

SIMULTANEOUS PACKING AND ROUTING OPTIMIZATION USING GEOMETRIC PROJECTION

Alex Jessee¹, Satya R. T. Peddada², Danny J. Lohan², James T. Allison², Kai A. James¹

¹Department of Aerospace Engineering

²Department 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.

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

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

70 1.1 Objectives and Contributions

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

- 76 1. We present a novel technique that supports simultaneous optimization of device placement and
77 interconnect paths, whereas existing methods treat device layout and interconnect routing separately
78 (e.g., optimal routing with fixed layout).
- 79 2. Physics-based objectives and constraints were incorporated into the optimization problem, in
80 addition to geometric constraints that prevent interference between devices and interconnects. Both
81 1D lumped parameter and 2D finite element physics models are used within a single optimization
82 problem to support physics-based evaluation.
- 83 3. We use the geometric projection method (GPM) of Norato *et al.* [32], which is an alternative to the
84 well-established SIMP (Solid Isotropic Material with Penalization) method design parameterization
85 [34] for solving the optimization problem.
- 86 4. We demonstrated the effectiveness of the proposed method via the solution of two device-routing
87 test cases that utilize physics-based simulations.

88 The new simultaneous approach makes significant system volume reduction possible. The projection
 89 method of Norato *et al.* [32] is extended to allow devices of arbitrary polygonal shape to be projected.

90 Sensitivity analysis for this projection is provided to allow the efficient use of gradient-based
 91 optimization methods. Examples presented later in this article consider thermal conduction on the
 92 continuum level using the finite element method, and a lumped parameter pipe flow model. The methods
 93 presented here, however, could be extended to model other combinations of physics; for example, thermal-
 94 electric or structural-fluid systems. Section 2 presents the models used to simulate the physics response
 95 of the system. Section 3 states the optimization problem and presents the derived function sensitivities.
 96 Finally, the method is demonstrated in Section 4 through the optimization of two device-routing systems.

97 2 Physics models

98 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:

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

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

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

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

$$KT = P \quad (4)$$

106 Equation (4) is solved for the temperature field vector T , where K is the global thermal stiffness matrix
 107 assembled from element stiffness matrices, k_{el} , defined in Eqn. (5), and P is the global load vector
 108 assembled from element load vectors, p_{el} , defined in Eqn. (6).

$$k_{el} = \int_{\Omega_e} \mathbf{B}^T \kappa \mathbf{B} d\Omega - \int_{\partial\Omega_h} h \mathbf{N} \mathbf{N}^T d\partial\Omega_h \quad (5)$$

$$p_{el} = \int_{\Omega_e} Q \mathbf{N} d\Omega + \int_{\partial\Omega_h} h T_{env} \mathbf{N} d\partial\Omega_h. \quad (6)$$

109 Here, \mathbf{N} and \mathbf{B} are element shape function and shape function gradients, respectively. These equations
 110 also include convection boundary conditions on the $\partial\Omega_h$ portion of the boundary. The temperature of the
 111 convecting fluid is T_{env} , and the convection coefficient is h (assumed constant here).
 112

113 Here the geometric projection method of Norato *et al* [32] is used. In the original projection method
 114 work, the new parameterization approach was used to optimize structures while ensuring that the resulting
 115 design could be made from stock materials, such as structural beams with standard shapes and sizes. We
 116 discovered that this method can be extended beneficially to the combined layout and routing optimization
 117 problem. The geometric projection method is used to create routing designs that can be manufactured
 118 out of standard circular cross-section pipes. The geometric parameterization involves design variables

119 that facilitate convenient derivation of lumped parameter model (Sec. 2.2) sensitivities. The remainder of
 120 this section will give a brief overview of the geometric projection method, and detail changes made for
 121 use with the routing and packaging problem.

