SIMULTANEOUS PACKING AND ROUTING OPTIMIZATION USING GEOMETRIC PROJECTION

Alex Jessee[1], **Satya R. T. Peddada**[2], **Danny J. Lohan**[2], **James T. Allison**[2], **Kai A. James** [1]
[1]Department of Aerospace Engineering
[2]Department of Industrial and Enterprise Systems Engineering
University of Illinois at Urbana-Champaign, Urbana, IL 61801
Email: {jessee2,speddad2,dlohan2,jtalliso,kaijames}@illinois.edu

Abstract

A new method for optimizing the layout of device-routing systems is presented. Gradient-based topology optimization techniques are used to simultaneously optimize both device locations and routing paths of device interconnects. In addition to geometric considerations, this method supports optimization based on system behavior by including physics-based objectives and constraints. Multiple physics domains are modeled using lumped parameter and finite element models. A geometric projection for devices of arbitrary polygonal shape is developed along with sensitivity analysis. Two thermal-fluid systems are optimized to demonstrate the use of this method.

1 Introduction

Power electronics circuits and fluid cooling systems (and many other types of other engineering systems) are composed of devices that exchange energy, as well as routing (interconnects) that facilitate energy transfer. One reason these systems are difficult to design is that they have many requirements, including: performance, cost, geometry, and volume restrictions. Identification of feasible designs can be exceptionally difficult in applications where space is limited, devices and interconnects involve complicated geometries, and system performance depends on spatial relationships and multiple physics couplings. Current practice relies largely upon human expertise, design rules, and manual design adjustments to solve these problems. This limits both the complexity of systems that can be designed involving non-trivial packing and routing decisions, as well as realization of potentially improved functionality or performance. Many previous efforts have focused on creating design automation methods to address elements of the integrated packing and routing problem described here. In this article, a design automation approach is presented that integrates several of these elements as a step toward more comprehensive solution of 3D packing and routing problems with both geometric and physics considerations.

Automated design methods for aspects of the device-routing layout problem have been developed and studied in the context of several applications, including electronic module layout design [1], vehicle assembly [2], layout of components in additive manufacturing [3], and automotive transmission design [4]. Optimal packaging approaches have incorporated metrics such as mass properties and spatial criteria [5], and have utilized solutions methods such as simulated annealing [6, 7] and pattern search (PS) [8, 9]. Many efforts have addressed the interconnect routing problem specifically, where device layout is held fixed. In addition to creation of general engineering system routing methods [10], application-specific efforts include pipe routing in ship engine rooms [11], aircraft engines (using genetic algorithms [12] or ant colony optimization (ACO) [13]), aerospace system routing [14], electrical wire routing in buildings [15], and oil industry equipment using ACO [16].

The above methods address only routing (geometrically placing) the device interconnects, or only device layout, but not both simultaneously. Also, they do not take into account performance considerations that require physics-based simulation for evaluation when making automated routing decisions.

In these existing methods, performance evaluation of these designs is left to human designers. The amount of time required for a designer to generate a feasible design and analyze its performance limits the ability of engineers to explore these complex design spaces within a constrained project timeline. These strategies can produce feasible designs, but they may not be optimal when considering all the system requirements, and the complexity of systems that can be considered is limited. In current practice, layout and routing problems are solved manually, which severely limits design capabilities for systems involving complex packing and routing tasks (especially in cases with strong physics interactions). In this research, computational methods will be presented that have the potential to generate designs faster, with better system performance, and for higher-complexity systems than those designed with methods that require significant human input.

Topology optimization, defined here as the optimal placement of material in a 2D or 3D geometric domain, does take into account models of physical behavior. This method has been used across a range of engineering domains, including to design structures for maximum stiffness [17], multi-material properties [18], or component geometries for optimal heat conduction properties [19, 20]. Problems that include multiple distinct physics domains have also been studied. De Kruijf *et al.*, Takezawa *et al.* and Kang & James performed optimization studies which included both structural and thermal conduction requirements [21–23]. The aerodynamic shape and internal structure of a wing have been optimized simultaneously [24–26] considering the interaction between aerodynamic loading and structural wing response. Topology optimization has also been used to optimize the placement of components and their supporting structure [27, 28]. This allows sections of specific geometry, such as a pattern of bolt holes, to be distributed optimally within a structure. Designs produced by topology optimization are often infeasible for traditional manufacturing methods (subtractive, formative), but often can be made using additive manufacturing [29]. The design of components that are more easily manufactured using traditional methods motivates the development of methods that optimize designs made from standard material sizes and shapes, typically using ground structure methods [30, 31]. The geometric projection methods in Refs. [32, 33] have also been suggested to optimize structures made from stock materials.

1.1 Objectives and Contributions

The primary objective of this work is to demonstrate the use of gradient-based topology optimization methods in optimal electro-thermal system layout problems. Optimal placement of devices (packing) and connections between devices (interconnect routing) are two separate NP-hard problems. The proposed method combines both the device placement and interconnect routing problems, in addition to using physics-based models for design comparison. The core contributions of this work are as follows:

1. We present a novel technique that supports simultaneous optimization of device placement and interconnect paths, whereas existing methods treat device layout and interconnect routing separately (e.g., optimal routing with fixed layout).

2. Physics-based objectives and constraints were incorporated into the optimization problem, in addition to geometric constraints that prevent interference between devices and interconnects. Both 1D lumped parameter and 2D finite element physics models are used within a single optimization problem to support physics-based evaluation.

3. We use the geometric projection method (GPM) of Norato et. *al* [32], which is an alternative to the well-established SIMP (Solid Isotropic Material with Penalization) method design parameterization [34] for solving the optimization problem.

4. We demonstrated the effectiveness of the proposed method via the solution of two device-routing test cases that utilize physics-based simulations.

The new simultaneous approach makes significant system volume reduction possible. The projection method of Norato et *al*. [32] is extended to allow devices of arbitrary polygonal shape to be projected. Sensitivity analysis for this projection is provided to allow the efficient use of gradient-based optimization methods. Examples presented later in this article consider thermal conduction on the continuum level using the finite element method, and a lumped parameter pipe flow model. The methods presented here, however, could be extended to model other combinations of physics; for example, thermal-electric or structural-fluid systems. Section 2 presents the models used to simulate the physics response of the system. Section 3 states the optimization problem and presents the derived function sensitivities. Finally, the method is demonstrated in Section 4 through the optimization of two device-routing systems.

2 Physics models

2.1 Steady state thermal conduction

Temperature distribution will be modeled on the continuum level using the finite element method. The strong form of the boundary value problem for heat conduction is given by:

$$\nabla \cdot (\boldsymbol{\kappa} \nabla T(\boldsymbol{x})) + Q = 0, \quad \boldsymbol{x} \text{ in } \Omega \tag{1}$$

$$T(\boldsymbol{x}) = T^*, \quad \boldsymbol{x} \text{ on } \Gamma_T \tag{2}$$

$$\boldsymbol{n} \cdot (\boldsymbol{\kappa} \nabla T(\boldsymbol{x})) = q^*, \quad \boldsymbol{x} \text{ on } \Gamma_q, \tag{3}$$

where $\boldsymbol{\kappa}$ is the matrix of thermal conduction coefficients, $T(\boldsymbol{x})$ is the temperature solution field, Q is heat flux per unit volume in the domain, and $\boldsymbol{n}$ is the unit normal to the domain boundary. Temperature, T^*, and heat flux, q^*, boundary conditions are applied on the Γ_T and Γ_q portions of the domain boundary, respectively. Detailed derivation of the finite element equations and implementation can be found in Ref. [35]. Here we will skip to the final equation that solves for temperatures at the nodes of the finite element mesh, which is obtained by discretizing the boundary value problem in Eqns. (1)-(3) using the finite element method.

$$\boldsymbol{K}\boldsymbol{T} = \boldsymbol{P} \tag{4}$$

Equation (4) is solved for the temperature field vector $\boldsymbol{T}$, where $\boldsymbol{K}$ is the global thermal stiffness matrix assembled from element stiffness matrices, $\boldsymbol{k}_{\text{el}}$, defined in Eqn. (5), and $\boldsymbol{P}$ is the global load vector assembled from element load vectors, $\boldsymbol{p}_{\text{el}}$, defined in Eqn. (6).

$$\boldsymbol{k}_{\text{el}} = \int_{\Omega_e} \boldsymbol{B}^T \boldsymbol{\kappa} \boldsymbol{B} d\Omega - \int_{\partial\Omega_h} h \boldsymbol{N} \boldsymbol{N}^T d\partial\Omega_h \tag{5}$$

$$\boldsymbol{p}_{\text{el}} = \int_{\Omega_e} Q \boldsymbol{N} d\Omega + \int_{\partial\Omega_h} h T_{\text{env}} \boldsymbol{N} d\partial\Omega_h. \tag{6}$$

Here, $\boldsymbol{N}$ and $\boldsymbol{B}$ are element shape function and shape function gradients, respectively. These equations also include convection boundary conditions on the $\partial\Omega_h$ portion of the boundary. The temperature of the convecting fluid is T_{env}, and the convection coefficient is h (assumed constant here).

Here the geometric projection method of Norato et *al* [32] is used. In the original projection method work, the new parameterization approach was used to optimize structures while ensuring that the resulting design could be made from stock materials, such as structural beams with standard shapes and sizes. We discovered that this method can be extended beneficially to the combined layout and routing optimization problem. The geometric projection method is used to create routing designs that can be manufactured out of standard circular cross-section pipes. The geometric parameterization involves design variables that facilitate convenient derivation of lumped parameter model (Sec. 2.2) sensitivities. The remainder of this section will give a brief overview of the geometric projection method, and detail changes made for use with the routing and packaging problem.

In the projection method, each element in the mesh is assigned a density parameter ρ_i with a value between zero and one. Solid material corresponds to $\rho_i = 1$, and void material corresponds to $\rho_i = 0$. The material properties for each element stiffness matrix $\boldsymbol{k}_i$ are scaled by $0 \le \rho_i \le 1$. Leaving out the convection boundary condition term, the element stiffness matrix is:

$$\boldsymbol{k}_i = (\rho_{\min} + (1 - \rho_{\min})\rho_i^p) \int_{\Omega_e} \boldsymbol{B}^T \boldsymbol{\kappa} \boldsymbol{B} d\Omega = (\rho_{\min} + (1 - \rho_{\min})\rho_i^p)\boldsymbol{k_0}, \tag{7}$$

where p is a penalization parameter used to penalize intermediate densities between 1 and 0, leading to a projection with less gray area between solid and void. The convection boundary condition term is independent of ρ_i, so can be omitted without loss of generality. If a regular mesh with all elements being the same shape and size is used, then the second form of Eqn. (7) can be applied to reduce computational expense, since the integral term is the same for all elements. A minimum density, $\rho_{\min}$, is enforced to prevent singularity in the global stiffness matrix. For structural problems, the smallest $\rho_{\min}$ that prevents ill-conditioning is used. In thermal problems, however, a physically-meaningful minimum density can be chosen to simulate the thermal conductivity of the surrounding medium, for example, air.

