

A Modular Conjugate Heat Transfer Optimization Framework for Thermal Management of Electric Aircraft

Christian Psenica^a, Lean Fang^a, Seth Zoppelt^a, Mark Leader^b, Ping He^{a,*}

^a*Department of Aerospace Engineering, Iowa State University*

^b*Propulsion Systems Analysis Branch, NASA Glenn Research Center*

Abstract

Conjugate heat transfer (CHT) analysis and optimization is a powerful method for improving thermal management, as it simultaneously resolves the temperature distribution in both fluid and solid domains. This paper presents a modular, discrete adjoint-based CHT optimization capability integrated within the OpenMDAO/MPhys framework. A unique feature of the proposed framework is its flexibility to extend to multi-disciplinary optimization, including aero-structural-thermal applications. The fluid domain is modeled using a finite-volume Computational Fluid Dynamics (CFD) solver, and the solid domain with a conduction heat transfer solver. A mixed Neumann-Dirichlet boundary condition is developed to enable full submersion of the solid geometry within the fluid domain, while ensuring consistent temperature and heat flux coupling at the CHT interface. Gradient-based optimization is performed; the gradients are efficiently computed using the discrete adjoint solvers implemented in DAFoam. To demonstrate the method, this paper considers two cases related to electric aircraft thermal management: a U-bend heat exchanger and an actively cooled battery pack. The U-bend case aims to minimize pressure loss while maximizing heat flux by changing the pipe geometry. The optimized design reduces pressure loss by 52.7% and increases total heat flux by 2.3%. In the battery pack case, a 3-by-3 cell configuration is cooled by ambient airflow, with constant heat generation prescribed in the cells. The battery casing shape serves as the design variable, and the objective function is a weighted sum of pressure loss and pack weight, subject to a maximum temperature constraint. The optimized design achieves a 44.6% reduction in pressure loss and a 1.5% reduction in weight, while satisfying the thermal constraint. To ensure the reliability of the optimized designs, this study validates coarse-mesh, steady-state predictions against fine-mesh unsteady simulations, demonstrating consistency within acceptable errors. This work demonstrates the potential of the developed framework to enable rapid, high-fidelity design of thermal management systems for electric aircraft.

Keywords: Conjugate Heat Transfer; Design Optimization; Discrete Adjoint; eVTOL Thermal Management

*Corresponding author

Email address: phe@iastate.edu (Ping He)

1 Nomenclature

2 *Symbols*

C_{PL}	Pressure loss coefficient
C_p	Specific heat at constant pressure
f	Objective function
$\mathbf{H}$	Influence from neighboring cell velocities and source term
N	Number of nodes in interpolation
Q	Heat flux; volumetric flow rate
$\mathbf{R}$	Residual vector
$\mathbf{S}_f$	Face area vector
$\mathbf{w}$	State variable vector
W	Battery weight
ΔT	Temperature difference
κ	Thermal conductivity
μ	Dynamic viscosity; mesh skewness parameter
ϕ	Face flux; heat flux
Ψ	Adjoint variable vector

3 *Subscripts and Superscripts*

$(\cdot)^{\text{fluid}}$	Fluid domain quantity
$(\cdot)^{\text{solid}}$	Solid domain quantity
$(\cdot)_f$	Face quantity
$(\cdot)_N$	Neighboring cell quantity
$(\cdot)_P$	Control volume cell quantity
$(\cdot)_{\text{ref}}$	Reference value
$(\cdot)_w$	Wall quantity

4 *Acronyms and Abbreviations*

BC	Boundary Condition
CFD	Computational Fluid Dynamics
CHT	Conjugate Heat Transfer
FFD	Free-Form Deformation
XDSM	Extended Design Structure Matrix

1. Introduction

Effective thermal management is a critical design challenge in electric aircraft, which lack the natural heat dissipation mechanisms found in conventional air-breathing engines [1]. Excessive heat buildup can compromise the performance and safety of key components such as batteries and electric motors. To ensure reliability, durability, and safe operation, these components must be kept cool using efficient and lightweight cooling strategies. Adjoint-based conjugate heat transfer (CHT) optimization offers a powerful framework for designing high-performance thermal management systems. Coupling computational fluid dynamics (CFD) for the fluid domain with a conduction solver for the solid domain enables realistic aero-thermal interaction modeling. As a result, significant research efforts have focused on developing CHT capabilities to support the design, analysis, and optimization of advanced heat exchangers.

Early studies involving CHT applied the approach to both analysis and design optimization. However, CHT modeling is inherently challenging due to the coupled interface between the fluid and solid domains, which can introduce numerical instabilities. To mitigate these challenges, many early methods simplified the problem by modeling only convective heat transfer with a constant wall temperature ([2–4]). While this approach avoids interface-related issues, it neglects conduction within the solid domain and enforces uniform wall temperatures, which may be unrealistic, particularly under low Reynolds number conditions where thermal gradients in the solid can be significant. To address this limitation, Mazaheri et al. [5] introduced a reduced conjugate heat transfer (RCHT) approach in the design optimization of a NASA C3X turbine blade. Although this method improved heat flux predictions, it required experimental data to determine the heat transfer coefficient. This dependency limits the method’s general applicability, as such data may not be readily available or may vary significantly between applications. As a result, the RCHT approach sacrifices extensibility and ease of use for improved modeling accuracy.

Another method of handling CHT is to re-formulate the modeling method to remove the CHT interface. The fluid-solid interactions can then be handled using various methods. As noted by Dbouk [6] in a comprehensive review of topology design methods for CHT, the Brinkman Penalization approach is one of the most common methods. This method drives the velocity within the solid domain to zero by adding a penalization term, αU , to the momentum equation of the Navier-Stokes (NS) Equations (where α is a large value and U is velocity). This method has been implemented in various topology optimization frameworks. Studies such as those performed by [7–9] developed/implemented a CHT density topology optimization capability. Taking this a step further, Sun et al. [10] developed a similar two-solid topology optimization model (optimization of a fluid and two solid domains simultaneously) for CHT. Other similar topology optimization frameworks have been developed more recently as well, such as the studies done by Meliga et al. [11] and Luo et al. [12], in which the level set method is used to resolve the CHT interface, rather than the Brinkman penalization approach. Meliga et al. [11] found that by leveraging the level set method coupled with anisotropic mesh adaptation (for regions captured by the level set method) one can cut down on computational cost as opposed to traditional topology optimization methods (which utilize isotropic grids).

To address numerical instabilities in CHT analysis, Verstraete et al. [13] developed stability criteria for partitioned CHT methods (where separate solvers are used for fluid and solid domains with careful boundary conditions to handle CHT). They analyzed four coupling strategies: flux forward temperature back (FFTB), where the fluid provides heat flux and the solid returns temperature; temperature forward flux back (TFFB), where the solid provides heat flux and the fluid returns temperature; heat transfer coefficient forward temperature back (hFTB), which uses convective heat transfer from the fluid to set solid boundary conditions with temperature returned; and their novel method, heat transfer coefficient forward flux back (hFFB). hFFB is similar to hFTB but returns heat flux instead of temperature. The study showed that CHT stability is primarily governed by the Biot number. Stability criteria were derived using a relaxation factor (β) for FFTB and TFFB, and using a virtual heat transfer coefficient for hFTB and hFFB, all validated on a flat plate test case. Importantly, the study found that methods applying temperature to the fluid domain become less stable with fewer iterations, while those applying heat flux enhance stability with hFFB being the most stable and fastest to converge. The partitioned CHT approach has been a common choice for application, as is the case with Anibal et al. [14] where an adjoint/automatic differentiation (AD) approach was used for a shape optimization of an electric aircraft motor heat sink. To handle the CHT problem, the FFTB method was utilized. This partitioned CHT approach has been used in other recent studies, such as [15–20], all of which perform an analysis and/or design optimization considering CHT.

In an effort to improve the potentially high computational time/cost needed for partitioned methods, researchers have worked on developing models for transient CHT that decouple the fluid and solid domains ([21, 22]), a feature enabled by the difference in time scales for each domain. Zhao et al. [21], specifically, introduced a novel dimensionless number, F_s , which represents the influence of convective heat transfer on the fluid flow’s temperature field. The simulation of the fluid domain is halted when the change (temporal) in F_s reaches a specified threshold, while the solid domain solution continues to run using the latest temperature distribution from the fluid domain. Other studies, like that of Du et al. [23], paused the simulations at predetermined times rather than using a variable parameter to decide, such as F_s . This method still employs separate solvers for the fluid and solid domains as partitioned methods do, but the authors noted that the introduction of this quasi-steady state heat transfer showed computational speed improvements up to 3.4 times that of partitioned methods. The main issues that arise with this method are the trade off between speed and accuracy (a decrease in computational time leads to a decrease in solution accuracy), careful implementation/tuning of F_s to take advantage of the speed benefits while maintaining accuracy, and a limitation to weak coupling between the domains which can degrade in accuracy for more complex flow.

Similar CHT studies have implemented various other methods as well. Narváez et al. [24] worked on the development of an enhanced Immersed Boundary Method (IBM) approach in order to help study CHT in turbulent wall-bounded flows for Direct Numerical Simulations (DNS) and Large Eddy Simulations (LES). A surrogate model was developed by He et al. [25] to help predict CHT characteristics to be used in design

optimization of a full-coverage film turbine vane. In similar work, a physics-informed neural network was developed [26] to handle CHT problems while trying to solve the issues of performance degradation, training instabilities, poor convergence, and low accuracy. In light of similar goals, improving stability and efficiency of CHT problems, Errera [27] worked on implementing adaptive coupling to balance between weak and strong coupling of the two domains. This was done by using a term named the numerical Biot number (a value for determining the intensity of fluid-solid thermal interaction).

Achieving an optimal design for heat transfer can be efficiently addressed using CHT optimization. This approach enables comprehensive design exploration by coupling CFD solvers for the fluid domain with conduction solvers for the solid domain, capturing realistic aerothermal interactions. Early studies primarily relied on gradient-free optimization algorithms [3, 5, 28, 29], but their computational cost grows exponentially with the number of design variables, limiting their applicability to simple geometries. In contrast, gradient-based methods, particularly those using adjoint solvers, alleviate this limitation by maintaining nearly constant computational cost regardless of design space dimensionality [2, 4]. Recent advancements in adjoint-based CHT optimization [14, 30–32] have further demonstrated the potential for scalable and efficient optimization in aerospace thermal management applications.

Despite the above progress, many existing gradient-based conjugate heat transfer (CHT) optimization studies rely on integrated or non-partitioned solvers. In these approaches, the fluid, solid, and fluid-solid coupling boundary conditions (BCs) are tightly integrated into a *single* primal and adjoint solver. While the integrated implementation can be computationally efficient, it significantly limits extensibility. Introducing a new CHT coupling BC or adding an additional discipline (e.g., structural mechanics) typically requires in-depth knowledge of the entire CHT primal and adjoint solver architecture. To break this limit, this paper proposes a fully modular CHT optimization framework, in which the fluid, solid, and CHT coupling are implemented as stand-alone solvers/components. These components, including their primal simulation and adjoint-derivative computation, are coupled using a modular multidisciplinary design optimization (MDO) library called OpenMDAO/MPhys [33, 34]. The most salient feature of the proposed framework is its flexibility: adding a new CHT coupling BC or discipline requires modifications only to the corresponding component, without necessitating knowledge of the internal implementations of the other components. This modular framework enables rapid extension of the CHT optimization capability to a wide range of applications (e.g., aero-structural-thermal optimization), something that is difficult to achieve with integrated (non-partitioned) solvers. Moreover, because all data exchange between the fluid and solid domains occurs entirely in memory, the computational efficiency of our framework matches that of integrated/non-partitioned solvers. To demonstrate the capability of this framework, we present two case studies. The first involves the optimization of a U-bend heat exchanger, based on the benchmark configuration proposed by the Von Karman Institute (VKI) for Fluid Dynamics [35]. The simulation models hot air entering the pipe and external cooling by ambient air. The objective function is a weighted sum of pressure loss and total heat flux (based on the average outlet temperature), and the design variable is the shape of the U-bend pipe. The

second case focuses on optimizing the shape of an actively cooled battery pack, where ambient air serves as the cooling medium. A heat source is prescribed for the battery cells, and the objective is to minimize a weighted sum of pressure loss and battery weight by optimizing the battery pack casing geometry. The proposed CHT optimization framework has been implemented into the DA Foam framework and is open to the public [36].

While this paper demonstrates the conjugate heat transfer optimization capability at a relatively low Reynolds number, the DA Foam framework has successfully conducted discrete adjoint-based optimization for higher Reynolds numbers up to 2 million [37–39]. The CHT optimization capabilities can be further extended to consider moving boundaries without requiring major modifications [40]. This capability allows for future applications such as eVTOL rotor optimization. Additionally, the DA Foam framework is readily extensible to topology optimization with CHT [41]. These features further underscore the broad applicability of the proposed optimization framework across various thermal management scenarios in electric aircraft design.

The remainder of the paper is organized as follows. Sec. 2 presents the proposed modular CHT optimization framework along with the theoretical background of its key components. Secs. 3.1 and 3.2 describe the aerothermal optimization setup, results, and detailed flow analysis for the U-bend and battery pack case studies, respectively. Sec. 4 summarizes the main findings and concludes the paper. Finally, Sec. 3.3 quantifies the coarse-mesh error in both cases using fine-mesh unsteady simulations.

2. Method

2.1. Conjugate heat transfer optimization framework

Fig. 1 shows the Extended Design Structure Matrix (XDSM) diagram of the CHT shape optimization framework (wrapped under the OpenMDAO/MPhys [34] framework). The optimization process begins with the pre-processing step; the mesh for both the fluid domain and solid domain is generated and, in addition, the free-form deformation control points (FFD points), which are used for geometry parameterization. pyGeo [42] is the open-source FFD package used to parameterize the design surface geometry. pyGeo constructs a set of point clouds within the FFD box, which is deformed by displacing the FFD points (these displacements serve as shape design variables). This FFD box fully contains the design surfaces in both the fluid and solid domains.

The iterative optimization process proceeds as follows: SNOPT [43], the optimizer, passes the design variable values (the displacements of the FFD points) to pyGeo for geometry parameterization. pyGeo outputs the updated surface mesh, which is given to the volume mesh deformation module, IDWarp [44]. Utilizing an inverse-distance weighted algorithm, IDWarp smoothly deforms the volume mesh in accordance with the deformed surface mesh from pyGeo. This newly deformed geometry gets passed on to the aerothermal analysis component. This component iteratively solves the coupled CHT problem, computes the objective and constraint functions (to feed into the optimizer), and finishes by passing the converged state variables

148 to the aerothermal adjoint component. The adjoint component uses DAFoam [45, 46] to compute the total
 149 derivatives for both the objective function and constraint functions. DAFoam develops Jacobian-free dis-
 150 crete adjoint methods [47] to efficiently compute gradients with a large number of design variables, allowing
 151 large-design freedom for CHT problems. The objective and constraint function(s) values and derivatives are
 152 then returned to the optimizer, which is used to update the design variables and repeat the process. This
 153 optimization process is run until convergence.

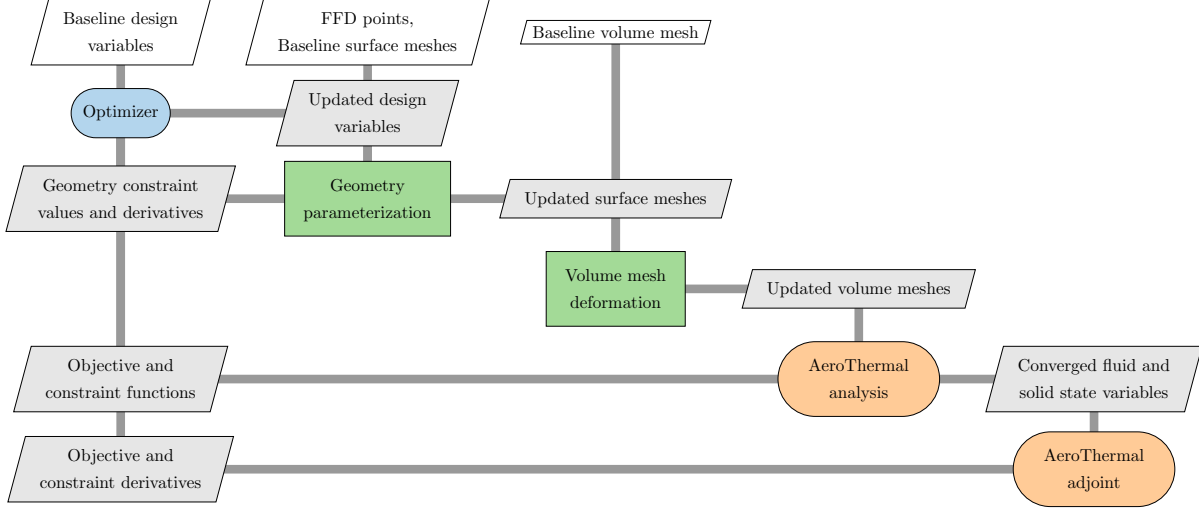


Figure 1: Schematic of the aerothermal optimization framework using the XDSM representation [48]. Here, the diagonal blocks are components, and the off-diagonal blocks are data transfer between components. A component can take inputs from the vertical direction and output data in the horizontal direction.

154 Many of the aforementioned components within this CHT framework have been elaborated on previously
 155 ([38, 49–53]). The focus and main contribution of our work lies within the development and implementation
 156 of the new CHT aerothermal analysis component. Fig. 2 shows the XDSM diagram for this component,
 157 which is a sub-iterative process of the optimization. This process begins on the fluid domain, which receives
 158 updated boundary conditions from the solid domain (reference Sec. 2.4 for details on the CHT boundary
 159 conditions). The fluid solver runs the CFD solution on the fluid domain, as elaborated on in Sec. 2.2,
 160 and upon convergence the transfer variables T_w^{fluid} and $(\kappa/d)_w^{\text{fluid}}$ (the near-wall temperature and near-wall
 161 thermal parameter, respectively) are extracted. These transfer variables are used for communication between
 162 the fluid and solid domain solvers via interpolation performed by the FUNtoFEM [54] package. This package
 163 interpolates the transfer variables from the fluid domain to the solid domain and then outputs an updated
 164 solid domain boundary condition. The updated solid domain boundary condition is passed to the solid
 165 conduction heat transfer solver (reference Sec. 2.3) which solves for the temperature profile of the solid
 166 domain before extracting the solid domain transfer variables (the near-wall temperature (T_w^{fluid}) and near-
 167 wall thermal parameter $(\kappa/d)_w^{\text{fluid}}$). In the same fashion as the interpolation from fluid to solid domains,
 168 the FUNtoFEM package is again used to interpolate the solid domain transfer variables back to the fluid

domain and to give the fluid solver updated boundary conditions. This process is run until convergence between the two domains is met. As mentioned in the introduction, this modular aerothermal analysis loop uses stand-alone components (they are not hard-coded into one solver), so it is straightforward to replace the existing CHT coupling BC with a new one.

