimization of pressure loss and maximization of heat transfer performance as applied for additive manufacturing". In Turbo Expo: Power for Land, Sea, and Air, Vol. 58653, American Society of Mechanical Engineers, p. V05BT21A006.
- [12] Dilgen, S. B., Dilgen, C. B., Fuhrman, D. R., Sigmund, O., and Lazarov, B. S., 2018. "Density based topology optimization of turbulent flow heat transfer systems". *Structural and Multidisciplinary Optimization*, **57**(5), pp. 1905–1918.
- [13] Fabbri, G., 1997. "A genetic algorithm for fin profile optimization". *International journal of heat and mass transfer*, **40**(9), pp. 2165–2172.
- [14] Hamadneh, N., Khan, W. A., Sathasivam, S., and Ong, H. C., 2013. "Design optimization of pin fin geometry using particle swarm optimization algorithm". *PloS one*, **8**(5), p. e66080.
- [15] Lynch, S. P., Thole, K. A., Kohli, A., and Lehane, C., 2011. "Computational predictions of heat transfer and film-cooling for a turbine blade with nonaxisymmetric endwall contouring". *Journal of turbomachinery*, **133**(4).
- [16] Rasmussen, C. E., 2003. "Gaussian processes in machine learning". In Summer school on machine learning, Springer, pp. 63–71.
- [17] Frazier, P. I., 2018. "A tutorial on bayesian optimization". *arXiv preprint arXiv:1807.02811*.
- [18] Singha, S., and Sinhamahapatra, K., 2010. "Flow past a circular cylinder between parallel walls at low reynolds numbers". *Ocean Engineering*, **37**(8-9), pp. 757–769.
- [19] Molnar, C., 2020. *Interpretable machine learning*. Lulu.com.