The density of each element is found by projecting geometric shapes onto the mesh. Norato et *al*. proposed bars with rounded ends as a shape which could be projected easily, and the same will be used here. Each bar involves three parameters: segment start and end points $\boldsymbol{x}_0$ and $\boldsymbol{x}_f$, and bar width w (Fig. 1). The parameter for out-of-plane thickness (that was presented in the original formulation) is left out here because the new method presented here requires that bars are not removed. This is important because the bars form a flow network, and bar removal could break flow paths. The signed distance between a bar q and an element with center at $\boldsymbol{p}$ is:

$$\phi_q(d_q(\boldsymbol{x}_{q_0}, \boldsymbol{x}_{q_f}, \boldsymbol{p}), w) = d_q(\boldsymbol{x}_{q_0}, \boldsymbol{x}_{q_f}, \boldsymbol{p}) - \frac{w}{2}, \tag{8}$$

where d_q is the distance between the segment q and point $\boldsymbol{p}$. See Ref. [32] for the distance calculation. A circle of radius r is placed at the element center. The density assigned to each element is the area of the circle covered by the bar divided by total area of the circle—see the shaded area of Fig. 1. The density as a function of signed distance is given by:

$$\rho_q(d_q(\boldsymbol{x}_{q_0}, \boldsymbol{x}_{q_f}, \boldsymbol{p}), r) = \begin{cases} 0 & \phi_q > r \\ \frac{1}{\pi r^2}\left[r^2 \cos^{-1}\left(\frac{\phi_q(d_q)}{r}\right) - \phi_q(d_q)\sqrt{r^2 - \phi_q(d_q)^2}\right] & -r \le \phi_q \le r \\ 1 & \phi_q < -r \end{cases} \tag{9}$$

The radius r determines the width of the grey area projected on to the mesh by the bar. A smaller radius will more accurately represent the bar geometry as a projection of mostly ones and zeros. To ensure that any element which touches a bar has a nonzero density, a radius that circumscribes the square elements is used in this paper. The radius must be less than half of the bar width in order for Eqn.(9) to correctly calculate the area intersected by the circle and bar.

In the combined layout and routing optimization problem, devices must also be included in the finite element analysis model. Devices are approximated as polygonal shapes with straight edges. Each device will be defined by a reference point, $\boldsymbol{c}_d$, and a set of vectors, $\boldsymbol{b}_i$, pointing from the reference point to polygon vertices. The device densities are calculated by first projecting each edge of the polygon as a rounded bar, and then filling in elements inside the polygon with density of 1 (Fig. 2). Densities for each

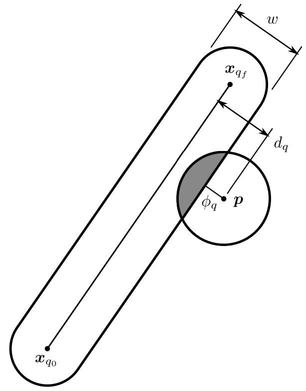


FIGURE 1: Bar projection

FIGURE 2: Device projection

edge $\tilde{\rho}_e$ are calculated using Eqn. (9). End points of edge segments, $\boldsymbol{x}_{e_0}$ and $\boldsymbol{x}_{e_f}$, are found by:

$$\boldsymbol{x}_{e_0} = \boldsymbol{x}_i = \boldsymbol{c_d} + \boldsymbol{b}_i \tag{10}$$

$$\boldsymbol{x}_{e_f} = \boldsymbol{x}_{i+1} = \boldsymbol{c_d} + \boldsymbol{b}_{i+1}. \tag{11}$$

The densities of all the edges in the device are then merged using a p-norm approximation of the maximum density, as quantified in Eqn. (12):

$$\rho_d(\boldsymbol{c_d}, \boldsymbol{p}) = \left[\sum_{e=1}^{N_e} (\tilde{\rho}_e(d_e(\boldsymbol{x}_{e_0}, \boldsymbol{x}_{e_f}, \boldsymbol{p})))^p \right]^{\frac{1}{p}}. \tag{12}$$

After merging edge densities, all elements with centers inside the polygon are assigned $\rho_d = 1$. Elements with centers inside the polygon can be found using the MATLAB function `inpoly()`, or by using the algorithm described in Ref. [36].

Finally, the density used in Eqn. (7) is calculated by merging densities of all bars and devices in Eqn. (13). In the temperature field solution, heat is being conducted between the devices and interconnects because the merged density field is used to calculate the stiffness matrix.

$$\rho_i = \left[\sum_{q=1}^{N_q} (\rho_q(d_q(\boldsymbol{x}_{q_0}, \boldsymbol{x}_{q_f}, \boldsymbol{p}_i)))^p + \sum_{d=1}^{N_d} (\rho_d(\boldsymbol{c}_d, \boldsymbol{p}_i))^p \right]^{\frac{1}{p}}. \tag{13}$$

Section 2.2 introduces pipe elbows which form a smooth radius at the intersection of two straight segments. These curved pipe segments are used in the pipe flow model but are not modeled in the projection. The projection at the intersection of two segments is therefore an approximation based on the assumption of straight pipes with an elbow radius of zero.

Devices may also add or remove heat from the domain. The projection in Eqn. (12) will be used to model this effect. Rather than assuming a constant internal heat generation Q across all devices, each device will have its own Q_d value. Element load vectors are then modified using this Q_d and the device density.

$$\boldsymbol{p}_e = \sum_{d=1}^{N_d} \rho_{\mathrm{de}}^{p} Q_d \int_{\Omega_e} \boldsymbol{N}\, d\Omega_e = \sum_{d=1}^{N_d} \rho_{\mathrm{de}}^{p} Q_d \boldsymbol{p}_0 \tag{14}$$

The convection boundary condition term of Eqn. (6) is omitted again as it will not be scaled with density.

2.2 Lumped parameter pipe flow model

This section presents a lumped parameter pipe flow model for the pressure throughout the flow loop. Pressure is a factor that influences pump power consumption, which is important to reduce. The lumped parameter model uses empirical relations to approximate flow loop sections using only a small number of parameters [37]. The lumped parameter model is computationally inexpensive compared to computational fluid dynamics (CFD) models, and provides suitable accuracy (important properties for design optimization). The following assumptions have been made in the pipe flow model presented here:

1. Flow is incompressible
2. All components are connected in series with no branches
3. Everything is in the same plane relative to ground (no height change)
4. Flow rate at the inlet is known
5. Flow is turbulent everywhere

Most of these assumptions could be relaxed if more accuracy is desired, with the penalty of increased computational expense and more complex sensitivity analysis.

We begin with Eqn. (15), which is derived in detail from an energy balance in Ref. [37]. Each term in this equation is formulated to have units of length. This equation relates head loss H_L to pressure, P, and velocity, V, at points 1 and 2 in the flow loop.

$$H_L = \frac{P_1 - P_2}{\rho_w} + \frac{V_1^2 - V_2^2}{2g} \tag{15}$$

Here, ρ_w is the weight density, and g is gravitational acceleration. Solving Eqn. (15) for the pressure difference (in terms of head) between two points produces:

$$\frac{P_1 - P_2}{\rho_w} = H_L - \frac{V_1^2 - V_2^2}{2g} \tag{16}$$

The flow rate in the system is known, so fluid velocity at any point can be calculated easily using:

$$V_i = \frac{Q}{A_i}. \tag{17}$$

Here, Q is the volumetric flow rate (uniform for a series flow loop), and A_i is the cross sectional area of the pipe at location i. Head loss is determined next, which is a proxy metric for energy loss between points 1 and 2 for reasons other than velocity change. Head loss, in units of length, is a standard metric used in describing flow system properties, including pump efficiency and system characterization curves [38]. Models for estimating head loss for many different pipe flow system components can be found in

Ref. [37], but here only two will be of interest: 1) losses due to friction between the fluid and pipe wall (sometimes called major loss), and 2) losses due to elbows connecting straight segments of pipe. Each straight segment of pipe and each elbow has a loss coefficient, K, assigned based on geometry. Total head loss for n_{pe} pipe elements in series can be calculated by combining the loss coefficients as follows:

$$H_L = \frac{\dot{w}^2}{2g\rho_w^2} \sum_{i=1}^{n_{\text{pe}}} \frac{K_i}{A_i^2}, \tag{18}$$

where $\dot{w}$ is the weight flow rate.

The loss coefficient for a straight segment of pipe with length l_i and diameter d_i is:

$$K_i^s = f_i \frac{l_i}{d_i}, \tag{19}$$

and for an elbow with bend angle α_i and bend radius r_i the loss coefficient is:

$$K_i^e = f_i \alpha_i \frac{r_i}{d_i} + (0.1 + 2.4 f_i) \sin\left(\frac{\alpha_i}{2}\right) + \frac{6.6 f_i \left(\sin\left(\frac{\alpha_i}{2}\right) + \sqrt{\sin\left(\frac{\alpha_i}{2}\right) + \varepsilon_s}\right)}{\left(\frac{r_i}{d_i}\right)^{4\alpha_i/\pi}} - f_i \frac{2 l_c}{d_i}. \tag{20}$$

See Fig. 3 for a description of elbow geometry. In Eqn. (20), a small perturbation ε_s has been added to make the expression differentiable at $\alpha = 0$. The first term in Eqn. (20) accounts for frictional losses across the elbow arc length. The final term reduces the loss coefficient to account for the length of straight pipe that is overlapped by the elbow, l_c. Implementing loss coefficient calculations in this way allows each pipe section to be modular. If the length of straight pipe were reduced directly at the straight loss coefficient calculation, information about the connecting pipe and elbow would be needed. The bend angle is found by defining two vectors—$\boldsymbol{a}$ and $\boldsymbol{b}$—based on the endpoints of two connected segments:

$$\boldsymbol{a} = \boldsymbol{x}_{a_f} - \boldsymbol{x}_{a_0} \tag{21}$$

$$\boldsymbol{b} = \boldsymbol{x}_{b_f} - \boldsymbol{x}_{b_0}. \tag{22}$$

From the definition of the dot product, we obtain:

$$\theta = \cos^{-1}(v), \tag{23}$$

where:

$$v = (1 - \varepsilon_c) \frac{\boldsymbol{a} \cdot \boldsymbol{b}}{||\boldsymbol{a}||||\boldsymbol{b}||}. \tag{24}$$

A perturbation ε_c is incorporated into Eqn. (24) to restrict the range such that $v \in [0, (1 - \varepsilon_c)]$. This is done to prevent the derivative of θ from being undefined when $v = 1$. The angle α in Eqn. (20) is defined as the supplementary angle of θ:

$$\alpha = \pi - \theta. \tag{25}$$

The clipped length is calculated using:

$$l_c = r_i \sqrt{\frac{1 + v}{1 - v}}. \tag{26}$$

The *friction factor* (f_i) appears in both loss coefficient equations. The friction factor is a function of Reynolds number, and can also account for pipe wall roughness. A variety of approximate models for

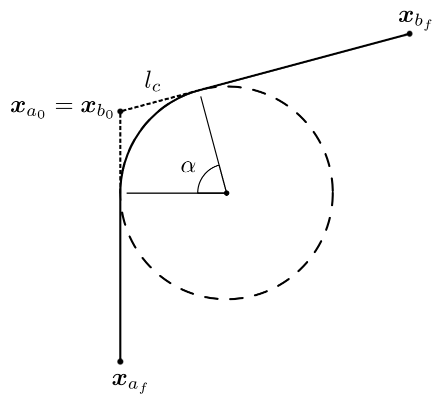


FIGURE 3: Pipe elbow geometry

friction factor have been developed based on experimental results. Here, the equation for turbulent flow in smooth pipes proposed by Blasius [39] is used to estimate f_i:

$$f_i = 0.3164 Re_i^{-0.25}, \tag{27}$$

with Reynolds number:

$$Re_i = \frac{V_i d_i \rho_m}{\mu}, \tag{28}$$

where ρ_m is mass density and μ is fluid viscosity. The use of the thermal conductivity and pipe flow models in optimization will be discussed in the following section.

3 Optimization problem and sensitivity analysis

This section presents the optimization problem formulation, as well as derivations for the sensitivities needed to use gradient-based optimization methods. The models presented in Section 2 lead to a specific choice of design variables which will be discussed in this section. Furthermore, this set of design variables can be used to define geometric constraints which are needed to prevent interference between different components of the system.

The system being optimized here consists of a number of devices and the interconnects routed between devices (see Figs. 6 or 11, for example).