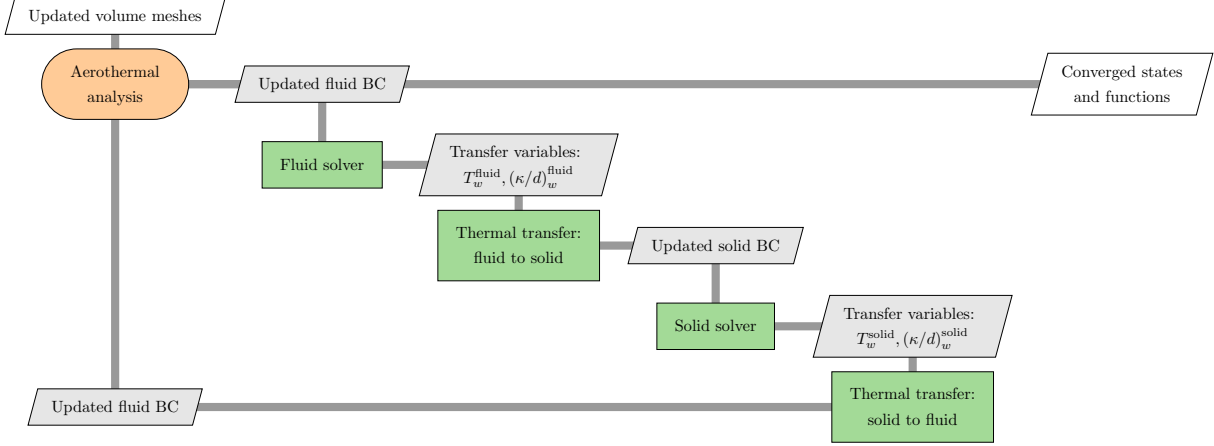


Figure 2: Schematic of the aerothermal analysis component considering conjugate heat transfer.

2.2. Finite-volume CFD simulation (fluid solver component)

Primal flow computation is performed by the DASSimpleFoam solver, which is a modified version of OpenFOAM's simpleFoam solver. The DASSimpleFoam solver adds a temperature scalar equation and solves the 3D steady-state incompressible Navier-Stokes equations (Eqs. 1 to 3).

$$\nabla \cdot \mathbf{U} = 0 \quad (1)$$

$$\nabla \cdot (\mathbf{U}\mathbf{U}) + \nabla p - \nabla \cdot \mu_{\text{eff}}(\nabla \mathbf{U} + \nabla \mathbf{U}^T) = 0 \quad (2)$$

$$\nabla \cdot (T\mathbf{U}) - \alpha_{\text{eff}}\Delta T = 0 \quad (3)$$

Here, $\mathbf{U} = [u, v, w]$ represents the velocity vector, T is the temperature, p is pressure, the effective viscosity, $\mu_{\text{eff}} = \mu + \mu_t$, is the sum of dynamic and turbulent eddy viscosity, and α_{eff} represents the effective thermal diffusivity.

Turbulence modeling is carried out using the Spalart-Allmaras model whose details of formulation and various terms are discussed in their paper [55].

$$\nabla \cdot (\mathbf{U}\tilde{v}) + \frac{1}{\sigma} \{ \nabla \cdot [(v + \tilde{v})\nabla \tilde{v}] + C_{b1}|\nabla \tilde{v}|^2 \} - C_{b1}\tilde{S}\tilde{v} + C_{w1}f_w \left(\frac{\tilde{v}}{d} \right)^2 = 0 \quad (4)$$

182 The SIMPLE algorithm (semi-implicit method for pressure linked equations) is used to solve the NS
 183 equations. First the momentum equation (Eq. 2) is discretized. Following this, an intermediate velocity field
 184 is solved (Eq. 6) which either uses a pressure field computed from the previous iteration or an initial guess,
 185 p^0 .

$$a_P \mathbf{U}_p + \sum_N a_N \mathbf{U}_N = -\nabla p^0 \quad (5)$$

186 Here, a represents a finite volume discretization coefficient while subscripts P and N denote the control
 187 volume cell and all neighboring cells, respectively. Adding a source term, $\mathbf{f}$, to Eq. 5 and rearranging the
 188 result gives Eq. 6:

$$a_P \mathbf{U}_p = - \sum_N a_N \mathbf{U}_N + \mathbf{f} - \nabla p^0 = \mathbf{H} - \nabla p^0 \quad (6)$$

189 Here, $\mathbf{H}$ is the influence from all of the neighboring cell velocities and the source term:

$$\mathbf{H} = - \sum_N a_N \mathbf{U}_N + \mathbf{f} \quad (7)$$

190 For linearizing the convective term, the face flux, ϕ , is used:

$$\int_S \mathbf{U} \mathbf{U} \cdot d\mathbf{S} = \sum_f \mathbf{U}_f [\mathbf{U}_f \cdot \mathbf{S}_f] = \sum_f \phi^0 \mathbf{U}_f \quad (8)$$

191 In Eq. 8, the face flux from the previous iteration (or the initial guess for face flux) is ϕ^0 . The f subscript
 192 represents the cell face.

193 However, when the intermediate velocity field is solved for in Eq. 6, the result does not satisfy Eq. 1, the
 194 continuity equation. Therefore, the next step is to correct the velocity and face flux by computing the new
 195 pressure field such that both Eq. 1 and Eq. 2 are satisfied.

196 Discretizing the continuity equation gives Eq. 9:

$$\int_S \mathbf{U} \cdot d\mathbf{S} = \sum_f [\mathbf{U}]_f \cdot \mathbf{S}_f = 0 \quad (9)$$

197 In Eq. 9 $\mathbf{U}_f$ is solved for via an interpolation of the cell-centered velocity, $\mathbf{U}_p$, a value obtained using
 198 Eq. 6, to the cell face:

$$[\mathbf{U}]_f = \left(\frac{\mathbf{H}}{a_p} \right)_f - \left(\frac{1}{a_p} \right)_f (\nabla p)_f \quad (10)$$

199 As discussed and proposed by Rhie and Chow [56], the momentum interpolation method is a mitigating
 200 factor to pressure field oscillations.

201 By substituting Eq. 10 into Eq. 9, the following Poisson equation is obtained:

$$\nabla \cdot \left(\frac{1}{a_p} \nabla p^1 \right) = \nabla \cdot \left(\frac{\mathbf{H}}{a_p} \right) \quad (11)$$

Eq. 11 is solved and used to give an updated pressure field (p^1). This updated pressure field is what is used to update the velocity field as well as the face flux:

$$\phi^1 = [\mathbf{U}]_f \cdot \mathbf{S}_f = \left[\left(\frac{\mathbf{H}}{a_p} \right)_f - \left(\frac{1}{a_p} \right)_f (\nabla p^1)_f \right] \cdot \mathbf{S}_f \quad (12)$$

$$\mathbf{U}^1 = \left(\frac{1}{a_p} \right) [\mathbf{H} - \nabla p^1] \quad (13)$$

This now gives a velocity and pressure field that satisfies both Eq. 1 and Eq. 2.

Finally, Eq. 3 (the passive scalar equation) is solved for the new temperature field (T). The turbulent viscosity, ν_t , is updated by solving Eq. 4 (the turbulence model equation). The above steps are run iteratively until the governing equations converge.

2.3. Finite-volume conduction heat transfer solver (solid solver component)

For simulating the temperature profile in the solid domain, this paper uses a finite-volume conduction heat transfer solver. The governing equation for this solver is the 3D steady-state heat equation:

$$\kappa \nabla^2 T_{\text{solid}} + S_{\text{heat}} = 0 \quad (14)$$

Here, T_{solid} is the solid domain temperature, κ represents thermal conductivity, and S_{heat} is the heat source.

2.4. Conjugate heat transfer boundary condition (data transfer between the fluid and solid components)

As investigated and elaborated on by Verstraete and Scholl [13], a pure Dirichlet or Neumann boundary condition can cause numerical instabilities or an ill-posed problem. As an example, implementing the commonly used fluid-domain-fixed-temperature and solid-domain-fixed-heat-flux boundary condition would create an ill-posed problem if the solid domain is fully submerged in the fluid domain. This would give a solid domain whose boundary conditions for all surfaces would be Neumann boundary conditions. Hence, the handling of the CHT interface must be dealt with, in the context of this modular approach, by using careful boundary conditions (CHT boundary conditions). This careful handling takes the form of a mixed or Robin-type boundary condition for both solid and fluid side interfaces. The mixed-type boundary condition is similar to that of OpenFOAM's `CoupledBaffleMixed` boundary condition:

$$T_{\text{wall}} = f T_{\text{ref}} + (1 - f)(T_{\text{cell}} + d \nabla T_{\text{ref}}) \quad (15)$$

In Eq. 15, T_{wall} is the temperature of the wall at the boundary, the temperature of the near-wall mesh cell center is T_{cell} , and d represents the normal distance between the wall surface and the near-wall mesh cell center. The remaining values: T_{ref} , f , and ∇T_{ref} are the reference temperature, user-prescribed value fraction,

and reference temperature gradient, respectively. The mixed-type boundary condition is then obtained by setting:

1. $T_{\text{ref}} = T_{\text{cell}}^{\text{neighbor}}$
2. $\nabla T_{\text{ref}} = 0$
3. $f = C^{\text{neighbor}} / (C^{\text{neighbor}} + C)$

The neighbor superscript denotes the value on the opposite side of the CHT interface (i.e., the neighboring domain) and setting $C^{\text{solid}} = \kappa/d^{\text{solid}}$ and $C^{\text{fluid}} = C_p \alpha_{\text{eff}}/d^{\text{fluid}}$ (the C parameter is defined for each domain).

All put together, the CHT boundary condition for the fluid domain becomes:

$$T_{\text{wall}}^{\text{fluid}} = \frac{C^{\text{solid}}}{C^{\text{solid}} + C^{\text{fluid}}} T_{\text{cell}}^{\text{solid}} + \frac{C^{\text{fluid}}}{C^{\text{solid}} + C^{\text{fluid}}} T_{\text{cell}}^{\text{fluid}} \quad (16)$$

The CHT boundary condition for the solid domain is similar:

$$T_{\text{wall}}^{\text{solid}} = \frac{C^{\text{fluid}}}{C^{\text{solid}} + C^{\text{fluid}}} T_{\text{cell}}^{\text{fluid}} + \frac{C^{\text{solid}}}{C^{\text{solid}} + C^{\text{fluid}}} T_{\text{cell}}^{\text{solid}} \quad (17)$$

The boundary conditions for both fluid and solid domains have the same right hand side. This is an important detail as this ensures temperature consistency of the wall at the CHT interface. Furthermore, it can be shown that heat flux is consistent at the interface as well. The heat fluxes are defined by Eqs. 18 and 19 for the fluid and solid domains, respectively:

$$\phi^{\text{fluid}} = C_p \alpha_{\text{eff}} \frac{T_{\text{cell}}^{\text{fluid}} - T_{\text{wall}}^{\text{fluid}}}{d^{\text{fluid}}} = C^{\text{fluid}} (T_{\text{cell}}^{\text{fluid}} - T_{\text{wall}}^{\text{fluid}}) \quad (18)$$

$$\phi^{\text{solid}} = \kappa \frac{T_{\text{cell}}^{\text{solid}} - T_{\text{wall}}^{\text{solid}}}{d^{\text{solid}}} = C^{\text{solid}} (T_{\text{cell}}^{\text{solid}} - T_{\text{wall}}^{\text{solid}}) \quad (19)$$

By substituting Eqs. 16 and 17 into Eqs. 18 and 19, then rearranging terms, it can be shown that the heat flux at the CHT interface is consistent; the heat flux from both domains is equal in magnitude and opposite in sign.

$$\phi^{\text{fluid}} = \frac{C^{\text{fluid}} C^{\text{solid}} (T_{\text{cell}}^{\text{fluid}} - T_{\text{cell}}^{\text{solid}})}{C^{\text{fluid}} + C^{\text{solid}}} = -\phi^{\text{solid}} \quad (20)$$

2.5. Interpolation of thermal variables (thermal transfer component)

The FUNtoFEM [54] package, as described previously, handles the interpolation of the thermal coupling variables between the solid and fluid domains. FUNtoFEM links a given solid node with the nearest N fluid nodes, and vice versa (the MELD transfer scheme). The weight of the N nodes is determined from their Euclidean distance. This robust package can handle various different setups, such as structured and unstructured meshes with either a conformal or non-conformal solid-fluid interface. The optimization work presented in this paper (Secs. 3.1 and 3.2) uses only conformal meshes (surface mesh cells from both fluid and solid domains perfectly align at the interface), hence $N = 1$.

2.6. Aerothermal coupled analysis and adjoint computation (aerothermal analysis component)

The aerothermal analysis and adjoint components are implemented into DAFoam, which provides the solid and fluid solver components and uses OpenMDAO/MPhys to handle the MDO coupling. This entire aerothermal coupling procedure, inclusive of all aforementioned components, is shown in finer detail in OpenMDAO's N2 diagram (Fig. 3). In Fig. 3, the dark blue boxes are groups and the light blue boxes represent the components of each group. The dark green boxes are the inputs of each component, while the light green boxes are for the outputs. The right-hand side of the N2 diagram shows the data flow throughout the model.

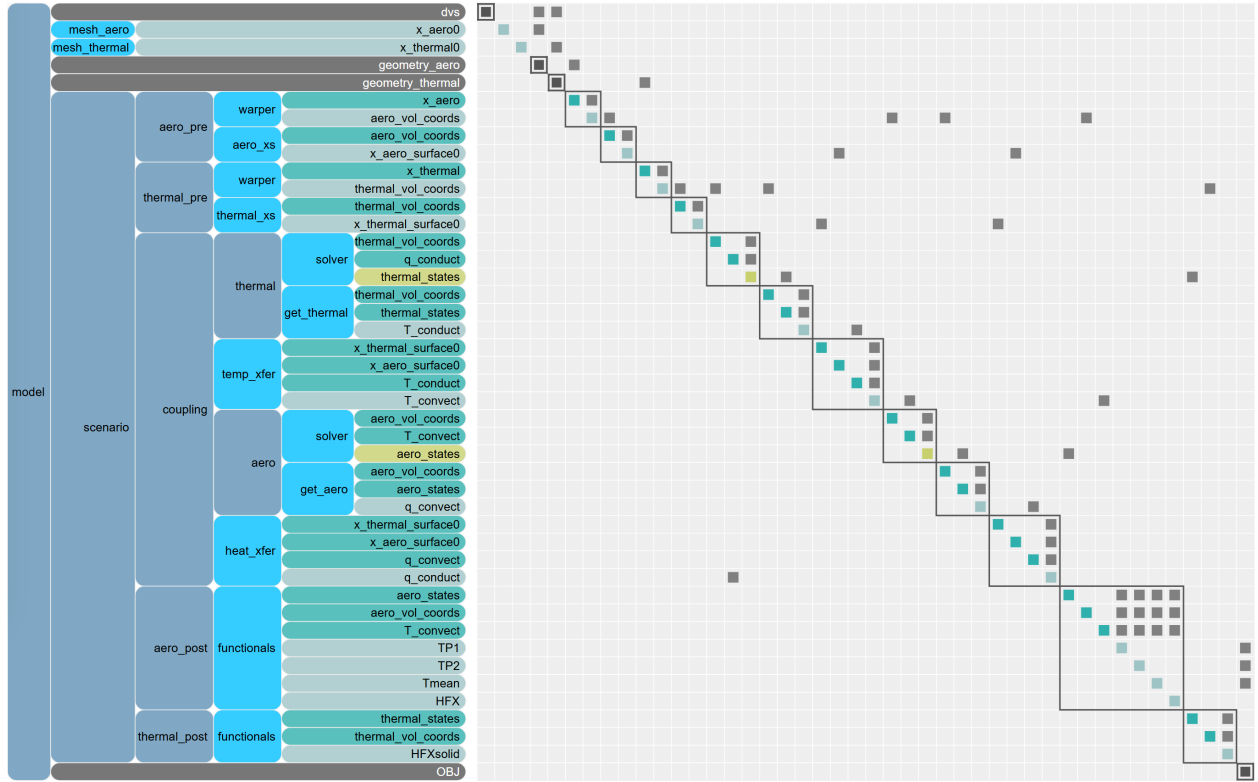


Figure 3: N2 diagram of the U-bend aerothermal optimization framework.

The use of OpenMDAO and MPhys is beneficial as traditional CHT approaches often use hard-coded solvers for the aerothermal coupling (creating difficulties in terms of extensibility to other applications). In contrast, the flexibility of both OpenMDAO and MPhys allows for a modular implementation of aerothermal coupling. The user, as required by MPhys, implements the methods used to calculate the outputs based on inputs for the components in Fig. 3. MPhys also computes the product of the state Jacobian matrix with a supplied vector. OpenMDAO then unifies the adjoint derivative computations using the MAUD algorithm [57]. Additionally, utilized in this work is OpenMDAO's built-in nonlinear block Gauss-Seidel solver:

$$\begin{bmatrix} A_{\text{fluid}} & C_{\text{s2f}} \\ C_{\text{f2s}} & A_{\text{solid}} \end{bmatrix} \begin{bmatrix} w_{\text{fluid}} \\ w_{\text{solid}} \end{bmatrix} = \begin{bmatrix} b_{\text{fluid}} \\ b_{\text{solid}} \end{bmatrix} \quad (21)$$

In this solver, A and b represent the nonlinear discrete matrix and vector, respectively. w denotes the state variable vector and C is the coupling variables. The Aitken under-relaxation method is implemented in order to update the states for each block iteration ensuring robust convergence. The coupled adjoint system is solved using the linear block Gauss-Seidel solver (also built into OpenMDAO):

$$\begin{bmatrix} \frac{\partial \mathbf{R}_{\text{fluid}}}{\partial \mathbf{w}_{\text{fluid}}}^T & \frac{\partial \mathbf{R}_{\text{solid}}}{\partial \mathbf{w}_{\text{fluid}}}^T \\ \frac{\partial \mathbf{R}_{\text{fluid}}}{\partial \mathbf{w}_{\text{solid}}}^T & \frac{\partial \mathbf{R}_{\text{solid}}}{\partial \mathbf{w}_{\text{solid}}}^T \end{bmatrix} \begin{bmatrix} \Psi_{\text{fluid}} \\ \Psi_{\text{solid}} \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial \mathbf{w}_{\text{fluid}}}^T \\ \frac{\partial f}{\partial \mathbf{w}_{\text{solid}}}^T \end{bmatrix} \quad (22)$$

For this linear solver, R is the residual, f is the function of interest (objective or constraint function), and w denotes the state variables. As previously mentioned, DAfoam [36, 46] serves as the fluid and solid adjoint solver. DAfoam solves these equations via a segregated approach with Aitken under-relaxation:

$$\frac{\partial \mathbf{R}_{\text{fluid}}}{\partial \mathbf{w}_{\text{fluid}}}^T \Psi_{\text{fluid}} = \frac{\partial f}{\partial \mathbf{w}_{\text{fluid}}}^T - \frac{\partial \mathbf{R}_{\text{solid}}}{\partial \mathbf{w}_{\text{fluid}}}^T \Psi_{\text{solid}}, \quad (23)$$

$$\frac{\partial \mathbf{R}_{\text{solid}}}{\partial \mathbf{w}_{\text{solid}}}^T \Psi_{\text{solid}} = \frac{\partial f}{\partial \mathbf{w}_{\text{solid}}}^T - \frac{\partial \mathbf{R}_{\text{fluid}}}{\partial \mathbf{w}_{\text{solid}}}^T \Psi_{\text{fluid}}, \quad (24)$$

DAfoam, which is an open-source framework for OpenFOAM, implements a Jacobian-free discrete adjoint method. Automatic differentiation is used for the matrix-vector products and partial derivatives (further details of this approach can be found in [47]). To solve the adjoint equation DAfoam employs the generalized minimal residual (GMRES) method which is an iterative (linear) solved found in the PETSc [58] library. The global preconditioner is from a nested preconditioning strategy using the additive Schwartz method. The local preconditioning is handled with the incomplete lower and upper (ILU) factorization method with one level of fill-in. Constructing $[\partial \mathbf{R} / \partial \mathbf{w}]_{RC}^T$ is done only once, at the first time step, and then is continuously reused for the adjoint equation. However, to help improve convergence [45], the construction of $[\partial \mathbf{R} / \partial \mathbf{w}]_{RC}^T$ is done so by approximating the residuals and their linearizations. As the cost of constructing $[\partial \mathbf{R} / \partial \mathbf{w}]_{RC}^T$ is roughly 30% of the adjoint runtime, this approach greatly reduces the total adjoint runtime. Furthermore, in a detailed analysis we previously performed [45], we showed that when comparing the flow runtime to adjoint runtime their ratio is about unity. This relationship holds regardless of Jacobian size (mesh cells) and number of cores used which validates the high level of computational efficiency of our framework.