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

$$\mathbf{k}_i = (\rho_{\min} + (1 - \rho_{\min})\rho_i^p) \int_{\Omega_e} \mathbf{B}^T \boldsymbol{\kappa} \mathbf{B} d\Omega = (\rho_{\min} + (1 - \rho_{\min})\rho_i^p) \mathbf{k}_0, \quad (7)$$

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

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

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

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

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

145 The radius r determines the width of the grey area projected on to the mesh by the bar. A smaller radius
 146 will more accurately represent the bar geometry as a projection of mostly ones and zeros. To ensure that
 147 any element which touches a bar has a nonzero density, a radius that circumscribes the square elements
 148 is used in this paper. The radius must be less than half of the bar width in order for Eqn.(9) to correctly
 149 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, \mathbf{c}_d , and a set of vectors, \mathbf{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

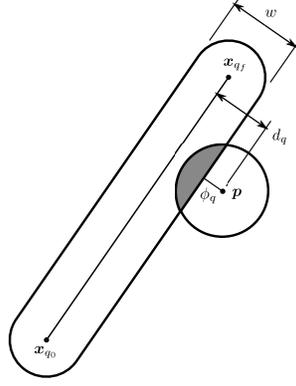


FIGURE 1: Bar projection



FIGURE 2: Device projection

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

$$\mathbf{x}_{e_0} = \mathbf{x}_i = \mathbf{c}_d + \mathbf{b}_i \quad (10)$$

$$\mathbf{x}_{e_f} = \mathbf{x}_{i+1} = \mathbf{c}_d + \mathbf{b}_{i+1}. \quad (11)$$

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

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

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

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

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

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

162 Devices may also add or remove heat from the domain. The projection in Eqn. (12) will be used to

163 model this effect. Rather than assuming a constant internal heat generation Q across all devices, each
 164 device will have its own Q_d value. Element load vectors are then modified using this Q_d and the device
 165 density.

$$\mathbf{p}_e = \sum_{d=1}^{N_d} \rho_{de}^p Q_d \int_{\Omega_e} \mathbf{N} d\Omega_e = \sum_{d=1}^{N_d} \rho_{de}^p Q_d \mathbf{p}_0 \quad (14)$$

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

167 2.2 Lumped parameter pipe flow model

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

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

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

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

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

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

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

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

$$V_i = \frac{Q}{A_i} \quad (17)$$

187 Here, Q is the volumetric flow rate (uniform for a series flow loop), and A_i is the cross sectional area of
 188 the pipe at location i . Head loss is determined next, which is a proxy metric for energy loss between
 189 points 1 and 2 for reasons other than velocity change. Head loss, in units of length, is a standard metric
 190 used in describing flow system properties, including pump efficiency and system characterization curves
 191 [38]. Models for estimating head loss for many different pipe flow system components can be found in

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

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

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

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

$$K_i^s = f_i \frac{l_i}{d_i}, \quad (19)$$

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

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

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— \mathbf{a} and \mathbf{b} —based on the endpoints of two connected segments:

$$\mathbf{a} = \mathbf{x}_{a_f} - \mathbf{x}_{a_0} \quad (21)$$

$$\mathbf{b} = \mathbf{x}_{b_f} - \mathbf{x}_{b_0}. \quad (22)$$

199 From the definition of the dot product, we obtain:

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

200 where:

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

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

$$\alpha = \pi - \theta. \quad (25)$$

204 The clipped length is calculated using:

$$l_c = r_i \sqrt{\frac{1+v}{1-v}}. \quad (26)$$

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

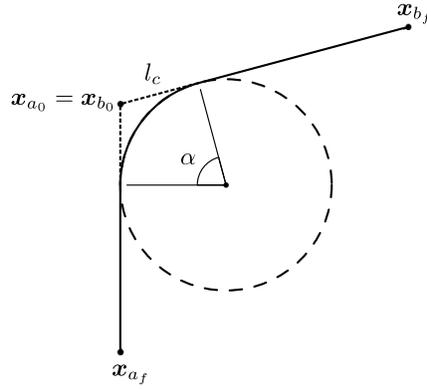


FIGURE 3: Pipe elbow geometry

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

$$f_i = 0.3164Re_i^{-0.25}, \quad (27)$$

209 with Reynolds number:

$$Re_i = \frac{V_i d_i \rho_m}{\mu}, \quad (28)$$

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

212 3 Optimization problem and sensitivity analysis

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

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

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

226 In addition to c_d and b_i introduced in Sec. 2, devices may have ports with location p_i relative to
 227 the reference point. Ports are the required locations for interconnect attachment to each device. As the
 228 reference point moves, the polygon and ports will move with it. Device shape, size, and port location are
 229 held fixed during the optimization, so the only design variable for each device is c_d . In some optimization
 230 studies, it may be useful to omit the reference point corresponding to a device from the set of optimization
 231 variables, holding the device fixed in a particular location. This can also be used to specify fixed inputs
 232 and outputs of the flow loop.