The goal is to find the optimal system layout. System layout is defined as the placement of the devices, the routing of device interconnects, as well as limited sizing parameters (such as interconnect diameter, assuming tubular connections between devices). Each device and interconnect are parameterized in a general manner to help simplify sensitivity calculations and support object-oriented code implementation. More complex devices and routing can be modeled without significant additional work due to the use of object-oriented programming.

In addition to $\boldsymbol{c}_d$ and $\boldsymbol{b}_i$ introduced in Sec. 2, devices may have ports with location $\boldsymbol{p}_i$ relative to the reference point. Ports are the required locations for interconnect attachment to each device. As the reference point moves, the polygon and ports will move with it. Device shape, size, and port location are held fixed during the optimization, so the only design variable for each device is $\boldsymbol{c}_d$. In some optimization studies, it may be useful to omit the reference point corresponding to a device from the set of optimization variables, holding the device fixed in a particular location. This can also be used to specify fixed inputs and outputs of the flow loop.

Pairs of ports are connected via physical interconnects, and device connection topology is assumed to be given (and unchanging) here. Each interconnect is represented here using one or more straight geometric segments. Increasing the number of segments in a connection supports consideration of approximately curved (and more complex) interconnect geometries, but increases computational expense. Interconnect segment i is associated with parameters for its start and end points, $\boldsymbol{x}_{i_0}$ and $\boldsymbol{x}_{i_f}$, respectively, as well as width, w_i. All of these quantities are optimization variables.

The models presented in Sec. 2 are reformulated in terms of the above parameterization to simplify sensitivity calculation. A tradeoff, however, exists between ease of sensitivity calculation and problem conditioning. Specifically, the connection between two interconnect segments, or a segment and a port, present a challenge. As defined above, each system element has its own independent parameters, so connections are free to be broken. There are two ways to solve this issue. The first method attempts to enforce constraints between connected points. Such constraints could be implemented as either linear equality or nonlinear inequality constraints, shown in Eqns. (29) and (30), respectively:

$$x_i - x_j = 0 \tag{29}$$

$$(x_i - x_j)^2 - \varepsilon \leq 0, \tag{30}$$

where x_i and x_j are copies of the same parameter across two different elements. Equation (30) constrains a norm of the parameter error to be within a tolerance ε, approximating the equality constraint in Eqn. (29).

While this may be useful for use with optimization algorithms that do not support equality constraints, it can also help in cases where satisfaction of equality constraints is difficult and causes optimization algorithm convergence problems. Equation (30) is a relaxation of the original equality constraint. A small value of ε provides an accurate approximation, but can degrade problem conditioning. Using a large value of ε improves problem conditioning, but reduces solution accuracy. Replacing linear with nonlinear constraints can also impact computational expense. A related alternative remedy is to increase constraint satisfaction tolerances for these consistency constraints within optimization algorithm settings, but this may not be a practical approach in general. In numerical experiments performed for this study, the value of ε was observed to be a critical parameter to tune the balance between problem difficulty in satisfying constraints (problem stiffness or conditioning), while maintaining geometric consistency.

A second approach to ensure geometric consistency is to define an implicit parameterization, making use of two design variable vectors: the *expanded* and *reduced* design variable vectors. Consider a system with n_d devices and n_s routing segments. The expanded design variable vector:

$$\boldsymbol{z}' := [\boldsymbol{c}_1, ..., \boldsymbol{c}_{n_d}, \boldsymbol{x}_{1_0}, \boldsymbol{x}_{1_f}, w_1, ..., \boldsymbol{x}_{n_{s0}}, \boldsymbol{x}_{n_{sf}}, w_{n_s}]^T$$

contains all the parameters discussed above for each element in the layout. The reduced design variable vector contains only the reference points of free devices, $\boldsymbol{c}_d^f$, locations where routing segments meet, $\boldsymbol{x}_i^f$, and the width of each connection, w_i. It is assumed that the width of all segments that connect two ports are the same (e.g., representing interconnects with uniform properties, such as electrical wiring or piping). The reduced design vector is then:

$$\boldsymbol{z} := [\boldsymbol{c}_1^f, ..., \boldsymbol{c}_{n_{\mathrm{df}}}^f, \boldsymbol{x}_1^f, ..., \boldsymbol{x}_{n_{sf}}^f, w_1, ..., w_{n_c}]^T,$$

where n_{df} is the number of free devices, n_{sf} is the number of points where routing segments meet, and n_c is the number of connections between pairs of ports. With these two vectors introduced, a mapping can be defined from the reduced to the expanded design variable vectors, defined in matrix form as:

$$\boldsymbol{z}' = \boldsymbol{M}\boldsymbol{z} + \boldsymbol{P} \tag{31}$$

where $\boldsymbol{M}$ is a binary mapping (or selection) matrix. It is derived by identifying which elements of $\boldsymbol{z}$ and

z' correspond to each other. The vector $\boldsymbol{P}$ represents fixed devices that are not in the reduced design variable vector, but have a fixed value throughout the optimization. Also, segment endpoints that are connected to device ports are mapped to the device reference point in the reduced design variable vector by including an offset corresponding to the port location $\boldsymbol{p}_i$. It should be noted that, in general, the mapping matrix is not invertible, so the inverse mapping based on Eqn. (31) may not be possible.

This mapping preserves the simplified sensitivity calculations described above, while eliminating the need for consistency constraints. All calculations of objective functions and constraints, and their sensitivities, are performed in terms of the expanded design variable vector. Next, the sensitivities are computed in terms of the reduced design variables by using the chain rule:

$$\frac{df}{d\boldsymbol{z}} = \frac{\partial f}{\partial \boldsymbol{z}'}\frac{d\boldsymbol{z}'}{d\boldsymbol{z}}, \tag{32}$$

$$\frac{d\boldsymbol{z}'}{d\boldsymbol{z}} = \boldsymbol{M}. \tag{33}$$

The reduced design variable vector is used as the optimization vector by the solution algorithm. To maintain correspondence between reduced and expanded vectors, the mapping defined in Eqn. (31) is applied each time the reduced design variable vector is updated.

Now that the parameterization of the design space has been determined, the complete optimization problem formulation can be presented:

$$\begin{aligned}
\min_{\boldsymbol{x}} \quad & f(\boldsymbol{x}, \boldsymbol{T}) && \text{(34a)}\\
\text{s.t.:} \quad & \boldsymbol{g}_{\text{phys}}(\boldsymbol{x}, \boldsymbol{T}) \leq 0 && \text{(34b)}\\
& \boldsymbol{g}_{\text{dd}}(\boldsymbol{x}) \leq 0 && \text{(34c)}\\
& \boldsymbol{g}_{\text{sd}}(\boldsymbol{x}) \leq 0 && \text{(34d)}\\
& \boldsymbol{g}_{\text{ss}}(\boldsymbol{x}) \leq 0 && \text{(34e)}\\
\text{where:} \quad & \boldsymbol{K}(\boldsymbol{x})\boldsymbol{T} = \boldsymbol{P}(\boldsymbol{x}) && \text{(34f)}
\end{aligned}$$

Here $f(\boldsymbol{x}, \boldsymbol{T})$ is the objective function and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{T})$ are constraint functions. In general, these functions may depend both on design ($\boldsymbol{x}$) and state ($\boldsymbol{T}$) variables. The function $f(\cdot)$ can be any one of the candidate objectives discussed later in Sec. 3.1. The constraints $\boldsymbol{g}_{\text{phys}}(\boldsymbol{x}, \boldsymbol{T})$ are constraints that depend both on design and on solutions to the physics models (i.e. the value of the state vector $\boldsymbol{T}$). The interference constraints $\boldsymbol{g}_{\text{dd}}(\boldsymbol{x})$, $\boldsymbol{g}_{\text{sd}}(\boldsymbol{x})$, and $\boldsymbol{g}_{\text{ss}}(\boldsymbol{x})$ prevent interference between two devices, one routing segment and one device, and two routing segments, respectively. These constraints are independent of any physics models, so they are all explicit functions of the design variables.

3.1 Objective function and physics-based constraints

This section presents objective function options and their derivatives. Objective functions and physics-based constraints are discussed together because they both depend on design and state variable values. In addition, these functions are interchangeable as either objective or constraint functions.

The first candidate function, $f_1(\cdot)$, relates to the solution of the lumped-parameter flow model:

$$f_1(\boldsymbol{x}, \boldsymbol{T}) = H_L \tag{35}$$

The objective is to minimize the head loss (H_L) in the flow loop as calculated in Eqn. (18). When head loss is used as an objective or constraint, the radius of each pipe elbow is also included a design variable. The elbow radii r_i are appended to the end of expanded and reduced design vectors (with a one-to-one mapping). The total derivative using the chain rule is:

$$\frac{dH_L}{d\boldsymbol{x}'} = \frac{\partial H_L}{\partial \boldsymbol{x}'} + \sum_{i=1}^{n_{pe}} \frac{\partial H_L}{\partial K_i}\frac{dK_i}{d\boldsymbol{x}'}. \tag{36}$$

The design variable vector contains device reference point coordinates, bar end coordinates, bar widths, and elbow radii. For each routing segment, the pipe diameter will be equal to the bar width. The only nonzero elements of the explicit derivative $\partial H_L/\partial \boldsymbol{x}'$ are those corresponding to bar width:

$$\frac{\partial H_L}{\partial d_i} = -\frac{\pi d_i \dot{w}^2}{2g\rho_w^2 A_i^2}\sum_{i=1}^{n_{pe}} K_i \tag{37}$$

For each lumped parameter element, whether a straight section or elbow, the following equation applies:

$$\frac{\partial H_L}{\partial K_i} = \frac{1}{A_i^2}\frac{\dot{w}^2}{2g\rho_w^2}. \tag{38}$$

The final derivative in Eqn. (36), $dK_i/d\boldsymbol{x}'$, depends whether the element is a straight or elbow section. For a straight section, design variables are segment end points, $\boldsymbol{x}_{i_0}$ and $\boldsymbol{x}_{i_f}$, and segment diameters d_i. The sensitives are given below:

$$\frac{dK_i^s}{d\boldsymbol{x}_{i_0}} = \frac{f_i}{d_i l_i}(\boldsymbol{x}_{i_0} - \boldsymbol{x}_{i_f}) \tag{39}$$

$$\frac{dK_i^s}{d\boldsymbol{x}_{i_f}} = \frac{f_i}{d_i l_i}(\boldsymbol{x}_{i_f} - \boldsymbol{x}_{i_0}) \tag{40}$$

$$\frac{dK_i^s}{dd_i} = -\frac{f_i l_i}{d_i^2} + \frac{l_i}{d_i}\frac{df_i}{dd_i} \tag{41}$$

$$\frac{df_i}{dd_i} = 0.25(0.3164)Re^{-1.25}\frac{4\dot{m}}{\pi\mu d_i^2}. \tag{42}$$

If the lumped parameter element is an elbow, the design variables are the four end points of connected segments, $\boldsymbol{x}_{a_0}$, $\boldsymbol{x}_{a_f}$, $\boldsymbol{x}_{b_0}$, and $\boldsymbol{x}_{b_f}$, diameter, d_i, and radius of the elbow, r_i. It is assumed that the diameters of connected segments are the same so there is only one diameter variable. The sensitivity of the elbow loss coefficient with respect to pipe diameter is:

$$\frac{dK_i^e}{dd_i} = \frac{\partial K_i^e}{\partial d_i} + \frac{\partial K_i^e}{\partial f_i}\frac{df_i}{dd_i}. \tag{43}$$