3. Results and discussion

To demonstrate the proposed modular CHT optimization framework, we consider two components (U-bend pipe and battery packs) in eVTOL (electric vertical take-off and landing) thermal management systems

in this section. We will elaborate on the optimization configurations for these two cases, conduct the CHT optimizations, and analyze the optimization results. We will then verify the optimized designs using high-resolution, unsteady simulations in the Appendix.

3.1. U-bend heat exchanger

3.1.1. CHT problem description

U-bend heat exchangers are widely used in aerospace applications due to their ability to enhance turbulent mixing, which promotes effective heat transfer. However, this improved performance often comes at the cost of increased pressure losses, which reduce overall system efficiency by elevating pump power requirements. To address these trade-offs, numerous studies have used CFD to investigate fluid flow and heat transfer characteristics, as well as to improve the design of channel and pipe heat exchangers. Initial research focused on ribbed straight channels [3, 28], with subsequent studies exploring more complex geometries such as ribbed angled channels and smooth-walled channels [5, 29, 52]. These investigations primarily analyzed the effects of channel shape, rib geometry, and rib arrangement on heat transfer performance. More recent efforts [2, 4] have employed adjoint-based methods for multi-objective optimization, simultaneously minimizing pressure loss and maximizing heat transfer.

As mentioned in Sec. 1, this paper uses a circular cross-section U-bend pipe as the baseline heat exchanger geometry (similar to that developed at the Von Karman Institute (VKI) for Fluid Dynamics [35]). The dimensions of the pipe are as follows: the inner radius is 42 mm, the outer radius is 45 mm, 850 mm in total height, and a total width of 200 mm. Each leg of the pipe (inlet and outlet leg) is 750 mm in height and is connected by a 180-degree U-bend section. The simulation domain (Fig. 4) consists of two sub-domains: the pipe itself (solid domain) and the far field and internal pipe flow domain (together encompassing the fluid domain). The far field fluid domain is a $2.4 \times 2 \times 1.5$ m box. This gives two CHT interfaces shown in green in Fig. 4. The first being the inner surface of the pipe and the second as the outer surface. The red surfaces denote the inlet and outlet, whereas the grey surfaces are the adiabatic walls.

The internal flow of the pipe is of a 310K hot fluid (air) with a 5 m/s velocity (prescribed at the inlet). The cold fluid, at 290K, flows over the exterior of the pipe (far field domain flow) with a prescribed inlet velocity value of 5 m/s. The fluid velocity, supplied by an internal pump, was chosen to give similar flow physics as previous studies ([2, 4, 59] to name a few). The initial condition of the pipe itself is a uniform 300K. Based on pipe diameter, the Reynolds number is 2.8×10^4 . Due to the low Mach number (~ 0.015), we consider the flow to be incompressible. Additionally, the flow is assumed to be fully turbulent to simplify the optimization.

The pipe was meshed using ANSYS Meshing, generating an unstructured mesh with a cell count of 670,000 for the fluid domain and 118,000 for the solid domain. The boundary layer was captured with four layers of prism mesh on both outer and inner surfaces of the pipe (Fig. 5 left) with an averaged y^+ of 24. Turbulence was handled with the Spalart-Allmaras model, which was used with a wall function to account

for the average y^+ . To ensure robust convergence of the steady-state CFD problem, a first-order upwind discretization scheme was used for the convective term.

The above-mentioned CFD configuration can introduce numerical errors when compared to a fine-mesh transient simulation with an averaged $y^+ \sim 1$. The motivation for these settings (coarse-mesh, steady-state CFD) is due to a significant reduction of computational cost. As previous studies ([60, 61]) have shown, design improvements can be predicted from coarse-mesh CFD and verified with fine-mesh CFD simulations. The absolute values from the coarse-mesh CFD may contain errors, but the process successfully captures the correct trends for design improvements. In the Appendix, the coarse-mesh CFD errors for both test cases (U-bend heat exchanger and the actively cooled battery pack) will be quantified using a fine-mesh CFD transient simulation with $y^+ \leq 1$.

The design surfaces for this optimization, the inner and outer surfaces of the pipe, are parameterized with 92 FFD points in total. Of these 92 points, 36 are used for design variables, which are shown in red in Fig. 5 (right). Each point is free to move independently in the x , y , and z directions. The outermost points are restricted to a maximum 40 mm displacement while the innermost have a 10 mm maximum displacement. Since each one of the x , y , and z coordinates of the FFD points is used as a design variable, there are a total of 108 design variables for this optimization. The large number of design variables used for this case is necessary as it gives sufficient design freedom to the optimizer. This is further supported in Appendix A where a different weighting scheme is used for the objective function, which results in a different optimized design. The objective function, $f = 0.9 \frac{T_M}{T_{M_{\text{ref}}}} + 0.1 \frac{C_{PL}}{C_{PL_{\text{ref}}}}$, is a weighted sum of the average outlet temperature

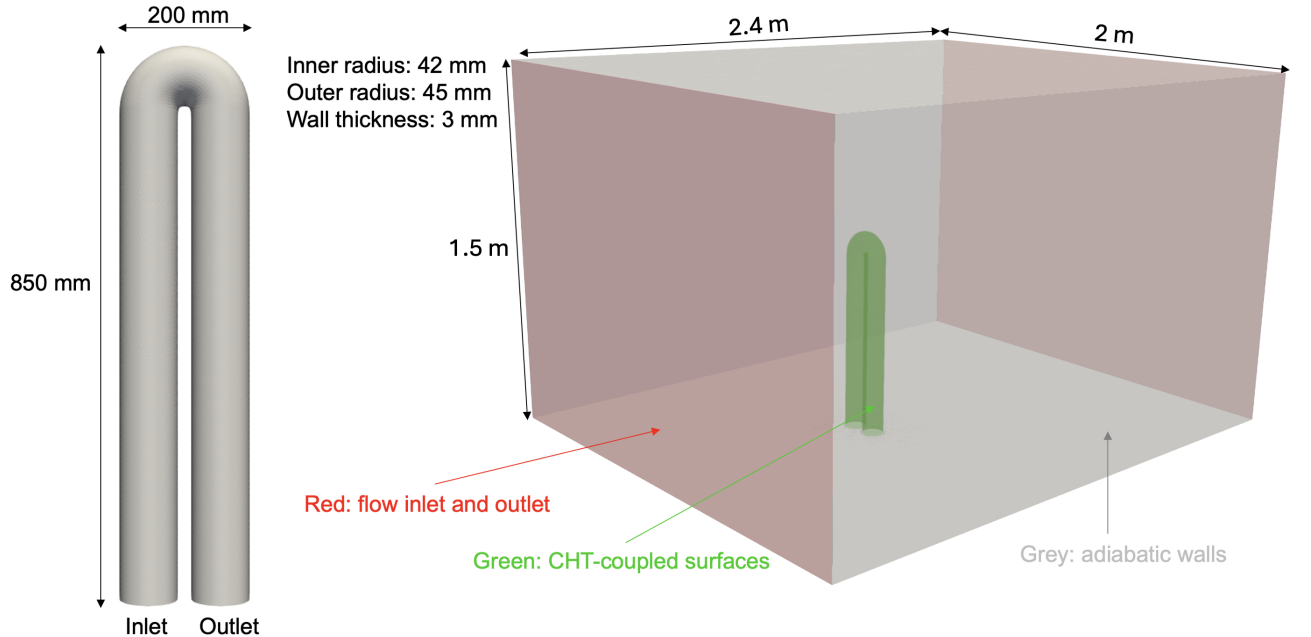


Figure 4: Schematic of the U-bend heat exchanger configuration.

and total pressure loss (denoted as T_M and C_{PL} , respectively). The “ref” subscripts denote the respective value for the baseline pipe geometry. This objective function aims to increase the total heat flux of the heat exchanger as well as improve pump efficiency. The rationale for minimizing pump power, which is equivalent to minimizing pressure loss, is that this would lead to an overall more efficient system (less power required to mitigate heat), and a lighter-weight pump could be used, reducing the gross takeoff weight of the aircraft. A detailed optimization formulation is outlined in Table 6. For this case, we optimize for only one weighting scheme, 90% total heat flux and 10% pressure loss, to showcase the effectiveness of our approach in handling a multi-objective CHT optimization.

Table 4: CHT optimization problem formulation for U-bend heat exchanger.

	Function/Variable	Description	Quantity
Minimize	$f = 0.9 \frac{T_M}{T_{M_{\text{ref}}}} + 0.1 \frac{C_{PL}}{C_{PL_{\text{ref}}}}$	Weighted sum function consisting of pressure loss and mean outlet temperature	1
with respect to	$-40 \leq \Delta x, \Delta y, \Delta z \leq 40$ mm	FFD shape displacement for outer portion of channel in the x, y , and z directions	54
	$-10 \leq \Delta x, \Delta y, \Delta z \leq 10$ mm	FFD shape displacement for inner portion of channel in the x, y , and z directions	54
	Total Design Variables		108

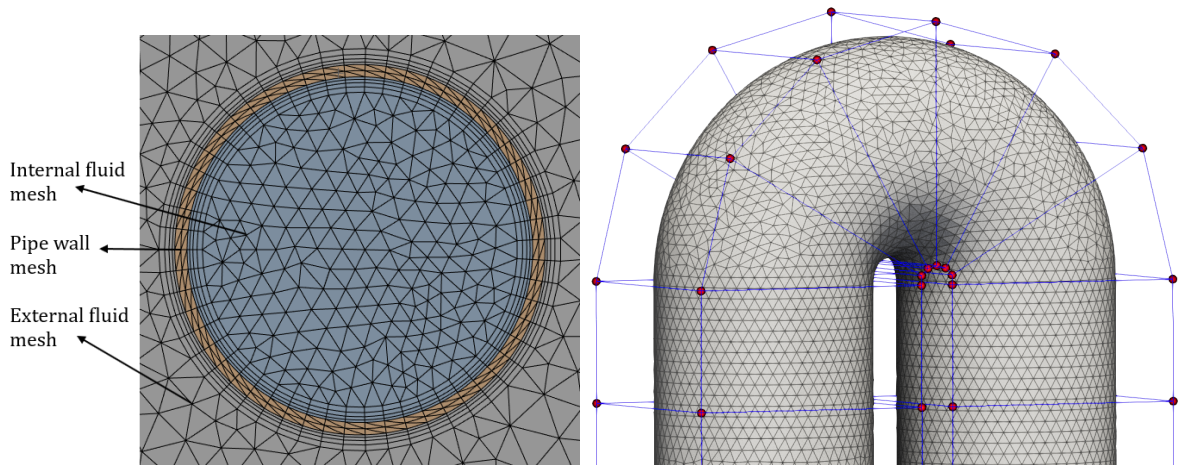


Figure 5: Meshes for the fluid and solid domains and the free form deformation control points (red).

Table 5: Summary of objective functions for the baseline and optimized designs.

	Pressure loss (Pa)	Total heat flux (W)
Baseline	22.31	122.85
Optimized	10.55	125.66
Difference	↓ 52.71%	↑ 2.29%

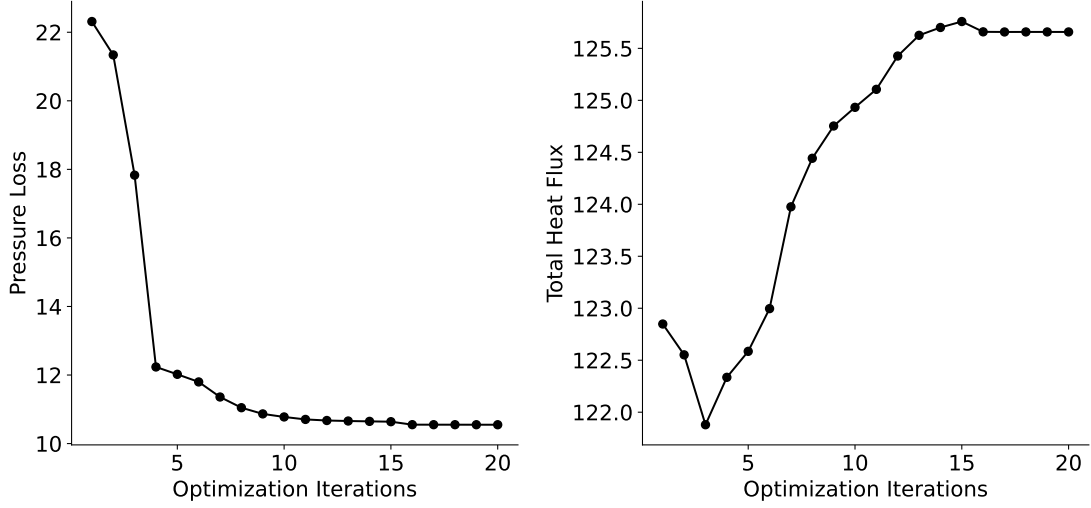


Figure 6: Optimization history for the objective functions of pressure loss (left) and total heat flux (right).

3.1.2. Aerothermal analysis for the baseline and optimized designs

The performance comparison between the baseline and optimized pipe is shown in Table A.10. The optimized pipe reduces pressure loss by 52.71% while simultaneously improving the total heat flux by 2.29%. The optimization ran for 20 iterations over the course of 3 days, using 72 processor cores total (Intel Skylake AVX512 CPU). The optimization history for each objective is shown in Fig. 6.

The optimization process as a whole sought to enlarge the channel, as can be seen in Fig. 7. More specifically, while the inner radius of curvature has remained mostly unchanged (though it did slightly increase), the outer radius of curvature (outer edge) has increased significantly and taken on a less uniform shape. The U-bend channel also noticeably grew in height, increasing by 36.025 mm.

The baseline pipe had a surface area of 441.2 mm², which increased by 16.45% to 513.8 mm² for the optimized geometry. As will be discussed, this change in geometry is expected as it is beneficial for both objectives. This size increase of the U-bend is a similar result to other studies [2, 4] (it should be noted, however, that the authors did not simulate a far field domain). Verstraete and Li [2] performed a shape optimization of a square cross-section U-bend heat exchanger using a multi-objective function of pressure loss and heat flux. In this study, the authors optimized pipe, particularly the “MinP” configuration, shares several physical details: the channel overall grew in size with the inner and outer radii increasing (though the change in inner radius for our pipe optimization is far less), the inlet leg takes on a sharper turn on the

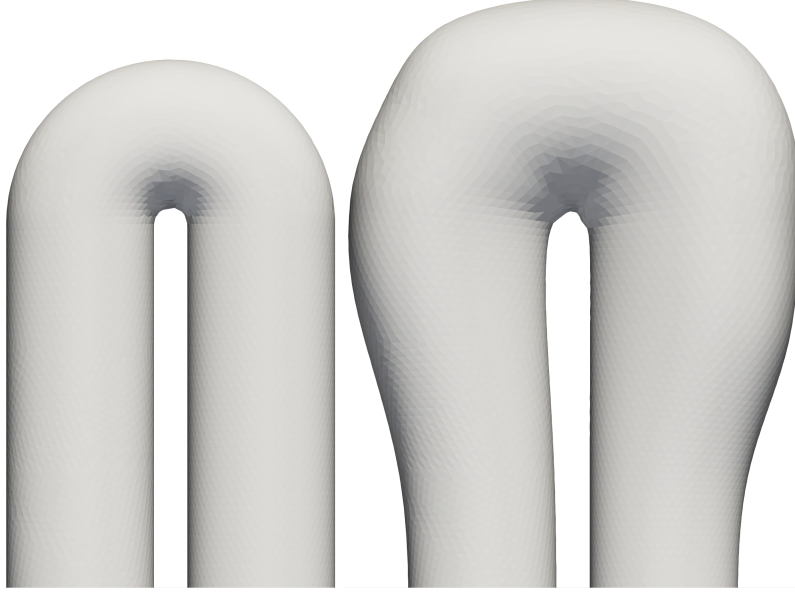


Figure 7: Baseline geometry (left) Vs. optimized geometry (right).

outer edge, and the outlet leg sticks out and features a straight outer edge.

The most noticeable difference in the optimized shape presented in this paper and that presented by Verstraete et al. [4] is that we have a smaller change to the inner radius of curvature, with the most change happening on the outer edge of the bend portion. Verstraete et al. generated an optimized shape that increases both radii significantly. It should be noted, however, that the results from this study show only a 39% reduction in pressure loss, even though the number of design variables used is much higher at 540. This is likely due to not giving proper design variable bounds, which restricts design freedom for the optimizer, resulting in a lower performance heat exchanger.

Regardless of differences, the results presented in this paper are expected; to make a more uniform pressure profile within the channel (which decreases pressure loss), the velocity gradient needs to be less sharp and be more gradual over the U-bend portion of the channel since this will create a less adverse pressure gradient. As internal pressure is more heavily influenced by geometry in comparison to heat flux, the optimized shape is an easily understood result as it still provides some turbulent flow mixing (which will enhance heat flux) but also greatly reduces this turbulent mixing from the baseline design (which reduces pressure loss).