233 Pairs of ports are connected via physical interconnects, and device connection topology is assumed
 234 to be given (and unchanging) here. Each interconnect is represented here using one or more straight

235 geometric segments. Increasing the number of segments in a connection supports consideration of
 236 approximately curved (and more complex) interconnect geometries, but increases computational expense.
 237 Interconnect segment i is associated with parameters for its start and end points, \mathbf{x}_{i_0} and \mathbf{x}_{i_f} , respectively,
 238 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:

$$x_i - x_j = 0 \quad (29)$$

$$(x_i - x_j)^2 - \varepsilon \leq 0, \quad (30)$$

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

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

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

$$\mathbf{z}' := [\mathbf{c}_1, \dots, \mathbf{c}_{n_d}, \mathbf{x}_{1_0}, \mathbf{x}_{1_f}, w_1, \dots, \mathbf{x}_{n_s_0}, \mathbf{x}_{n_s_f}, w_{n_s}]^T$$

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

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

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

$$\mathbf{z}' = \mathbf{M}\mathbf{z} + \mathbf{P} \quad (31)$$

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

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

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

$$\frac{df}{dz} = \frac{\partial f}{\partial z'} \frac{dz'}{dz}, \quad (32)$$

273

$$\frac{dz'}{dz} = M. \quad (33)$$

274 The reduced design variable vector is used as the optimization vector by the solution algorithm. To
 275 maintain correspondence between reduced and expanded vectors, the mapping defined in Eqn. (31) is
 276 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:

$$\min_x f(\mathbf{x}, \mathbf{T}) \quad (34a)$$

$$\text{s.t.} \quad \mathbf{g}_{\text{phys}}(\mathbf{x}, \mathbf{T}) \leq 0 \quad (34b)$$

$$\mathbf{g}_{\text{dd}}(\mathbf{x}) \leq 0 \quad (34c)$$

$$\mathbf{g}_{\text{sd}}(\mathbf{x}) \leq 0 \quad (34d)$$

$$\mathbf{g}_{\text{ss}}(\mathbf{x}) \leq 0 \quad (34e)$$

$$\text{where:} \quad \mathbf{K}(\mathbf{x})\mathbf{T} = \mathbf{P}(\mathbf{x}) \quad (34f)$$

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

284 3.1 Objective function and physics-based constraints

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

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

$$f_1(\mathbf{x}, \mathbf{T}) = H_L \quad (35)$$

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

292 mapping). The total derivative using the chain rule is:

$$\frac{dH_L}{d\mathbf{x}'} = \frac{\partial H_L}{\partial \mathbf{x}'} + \sum_{i=1}^{n_{pe}} \frac{\partial H_L}{\partial K_i} \frac{dK_i}{d\mathbf{x}'} \quad (36)$$

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

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

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

$$\frac{\partial H_L}{\partial K_i} = \frac{1}{A_i^2} \frac{\dot{w}^2}{2g\rho_w^2} \quad (38)$$

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

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

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

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

297

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

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

$$\frac{dK_i^e}{dd_i} = \frac{\partial K_i^e}{\partial d_i} + \frac{\partial K_i^e}{\partial f_i} \frac{df_i}{dd_i} \quad (43)$$

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

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

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

302 With respect to elbow radius, the sensitivity is:

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

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

$$\frac{dK_i^e}{d\mathbf{x}_{a_0}} = \frac{\partial K_i^e}{\partial \alpha_i} \frac{d\alpha_i}{d\mathbf{x}_{a_0}} - \frac{2f_i}{d_i} \frac{dl_c}{d\mathbf{x}_{a_0}}, \quad (47)$$

with:

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

303 The derivatives of angle α are:

$$\frac{d\alpha_i}{d\mathbf{x}_{a_0}} = \frac{\partial \alpha_i}{\partial v} \frac{dv}{d\mathbf{x}_{a_0}}, \quad (49)$$

304 with:

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

305 The required derivatives of the clipped length are as follows:

$$\frac{dl_c}{dr_i} = \sqrt{\frac{1+v}{1-v}} \quad (51)$$

306

$$\frac{dl_c}{d\mathbf{x}_{a_0}} = \frac{\partial l_c}{\partial v} \frac{dv}{d\mathbf{x}_{a_0}} \quad (52)$$

307 where:

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

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

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

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

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

$$f_2(\mathbf{x}, \mathbf{T}) = T(\mathbf{c}_d). \quad (56)$$

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

$$\frac{df_2}{dx'_i} = \frac{\partial f_2}{\partial x'_i} + \Psi^T \frac{d\mathbf{R}}{dx'_i}, \quad (57)$$

312 where Ψ is the *adjoint vector*, which can be calculated with the following equation:

$$\Psi = \left[\frac{\partial \mathbf{R}}{\partial \mathbf{y}} \right]^{-T} \left[\frac{\partial f_2}{\partial \mathbf{y}} \right]^T. \quad (58)$$

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

$$\mathbf{R} = \mathbf{K}\mathbf{T} - \mathbf{P} = \mathbf{0} \quad (59)$$

314 The vector \mathbf{y} in Eqn. (58) is the unknown vector. In the finite element problem the unknowns are
 315 $\mathbf{y} = [\mathbf{P}^p, \mathbf{T}^f]^T$. The vector \mathbf{P}^p is the flux at nodes where prescribed temperature boundary conditions
 316 are applied, and \mathbf{T}^f are temperatures at all remaining nodes. Equation (59) is partitioned into blocks,
 317 p and f , corresponding to the *prescribed* and *free* (unknown) degrees of freedom, respectively. The
 318 derivative of the residual with respect to the unknown vector results in a partitioned matrix:

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

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

$$T(c_d) = \mathbf{N}^T(\xi, \eta) \mathbf{T}_{\text{el}}, \quad (61)$$

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

$$\frac{\partial f_2}{\partial \mathbf{x}'} = \mathbf{B}^T \mathbf{T}_{\text{el}} \quad (62)$$

325

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

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

$$\frac{d\mathbf{R}}{dx'_i} = \frac{\partial \mathbf{R}}{\partial \rho_e} \frac{d\rho_e}{dx'_i} \quad (64)$$

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

$$\frac{df_2}{dx'_i} = \frac{\partial f_2}{\partial x'_i} + \sum_{e=1}^{N_e} p(1 - \rho_{\min}) \rho_e^{(p-1)} \Psi_e^T \mathbf{k}_0 \mathbf{T}_e \frac{d\rho_e}{dx'_i} + \sum_{d=1}^{N_d} Q_d \sum_{e=1}^{N_e} p \rho_{\text{de}}^{(p-1)} \Psi_e^T \mathbf{p}_0 \frac{d\rho_{\text{de}}}{dx'_i}, \quad (65)$$

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

331 The derivative of the geometric projection is the final part of Eqn. (65). The derivative of density

332 resulting from the merge in Eqn. (13) is:

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

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

$$\frac{d\rho_d}{dc_d} = \rho_d^{1-p} \sum_{e=1}^{N_e} (\tilde{\rho}_e)^{p-1} \frac{\partial \tilde{\rho}_e}{\partial c_d}, \quad (67)$$

335 where:

$$\frac{\partial \tilde{\rho}_e}{\partial c_d} = \frac{\partial \tilde{\rho}_e}{\partial \mathbf{x}_{e_0}} \frac{\partial \mathbf{x}_{e_0}}{\partial c_d} + \frac{\partial \tilde{\rho}_e}{\partial \mathbf{x}_{e_f}} \frac{\partial \mathbf{x}_{e_0}}{\partial c_d}. \quad (68)$$

336 Derivatives with respect to edge ends can again be found in Norato et al. [32], and appear here as
337 $\partial \tilde{\rho}_e / \partial \mathbf{x}_{e_0}$ and $\partial \tilde{\rho}_e / \partial \mathbf{x}_{f_0}$. From the definition of edge endpoints in Eqn. (10) and (11), derivatives $\partial \mathbf{x}_{e_0} / \partial \mathbf{c}$
338 and $\partial \mathbf{x}_{e_0} / \partial \mathbf{c}$ are the identity matrix. The density derivative of any element inside the device polygon is
339 set to $\partial \rho_d / \partial \mathbf{c} = 0$. To justify making the sensitivity of interior elements zero, imagine perturbing the
340 center of the device by a small amount. Most elements inside the polygon are still inside the polygon, so
341 there is no change in the density. Some elements near the edges may have switched from being inside
342 the polygon to in the edge bar, or vice versa. These elements will be near the bar center and should still
343 have full density.