Equation (42) can be used again here. The partial derivatives are:

$$\frac{\partial K_i^e}{\partial d_i} = -f_i\alpha_i\frac{r_i}{d_i^2} + 6.6\left(\frac{4\alpha_i}{\pi}\right)f_i\left(\sin\left(\frac{\alpha_i}{2}\right) + \sqrt{\sin\left(\frac{\alpha_i}{2}\right) + \varepsilon_s}\right)r^{(-4\alpha_i/\pi)}d^{(4\alpha_i/\pi)-1} + f_i\frac{2l_c}{d_i^2} \tag{44}$$

$$\frac{\partial K_i^e}{\partial f_i} = \alpha_i\frac{r_i}{d_i} + 2.4\sin\left(\frac{\alpha_i}{2}\right) + \frac{6.6\left(\sin\left(\frac{\alpha_i}{2}\right) + \sqrt{\sin\left(\frac{\alpha_i}{2}\right) + \varepsilon_s}\right)}{\left(\frac{r}{d}\right)^{4\alpha_i/\pi}} - \frac{2l_c}{d_i} \tag{45}$$

With respect to elbow radius, the sensitivity is:

$$\frac{dK_i^e}{dr_i} = f_i\alpha_i\frac{1}{d_i} - 6.6\frac{4\alpha_i}{\pi}f_e\left(\sin\left(\frac{\alpha_i}{2}\right) + \sqrt{\sin\left(\frac{\alpha_i}{2}\right) + \varepsilon_s}\right)r^{(-4\alpha_i/\pi)-1}d^{(4\alpha_i/\pi)} - \frac{2f_i}{d_i}\frac{dl_c}{dr_i}. \tag{46}$$

The chain rule can be used to calculate sensitivities with respect to the four segment end points:

$$\frac{dK_i^e}{d\boldsymbol{x}_{a_0}} = \frac{\partial K_i^e}{\partial\alpha_i}\frac{d\alpha_i}{d\boldsymbol{x}_{a_0}} - \frac{2f_i}{d_i}\frac{dl_c}{d\boldsymbol{x}_{a_0}}, \tag{47}$$

with:

$$\begin{aligned}\frac{\partial K_i^e}{\partial\alpha_i} = f_i\frac{r_i}{d_i} + \frac{1}{2}(0.1 + 2.4f_e)\cos\left(\frac{\alpha_i}{2}\right)\\ + \frac{26.4}{\pi}f_i\left(\sin\left(\frac{\alpha_i}{2}\right) + \sqrt{\sin\left(\frac{\alpha_i}{2}\right) + \varepsilon_s}\right)\left(\frac{d}{r}\right)^{(4\alpha_i/\pi)}\ln\left(\frac{d}{r}\right)\\ + 6.6f_i\left(\frac{d}{r}\right)^{(4\alpha_i/\pi)}\left(\frac{1}{4}\frac{1}{\sqrt{\sin\left(\frac{\alpha_i}{2}\right) + \varepsilon_s}}\cos\left(\frac{\alpha_i}{2}\right) + \frac{1}{2}\cos\left(\frac{\alpha_i}{2}\right)\right).\end{aligned} \tag{48}$$

The derivatives of angle α are:

$$\frac{d\alpha_i}{d\boldsymbol{x}_{a_0}} = \frac{\partial\alpha_i}{\partial v}\frac{dv}{d\boldsymbol{x}_{\boldsymbol{a_0}}}, \tag{49}$$

with:

$$\frac{\partial\alpha_i}{\partial v} = \frac{1}{\sqrt{1 - v^2}}. \tag{50}$$

The required derivatives of the clipped length are as follows:

$$\frac{dl_c}{dr_i} = \sqrt{\frac{1 + v}{1 - v}} \tag{51}$$

$$\frac{dl_c}{d\boldsymbol{x}_{a_0}} = \frac{\partial l_c}{\partial v}\frac{dv}{d\boldsymbol{x}_{a_0}} \tag{52}$$

where:

$$\frac{\partial l_c}{\partial v} = \frac{r_i}{(1 - v)^2}\sqrt{\frac{1 - v}{1 + v}}. \tag{53}$$

The calculations in Eqns. (49) and (52) require derivatives of v with respect to the four bar end points:

$$\frac{dv}{d\boldsymbol{x}_{a_f}} = -\frac{dv}{d\boldsymbol{x}_{a_0}} = (1 - \varepsilon_c)\left[\frac{\boldsymbol{b}}{||\boldsymbol{a}||||\boldsymbol{b}||} - \frac{1}{||\boldsymbol{a}||^3||\boldsymbol{b}||}(\boldsymbol{a}\cdot\boldsymbol{b})\boldsymbol{a}\right], \tag{54}$$

$$\frac{dv}{d\boldsymbol{x}_{b_f}} = -\frac{dv}{d\boldsymbol{x}_{b_0}} = (1 - \varepsilon_c)\left[\frac{\boldsymbol{a}}{||\boldsymbol{a}||||\boldsymbol{b}||} - \frac{1}{||\boldsymbol{a}||||\boldsymbol{b}||^3}(\boldsymbol{a}\cdot\boldsymbol{b})\boldsymbol{b}\right]. \tag{55}$$

A second objective function option is to minimize one of the device temperatures. Temperature at the device center is used here, so the objective function is:

$$f_2(\boldsymbol{x}, \boldsymbol{T}) = T(\boldsymbol{c}_d). \tag{56}$$

This depends on the solution of the thermal conduction physics problem. Sensitivities of functions depending on solution of a finite element problem can be calculated using the adjoint method [40]:

$$\frac{df_2}{dx_i'} = \frac{\partial f_2}{\partial x_i'} + \boldsymbol{\Psi}^T \frac{d\boldsymbol{R}}{dx_i'}, \tag{57}$$

where $\boldsymbol{\Psi}$ is the *adjoint vector*, which can be calculated with the following equation:

$$\boldsymbol{\Psi} = \left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{y}}\right]^{-T} \left[\frac{\partial f_2}{\partial \boldsymbol{y}}\right]^T. \tag{58}$$

The residual $\boldsymbol{R}$ comes from manipulating Eqn. (4) so that one side is zero:

$$\boldsymbol{R} = \boldsymbol{K}\boldsymbol{T} - \boldsymbol{P} = \boldsymbol{0} \tag{59}$$

The vector $\boldsymbol{y}$ in Eqn. (58) is the unknown vector. In the finite element problem the unknowns are $\boldsymbol{y} = [\boldsymbol{P}^p, \boldsymbol{T}^f]^T$. The vector $\boldsymbol{P}^p$ is the flux at nodes where prescribed temperature boundary conditions are applied, and $\boldsymbol{T}^f$ are temperatures at all remaining nodes. Equation (59) is partitioned into blocks, p and f, corresponding to the *prescribed* and *free* (unknown) degrees of freedom, respectively. The derivative of the residual with respect to the unknown vector results in a partitioned matrix:

$$\left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{y}}\right] = \begin{bmatrix} -\boldsymbol{I} & \boldsymbol{K}^{pf} \\ \boldsymbol{0} & \boldsymbol{K}^{ff} \end{bmatrix}. \tag{60}$$

The temperature at an arbitrary location in the domain needs to be calculated by interpolation from nodal temperatures:

$$T(\boldsymbol{c}_d) = \boldsymbol{N}^T(\xi, \eta)\boldsymbol{T}_{\text{el}}, \tag{61}$$

where ξ and η are the location of $\boldsymbol{c}_d$ in the local element coordinate system, and $\boldsymbol{T}_{\text{el}}$ is the vector of element nodal temperatures. The only nonzero derivative with respect to design variables $\boldsymbol{x}'$ is for the device reference point. Using the definition of the matrix $\boldsymbol{B} = d\boldsymbol{N}/d\boldsymbol{x}$, explicit derivatives of $f_2(\cdot)$ can then be calculated:

$$\frac{\partial f_2}{\partial \boldsymbol{x}'} = \boldsymbol{B}^T \boldsymbol{T}_{\text{el}} \tag{62}$$

$$\frac{\partial f_2}{\partial \boldsymbol{y}} = \boldsymbol{N}(\xi, \eta) \tag{63}$$

The derivative of the residual with respect to design variables is calculated using the chain rule.

$$\frac{d\boldsymbol{R}}{dx_i'} = \frac{\partial \boldsymbol{R}}{\partial \rho_e} \frac{d\rho_e}{dx_i'} \tag{64}$$

Finally, taking into account Eqns. (4) and (5), the total derivative is:

$$\frac{df_2}{dx_i'} = \frac{\partial f_2}{\partial x_i'} + \sum_{e=1}^{N_e} p(1-\rho_{\min})\rho_e^{(p-1)} \boldsymbol{\Psi}_e^T \boldsymbol{k}_0\, \boldsymbol{T}_e \frac{d\rho_e}{dx_i'} + \sum_{d=1}^{N_d} Q_d \sum_{e=1}^{N_e} p\rho_{\text{de}}^{(p-1)} \boldsymbol{\Psi}_e^T \boldsymbol{p}_0 \frac{d\rho_{\text{de}}}{dx_i'}, \tag{65}$$

where N_e is the number of finite elements in the mesh, ρ_e is the element density from Eqn. (13), ρ_{de} is the device element density from Eqn. (12), and $\boldsymbol{\Psi}_e$ and $\boldsymbol{T}_e$ are the adjoint and temperature vectors corresponding to element degrees of freedom.

The derivative of the geometric projection is the final part of Eqn. (65). The derivative of density resulting from the merge in Eqn. (13) is:

$$\frac{d\rho}{dx'_i} = \sum_{q=1}^{N_q} \left(\frac{\rho_q}{\rho}\right)^{p-1} \frac{d\rho_q}{dx'_i} + \sum_{d=1}^{N_d} \left(\frac{\rho_d}{\rho}\right)^{p-1} \frac{d\rho_d}{dx'_i}. \tag{66}$$

See Ref. [32] for derivatives with respect to bar ends. For devices, the derivative with respect to device centers must be calculated using:

$$\frac{d\rho_d}{d\boldsymbol{c}_d} = \rho_d^{1-p} \sum_{e=1}^{N_e} (\tilde{\rho}_e)^{p-1} \frac{\partial \tilde{\rho}_e}{\partial \boldsymbol{c}_d}, \tag{67}$$

where:

$$\frac{\partial \tilde{\rho}_e}{\partial \boldsymbol{c}_d} = \frac{\partial \tilde{\rho}_e}{\partial \boldsymbol{x}_{e_0}} \frac{\partial \boldsymbol{x}_{e_0}}{\partial \boldsymbol{c}_d} + \frac{\partial \tilde{\rho}_e}{\partial \boldsymbol{x}_{e_f}} \frac{\partial \boldsymbol{x}_{e_0}}{\partial \boldsymbol{c}_d}. \tag{68}$$

Derivatives with respect to edge ends can again be found in Norato et *al*. [32], and appear here as $\partial\tilde{\rho}_e/\partial\boldsymbol{x}_{e_0}$ and $\partial\tilde{\rho}_e/\partial\boldsymbol{x}_{f_0}$. From the definition of edge endpoints in Eqn. (10) and (11), derivatives $\partial\boldsymbol{x}_{e_0}/\partial\boldsymbol{c}$ and $\partial\boldsymbol{x}_{e_0}/\partial\boldsymbol{c}$ are the identity matrix. The density derivative of any element inside the device polygon is set to $\partial\rho_d/\partial\boldsymbol{c} = \boldsymbol{0}$. To justify making the sensitivity of interior elements zero, imagine perturbing the center of the device by a small amount. Most elements inside the polygon are still inside the polygon, so there is no change in the density. Some elements near the edges may have switched from being inside the polygon to in the edge bar, or vice versa. These elements will be near the bar center and should still have full density.