As mentioned previously, it is expected that the velocity profile will become more uniform throughout the optimization process. A more uniform velocity profile reduces pressure loss from frictional forces and turbulent flow. Fig. 8 shows the velocity contour overlaid by the flow streamlines between the baseline and the optimized U-bend pipe at the mid-plane. The most noticeable change in the velocity profile is the absence of the large turbulent region just past the inner U-bend portion. The sharp U-bend turn of the baseline design generates an adverse pressure gradient along the streamwise direction, ultimately resulting

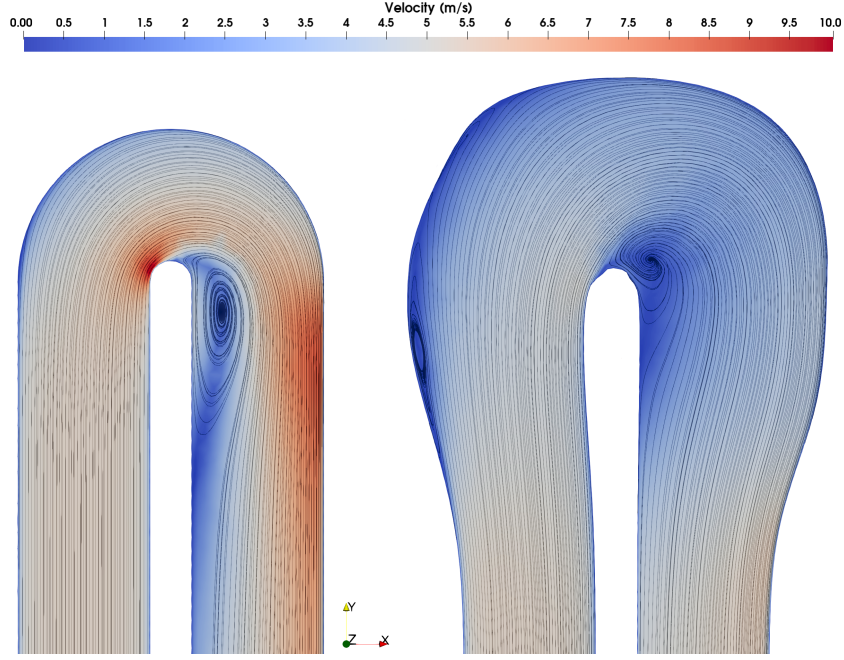


Figure 8: Baseline (left) Vs. optimized geometry (right) internal velocity streamlines at the symmetry plane.

in strong flow separation. Though this flow feature is beneficial to heat transfer as it encourages turbulent mixing between hot and cold regions within the pipe, it significantly increases the pressure loss. This is a common trend found amongst similar studies where a decrease in pressure loss is achieved by a more uniform velocity profile/reduction in flow separation ([2–4, 11, 16, 52] to name a few).

The optimized pipe still exhibits some flow separation just beyond the inner U-bend portion, as there is a thermal objective function being considered (total heat flux), but it has been greatly reduced. Additionally, there is a slight region of flow separation predicted by the CFD on the optimized pipe along the outer surface before the U-bend. This small region of flow separation can enhance, though minimally, the local heat flux in that region.

This change in velocity profile is achieved by the increase in the radius of curvature of the optimized U-bend pipe. The inner radius of curvature is mostly the same, but the outer radius of curvature is noticeably larger. This increase allows for a more gradual velocity gradient than the baseline design, which decreases the mean acceleration of the fluid. This is the mechanism responsible for reducing flow separation and maintaining fluid pressure.

The far-field flow helps drive the overall heat transfer by cooling the U-bend pipe and therefore helping to maintain a greater temperature gradient. The region of turbulent flow just beyond the exterior surface of the channel, Fig. 9, shows a decrease in flow separation due to the increase in overall radius of the pipe over the U-bend. The baseline design shows two significant regions of flow separation, while the optimized design shows only one heavily reduced region of turbulent flow. Although this change is substantial, we do not observe a clear difference in heat flux between the baseline and optimized design due to the pipe’s exterior

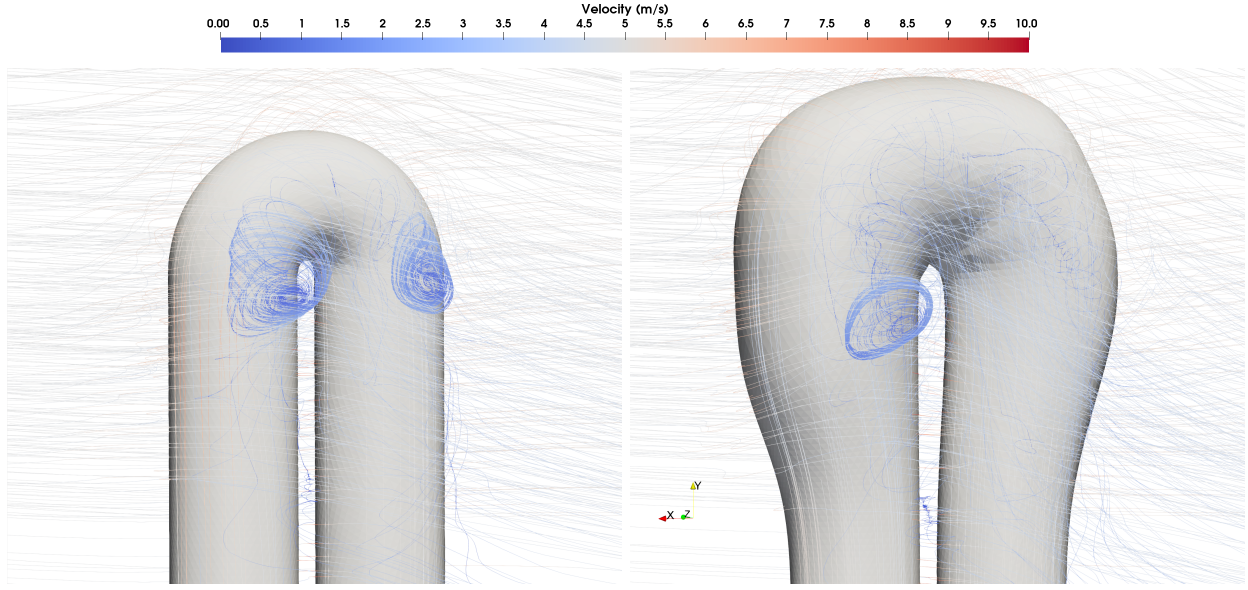


Figure 9: Baseline (left) Vs. optimized geometry (right) external velocity streamlines.

geometry/flow.

From the more uniform velocity profile, the pressure profile (Fig. 10) follows suit and becomes far more uniform in the optimized pipe. More specifically, the large reduction in flow separation just beyond the inner U-bend directly corresponds to a large increase in local pressure. The inlet leg underwent the opposite change, decreasing in pressure from 35.07 Pa to 23.29 Pa (a 33.6% drop). The change in the outlet leg pressure is negligible, decreasing by 0.16% (12.76 Pa for the baseline design to 12.74 Pa on the optimized design). The combined effect of this pressure change leads to a 52.71% drop in total pressure.

The 52.71% drop in pressure loss is considerably greater than that of similar studies ([2, 4]). In Verstraete and Li [2], the authors achieved a 30.81% pressure loss reduction while the later study [4] saw a slight increase to 39%. These differences can be attributed, partly, to differences in the weighting of the objective function and design freedom. For example, Verstraete and Li [2] bounded the geometry deformation to a predetermined bounding box (the pipe could not be deformed in such a way that it would exceed this box). The result of this was an optimized U-bend pipe which featured a flat top rather than allowing the top of the pipe to take on a larger curve, which helps reduce pressure loss. He et al. [52], as a counter example, did not have such restrictions and saw a 55.5% decrease in pressure loss when optimizing a square cross-section U-bend under the same weighted objective function.

However, the trend of creating a more uniform velocity profile to decrease pressure loss is common in other types of similar studies as well ([7, 17] to name a couple). Subramaniam et al. [7] performed a topology optimization of a passage with the goal of enhancing heat flux while mitigating pressure loss. From this optimization, it was found that for large weights given to pressure loss, the optimizer seeks out a uniform velocity profile, which in turn creates a (more) uniform pressure profile. Though the weight given to pressure

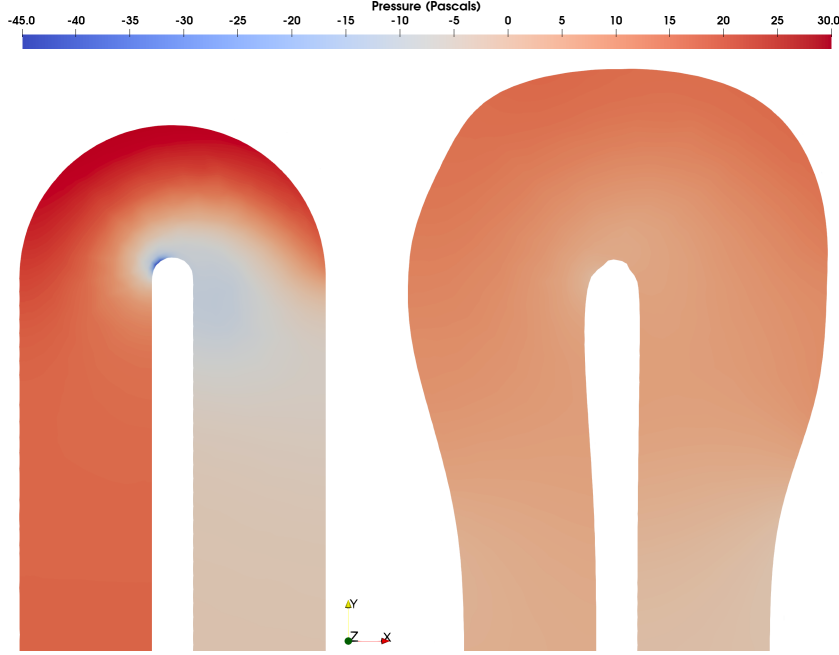


Figure 10: Baseline (left) Vs. optimized geometry (right) internal pressure profile at the symmetry plane.

loss (90%) in our study is much less than that given to heat flux (10%), the changes in the pipe's geometry are in favor of pressure loss since pressure loss is more easily influenced by changes in geometry than heat flux is.

Though this decrease in pressure loss is favorable as it leads to a more efficient design in terms of pump power, it tends to decrease the total heat flux. The regions of greatest heat flux on the baseline pipe are a small zone of low-pressure right before the inner U-bend and a large region past the U-bend, which corresponds to the region of large flow separation from Fig. 8 (left). Both of these notable regions are missing from the optimized pipe. As noted by other studies ([2, 3, 28, 29, 52]), and holds true in this present study, the objectives of heat flux and pressure loss naturally oppose one another; turbulent flow will increase the pressure loss ultimately due to the sharp reduction in velocity but will also enhance heat flux since the turbulent flow allows more mixing of the air which helps maintain a greater temperature gradient.

Despite the loss in flow separation, the total heat flux increased from the baseline design to the optimized design by 2.29%. The changes in heat transfer performance can be seen in Fig. 11, which shows the Nusselt number contour for the inner surface of the U-bend pipe. The Nusselt number is defined as $Nu = QD/(k\Delta T)$ where: Q denotes heat flux, D is the inner pipe diameter, k is the thermal conductivity of air, and $\Delta T = 10$ K is the temperature difference between the inner pipe surface and the internal fluid flow. The contours are visibly different from one another. In the optimized design, it is clear to see a net decrease in the Nusselt number. The heat flux (that is, per unit area) has decreased for the optimized design, but the increase in surface area (16.45%) acts as a mitigating factor, allowing the total heat flux to increase.

The trade-off between the objective functions of total heat flux and pressure can be seen in Fig.12

where the performance factor, $Nu/(fRe)$, is shown over the optimization process. Here we define $f = C_{PL}/(0.5(l/D)\rho U_m^2)$ as the friction factor where C_{PL} is the coefficient of pressure loss, l is the length of the pipe, D is the diameter, ρ is the fluid density, and U_m is the average outlet velocity. The performance factor increases by nearly 50% from the baseline to optimized design, which is in accordance with the reduction in pressure loss. This indicates a high performance increase in pressure loss over total heat flux.

The increase in heat transfer is less than that of similar studies [2, 4, 52], where heat flux increased anywhere from 8% to 42%. Again, these changes can be attributed to differences in setup. Besides the authors using a square cross-section instead of a circular the model was also convection-only. A pure convection model, considering the application, is less accurate than a CHT model since conduction effects are not ignored and the boundary condition for the wall is more realistic (not a fixed wall temperature).

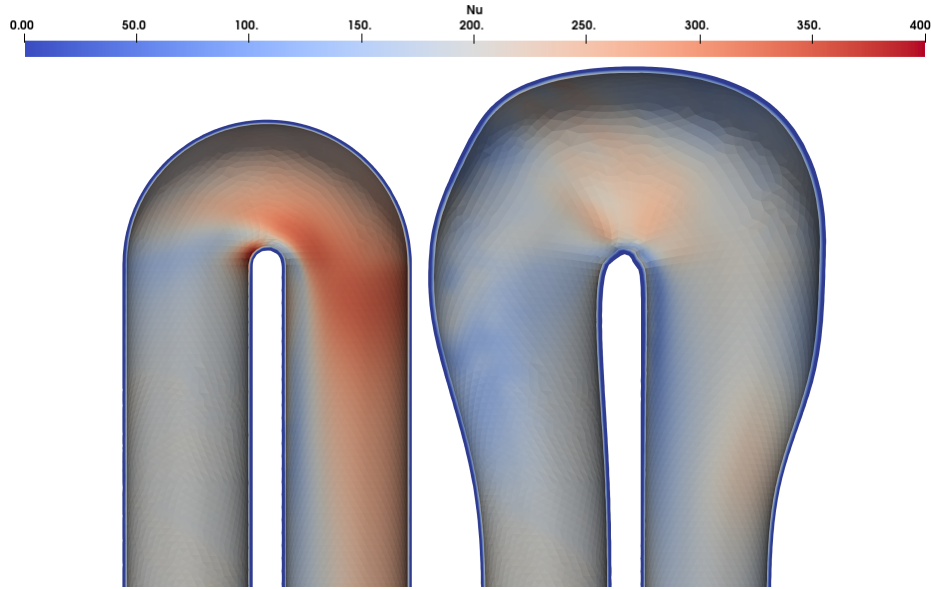


Figure 11: Baseline (left) Vs. optimized geometry (right) internal-surface heat flux profile.

3.2. Battery pack

3.2.1. CHT problem description

Battery packs are critical subsystems that require careful thermal management design. Cooling strategies can be either passive or active. Passive cooling avoids the need for power-consuming pumps, but it necessitates a larger surface area for heat rejection to the environment, which increases the overall weight of the battery pack. In contrast, active cooling employs forced convection, typically using ambient air, which is readily available on aircraft, to reject heat more efficiently. While this approach can significantly reduce battery pack weight, it requires a well-designed cooling system to compensate for the added weight and power demands of the cooling pump [62]. The CHT optimization method has been applied to analyze the aerothermal performance of various battery pack geometries and flow conditions [63, 64]. Some studies have extended this analysis to optimize performance [65], though they relied on gradient-free optimization algorithms. To

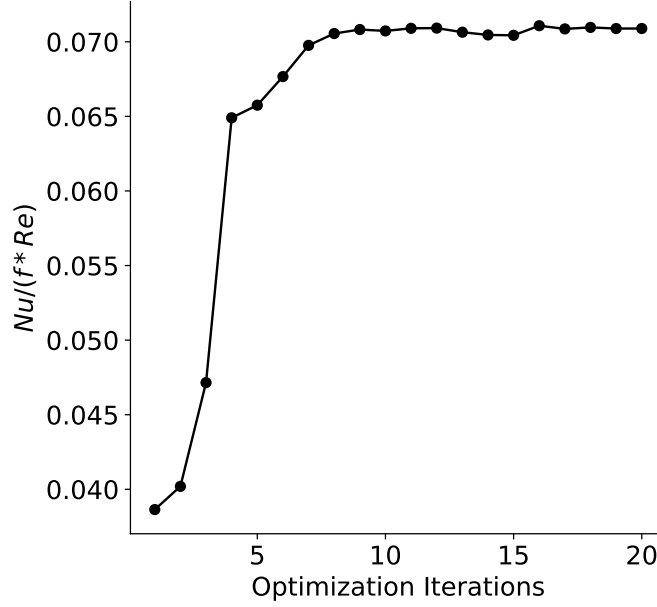


Figure 12: Performance balance between Nusselt number and friction loss.

enable exploration of a larger design space more efficiently, many previous works employed a gradient-based optimization approach using adjoint-based sensitivity analysis, as demonstrated by Gkaragkounis et al. [30], Gallorini et al. [31], and Imam-Lawal et al. [32].

In the second case, we examine an actively cooled battery pack configuration composed of a 3×3 grid of battery cells, as illustrated in Fig. 13. Each cell is an 18650 lithium-ion battery, measuring 9 mm in radius and 65 mm in height. The battery pack is arranged such that every cell is enclosed in casing material, and the entire cell grid is surrounded on all sides by flat aluminum plates, each 3 mm thick. Air channels are provided between the cells to allow cooling airflow. For the purpose of optimization, the geometry of the battery casing is treated as the design variable, aiming to enhance overall performance while maintaining ideal battery temperatures. The baseline configuration features a casing with a cylindrical cross-section and a radius of 15 mm. Spacing between the centers of adjacent cells is set at 50 mm in both directions. Aluminum is used for both the casing material and the enclosing top and bottom plates.

A simulation domain measuring 240×170 mm is employed to perform the CFD analysis. This enlarged domain in the streamwise direction increases the effective cooling surface area artificially. To account for this, it is assumed that heat generation from the battery cells is transferred only through the cylindrical surfaces and specific portions of the top and bottom plates, indicated by the green square in Fig. 13 (left) and the green-colored surfaces in Fig. 13 (right). All remaining areas of the top and bottom plates, as well as the side boundaries, are treated as adiabatic, shown as grey surfaces in Fig. 13 (right).

As was the case with the U-bend pipe, unstructured meshes for both the fluid and solid domains were generated using ANSYS Meshing, shown in Fig. 14. The fluid domain consists of approximately 400,000 cells, while the solid domain contains roughly 320,000 cells. To resolve the boundary layer behavior within

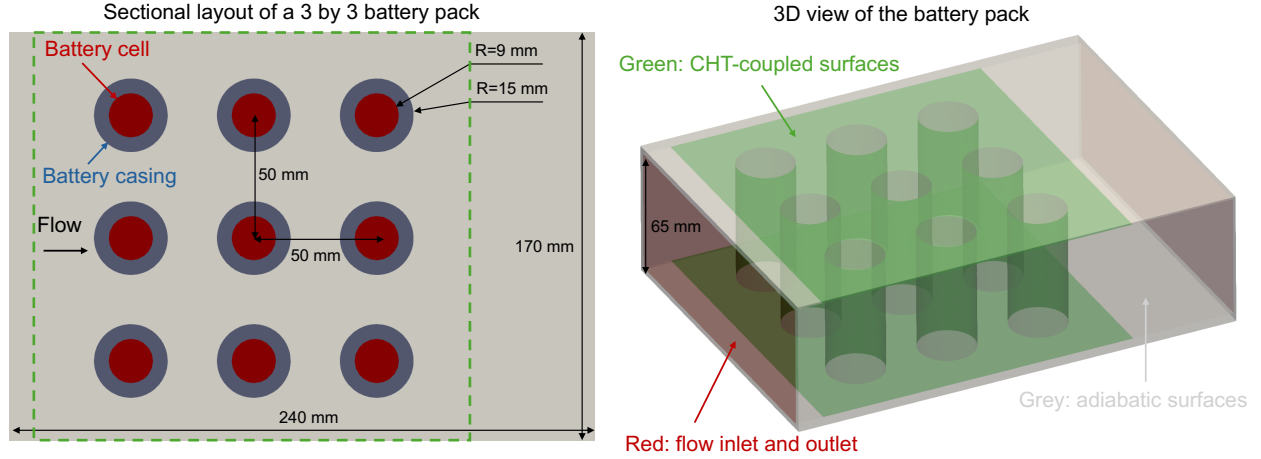


Figure 13: Schematic of the active cooling battery pack configuration.

the fluid region, similar to the approach taken in the U-bend case, three layers of prism cells are applied near all wall surfaces, with an average y^+ value of 30. The Spalart-Allmaras turbulence model was employed, using wall functions, and the convective terms are discretized using a first-order upwind scheme.