344 3.2 Geometric constraints

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

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

$$\frac{dg_{dd}}{d\mathbf{c}_i} = 2(\mathbf{c}_j - \mathbf{c}_i) \quad (70)$$

$$\frac{dg_{dd}}{d\mathbf{c}_j} = -2(\mathbf{c}_j - \mathbf{c}_i) \quad (71)$$

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

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

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

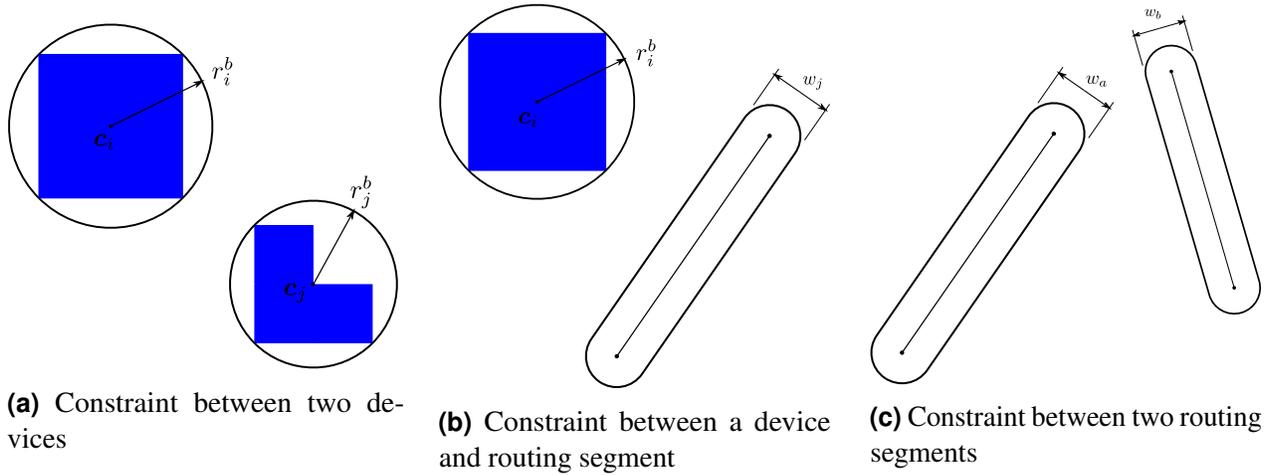


FIGURE 4: Geometric constraints

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

$$\frac{dd_{ij}}{dc_i} = \begin{cases} \frac{b}{\|b\|} & \mathbf{a} \cdot \mathbf{b} \leq 0 \\ \frac{1}{\|g\|} \left(\mathbf{I} - \frac{1}{\|a\|^2} (\mathbf{a} \otimes \mathbf{a}) \right) \mathbf{g} & 0 < \mathbf{a} \cdot \mathbf{b} < \mathbf{a} \cdot \mathbf{a} \\ \frac{e}{\|e\|} & \mathbf{a} \cdot \mathbf{b} \geq \mathbf{a} \cdot \mathbf{a}. \end{cases} \quad (73)$$