3.2 Geometric constraints

The interference constraint functions will be presented below along with sensitivity analysis. Any constraint involving a device is not enforced on the device boundary, but on a bounding circle with radius r^b centered at the device reference point, see Figs. 4a or 4b. The radius can be found by calculating the maximum vertex distance from the device reference point. The constraint between two devices i and j, and its sensitivity, are shown below (see also Fig. 4a for an illustration of the constraint).

$$g_{\mathrm{dd}}(\boldsymbol{x}) = (r_i^b + r_j^b)^2 - \|\boldsymbol{c}_j - \boldsymbol{c}_i\|^2 \leq 0 \tag{69}$$

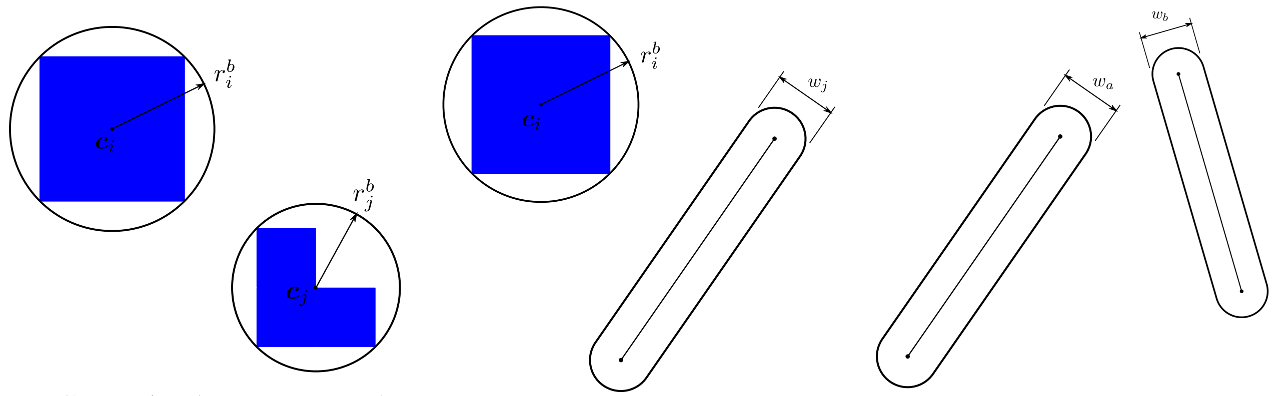
$$\frac{dg_{\mathrm{dd}}}{d\boldsymbol{c}_i} = 2(\boldsymbol{c}_j - \boldsymbol{c}_i) \tag{70}$$

$$\frac{dg_{\mathrm{dd}}}{d\boldsymbol{c}_j} = -2(\boldsymbol{c}_j - \boldsymbol{c}_i) \tag{71}$$

For the constraint between a device i and a segment of routing j, we will use previous results from projecting a bar onto the mesh. As an intermediate step in the projection, the distance between a line segment and a point was found. Here, the distance will be found between the line segment and a device reference point rather than a mesh element center. The constraint function is:

$$g_{\mathrm{sd}} = \frac{w_j}{2} + r_i^b - d_{ij} \leq 0. \tag{72}$$

Sensitivities $dd_{ij}/d\boldsymbol{x}_{j_0}$ and $dd_{ij}/d\boldsymbol{x}_{j_f}$ are already known from previous results, as well as $dg_{\mathrm{sd}}/dw_j = \frac{1}{2}$. The device reference point is a design variable, whereas the element centers were not design variables in



(a) Constraint between two devices

(b) Constraint between a device and routing segment

(c) Constraint between two routing segments

FIGURE 4: Geometric constraints

previous results, so an additional sensitivity needs to be calculated:

$$\frac{dd_{ij}}{d\boldsymbol{c}_i} = \begin{cases} \frac{\boldsymbol{b}}{\|\boldsymbol{b}\|} & \boldsymbol{a}\cdot\boldsymbol{b} \leq 0 \\ \frac{1}{\|\boldsymbol{g}\|}\Big(\boldsymbol{I} - \frac{1}{\|\boldsymbol{a}\|^2}(\boldsymbol{a}\otimes\boldsymbol{a})\Big)\boldsymbol{g} & 0 < \boldsymbol{a}\cdot\boldsymbol{b} < \boldsymbol{a}\cdot\boldsymbol{a} \\ \frac{\boldsymbol{e}}{\|\boldsymbol{e}\|} & \boldsymbol{a}\cdot\boldsymbol{b} \geq \boldsymbol{a}\cdot\boldsymbol{a}. \end{cases} \tag{73}$$

A constraint to prevent interference between two routing segments requires finding the distance between two line segments. Reference [41] describes an algorithm for calculating the distance between two line segments. First, the two segments are extended into infinite lines, and the minimum distance is found. If the minimum occurs at a point on the line outside of the segment endpoints, then a series of cases must be tested to find the distance between endpoints and the other segment. MATLAB code to compute the minimum distance between two segments, and the derivative with respect to the segment endpoints, is included in the Appendix. The constraint to avoid interference between routing segments a and b with minimum distance d_{ab} between them is:

$$g_{\text{ss}} = \left(\frac{w_a}{2} + \frac{w_b}{2}\right)^2 - d_{\text{ab}}^2 \leq 0. \tag{74}$$

The squared distance is used to avoid undefined derivatives when the distance is zero. The sensitivity with respect to segment end points can be found in the MATLAB code in the Appendix. Sensitivity with respect to the bar widths are:

$$\frac{dg_{\text{ss}}}{dw_a} = \frac{dg_{\text{ss}}}{dw_b} = \left(\frac{w_a}{2} + \frac{w_b}{2}\right). \tag{75}$$

4 Results

The above method will be used to optimize the device layout and interconnect routing of two different systems. The first system consists of three identical devices connected in a loop. A comparison of the results using three different objectives functions will be made. A second system with unique devices and fixed input and output locations will be optimized. Both the system architecture and the geometric topology of the system are fixed during the optimization of both of these examples. The system architecture specifies what ports on which components are connected to specific ports on other components within the system. For each system architecture, many geometric topologies may exist; e.g.,

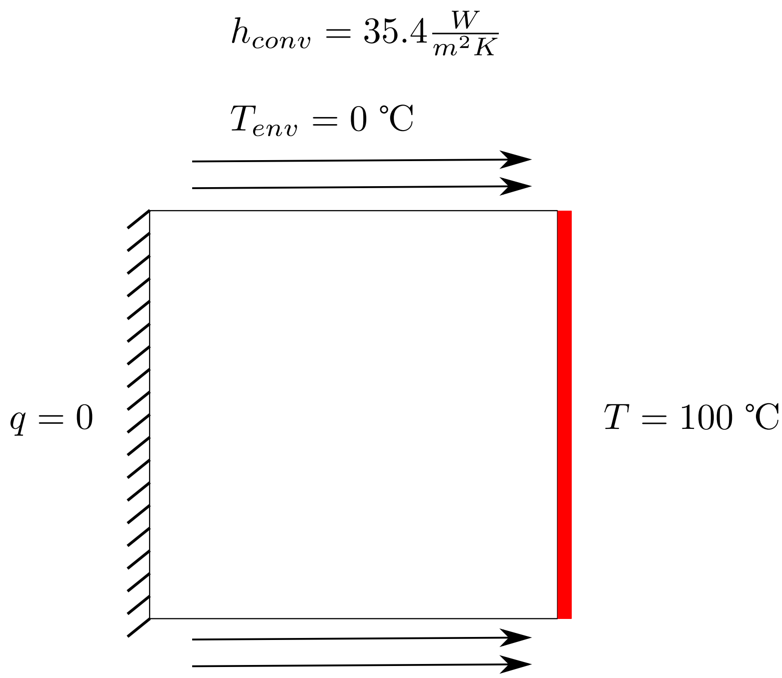


FIGURE 5: Boundary condition for thermal finite element analysis

if an interconnect links ports A and B, many options may exist for how this interconnect passes around various other interconnects and devices in the system. Exploring all possible geometric topologies for a single system architecture becomes exponentially more complex as the number of devices increases. Generating and selecting distinct geometric topologies is a topic of ongoing work, and is beyond the scope of this article. Boundary conditions for the thermal problem will be the same for both systems. Top and bottom edges of the domain will have convection boundary conditions with a convection coefficient of h = 35.4 W/(m^{2}K) and environment temperature of 0 °C. The right edge has a fixed temperature of 100 °C, and the left edge is insulated. The MATLAB sequential quadratic programming (SQP) optimization solver was used for all solutions presented below.

The system with the initial condition depicted in Fig. 6 was optimized according to three different objective functions. The system has three 0.06 m × 0.06 m square devices, each generating 3,000 W/m^2 of heat. The devices are connected in a loop. Each connection has one free intersection point. A short fixed pipe segment is attached to each port. This allows constraints between devices and all free pipe segments to be enforced. The thermal conductivity of the solid material is 54 W/(mK), and 0.032 W/(mK) is used for void, resulting in $\rho_{\min} = 5.86 \times 10^{-4}$. The domain for the optimization is 0.5 m × 0.5 m. A projection radius of 2.36 mm, circumscribing the finite element, was used, and the penalization parameter was 3. Properties of fluid flow in the loop are listed in Table 1. Three optimization studies were completed: Tests A, B, and C. The objective of Test A is to minimize head loss in the loop. The problem statement is:

$$\begin{aligned}
\min_{\boldsymbol{x}} \quad & f(\cdot) = H_l && \text{(76a)}\\
\text{subject to:} \quad & T(\boldsymbol{c}_1), T(\boldsymbol{c}_2), T(\boldsymbol{c}_3) \leq T_c && \text{(76b)}\\
& \boldsymbol{g}_{\text{geo}} \leq 0 && \text{(76c)}\\
\text{where:} \quad & \boldsymbol{K}(\boldsymbol{x})\boldsymbol{T} = \boldsymbol{P}(\boldsymbol{x}). && \text{(76d)}
\end{aligned}$$

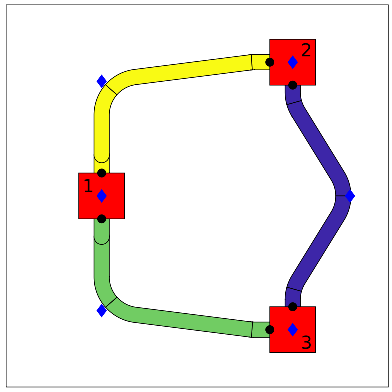


FIGURE 6: Initial system layout for test A, B, and C. Device numbers shown. Blue diamonds are design variable locations.