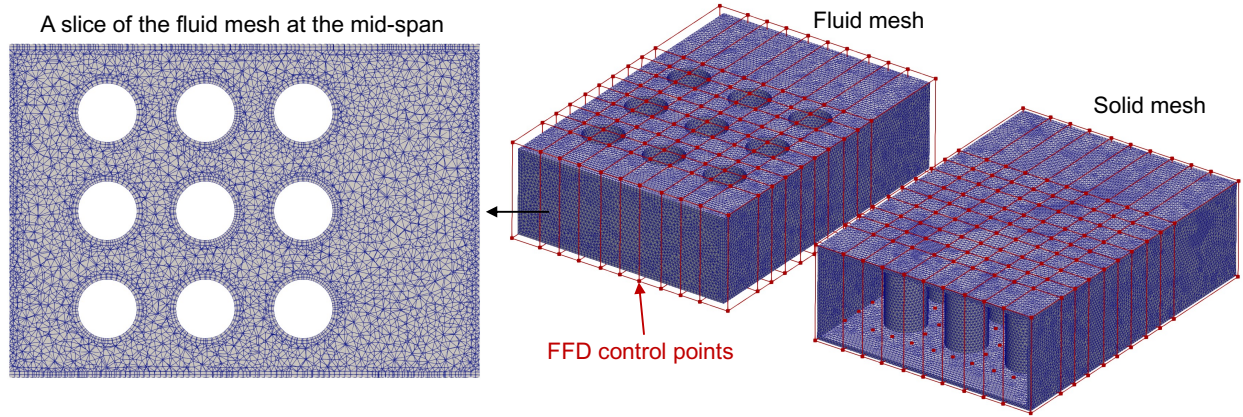


Figure 14: Meshes for the fluid and solid domains and the free-form deformation control points (red).

To replicate heat generation within the solid domain, a constant heat source is applied to each battery cell, as depicted in Fig. 15. Based on a nominal power output of 57.6 W for the 18650 battery cell and an assumed efficiency of approximately 82%, the resulting heat generation per cell is calculated to be 10 W. The heat source is modeled as a cylinder with a radius of 9 mm and a height of 65 mm, occupying the same spatial region as the physical battery cell. To mitigate abrupt spatial variation, a Gaussian smoothing function is applied to allow the heat source to decay gradually in the radial direction. Since the casing geometry evolves throughout the optimization process, the mesh cells corresponding to the battery cell region may vary. To address this, the heat source distribution is recalculated and reassigned to the appropriate solid mesh cells during each optimization iteration. This approach preserves the cylindrical shape of the heat source and ensures that its total magnitude remains constant throughout the optimization.

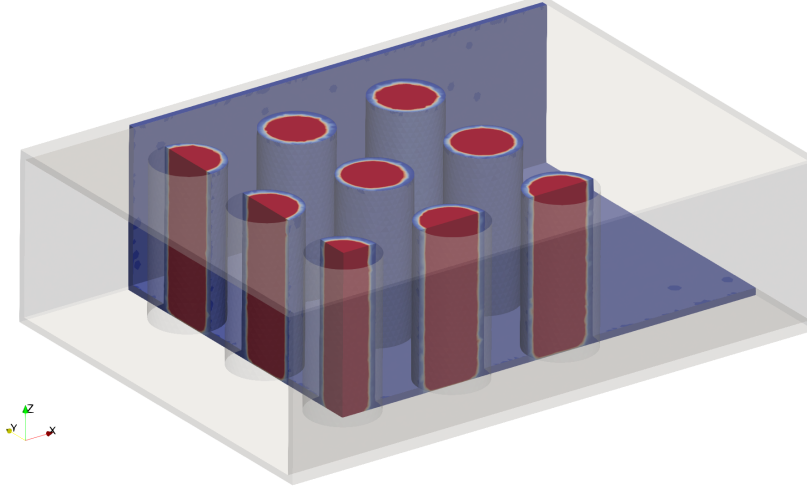


Figure 15: Heat source term (red) added to the battery cells. Some battery cells are cut through for better illustrations.

The inlet flow temperature is set to 300 K. The battery's maximum allowable temperature is 328 K, but applying a safety factor of 1.75 based on the permissible temperature rise results in a maximum temperature constraint ($T_{\max}$) of 317 K. With an inlet velocity of 7.8 m/s, the baseline design meets this feasibility requirement, giving a maximum temperature $T_{\max} \simeq 317$ K. The Reynolds number, calculated using the battery casing diameter, is approximately 1.3×10^4 . Table 6 provides a summary of the CHT optimization problem formulation. The objective function is similar to that used in the U-bend optimization, a weighted sum of pressure loss and battery pack weight, expressed as $f = 0.8 \frac{C_{PL}}{C_{PL_{\text{ref}}}} + 0.2 \frac{W}{W_{\text{ref}}}$. Here, C_{PL} denotes pressure and W represents battery pack weight. Pressure loss is proportional to the volumetric flow rate Q and pump efficiency η , both of which are assumed constant throughout the optimization due to the fixed inlet velocity. This optimization, consistent with the U-bend optimization, performs a single run using weights of 80% for pressure loss and 20% for weight.

The battery casing shape serves as the design variable in this study. All battery cells share an identical shape, which remains constant along the spanwise (z) direction. Each cell's geometry is controlled by 8 FFD points, able to displace in the x and y directions. Symmetry constraints are applied in the y direction for the battery cell shape, reducing the total number of design degrees of freedom to seven. While this limits the design space compared with the U-bend case, the optimization still achieves significant performance improvements, demonstrating that the geometry parameterization captures the essential geometric features that impact performance. In the future, we will allow more design freedom, such as variable geometry along the spanwise (z) direction of the castings, independent battery casting shapes, and variable inlet velocity.

To guarantee realistic design outcomes, six geometric and physical constraints are enforced. A maximum temperature limit is applied to the battery cell surfaces; since downstream cells typically experience higher temperatures, this constraint is only imposed on the cells in the last column. To maintain adequate space within the casing for each battery cell, a radius constraint ensures that the casing thickness remains greater

Table 6: CHT optimization problem formulation for battery pack.

	Function/Variable	Description	Quantity
Minimize	$f = 0.8 \frac{C_{PL}}{C_{PL_{ref}}} + 0.2 \frac{W}{W_{ref}}$	Weighted sum function consisting of pressure loss and battery weight	1
with respect to	$-5 \leq \Delta x \leq 5 \text{ mm}$	FFD shape displacement in the x direction	4
	$-5 \leq \Delta y \leq 5 \text{ mm}$	FFD shape displacement in the y direction	3
	Total Design Variables		7
subject to	$T_{max} < 317 \text{ K}$	Max temperature constraint	1
	$\theta_{fluid} \leq 75^\circ$	Max mesh non-orthogonality for fluid	1
	$\mu_{fluid} \leq 5$	Max mesh skewness for fluid	1
	$\theta_{solid} \leq 75^\circ$	Max mesh non-orthogonality for solid	1
	$\mu_{solid} \leq 5$	Max mesh skewness for solid	1
	$r \geq 10 \text{ mm}$	Min radius constraint	1
	Total Constraint Functions		6

than 1 mm throughout the optimization process. Additionally, due to the complexity of the mesh, four mesh quality constraints are implemented for both the fluid and solid domains.

Table 7: Summary of objective functions and constraints for the baseline and optimized designs.

	Pressure loss (Pa)	Battery weight (Kg)	Max temperature (K)
Baseline	90.2	1.1	316.93
Optimized	50.0	1.084	317.07
Difference	↓ 44.6%	↓ 1.5%	↑ 0.04%

Fig. 16 presents the optimization history for both the objective function and the maximum temperature constraint. The objective function decreases by 36% while satisfying the temperature constraint. The optimization was carried out over 20 iterations and lasted for 7 hours, using 192 processor cores (Intel Xeon 8160 CPU). Table 7 summarizes the objective function values and constraint satisfaction for both the baseline and optimized designs, providing a detailed breakdown of the improvements achieved. Reductions are observed in both pressure loss and battery weight; however, the pressure loss decreases significantly more (44.6%), which aligns with its higher weighting in the objective function.

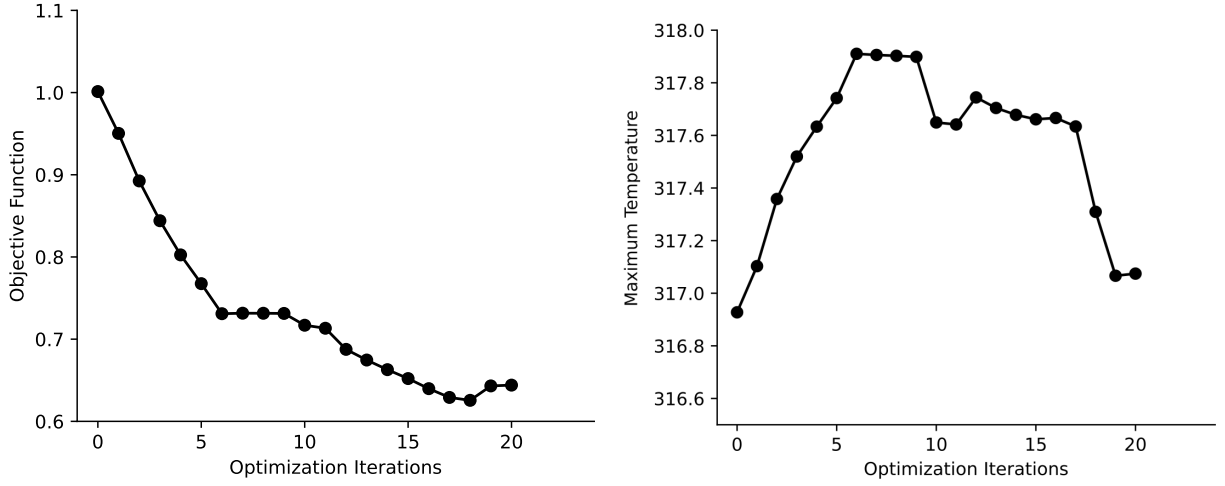


Figure 16: Optimization history for the weighted sum objective function (left) and the maximum temperature constraint, in Kelvin (right).

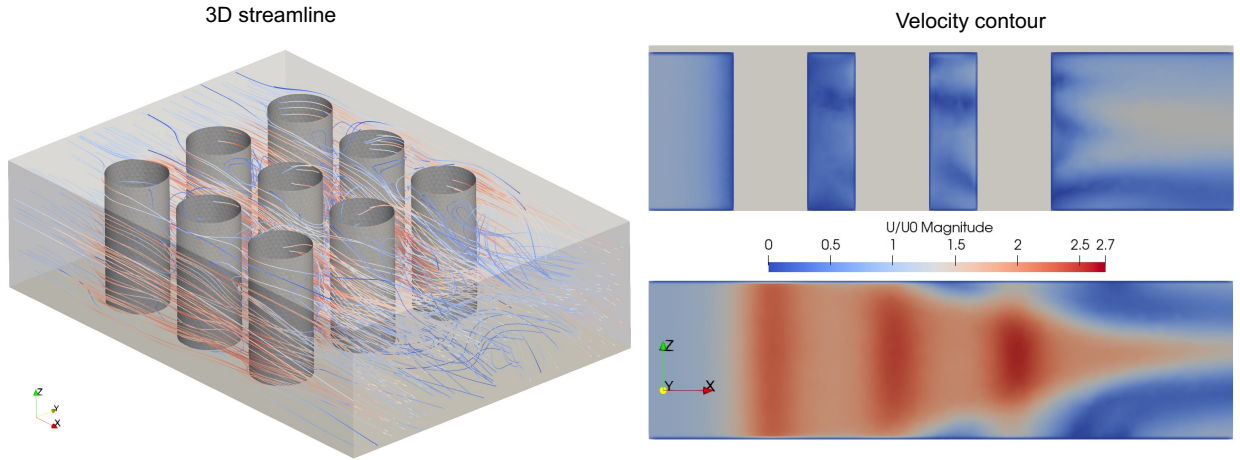


Figure 17: Left: 3D streamline of flow over the battery pack (baseline design). Right: velocity slices at different lateral locations (top: slice cut through the battery cell center, bottom: slice between the battery cells).

3.2.2. Aerothermal analysis for the baseline and optimized designs

To gain insight into the mechanisms driving performance improvements, the flow fields of the baseline and optimized designs are compared. Fig. 17 illustrates 3D streamlines colored by velocity magnitude, along with velocity contours at various lateral cross-sections. As depicted in Fig. 17 (left), the flow accelerates through the narrow gaps between battery cells and subsequently separates, producing downstream wake vortices. These vortices interact with the top and bottom walls and evolve into complex three-dimensional vortex structures farther downstream. This behavior is more apparent in Fig. 17 (bottom-right), where regions of low-speed, recirculating flow are observed near the top and bottom boundaries.

Fig. 18 presents velocity contours at various spanwise locations for both the baseline and optimized designs. In the baseline case, a large velocity region is observed downstream of the third column of battery

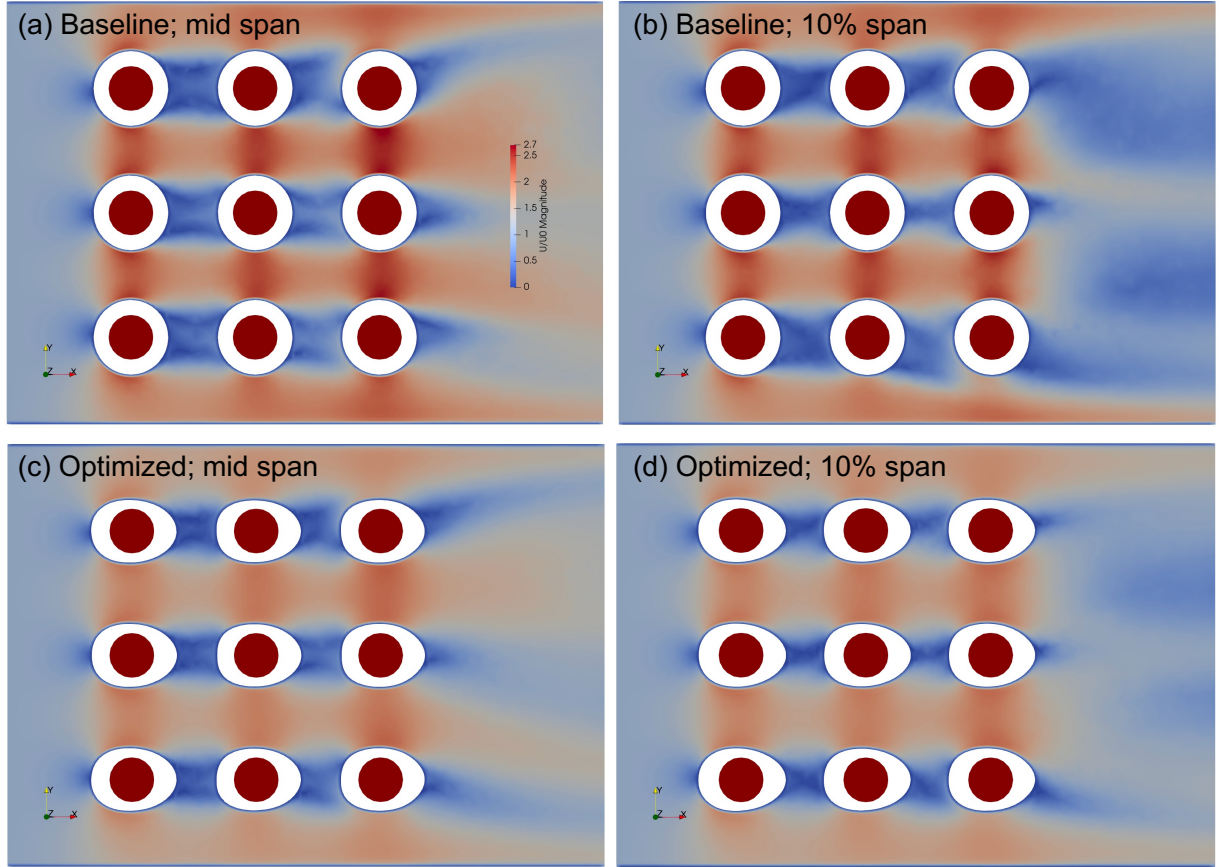


Figure 18: Velocity contours at different spanwise locations. Left: 10% of the span. Right: 50% of the span. Top: baseline design. Bottom: Optimized design. The red circles represent the battery cells.

cells at mid-span. However, near the bottom wall (at 10% span), a significantly lower velocity region appears, attributed to wall effects from the top and bottom surfaces. The optimized design introduces a more streamlined battery casing geometry, effectively diminishing the extent of the wake regions at both spanwise locations. These smaller, weaker wakes contribute to a reduction in overall pressure loss. This observation aligns with prior findings, such as the battery pack optimization study by Afzal et al. [65], in which the objective function combined average Nusselt number, maximum battery temperature, and fluid friction coefficient. Among several design parameters, battery cell spacing was varied. It was found that increasing the spacing led to a more streamlined velocity field and a reduction in the friction coefficient, which is directly related to pressure loss. The outcome is comparable to this current study, where a shape optimization achieved a similar streamlined flow field. In both studies, the CFD agreed on an optimal velocity profile, only differing in method.

As observed in the U-bend optimization case, the objective function, particularly pressure loss, and the maximum temperature constraint are competing with each other ([2, 3, 28, 29, 52]). The optimizer tends to favor a thin, streamlined battery casing geometry to reduce both pressure loss and overall weight. However,

such a design inherently reduces the available cooling surface area, potentially leading to elevated battery cell temperatures. To balance these competing effects, the optimizer identifies a battery casing shape that is as aerodynamic and lightweight as possible, while still maintaining temperatures within the specified limit.

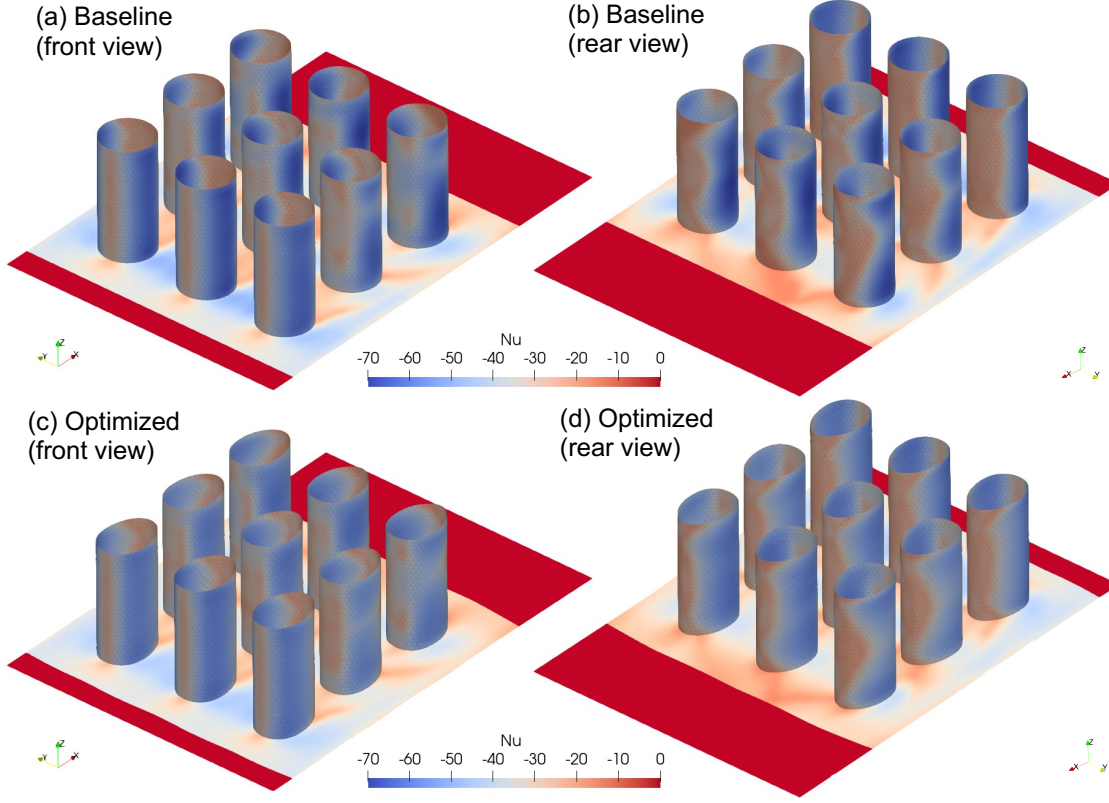


Figure 19: The Nusselt number contours on the battery casing surfaces and bottom walls. Top: baseline design. Bottom: Optimized design. The left sub-figures show a view from the front (flow entrance), and the right sub-figures show a view from the rear (flow exit).