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

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

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

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

368 4 Results

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

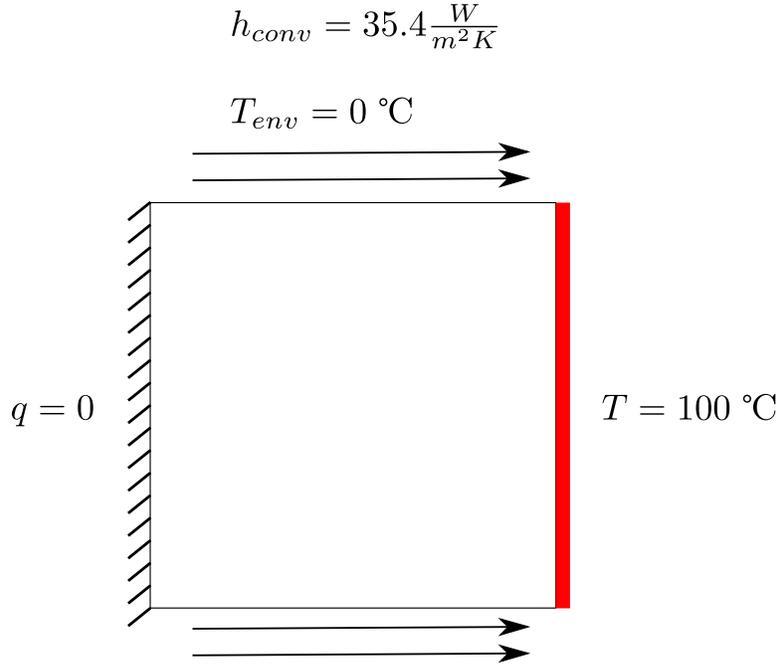


FIGURE 5: Boundary condition for thermal finite element analysis

376 if an interconnect links ports A and B , many options may exist for how this interconnect passes around
 377 various other interconnects and devices in the system. Exploring all possible geometric topologies for
 378 a single system architecture becomes exponentially more complex as the number of devices increases.
 379 Generating and selecting distinct geometric topologies is a topic of ongoing work, and is beyond the
 380 scope of this article. Boundary conditions for the thermal problem will be the same for both systems. Top
 381 and bottom edges of the domain will have convection boundary conditions with a convection coefficient
 382 of $h = 35.4 \text{ W}/(\text{m}^2\text{K})$ and environment temperature of $0 \text{ } ^\circ\text{C}$. The right edge has a fixed temperature of 100
 383 $^\circ\text{C}$, and the left edge is insulated. The MATLAB sequential quadratic programming (SQP) optimization
 384 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 $0.06 \text{ m} \times 0.06 \text{ m}$ square devices, each generating $3,000 \text{ W}/\text{m}^2$ 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 $54 \text{ W}/(\text{mK})$, and $0.032 \text{ W}/(\text{mK})$ is used for void, resulting in $\rho_{\min} = 5.86 \times 10^{-4}$. The domain for the optimization is $0.5 \text{ m} \times 0.5 \text{ m}$. 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:

$$\min_x f(\cdot) = H_l \tag{76a}$$

$$\text{subject to: } T(c_1), T(c_2), T(c_3) \leq T_c \tag{76b}$$

$$g_{\text{geo}} \leq 0 \tag{76c}$$

$$\text{where: } \mathbf{K}(\mathbf{x})\mathbf{T} = \mathbf{P}(\mathbf{x}). \tag{76d}$$

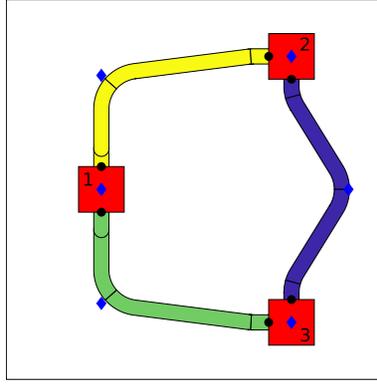


FIGURE 6: Initial system layout for test A, B, and C. Device numbers shown. Blue diamonds are design variable locations.

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

$$\min_x f(\cdot) = T(c_3) \quad (77a)$$

$$\text{subject to: } T(c_1), T(c_2) \leq T_c \quad (77b)$$

$$H_l \leq H_{l_c} \quad (77c)$$

$$g_{\text{geo}} \leq 0 \quad (77d)$$

$$\text{where: } \mathbf{K}(\mathbf{x})\mathbf{T} = \mathbf{P}(\mathbf{x}). \quad (77e)$$

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

$$\min_x f(\cdot) = (\max(\bar{x}) - \min(\bar{x}))(\max(\bar{y}) - \min(\bar{y})) \quad (78a)$$

$$\text{subject to: } T(c_1), T(c_2), T(c_3) \leq T_c \quad (78b)$$

$$H_l \leq H_{l_c} \quad (78c)$$

$$g_{\text{geo}} \leq 0 \quad (78d)$$

$$\text{where: } \mathbf{K}(\mathbf{x})\mathbf{T} = \mathbf{P}(\mathbf{x}). \quad (78e)$$