Test B minimizes the temperature of device 3, and the problem statement is:

$$\min_{\boldsymbol{x}} \quad f(\cdot) = T(\boldsymbol{c}_3) \tag{77a}$$
$$\text{subject to:} \quad T(\boldsymbol{c}_1), T(\boldsymbol{c}_2) \leq T_c \tag{77b}$$
$$H_l \leq H_{l_c} \tag{77c}$$
$$\boldsymbol{g}_{\text{geo}} \leq 0 \tag{77d}$$
$$\text{where:} \quad \boldsymbol{K}(\boldsymbol{x})\boldsymbol{T} = \boldsymbol{P}(\boldsymbol{x}). \tag{77e}$$

The problem statement for Test C, minimizing the system bounding box area, is:

$$\min_{\boldsymbol{x}} \quad f(\cdot) = (\max(\bar{\mathrm{x}}) - \min(\bar{\mathrm{x}}))(\max(\bar{\mathrm{y}}) - \min(\bar{\mathrm{y}})) \tag{78a}$$
$$\text{subject to:} \quad T(\boldsymbol{c}_1), T(\boldsymbol{c}_2), T(\boldsymbol{c}_3) \leq T_c \tag{78b}$$
$$H_l \leq H_{l_c} \tag{78c}$$
$$\boldsymbol{g}_{\text{geo}} \leq 0 \tag{78d}$$
$$\text{where:} \quad \boldsymbol{K}(\boldsymbol{x})\boldsymbol{T} = \boldsymbol{P}(\boldsymbol{x}). \tag{78e}$$

Where $\bar{\mathrm{x}}$ and $\bar{\mathrm{y}}$ are the set of x and y coordinates of device reference points and bar segment end points. The critical device temperature, T_c, for all three devices is 30 °C. Head loss constraints, H_{l_c}, of 4 m and 5 m will be used to quantify the effect of this constraint on the outcome. Here all geometric constraints from Sec. 3.2 have been lumped together into $\boldsymbol{g}_{\text{geo}}$. The function in Eqn. (78a) represents a rectangular bounding box containing all free devices and interconnects that is aligned with the x and y axes.

$$\min_{\boldsymbol{x}} \quad A(\cdot) \qquad (\textbf{Objective:} \text{ Bounding box area})$$
$$\text{subject to:} \quad T_{\boldsymbol{d}_{i=1\ldots4}} \leq T_{\boldsymbol{d}_{max}} \qquad \text{(Max. device temp. constraint)}$$
$$T_{\boldsymbol{f}} \leq T_{\boldsymbol{f}_{max}} \qquad \text{(Max. fluid temp. constraint)}$$
$$H_l \leq H_{l_c} \qquad \text{(Max. head loss constraint)}$$
$$\boldsymbol{g}_{\text{geo}} \leq 0 \qquad \text{(Geometric constraints)}$$
$$\text{where:} \quad \boldsymbol{K}(\boldsymbol{x})\boldsymbol{T} = \boldsymbol{P}(\boldsymbol{x}). \qquad \text{(Physics-based model eqns.)}$$

The minimum head loss for optimization problem Test A is 2.063 m with the layout shown in Fig. 7.

	symbol	value
density	ρ_m	1072 kg/m^3
viscosity	μ	0.068 kg/(m s)
flow rate	Q	0.001 m^3/s
max pipe diameter	-	0.02 m

TABLE 1: Fluid flow properties

Table 2 compares the sensitivities of the temperature constraints using the adjoint method and finite difference method. The sensitivities were obtained from the 53rd iteration (close to optimal solution) of the optimization problem in Fig. 7. A total of nine sensitivities are compared. They are sensitivities of each of the three temperature constraints with respect to three different design variables.

To satisfy device temperature constraints, one of the routing interconnects touches the convection boundary. This conducts heat from the devices through the routing to the boundary where it can be dissipated. The optimization finds a balance between smooth bends and reducing pipe length to reduce head loss in a way that is best for system performance. A multi-start approach was used in test A of the first example to improve the probability of finding global optima. As shown in Fig. 8, we use six initial layouts (with different device locations and interconnect nodal positions) of which five converged to a feasible solution. The five that converged had the same layout of devices and interconnects. There were two differences between the different optimal solutions. One is whether the layout connected to the top or bottom boundary; since the boundary conditions are symmetric, these two solutions are functionally identical. The second difference is the horizontal location of the layout. The objective function changes slightly when the layout is moved along the boundary. It appears that the solution can get stuck in a local optima based on where the interconnect initially touches the boundary. It is also possible that if the convergence tolerance is reduced, the solutions all may converge to the same solution. As discussed in the problem definition, determining starting points for the optimization, including device layouts, geometric topologies, and interconnect route shapes, is a complex problem which requires more investigation.

Table 3 contains the objective function values for Test B with two different head loss constraint values. The optimal layouts are shown in Fig. 9. Both layouts have an interconnect touching the top boundary to dissipate heat from devices. The layout for the 4 m head loss constraint requires smoother elbows, which comes at the expense of a higher device 3 temperature. More specifically, device 3 has a 80.4% higher temperature than the system with a 5.0 m head loss constraint when a 4.0 m head loss constraint is used. Results of Test C are given in Table 4, and the corresponding final layout designs are shown in Fig. 10. Both layouts are similar, except that the system with the 5.0 m head loss constraint can pack the devices closer together by taking advantage of sharper bends. The result is a 16.6% increase in the bounding box area for the system when a 4.0 m head loss constraint is applied. Each objective function produces significantly different optimal layouts. This highlights the importance appropriate objective function selection.

A notional power electronics cooling system for an unmanned aerial vehicle (UAV) was optimized using the method presented above. The initial system layout is depicted in Fig. 11, and corresponding device properties are given in Table 5. The system consists of two battery packs, an AC/DC converter, and a heat exchanger. The battery packs and AC/DC converter add heat to the system, and the heat exchanger removes heat. A fixed-location inlet and outlet for the fluid loop are placed on the left edge. Boundary conditions and flow properties are same as the first problem, except the domain has been enlarged to 1 m $\times$ 1 m to allow space for more components. The maximum pipe diameter is also increased to 0.03 m. There are two free points in each connection to allow for more complex interconnect routing paths. The optimization was solved using both head loss and bounding box objective functions.

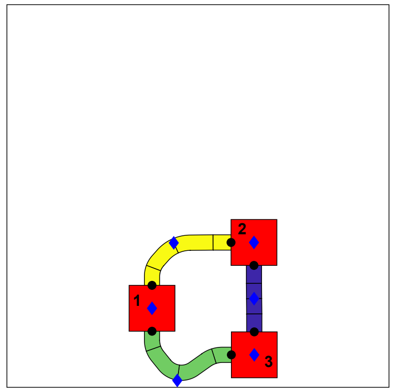


FIGURE 7: Optimal layout of Test A

Analytical sensitivities	Finite difference sensitivities	Relative errors
-0.000001089314857	-0.000001089314955	8.996×10^{-8}
-0.000201888612579	-0.000201888612558	-1.040×10^{-10}
-0.000128819188526	-0.000128819188535	6.987×10^{-11}
-0.000015775640565	-0.000015775640533	-2.028×10^{-9}
-0.000000314345463	-0.000000314345463	0.0
0.0000000005125052	0.0000000005125089	-7.219×10^{-6}
-0.000006785843214	-0.000006785843276	9.137×10^{-9}
-0.000005674234221	-0.000005674234272	8.988×10^{-9}
-0.000053662559067	-0.000053662559068	1.863×10^{-11}

TABLE 2: Comparison of analytical and finite difference sensitivities for device temperature constraints. The sensitivities are obtained from the 53^{rd} iteration of the optimization problem presented in Fig. 7. Sensitivities for each of the three temperature constraints with respect to three different design variables are presented.

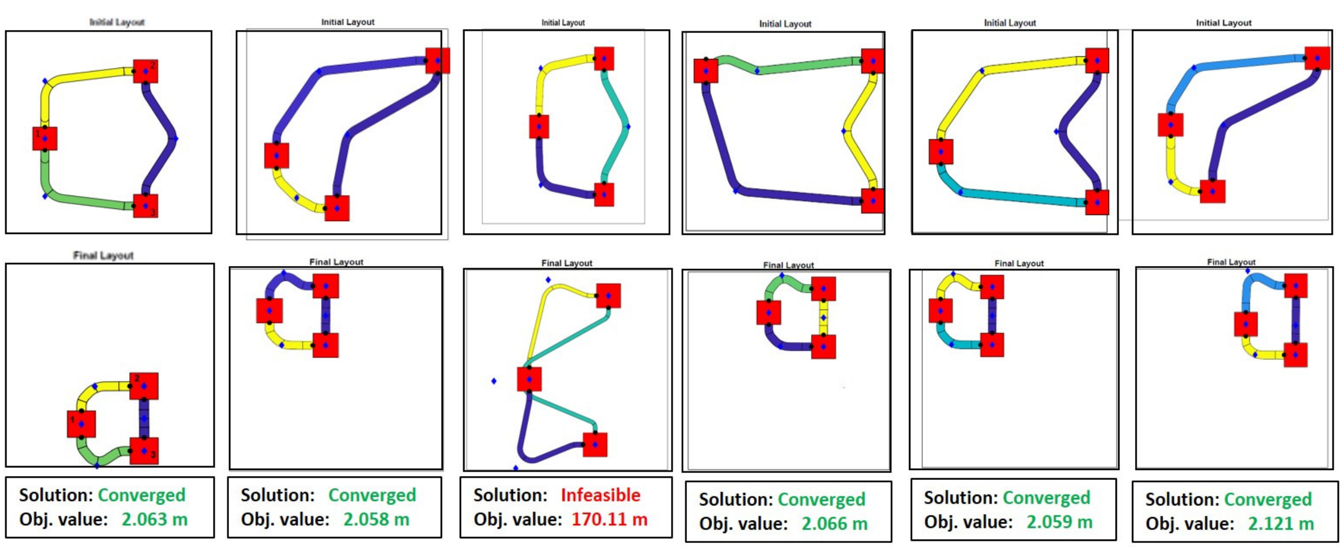
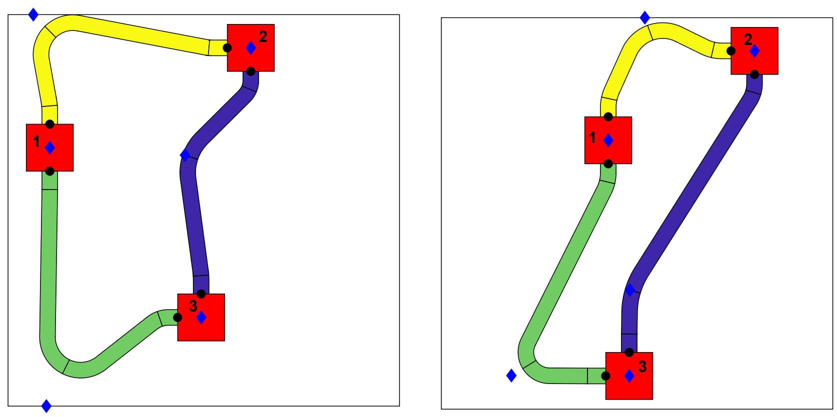


FIGURE 8: Six different initial layouts (first row) for Test A with simple head loss objective function and their corresponding final layouts (second row) along with their objective function values.

Head Loss (m)	objective (°C)	% increase
≤ 5	10.16	-
≤ 4	18.33	80.4

TABLE 3: Results of Test B



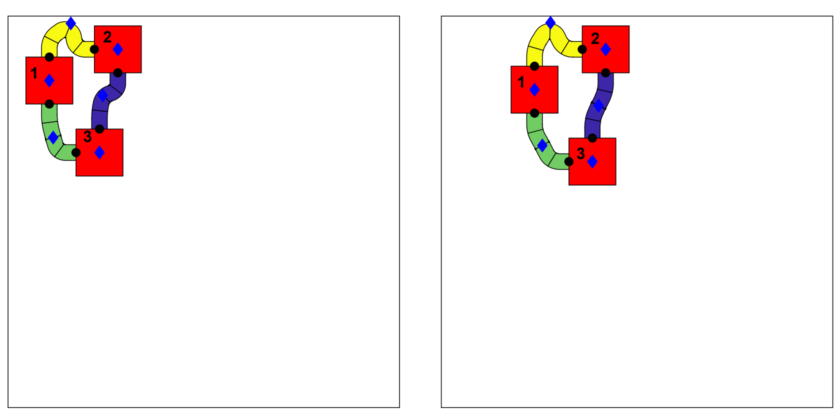
(a) 5 m head loss constraint

(b) 4 m head loss constraint

FIGURE 9: Final layouts for Test B.

Head Loss (m)	objective (m^2)	% increase
≤ 5	0.0156	-
≤ 4	0.0182	16.6

TABLE 4: Results of Test C



(a) 5 m head loss constraint

(b) 4 m head loss constraint

FIGURE 10: Final layouts for Test C.