The resulting velocity distribution leads to a notable Nusselt number pattern, as shown in Fig. 19. Similar to the U-bend pipe case, the Nusselt number is computed as $Nu = QD/(k\Delta T)$, where D represents the battery cell diameter and $\Delta T = 17$ K is the reference temperature difference between the battery surface and the freestream.

Bisoi et al. [17] conducted a CHT analysis of flow through a channel obstructed by cylinders and reported enhanced heat flux near the leading edges of the cylinders as well as along the top and bottom channel walls. Similar behavior is observed in the present study, where two distinct high Nusselt number (high heat flux) regions are identified on the battery surfaces. The first region appears near the leading edge of the battery cylinder, resulting from flow impingement at relatively high velocity. The second occurs near the trailing edge, where increased turbulence intensity from the flow separation enhances heat transfer. High Nusselt number regions are also observed along the bottom wall, specifically in areas downstream of the battery cylinders. This is attributed to strong mixing between wake vortices and the boundary layer flow near

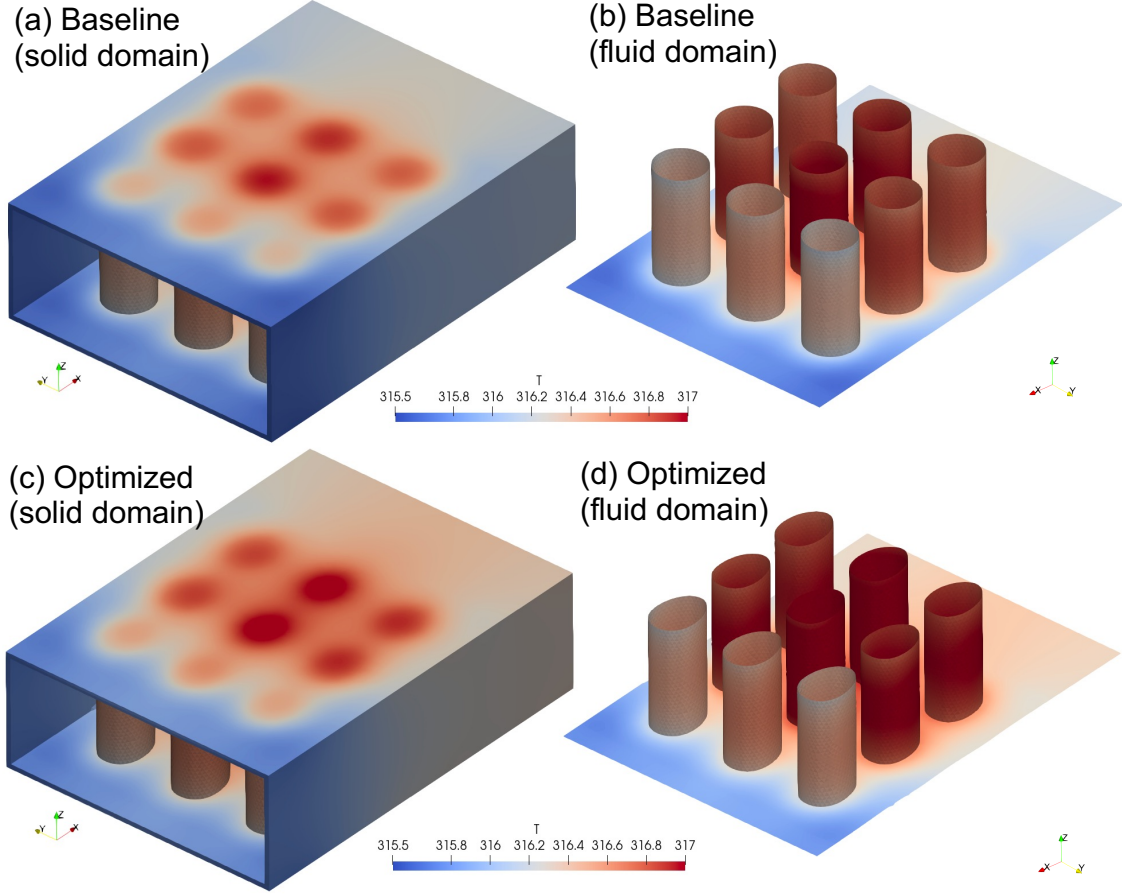


Figure 20: Temperature contours for the fluid and solid domains. Top: baseline design. Bottom: Optimized design. The left sub-figures include the pack enclosure, while the right sub-figures show the cells exposed.

the bottom wall, as previously discussed. Overall, the Nusselt number distributions for the baseline and optimized designs remain similar, which is expected, since the maximum temperature constraint enforces similar heat transfer performance between the two designs.

Fig. 20 shows the temperature contours for both the fluid and solid domains. A temperature difference of roughly 1.5 K is observed between the upstream and downstream regions, which is expected as the cooling air absorbs heat while passing through the battery pack, reducing its cooling effectiveness downstream. As expected, high temperatures are found near the battery cells adjacent to the top wall (see Fig. 17, left), with downstream cells exhibiting higher temperatures compared to those upstream. This trend justifies the maximum temperature constraint setup; the constraint is needed for only a single battery cell in the third row. Although the baseline and optimized designs have similar temperature distributions, the optimized design shows slightly higher overall temperatures, consistent with the results presented in the last column of Table 7.

3.3. Verification of optimized designs using unsteady simulations with high-resolution mesh

The mesh employed in both above case studies is relatively coarse, with an average y^+ value of 24 for the U-bend case and 30 for the battery pack. While this coarseness contributes to faster optimization run times, it inevitably introduces some level of numerical error. To understand how accurately the optimizer captures the design space and the underlying trends in performance improvement using coarse-mesh CFD, a comparative analysis using fine-mesh simulations is presented.

The baseline mesh is deformed into the optimized mesh using the pyGeo and IDWarp packages, as shown in Figs. 23, 24, ensuring that both meshes share identical cell counts and topology. The boundary conditions applied in the fine-mesh simulations are consistent with those used in the coarse-mesh cases, with the exception of the following modifications:

1. Rather than a CHT boundary condition, a constant wall temperature boundary condition is used.
2. The simulations are now unsteady (using pimpleFoam) with a maximum Courant number of 0.8.
3. To discretize the convective term a second order upwind scheme is used.

The fluid-only, constant-wall-temperature analysis can significantly reduce the computational cost compared with an unsteady CHT analysis. However, the main limitation is that it excludes conduction heat transfer, which typically leads to an inaccurate prediction of heat flux. However, in design optimization, we are primarily concerned with relative changes in performance rather than the absolute values. Thus, a fluid-only verification study should still capture the correct trends in design improvement.

3.3.1. U-bend optimization

The mesh density for the U-bend case has been increased to 2.24 million with an average y^+ of 0.35, and a maximum of 1.4. In an effort to save computational time and due to the fact that the far field flow helps heat transfer but does not enhance it between the baseline and optimized designs, only the internal fluid domain is simulated. Fig. 23 (top) shows the outer mesh of the pipe's surface next to a y^+ contour. 2D slices of the internal mesh are shown in Fig. 23 (bottom).

The time series plots for both total heat flux and pressure loss are shown in Fig. 22. The time-averaged values for both pressure loss and total heat flux converged to their respective values around $t = 0.4$ seconds. The baseline and optimized values for pressure loss remain mostly unchanged between the coarse-mesh and fine-mesh CFD simulations. This is easily understood by examining the velocity profile at the mid-plane as shown in Fig. 21. Comparing Fig. 21 to Fig. 8, it can be seen that the two velocity profiles are similar. The main flow feature, the large region of separated turbulent flow just past the inner U-bend, is still present but reduced in intensity for the fine-mesh CFD. This leads to a slightly higher pressure in the outlet leg of the U-bend pipe. The other objective function, total heat flux, has decreased in magnitude due to removing the far field domain and has a smaller performance increase for the fine-mesh CFD over the coarse-mesh CFD. Table 8 summarizes the performance enhancements between the coarse-mesh and fine-mesh CFD simulations.

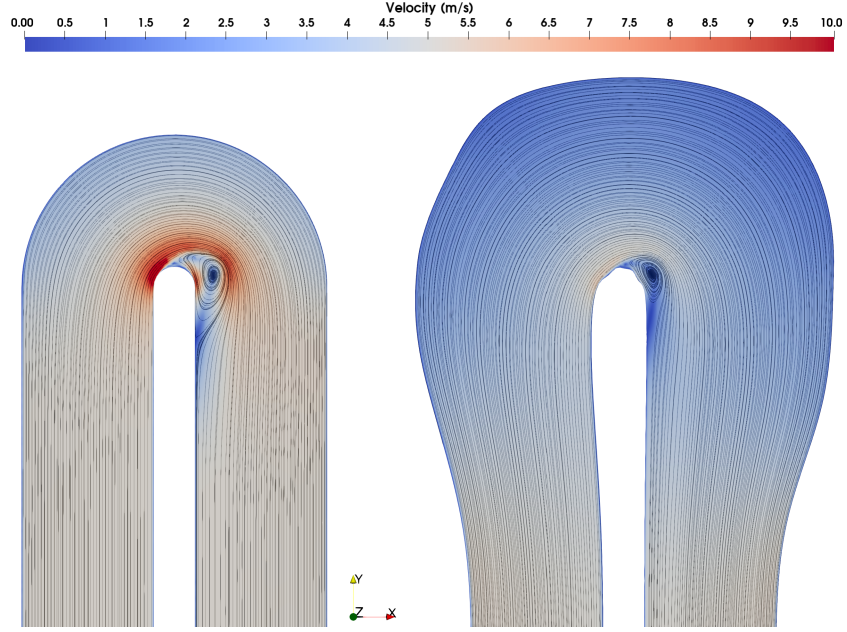


Figure 21: Baseline (left) Vs. optimized geometry (right) internal velocity streamlines at the symmetry plane.

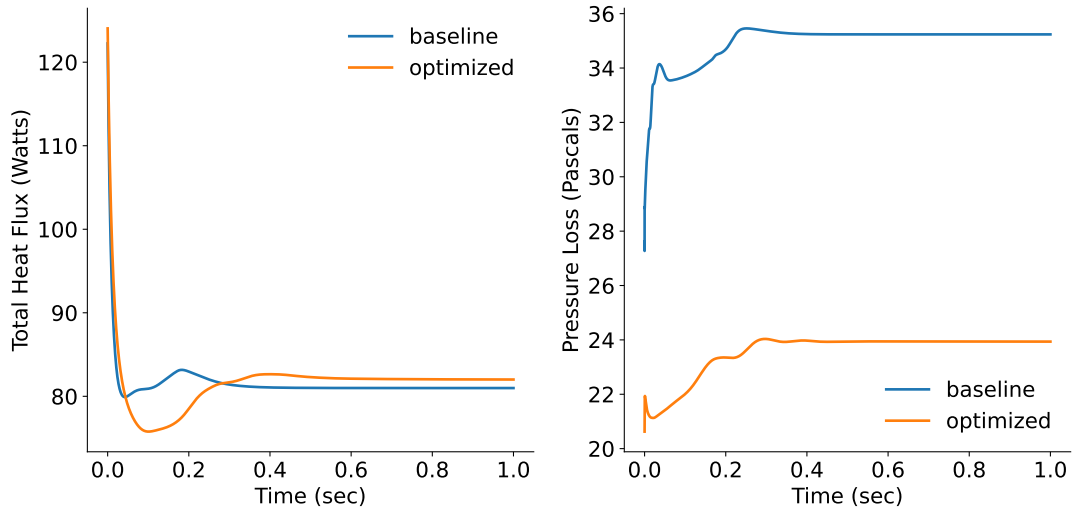


Figure 22: Time-series of heat flux (left) and pressure loss (right) for the baseline and optimized designs (fine mesh).

The fine-mesh CFD predicts the pressure loss to drop by 50.11% as opposed to the 52.71% drop from
 the coarse-mesh CFD. This difference of only 2.6% shows a strong agreement between the fine and course
 results. The total heat flux had a similar error margin with the coarse-mesh CFD giving a 2.29% increase
 in total heat flux, higher than the 1.27% enhancement given from the fine-mesh CFD. This 1.02% difference
 shows a similar trend as pressure loss; the fine-mesh and coarse-mesh CFD simulations are both close in
 value, ultimately validating our results.

Table 8: Comparison of pressure loss and total heat flux for the baseline and optimized designs.

	Pressure loss (Pa)	Total heat flux (W)
Baseline (coarse mesh)	22.31	122.85
Optimized (coarse mesh)	10.55	125.66
Difference	↓ 52.71%	↑ 2.29%
Baseline (fine mesh)	22.17	80.98
Optimized (fine mesh)	11.05	82.009
Difference	↓ 50.11%	↑ 1.27%

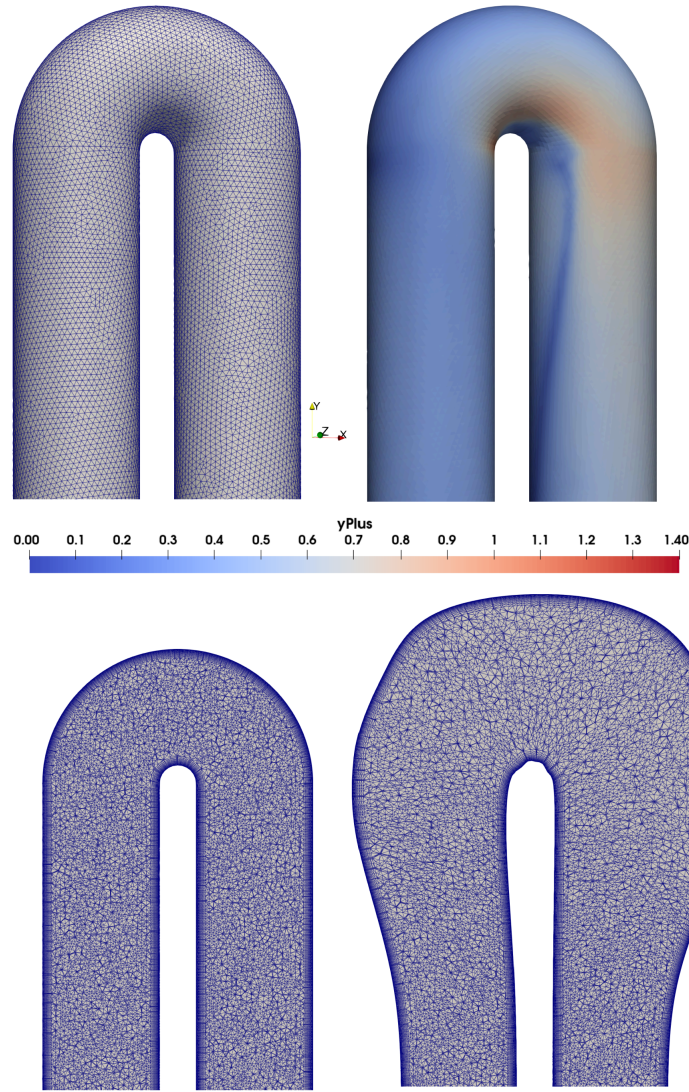


Figure 23: 3D view (top) and 2D view (bottom) of the fine mesh (2.24 million cells)

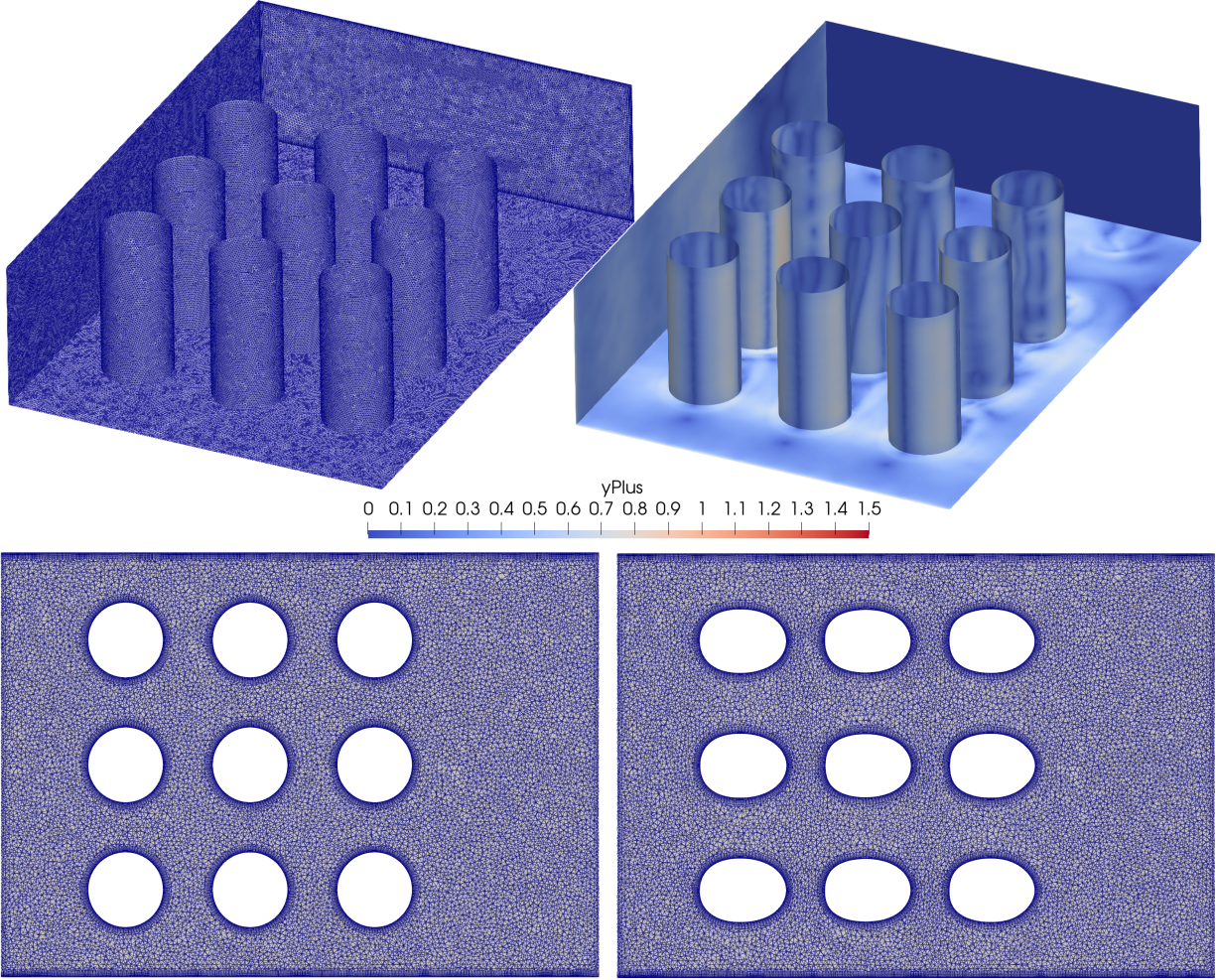


Figure 24: Top: 3D view of the fine mesh (9.7 million cells) along with the y^+ contour. Bottom: 2D mesh slices at the midspan.