385 Where \bar{x} and \bar{y} are the set of x and y coordinates of device reference points and bar segment end points.
 386 The critical device temperature, T_c , for all three devices is 30 °C. Head loss constraints, H_{l_c} , of 4 m and
 387 5 m will be used to quantify the effect of this constraint on the outcome. Here all geometric constraints
 388 from Sec. 3.2 have been lumped together into g_{geo} . The function in Eqn. (78a) represents a rectangular
 389 bounding box containing all free devices and interconnects that is aligned with the x and y axes.

$$\min_x A(\cdot)$$

(Objective: Bounding box area)

$$\text{subject to: } T_{d_{i=1..4}} \leq T_{d_{max}}$$

(Max. device temp. constraint)

$$T_f \leq T_{f_{max}}$$

(Max. fluid temp. constraint)

$$H_l \leq H_{l_c}$$

(Max. head loss constraint)

$$g_{\text{geo}} \leq 0$$

(Geometric constraints)

$$\text{where: } \mathbf{K}(\mathbf{x})\mathbf{T} = \mathbf{P}(\mathbf{x}).$$

(Physics-based model eqns.)

390 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

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

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

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

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

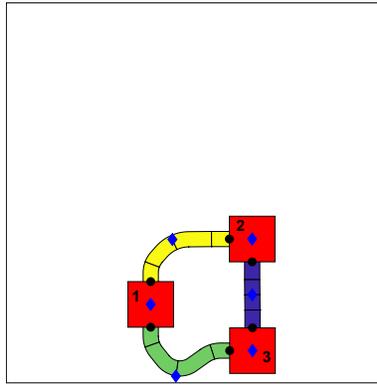


FIGURE 7: Optimal layout of Test A

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 53rd 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.

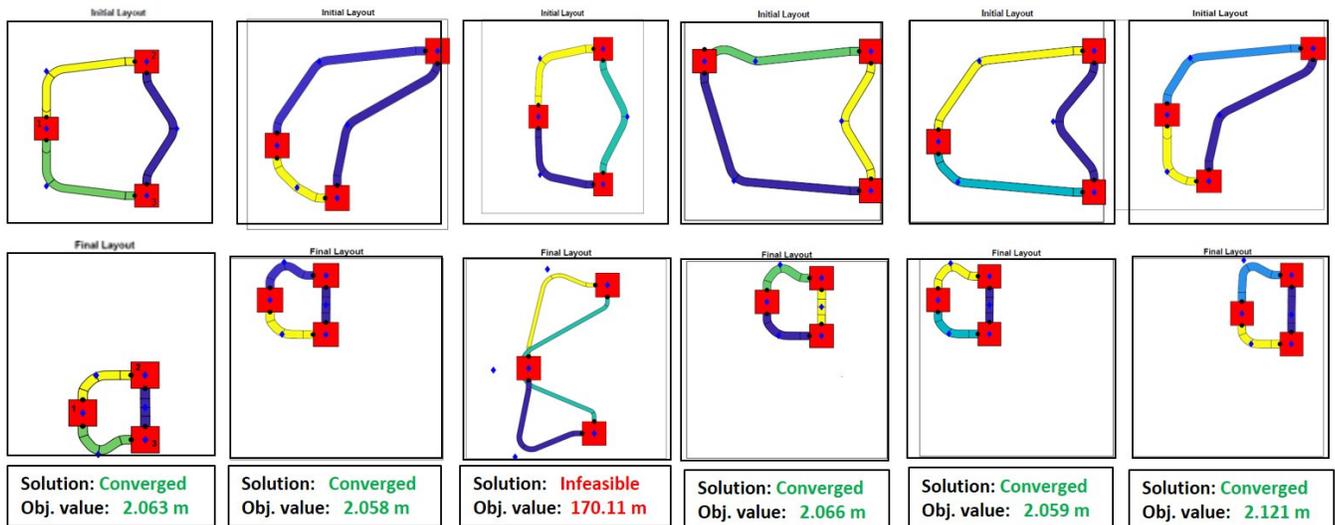


FIGURE 8: Six different initial layouts (first row) for Test A with simple head loss objective function and their corresponding final layouts (second row) along with their objective function values.