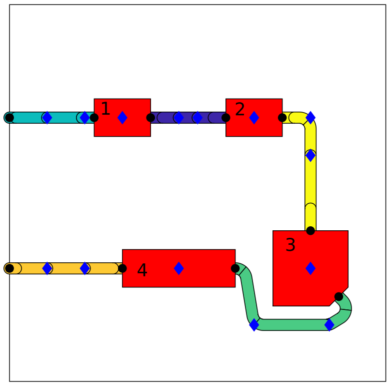
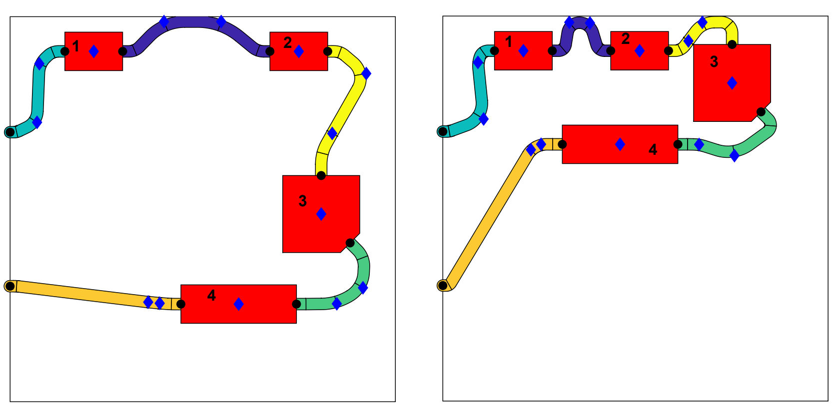


FIGURE 11: Initial layout of power electronics cooling system

Device number	Description	Q_d (W/m^2)	T_{max} (°C)
1	Battery	5000	30
2	Battery	5000	30
3	AC/DC converter	1000	70
4	Heat exchanger	-2000	-

TABLE 5: Device properties

Both optimization problems enforce device temperature constraints as described in the device properties table. In addition, the optimization for minimizing the bounding box had a head loss constraint of 1.5 m. Resulting layouts are shown in Fig. 12, and some corresponding values from the final layouts are listed in Table 6. The objective function and first order optimality value history for the two optimizations are shown in Figs. 13 and 14. In the head loss optimization, device 1 and 2 temperature constraints were active. In the bounding box optimization, the head loss constraint and the device 1 temperature constraint were active. As expected, using sharp angles at the elbows enables designs with smaller bounding boxes. The head loss objective layout has a higher total piping length, but lower head loss. This suggests that elbow geometry is the dominant contributor to head loss.



(a) Optimal layout for pressure objective

(b) Optimal layout for bounding box objective

FIGURE 12: Optimal layouts of power electronics cooling system

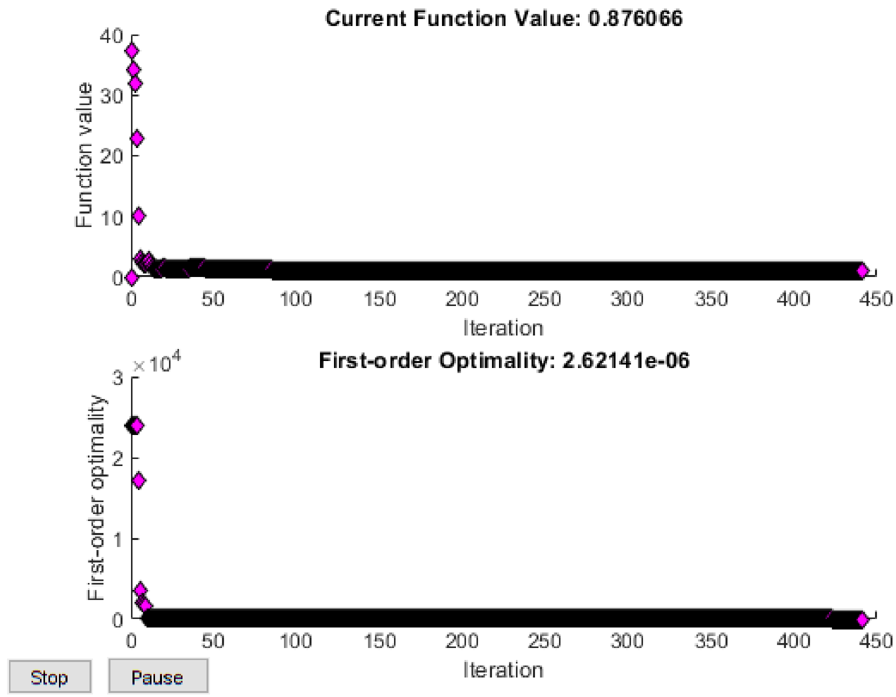


FIGURE 13: Objective function value and first order optimality condition value for pressure objective

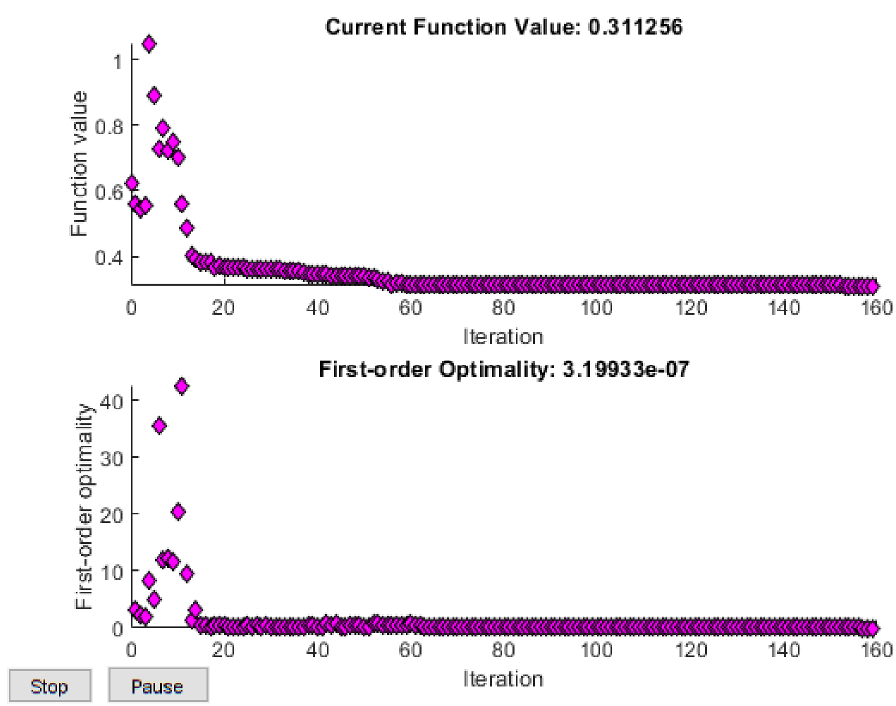


FIGURE 14: Objective function value and first order optimality condition value for bounding box objective

objective	H_l (m)	bounding box (m^2)	T_1 (°C)	T_2 (°C)	T_3 (°C)
head loss	0.876	0.711	30.0	30.0	27.3
bounding box	1.50	0.311	30.0	22.9	17.2

TABLE 6: Power electronics cooling system optimization results. Objective function values are highlighted in gray.

5 Discussion

The above examples demonstrate our technique for simultaneous optimization of device placement and interconnect routing. Starting from the initial condition, both the devices and interconnects move through the domain in search of a locally-optimal solution. A multi-start approach was used in test A of the first example to improve the probability of finding global optima. As discussed in the problem definition determining starting points for the optimization is a complex problem which requires more investigation. Connections in the flow loop are maintained throughout the optimization by using the mapping method described in section 3, and interference between components is prevented via geometric constraints. In addition to finding a physically-feasible solution, this technique accounts for physics considerations. Here we have used 1D lumped parameter and 2D finite element physics models in the calculation of objective functions and constraints. The above results indicate that using these physics-based comparison metrics do indeed influence optimal system layouts. Incorporating physics modeling within the optimization has several potential benefits. It enables discovery of solutions that more accurately reflect realistic system design needs compared to methods that use only geometric metrics. In addition, incorporating these more comprehensive models into early-stage design phases can help reduce the number of overall design cycle iterations, revealing possibly important physics and design interactions before embarking on detailed design or prototyping.

As presented here, this technique can be used successfully for generating optimal layouts of systems that are well-approximated in 2D space. This is an important step beyond previous work by combining layout, routing, and physics into a single problem, but is a starting point for important additional capabilities in solving more comprehensive related problems. In previous work, most studies treated device layout and interconnect routing separately, and also separated the geometric layout optimization from physics-based system evaluation. Several fundamental advances are needed beyond what is presented here to more completely address pressing needs identified across a range of related industries. For example, a number of assumptions were made here, as described in Section 2.2; relaxing these assumptions (e.g., coupling between 1D and 2D physics) will lead to solutions that more accurately reflect real design intent, but will introduce computational challenges. Notably, these couplings may result in nonlinearities which would complicate the strategies presented here. Increasing the complexity of the 2D design space, such non-convex or disconnected domains and allowing devices to rotate, should be explored. In addition, this paper has demonstrated some important aspects of modeling these type of systems, such as preventing interference between devices and routing, modeling devices of complex shape, and implementing constraints based on physics models in multiple dimensions. Most notably, transitioning to 3D packing and routing problems will introduce fundamentally new challenges that do not exist in the 2D problem, such as a vast space of distinct geometric topologies that must be explored. Other creative formulation and solution strategies may be required for these more complete problems.

6 Conclusion

A novel method for simultaneously optimizing the 2D placement of devices and the device interconnect routing paths was presented. Physics-based objectives and constraints were incorporated into the optimization problem, in addition to geometric constraints preventing interference between components. Both 1D lumped parameter and 2D finite element physics models can be used within a single optimization problem. A geometric projection of arbitrary polygons, as well as required sensitivity calculations, were presented that support optimization based on a finite element mesh. A set of design variables was developed that links directly to both physics models and geometric functions. By using a mapping between expanded and reduced design variables, sensitivities can be calculated more easily, and connections between interconnect segments are enforced without requiring additional constraints.

The method was first used to optimize a simple three-device system according to different objective functions. Each objective function resulted in a significantly different optimal layout. This highlights the need for the system designer to understand which aspects of the system are important when developing the optimization problem formulation. A second system with fixed interaction points, complex device shapes, and more routing segments was then optimized. These features are all useful in designing a real device-routing system. The method presented here could lead to faster development of systems, which are smaller and perform better than those designed by conventional design methods.

7 Acknowledgements

This work was supported by the National Science Foundation Engineering Research Center for Power Optimization of Electro-Thermal Systems (POETS) with cooperative agreement EEC-1449548. The authors also sincerely thank Dr. Larry Zeidner and Dr. Neal Herring from the United Technologies Research Center (UTRC), as well as Dr. Joseph Zimmerman from CU Aerospace, for providing valuable feedback while serving as industry partner liaisons for this work.

References

[1] Schafer, M., and Lengauer, T., 1999. "Automated Layout Generation and Wiring Area Estimation for 3D Electronic Modules ". *Journal of Mechanical Design,* ***123***(3), 05, pp. 330–336.

[2] Dong, H., Guarneri, P., and Fadel, G., 2011. "Bi-level Approach to Vehicle Component Layout With Shape Morphing". *Journal of Mechanical Design,* ***133***(4), 05. 041008.

[3] Panesar, A., Brackett, D., Ashcroft, I., Wildman, R., and Hague, R., 2015. "Design Framework for Multifunctional Additive Manufacturing: Placement and Routing of Three-Dimensional Printed Circuit Volumes". *Journal of Mechanical Design,* ***137***(11), 10. 111414.

[4] Yin, S., Cagan, J., and Hodges, P., 2004. "Layout Optimization of Shapeable Components With Extended Pattern Search Applied to Transmission Design ". *Journal of Mechanical Design,* ***126***(1), 03, pp. 188–191.