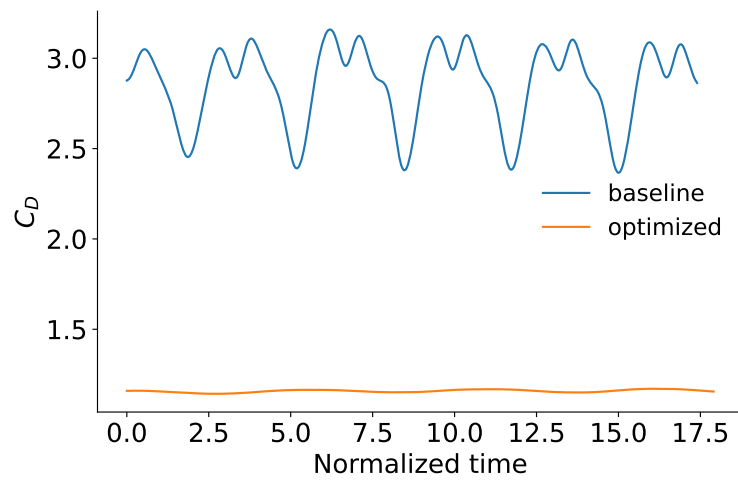


Figure 25: Time-series of drag coefficient for the baseline and optimized designs (fine mesh).

3.3.2. Battery pack optimization

The new mesh density for the battery case is increased to 9.7 million cells. Fig. 24 (top) is the 3D view of the mesh next to the y^+ contour. The maximum y^+ is 0.98 with an average value of 0.35. A slice of the 2D mesh is shown in Fig. 24 (bottom).

Fig. 25 presents the time series of the drag coefficient (C_D) for both the baseline and optimized designs. As expected, the baseline configuration has a significantly higher time-averaged C_D compared to the optimized design. This increased drag is attributed to strong flow unsteadiness caused by vortex shedding behind the battery cells. In contrast, the optimized design shows reduced oscillation in C_D , a result of its more streamlined geometry. This behavior is further shown in the velocity contours at mid-span shown in Fig. 26. For the baseline design, the instantaneous flow field (right column) deviates notably from the mean flow field (left column), whereas the optimized design demonstrates much smaller differences between instantaneous and mean flows.

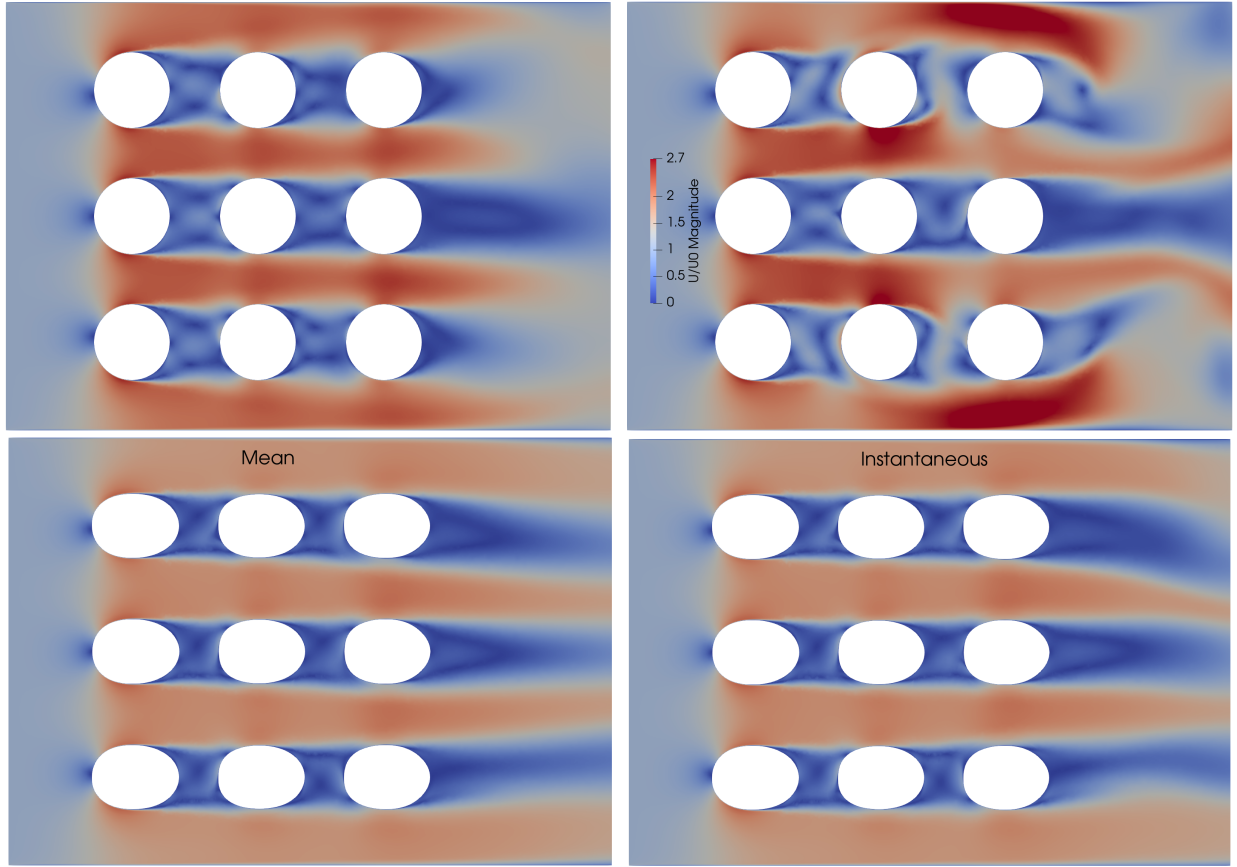


Figure 26: Velocity contour at the midspan for the fine-mesh simulation. Top: baseline design. Bottom: optimized design.

Finally, the pressure loss and total heat flux for the baseline and optimized designs are summarized in Table 9, with results from both the coarse- and fine-mesh simulations shown. The fine-mesh simulation predicts a 9.3% greater pressure loss reduction (the objective function in the CHT optimization) than the

Table 9: Comparison of pressure loss and total heat flux for the baseline and optimized designs.

	Pressure loss (Pa)	Total heat flux (W)
Baseline (coarse mesh)	90.2	89.8
Optimized (coarse mesh)	50.0	89.6
Difference	↓ 44.6%	↓ 0.2%
Baseline (fine mesh)	105.6	107.9
Optimized (fine mesh)	48.7	95.5
Difference	↓ 53.9%	↓ 11.5%

coarse mesh, indicating that the coarse-mesh optimization correctly captures the design improvement trend. In contrast, the coarse-mesh simulation predicts roughly the same total heat fluxes for the baseline and optimized designs, while the fine-mesh simulation reveals an 11.5% reduction in total heat flux for the optimized design. This reduction in heat flux would lead to an increase in battery temperature, rather than remaining the same. The discrepancy in heat flux performance for the optimized design can be attributed to two aspects: (1) The battery pack case has strong flow unsteadiness, which is ignored in our steady-state CHT optimization, and (2) In the verification, we simulate only the fluid domain; the battery casing’s transient thermal resistance is not considered. A detailed analysis of transient CHT flow for the battery cases is beyond the scope of this paper and will be conducted in the future work. Overall, the coarse-mesh CFD introduces approximately 10% error in predicting both pressure loss and total heat flux. This level of error is acceptable, as it falls within the margin of design improvement ($> 40\%$ pressure loss reduction).

4. Summary

This paper introduces a recently developed discrete adjoint-based conjugate heat transfer (CHT) optimization framework, integrated within the DA Foam and OpenMDAO/MPhys frameworks. The most salient feature of the proposed framework is its extensibility, allowing not only efficient aerothermal optimization but also providing a scalable foundation for future multidisciplinary design optimization (MDO) involving additional physics, such as structural mechanics. The CHT simulations employ a finite-volume computational fluid dynamics (CFD) solver for the fluid domain and a conduction heat transfer solver for the solid domain. A mixed-type boundary condition is applied at the solid-fluid interface to ensure accurate coupling of temperature and heat flux between the two domains.

Two case studies are presented to demonstrate the flexibility and effectiveness of the proposed framework. The first case involves the optimization of a U-bend cooling channel, a typical component in aerospace heat exchangers. The objective is to balance pressure loss and total heat transfer by minimizing a weighted sum objective function composed of 90% average outlet temperature and 10% pressure loss, with the U-

bend pipe’s geometry serving as the design variable. The optimized configuration yields a 52.71% reduction in pressure loss while improving total heat flux by 2.29%. These improvements are primarily attributed to greater pipe curvature and reduced internal flow separation, which collectively lower turbulence intensity and velocity peaks, thereby decreasing pressure loss. Additionally, a 16.45% increase in surface area contributes to the observed enhancement in heat transfer.

The second case study focuses on a battery pack cooled by forced convection using ambient air. Here, the battery casing shape is defined as the design variable, and the objective function is a weighted sum of pressure loss and battery pack weight, subject to a maximum temperature constraint. The optimized design results in a 44.6% reduction in pressure loss and a 1.5% reduction in battery weight, while still satisfying the thermal constraint. These gains are achieved through a more aerodynamic casing geometry, which promotes streamlined flow and improved thermal performance.

Coarse-mesh CFD simulations are employed during the optimization process, with their accuracy assessed through comparison with fine-mesh unsteady simulations. The results indicate acceptable error margins for both case studies. For the U-bend optimization, the coarse-mesh predictions deviated by 2.6% in pressure loss and 1.02% in total heat flux. In the battery pack study, errors of 9.3% in pressure loss and 11.3% in total heat flux were observed. These discrepancies remain within the margin of improvement and have minimal impact on the optimal geometries obtained through optimization.

The primary limitation of the shape optimization approach presented in this study is its reliance on a predefined baseline geometry, which constrains the optimizer to explore only smooth geometric variations within the parameterized design space. This inherently limits the range of optimized configurations that can be achieved, particularly for cases where a more radical geometry may be beneficial. In the future, we will extend the DA Foam framework to conduct CHT topology optimization, which removes this geometric constraint and serves as an alternative solution approach well-suited for MDO of unconventional or application-specific geometries. Other improvements include extending the CHT optimization framework to consider transient heat transfer, as the current implementation is limited to steady-state flow.

Collectively, this work highlights the effectiveness of gradient-based CHT optimization in handling complex thermal system management within the DA Foam framework. The developed framework demonstrates strong potential for enabling rapid, high-fidelity design of lightweight, high-performance thermal management solutions for electric aircraft, particularly for emerging platforms such as electric vertical take-off and landing (eVTOL) vehicles.

5. Acknowledgement

This material is based upon work supported by the National Aeronautics and Space Administration Established Program to Stimulate Competitive Research (NASA-EPSCoR) under Grant No. 80NSSC23M0159. This work used the Stampede 3 supercomputer at Texas Advanced Computing Center through allocation ATM-140019 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (AC-

CESS) program, which is supported by U.S. National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296. Portions of this manuscript benefited from the use of artificial intelligence tools, which were employed primarily to improve readability, grammar, and clarity of language. All intellectual content, analysis, and conclusions remain the sole work of the authors.

References

- [1] B. J. Brelje, J. R. Martins, Electric, hybrid, and turboelectric fixed-wing aircraft: A review of concepts, models, and design approaches, *Progress in Aerospace Sciences* 104 (2019) 1–19. doi:10.1016/j.paerosci.2018.06.004, publisher: Elsevier Ltd.
- [2] T. Verstraete, J. Li, Multi-objective optimization of a U-bend for minimal pressure loss and maximal heat transfer performance in internal cooling channels, in: *ASME turbo expo 2013: Turbine technical conference and exposition*, American Society of Mechanical Engineers, 2013.
- [3] K. Y. Kim, Y. M. Lee, Design optimization of internal cooling passage with V-shaped ribs, *Numerical Heat Transfer; Part A: Applications* 51 (2007) 1103–1118. doi:10.1080/10407780601112860.
- [4] T. Verstraete, L. Müller, J.-D. Müller, Adjoint-based design optimisation of an internal cooling channel U-Bend for minimised pressure losses, *International Journal of Turbomachinery, Propulsion and Power* 2 (2017) 10. Publisher: Multidisciplinary Digital Publishing Institute.
- [5] K. Mazaheri, M. Zeinalpour, H. R. Bokaei, Turbine blade cooling passages optimization using reduced conjugate heat transfer methodology, *Applied Thermal Engineering* 103 (2016) 1228–1236. Publisher: Elsevier.
- [6] T. Dbouk, A review about the engineering design of optimal heat transfer systems using topology optimization, *Applied Thermal Engineering* 112 (2017) 841–854. doi:10.1016/j.applthermaleng.2016.10.134.
- [7] V. Subramaniam, T. Dbouk, J.-L. Harion, Topology optimization of conjugate heat transfer systems: A competition between heat transfer enhancement and pressure drop reduction, *International Journal of Heat and Fluid Flow* 75 (2019) 165–184. doi:10.1016/j.ijheatfluidflow.2019.01.002.
- [8] A. Liu, G. Wang, D. Wang, X. Peng, H. Yuan, Study on the thermal and hydraulic performance of fin-and-tube heat exchanger based on topology optimization, *Applied Thermal Engineering* 197 (2021) 117380. doi:10.1016/j.applthermaleng.2021.117380.
- [9] X. Xiang, R. Yang, H. Sun, D. Hoffman, Performance improvement of a hybrid battery thermal management system with pseudo-3D topology optimization, *International Journal of Heat and Mass Transfer* 240 (2025) 126632. doi:10.1016/j.ijheatmasstransfer.2024.126632.

- [10] C. Sun, W. Wang, X.-W. Tian, X. Zeng, S.-H. Qian, Y.-Z. Cai, X.-H. Wang, Thermal design of composite cold plates by topology optimization, *International Journal of Mechanical Sciences* 259 (2023) 108594. doi:10.1016/j.ijmecsci.2023.108594.
- [11] P. Meliga, W. Abdel Nour, D. Laboureur, D. Serret, E. Hachem, Multi-objective topology optimization of conjugate heat transfer using level sets and anisotropic mesh adaptation, *Fluids* 9 (2024).
- [12] J.-W. Luo, L. Chen, Y. Xia, X. Zheng, W.-Q. Tao, Topology optimization of natural convection using porous metal foam based on the adjoint lattice boltzmann method and level set method, *Computers Fluids* 265 (2023) 106007. doi:10.1016/j.compfluid.2023.106007.
- [13] T. Verstraete, S. Scholl, Stability analysis of partitioned methods for predicting conjugate heat transfer, *International Journal of Heat and Mass Transfer* 101 (2016) 852–869. doi:j.ijheatmasstransfer.2016.05.041.
- [14] J. Anibal, C. A. Mader, J. R. R. A. Martins, Aerodynamic shape optimization of an electric aircraft motor surface heat exchanger with conjugate heat transfer constraint, *International Journal of Heat and Mass Transfer* 189 (2022) 122689. doi:10.1016/j.ijheatmasstransfer.2022.122689.
- [15] J. S. Kim, S. H. Seo, C. G. Ban, K. H. Kim, Conjugate heat transfer analysis of a ram/scramjet with thermal decomposition of the regenerative cooling channel, *International Journal of Heat and Mass Transfer* 244 (2025) 126901. doi:10.1016/j.ijheatmasstransfer.2025.126901.
- [16] M. Chávez-Modena, L. M. González, E. Valero, Numerical optimization of the fin shape experiments of a heat conjugate problem surface air/oil heat exchanger (SACOC), *International Journal of Heat and Mass Transfer* 182 (2022) 121971. doi:10.1016/j.ijheatmasstransfer.2021.121971.
- [17] R. Bisoi, S. Jogee, K. Anupindi, Conjugate heat transfer analysis of flow over an array of cylinders placed in a parallel channel, *International Journal of Heat and Mass Transfer* 239 (2025) 126541. doi:10.1016/j.ijheatmasstransfer.2024.126541.
- [18] Y. Yuan, J. Cao, Z. Zhang, Z. Xiao, X. Wang, Numerical simulation and multi-objective optimization design of conjugate heat transfer in novel double shell-passes multi-layer helically coiled tubes heat exchangers, *Case Studies in Thermal Engineering* 62 (2024) 105155. doi:10.1016/j.csite.2024.105155.
- [19] T. Hickling, L. He, A multiscale framework for unsteady conjugate heat transfer with turbulence resolving methods — with application to rotating cavities, *International Journal of Heat and Fluid Flow* 103 (2023) 109174. doi:10.1016/j.ijheatfluidflow.2023.109174.
- [20] A. Masoud Ali, A. Rona, M. Angelino, Numerical investigation of various twisted tapes enhancing a circular microchannel heat sink performance, *International Journal of Heat and Fluid Flow* 98 (2022) 109065. doi:10.1016/j.ijheatfluidflow.2022.109065.

- [21] B. Zhao, Y. Zhou, C. Ding, S. Dong, A fast fluid–solid coupled heat transfer algorithm based on dividing the development stages of the flow field, *International Journal of Heat and Fluid Flow* 107 (2024) 109387. doi:10.1016/j.ijheatfluidflow.2024.109387.
- [22] F. Cai, H. Zhou, M. Yao, Z. Ren, An accelerated time-scale decomposition algorithm for transient thermal fluid-solid coupled simulations in multi-medium systems, *International Journal of Heat and Fluid Flow* 107 (2024) 109364. doi:10.1016/j.ijheatfluidflow.2024.109364.
- [23] C. Du, Z. Chen, W. Li, Z. Gao, Studies on acceleration algorithms for numerical simulations of hypersonic conjugate heat transfer problems, *Computers & Fluids* 275 (2024) 106260. doi:10.1016/j.compfluid.2024.106260.
- [24] G. Narváez, E. Lamballais, E. Schettini, Simulation of turbulent flow subjected to conjugate heat transfer via a dual immersed boundary method, *Computers & Fluids* 229 (2021) 105101. doi:10.1016/j.compfluid.2021.105101.
- [25] Q. He, W. Zhao, Z. Chi, S. Zang, Application of deep-learning method in the conjugate heat transfer optimization of full-coverage film cooling on turbine vanes, *International Journal of Heat and Mass Transfer* 195 (2022) 123148. doi:10.1016/j.ijheatmasstransfer.2022.123148.
- [26] J. Lee, S. Shin, H. Choi, A. Lee, B. Park, S. Lee, Extended multiphysics-informed neural network for conjugate heat transfer problems, *International Journal of Heat and Mass Transfer* 246 (2025) 127098. doi:10.1016/j.ijheatmasstransfer.2025.127098.
- [27] M.-P. Errera, Advanced numerical methods for conjugate heat transfer problems, *Computers & Fluids* 292 (2025) 106594. doi:10.1016/j.compfluid.2025.106594.
- [28] H.-M. Kim, K.-Y. Kim, Design optimization of rib-roughened channel to enhance turbulent heat transfer, *International Journal of Heat and Mass Transfer* 47 (2004) 5159–5168. Publisher: Elsevier.
- [29] Y.-T. Yang, H.-W. Tang, C.-J. Wong, Numerical simulation and optimization of turbulent fluids in a three-dimensional angled ribbed channel, *Numerical Heat Transfer, Part A: Applications* 70 (2016) 532–545. Publisher: Taylor & Francis.
- [30] K. T. Gkaragkounis, E. M. Papoutsis-Kiachagias, K. C. Giannakoglou, The continuous adjoint method for shape optimization in Conjugate Heat Transfer problems with turbulent incompressible flows, *Applied Thermal Engineering* 140 (2018) 351–362. doi:10.1016/j.applthermaleng.2018.05.054, publisher: Elsevier.
- [31] E. Gallorini, J. Hèlie, F. Piscaglia, A multi region adjoint-based solver for topology optimization in conjugate heat transfer problems, *Computers & Fluids* 266 (2023) 106042. doi:10.1016/j.compfluid.2023.106042.

- [32] O. Imam-Lawal, T. Verstraete, J.-D. Müller, Discrete adjoint for coupled conjugate heat transfer, *Structural and Multidisciplinary Optimization* 66 (2023) 107. doi:10.1007/s00158-023-03549-1.
- [33] J. S. Gray, J. T. Hwang, J. R. R. A. Martins, K. T. Moore, B. A. Naylor, OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization, *Structural and Multidisciplinary Optimization* 59 (2019) 1075–1104. doi:10.1007/s00158-019-02211-z.
- [34] A. Yildirim, K. E. Jacobson, J. L. Anibal, B. K. Stanford, J. S. Gray, C. A. Mader, J. R. R. A. Martins, G. J. Kennedy, MPhys: a modular multiphysics library for coupled simulation and adjoint derivative computation, *Structural and Multidisciplinary Optimization* 68 (2025) 15. doi:10.1007/s00158-024-03900-0.
- [35] T. Verstraete, The VKI U-Bend optimization test case, Technical Report, 2016. URL: <http://aboutflow.sems.qmul.ac.uk/events/munich2016/benchmark/testcase1/>.
- [36] DAFoam documentation, 2022. URL: <https://dafoam.github.io>.
- [37] S. A. Zoppelt, H. Koyuncuoglu, P. He, High-fidelity aerostructural optimization benchmark for aircraft propellers in hover, *AIAA SCITECH 2024 Forum* (2024). URL: <https://arc.aiaa.org/doi/abs/10.2514/6.2024-2773>. doi:10.2514/6.2024-2773.
- [38] H. U. Koyuncuoglu, P. He, Simultaneous wing shape and actuator parameter optimization using the adjoint method, *Aerospace Science and Technology* 130 (2022) 107876. Publisher: Elsevier.
- [39] K. T. Tran, P. He, Unsteady aerodynamic optimization of airfoils considering shape and propeller parameters, *AIAA AVIATION 2021 FORUM* (2021). URL: <https://arc.aiaa.org/doi/abs/10.2514/6.2021-3033>. doi:10.2514/6.2021-3033. arXiv:<https://arc.aiaa.org/doi/pdf/10.2514/6.2021-3033>.
- [40] Z. Li, L. Fang, A. Sharma, P. He, Field inversion machine learning for time-resolved unsteady flows in airfoil dynamic stall, *arXiv preprint arXiv:2511.18276* (2025).
- [41] Dafoam tutorial: Conjugate heat transfer topology optimization of a channel, 2025. URL: https://dafoam.github.io/docs/mydoc_tutorial_topo_channel_cht.html.
- [42] G. K. W. Kenway, G. J. Kennedy, J. R. R. A. Martins, A CAD-Free approach to high-fidelity aerostructural optimization, in: *Proceedings of the 13th AIAA/ISSMO multidisciplinary analysis optimization conference*, Fort Worth, TX, 2010. doi:10.2514/6.2010-9231.
- [43] P. E. Gill, W. Murray, M. A. Saunders, SNOPT: An SQP algorithm for large-scale constrained optimization, *SIAM Review* 47 (2005) 99–131. doi:10.1137/S0036144504446096, publisher: SIAM.

- [44] N. R. Secco, G. K. W. Kenway, P. He, C. A. Mader, J. R. R. A. Martins, Efficient mesh generation and deformation for aerodynamic shape optimization, *AIAA Journal* 59 (2021) 1151–1168. doi:10.2514/1.J059491.
- [45] P. He, C. A. Mader, J. R. R. A. Martins, K. J. Maki, An aerodynamic design optimization framework using a discrete adjoint approach with OpenFOAM, *Computers & Fluids* 168 (2018) 285–303. doi:10.1016/j.compfluid.2018.04.012.
- [46] P. He, C. A. Mader, J. R. R. A. Martins, K. J. Maki, DAfoam: An open-source adjoint framework for multidisciplinary design optimization with OpenFOAM, *AIAA Journal* 58 (2020) 1304–1319. doi:10.2514/1.J058853.
- [47] G. K. Kenway, C. A. Mader, P. He, J. R. Martins, Effective adjoint approaches for computational fluid dynamics, *Progress in Aerospace Sciences* 110 (2019) 100542. doi:10.1016/j.paerosci.2019.05.002, publisher: Pergamon.
- [48] A. B. Lambe, J. R. R. A. Martins, Extensions to the design structure matrix for the description of multidisciplinary design, analysis, and optimization processes, *Structural and Multidisciplinary Optimization* 46 (2012) 273–284. doi:10.1007/s00158-012-0763-y.
- [49] B. Pacini, M. Prajapati, K. Duraisamy, J. R. Martins, P. He, Towards mixed-fidelity aero-structural-acoustic optimization for urban air mobility vehicle design, in: *AIAA AVIATION 2023 forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2023. doi:10.2514/6.2023-3905.
- [50] P. He, H. Koyuncuoglu, H. Hu, A. Dhulipalla, H. Hu, H. Hu, High-fidelity aerodynamic and aerostuctural optimization of UAV propellers using the adjoint method, in: *AIAA SCITECH 2023 forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2023. doi:10.2514/6.2023-0531.
- [51] H. Koyuncuoglu, P. He, CFD based multi-component aerodynamic optimization for wing propeller coupling, in: *AIAA SCITECH 2023 forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2023. doi:10.2514/6.2023-1844.
- [52] P. He, J. R. R. A. Martins, C. A. Mader, K. Maki, Aerothermal optimization of a ribbed U-Bend cooling channel using the adjoint method, *International Journal of Heat and Mass Transfer* 140 (2019) 152–172. doi:10.1016/j.ijheatmasstransfer.2019.05.075.
- [53] P. He, C. A. Mader, J. R. R. A. Martins, K. J. Maki, Aerothermal optimization of internal cooling passages using a discrete adjoint method, in: *AIAA/ASME joint thermophysics and heat transfer conference*, Atlanta, GA, 2018. doi:10.2514/6.2018-4080.
- [54] FUNtoFEM code, 2022. URL: <https://github.com/smdogroup/funtofem>.

- [55] P. Spalart, S. Allmaras, A one-equation turbulence model for aerodynamic flows, in: 30th aerospace sciences meeting and exhibit, 1992. doi:10.2514/6.1992-439.
- [56] C. M. Rhie, W. L. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, *AIAA Journal* 21 (1983) 1525–1532. doi:10.2514/3.8284.
- [57] J. R. R. A. Martins, J. T. Hwang, Review and unification of methods for computing derivatives of multidisciplinary computational models, *AIAA Journal* 51 (2013) 2582–2599. doi:10.2514/1.J052184.
- [58] S. Balay, K. Buschelman, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, H. Zhang, PETSc Web page, 2009.
- [59] S. Willeke, T. Verstraete, Adjoint optimization of an internal cooling channel U-bend, in: ASME turbo expo 2015: Turbine technical conference and exposition, American Society of Mechanical Engineers, 2015, pp. V05AT11A029–V05AT11A029.
- [60] Z. Lyu, G. K. W. Kenway, J. R. R. A. Martins, Aerodynamic shape optimization investigations of the Common Research Model wing benchmark, *AIAA Journal* 53 (2015) 968–985. doi:10.2514/1.J053318.
- [61] J. Park, B. G. Knight, Y. Liao, M. Mangano, B. Pacini, K. J. Maki, J. R. Martins, J. Sun, Y. Pan, Cfd-based design optimization of ducted hydrokinetic turbines, *Scientific Reports* 13 (2023) 17968. doi:10.1038/s41598-023-43724-4.
- [62] P. Qin, J. Sun, X. Yang, Q. Wang, Battery thermal management system based on the forced-air convection: A review, *ETransportation* 7 (2021) 100097. doi:10.1016/j.etrans.2020.100097.
- [63] D. Chalise, K. Shah, R. Prasher, A. Jain, Conjugate heat transfer analysis of thermal management of a li-ion battery pack, *Journal of Electrochemical Energy Conversion and Storage* 15 (2018) 011008. doi:10.1115/1.4038258.
- [64] I. Mokashi, A. Afzal, Q. Al-Mdallal, L. S. Sundar, S. A. Khan, N. A. Abdullah, M. H. Azami, C. A. Saleel, Effect of non-conjugate and conjugate condition on heat transfer from battery pack, *Alexandria Engineering Journal* 61 (2022) 3131–3145. doi:10.1016/j.aej.2021.08.042.
- [65] A. Afzal, A. M. Samee, R. Jilte, M. T. Islam, A. M. Manokar, K. A. Razak, Battery thermal management: An optimization study of parallelized conjugate numerical analysis using cuckoo search and artificial bee colony algorithm, *International Journal of Heat and Mass Transfer* 166 (2021) 120798. doi:10.1016/j.ijheatmasstransfer.2020.120798.

Appendix A. Effects of objective weights on U-bend heat exchanger optimization

The U-bend heat exchanger optimization presented in this study had one weighting scheme: 10% weight for pressure loss performance and 90% weight for heat flux. To show how the optimizer adjusts the design

given various weighting schemes, we present here the same U-bend optimization for a heat flux weight of 10%. Fig. A.27 shows the optimized U-bend geometry for these cases, where each U-bend is denoted by its percent weight for heat flux. The 10% weight takes on a different shape, having no protrusion on the outer portion of the inlet leg and an overall smoother bend section to the channel, which generates a more even velocity and pressure profile. In comparison to the 90% weight channel, the 10% weight channel exhibits a slimmer design to further reduce friction drag due to its smaller surface area.

Fig. A.28 shows the velocity contours and flow streamlines. The 10% weight has a more even velocity profile, with no large areas of flow separation, compared with the 90% weight. This is in part due to the dramatic increase in the inner radius of curvature for the 10% weight. Although this radius of curvature was still increased over the baseline design for the 90% weight, it was kept slimmer, which creates a tighter bend and therefore more flow separation.

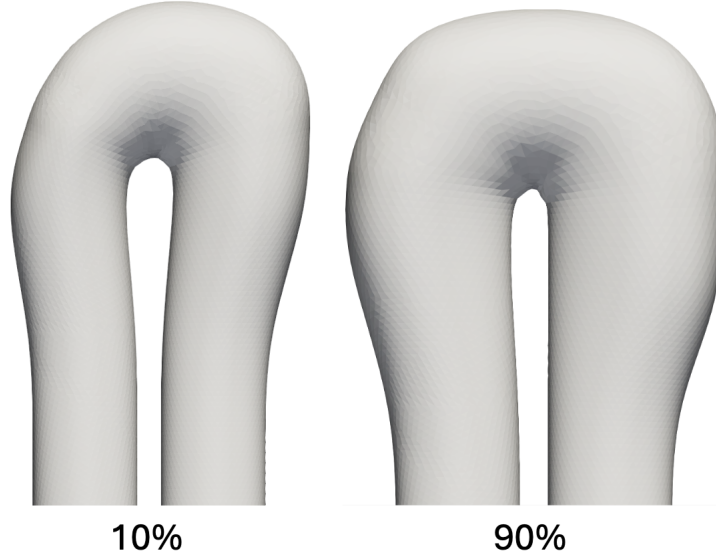


Figure A.27: Optimized U-bend heat exchanger for various percentage heat flux weights.

The performance comparison for these U-bend pipes is summarized in Table A.10. The 10% weight offers a greater decrease in pressure loss, as expected, but starts to degrade the heat transfer performance, decreasing by 0.95%. The pressure loss decrease is close to that of the 90% weight, indicating the upper bound of pressure loss performance gain.

Table A.10: Summary of objective functions for the baseline and optimized designs with various weighting schemes.

	Pressure loss (Pa)	Total heat flux (W)
10%	↓ 57.83%	↓ 0.95%
90%	↓ 52.71%	↑ 2.29%

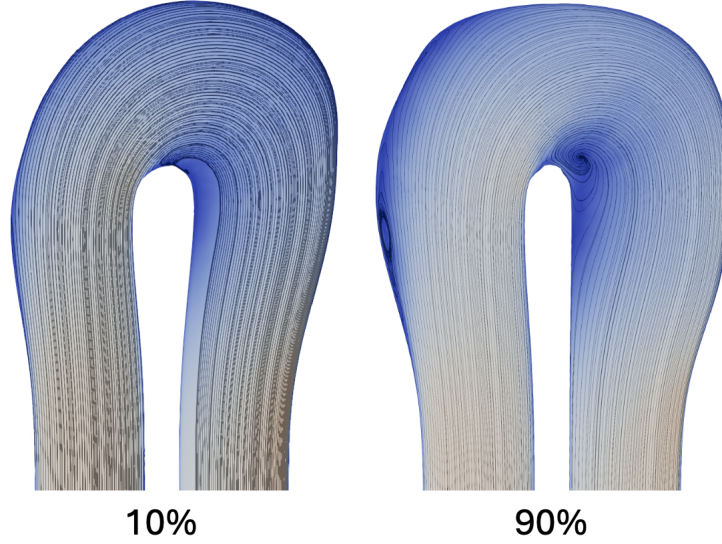


Figure A.28: Internal streamlines for various heat flux weights on U-bend optimization.

Appendix B. Mesh convergence study

To quantify discretization error in the numerical simulations, a mesh-convergence study was performed for the U-bend case (Fig. B.29). A similar study was not conducted for the battery-pack case because a fine mesh leads to strong flow unsteadiness and significant oscillations in the objective functions. Note that this behavior is the main reason we used a coarse mesh for our CHT optimization.

The three meshes that were tested have 0.77 million cells (0.68 and 0.09 million cells for the fluid and solid domains, respectively), 2.54 million (2.3 and 0.24 million cells for the fluid and solid domains, respectively), and 4.3 million (3.8 and 0.5 million cells for the fluid and solid domains, respectively). From Fig. B.29, the heat transfer performance is slightly over-predicted for coarser meshes, with the Nusselt number reaching nearly 89 at 0.77 million mesh cells. The Nusselt number begins to level off once the mesh density reaches 2 million, where the Nusselt number decreases by roughly 1 when increasing the mesh density from 2.54 million cells to 4.3 million cells. The pressure loss is less sensitive to the mesh density. From 0.77 million cells to 4.3 million, the change in pressure loss is roughly 1 Pascal.

Appendix C. Verification of adjoint derivatives

To verify the accuracy of our discrete adjoint approach, we compare several adjoint derivatives for pressure loss (dC_{PL}/dx_i) and mean outlet temperature (dT_M/dx_i) to their respective finite difference approximations, shown in Table C.11. We use a relatively large step size of 2×10^{-3} to account for the oscillation of the objective function due to flow unsteadiness. The derivatives are numbered, given as d/dx_i where i represents the index of the design variable. The derivatives tested are from the optimized U-bend channel presented in Sec. 3.1. The average error for the pressure loss derivatives is 0.937% and for the mean outlet temperature,

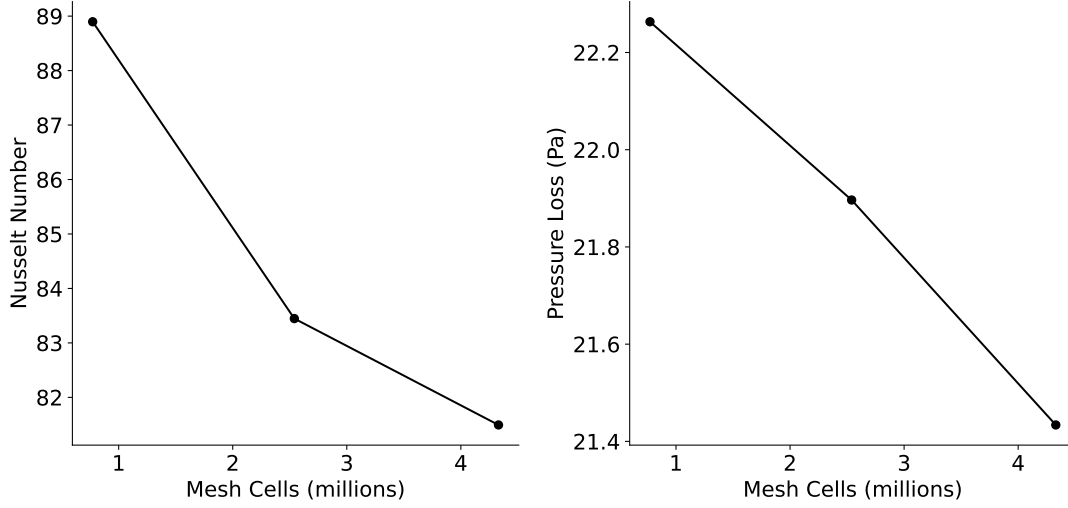


Figure B.29: Mesh convergence study showing Nusselt number and pressure loss as a function of mesh cells.

922 1.8%. The slightly higher average error for the mean outlet temperature derivatives is attributed to the
 923 complex coupling between the fluid and solid domains. However, the error between the finite difference
 924 approach and our discrete adjoint approach remains well within an acceptable error margin, supporting the
 925 robust CHTo optimization of our framework.

Table C.11: Verification of the adjoint accuracy using finite difference approximations

Quantity	dC_{PL}/dx_1	dC_{PL}/dx_2	dC_{PL}/dx_3	dC_{PL}/dx_4	dC_{PL}/dx_5	dC_{PL}/dx_6
Adjoint	-7.893	-5.535	-15.761	-12.678	-4.727	-4.622
FD-Ref	-7.788	-5.472	-15.537	-12.786	-4.727	-4.661
%Error	1.348%	1.151%	1.441%	-0.844%	0.004%	-0.836%
Quantity	dT_M/dx_1	dT_M/dx_2	dT_M/dx_3	dT_M/dx_4	dT_M/dx_5	dT_M/dx_6
Adjoint	0.693	0.498	1.211	0.845	0.408	0.456
FD-Ref	0.706	0.478	1.184	0.831	0.408	0.453
%Error	-1.841%	4.184%	2.280%	1.684%	0.164%	0.662%