[5] Landon, M. D., and Balling, R. J., 1994. "Optimal Packaging of Complex Parametric Solids According to Mass Property Criteria". *Journal of Mechanical Design,* ***116***(2), 06, pp. 375–381.

[6] Szykman, S., Cagan, J., and Weisser, P., 1998. "An Integrated Approach to Optimal Three Dimensional Layout and Routing". *Journal of Mechanical Design,* ***120***(3), 09, pp. 510–512.

[7] Szykman, S., and Cagan, J., 1997. "Constrained Three-Dimensional Component Layout Using Simulated Annealing". *Journal of Mechanical Design,* ***119***(1), 03, pp. 28–35.

[8] Aladahalli, C., Cagan, J., and Shimada, K., 2006. "Objective Function Effect Based Pattern Search—Theoretical Framework Inspired by 3D Component Layout". *Journal of Mechanical Design,* ***129***(3), 03, pp. 243–254.

[9] Yin, S., and Cagan, J., 2000. "An Extended Pattern Search Algorithm for Three-Dimensional Component Layout ". *Journal of Mechanical Design,* ***122***(1), 01, pp. 102–108.
[10] Szykman, S., and Cagan, J., 1996. "Synthesis of Optimal Nonorthogonal Routes". *Journal of Mechanical Design,* ***118***(3), 09, pp. 419–424.
[11] Park, J.-H., and Storch, R., 2002. "Pipe-routing algorithm development: case study of a ship engine room design". *Expert Syst. Appl. (UK),* ***23***(3), pp. 299 – 309.
[12] Ren, T., Zhu, Z.-L., Dimirovski, G., Gao, Z.-H., Sun, X.-H., and Yu, H., 2014. "A new pipe routing method for aero-engines based on genetic algorithm". *Proceedings of the Institution of Mechanical Engineers, Part G (Journal of Aerospace Engineering),* ***228***(3), pp. 424 – 34.
[13] Qu, Y., Jiang, D., Gao, G., and Huo, Y., 2016. "Pipe routing approach for aircraft engines based on ant colony optimization". *Journal of Aerospace Engineering,* ***29***(3), p. 04015057.
[14] Van der Velden, C., Bil, C., Yu, X., and Smith, A., 2007. "An intelligent system for automatic layout routing in aerospace design". *Innovations in Systems and Software Engineering,* ***3***(2), pp. 117 – 128.
[15] Liu, C., 2018. "Optimal design of high-rise building wiring based on ant colony optimization". *Cluster Computing*, pp. 1 – 8.
[16] Wu, L., Tian, X., Wang, H., Liu, Q., and Xiao, W., 2019. "Improved ant colony optimization algorithm and its application to solve pipe routing design". *Assembly Automation,* ***39***(1), pp. 45 – 57.
[17] Sigmund, O., 2001. "A 99 line topology optimization code written in matlab". *Structural and Multidisciplinary Optimization,* ***21***(2), 04, pp. 120–127.
[18] Kazemi, H., Vaziri, A., and Norato, J. A., 2018. "Topology Optimization of Structures Made of Discrete Geometric Components With Different Materials". *Journal of Mechanical Design,* ***140***(11), 09. 111401.
[19] Iga, A., Nishiwaki, S., Izui, K., and Yoshimura, M., 2009. "Topology optimization for thermal conductors considering design-dependent effects, including heat conduction and convection". *International Journal of Heat and Mass Transfer,* ***52***(11-12), pp. 2721 – 2732.
[20] Dirker, J., and Meyer, J. P., 2013. "Topology optimization for an internal heat-conduction cooling scheme in a square domain for high heat flux applications". *Journal of Heat Transfer,* ***135***(11).
[21] de Kruijf, N., Zhou, S., Li, Q., and Mai, Y.-W., 2007. "Topological design of structures and composite materials with multiobjectives". *International Journal of Solids and Structures,* ***44***(22-23), pp. 7092 – 109.
[22] Takezawa, A., Yoon, G. H., Jeong, S. H., Kobashi, M., and Kitamura, M., 2014. "Structural topology optimization with strength and heat conduction constraints". *Computer Methods in Applied Mechanics and Engineering,* ***276***, pp. 341 – 61.
[23] Kang, Z., and James, K. A., 2019. "Multimaterial topology design for optimal elastic and thermal response with material-specific temperature constraints". *International Journal for Numerical Methods in Engineering,* ***117***(10), pp. 1019–1037.
[24] James, K., Kennedy, G., and Martins, J., 2014. "Concurrent aerostructural topology optimization of a wing box". *Computers &; Structures,* ***134***, pp. 1 – 17.
[25] Dunning, P., Stanford, B., and Kim, H., 2015. "Coupled aerostructural topology optimization using a level set method for 3d aircraft wings". *Structural and Multidisciplinary Optimization,* ***51***(5), pp. 1113 – 32.
[26] Oktay, E., Akay, H., and Merttopcuoglu, O., 2011. "Parallelized structural topology optimization and cfd coupling for design of aircraft wing structures". *Comput. Fluids (UK),* ***49***(1), pp. 141 – 5.
[27] Zhu, J., Zhang, W., Beckers, P., Chen, Y., and Guo, Z., 2008. "Simultaneous design of components layout and supporting structures using coupled shape and topology optimization technique". *Structural and Multidisciplinary Optimization,* ***36***(1), pp. 29 – 41.

[28] Zhu, J.-H., Guo, W.-J., Zhang, W.-H., and Liu, T., 2017. “Integrated layout and topology optimization design of multi-frame and multi-component fuselage structure systems”. *Structural and Multidisciplinary Optimization,* ***56***(1), pp. 21 – 45.

[29] Zegard, T., and Paulino, G. H., 2016. “Bridging topology optimization and additive manufacturing”. *Structural and Multidisciplinary Optimization,* ***53***(1), pp. 175 – 192.

[30] Tejani, G. G., Savsani, V. J., Patel, V. K., and Savsani, P. V., 2018. “Size, shape, and topology optimization of planar and space trusses using mutation-based improved metaheuristics”. *Journal of Computational Design and Engineering,* ***5***(2), pp. 198 – 214.

[31] Zhang, X. S., Paulino, G. H., and Ramos, A. S., 2018. “Multi-material topology optimization with multiple volume constraints: a general approach applied to ground structures with material nonlinearity”. *Structural and Multidisciplinary Optimization,* ***57***(1), pp. 161 – 182.

[32] Norato, J., Bell, B., and Tortorelli, D., 2015. “A geometry projection method for continuum-based topology optimization with discrete elements”. *Computer Methods in Applied Mechanics and Engineering,* ***293***, pp. 306 – 27.

[33] Zhang, S., Norato, J. A., Gain, A. L., and Lyu, N., 2016. “A geometry projection method for the topology optimization of plate structures”. *Structural and Multidisciplinary Optimization,* ***54***(5), pp. 1173 – 1190.

[34] Bendsøe, M., 1989. “Optimal shape design as a material distribution problem”. *Structural optimization,* ***1***(4), p. 193–202.

[35] Hughes, T. J. R., 2000. *The Finite Element Method*. Dover Publications.

[36] Hormann, K., and Agathos, A., 2001. “The point in polygon problem for arbitrary polygons”. *Computational Geometry: Theory and Applications,* ***20***(3), pp. 131 – 44.

[37] Rennels, D. C., and Hudson, H. M., 2012. *Pipe Flow: A Practical and Comprehensive Guide*. "".

[38] Tarquin, A. J., and Dowdy, J., 1989. “Optimal pump operation in water distribution”. *Journal of Hydraulic Engineering,* ***115***(2), pp. 158–168.

[39] Nakayama, Y., 2018. “Chapter 7 - flow in pipes”. In *Introduction to Fluid Mechanics (Second Edition)*, Y. Nakayama, ed., second edition ed. Butterworth-Heinemann, pp. 135 – 161.

[40] Martins, J., and Hwang, J., 2013. “Review and unification of methods for computing derivatives of multidisciplinary computational models”. *AIAA Journal,* ***51***(11), pp. 2582 – 99.

[41] Sunday, D., 2012. Distance between 3d lines & segments.

A Distance between segments MATLAB code

```
function [sqDist, dd_dx] = sqDistance(xa0, xaf, xb0, xbf)
  %segmentDistance calculates minimum squared distance between
  % two line segments
  % xa0, xaf - end points of first segment - row vector
  % xb0, xbf - end points of second segment - row vector
  % returns dd_dx in order [xa0, xaf, xb0, xbf]

  dim = size(xa0,2);
  I = eye(dim,dim);
  z = zeros(dim,dim);

  u = xaf - xa0;
  dudx = [-I I z z];
  v = xbf - xb0;
  dvdx = [z z -I I];
  w = xa0 - xb0;
  dwdx = [I z -I z];
  a = u*u';
  b = u*v';
  c = v*v';
  d = u*w';
  e = v*w';
  dadx = 2*u*dudx;
  dbdx = v*dudx + u*dvdx;
  dcdx = 2*v*dvdx;
  dddx = w*dudx + u*dwdx;
  dedx = w*dvdx + v*dwdx;

  D = a*c-b^2;
  dDdx = a*dcdx + c*dadx - 2*b*dbdx;

  sD = D;
  tD = D;
  dsddx = dDdx;
  dtddx = dDdx;

  eps = 1e-5; % tolerance for nearly parallel
  if D < eps
    % segments are nearly parallel
    sN = 0;
    sD = 1;
    tN = e;
    tD = c;
    dsndx = zeros(1,4*dim);
```

```
      dsddx = zeros(1,4*dim);
      dtndx = dedx;
      dtddx = dcdx;
    else
      sN = (b*e-c*d);
      tN = (a*e-b*d);
      dsndx = e*dbdx + b*dbdx - d*dcdx - c*dddx;
      dtndx = a*dedx + e*dadx - d*dbdx - b*dddx;
      if sN < 0
        sN = 0;
        tN = e;
        tD = c;
        dsndx = zeros(1,4*dim);
        dtndx = dedx;
        dtddx = dcdx;
      elseif sN > sD
        sN = sD;
        tN = e+b;
        tD = c;
        dsndx = dsddx;
        dtndx = dedx + dbdx;
        dtddx = dcdx;
      end
    end

    if tN < 0
      tN = 0;
      dtndx = zeros(1,4*dim);
      if -d < 0
        sN = 0;
        dsndx = zeros(1,4*dim);
      elseif -d > a
        sN = sD;
        dsndx = dsddx;
      else
        sN = -d;
        sD = a;
        dsndx = -dddx;
        dsddx = dadx;
      end
    elseif tN > tD
      tN = tD;
      dtndx = dtddx;
      if (-d+b) < 0
        sN = 0;
        dsndx = zeros(1,4*dim);
      elseif (-d+b) > a
        sN = sD;
```

```
      dsndx = dsddx;
    else
      sN = -d + b;
      sD = a;
      dsndx = -dddx + dbdx;
      dsddx = dadx;
    end
  end

  if abs(sN) < eps
    sC = 0;
    dscdx = zeros(1,4*dim);
  else
    sC = sN/sD;
    dscdx = (1/sD)*dsndx - (sN/(sD^2))*dsddx;
  end

  if abs(tN) < eps
    tC = 0;
    dtcdx = zeros(1,4*dim);
  else
    tC = tN/tD;
    dtcdx = (1/tD)*dtndx - (tN/(tD^2))*dtddx;
  end

  dP = w+(sC*u)-(tC*v);
  dPdw = I;
  dPdu = sC*I;
  dPdv = -tC*I;
  dPdsc = u';
  dPdtc = -v';
  ddP_dx = [dPdw dPdu dPdv dPdsc dPdtc]*[dwdx; dudx; dvdx; dscdx; dtcdx];
  sqDist = dP*dP';
  dd_dx = 2*dP*ddP_dx;
end
```