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

TABLE 3: Results of Test B

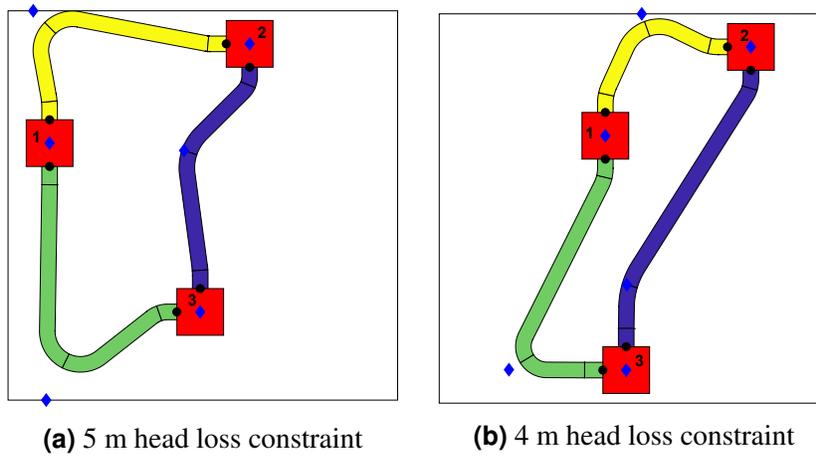


FIGURE 9: Final layouts for Test B.

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

TABLE 4: Results of Test C

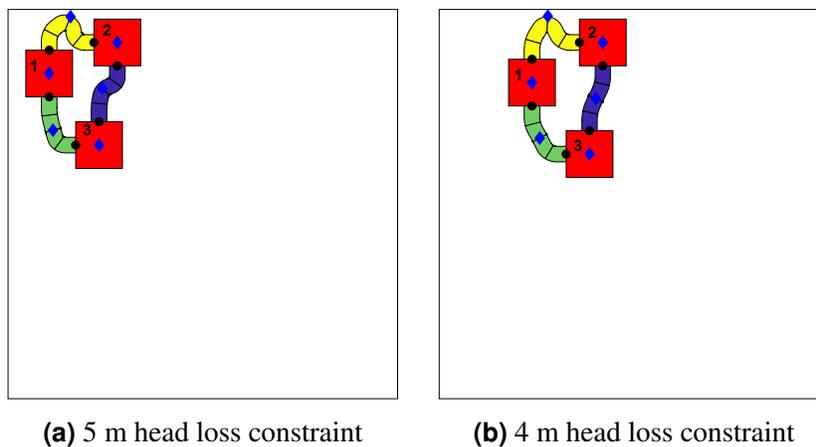


FIGURE 10: Final layouts for Test C.

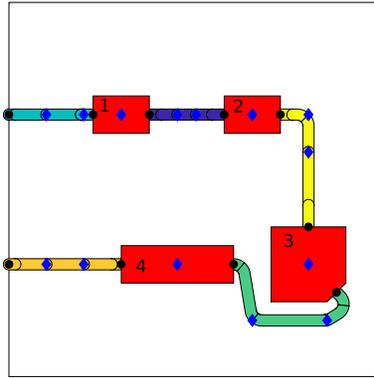
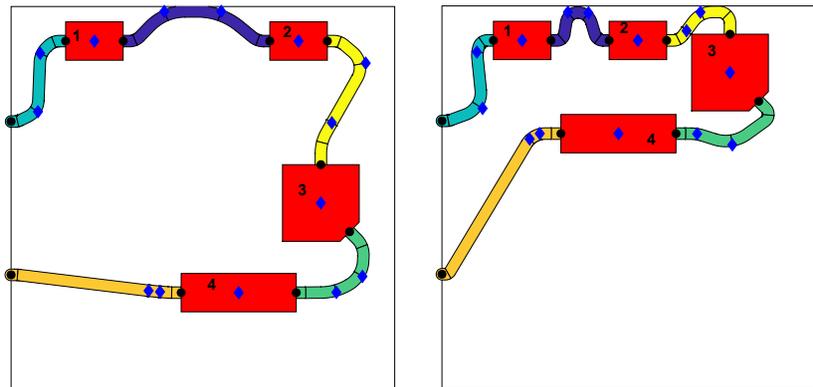


FIGURE 11: Initial layout of power electronics cooling system

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

430 Both optimization problems enforce device temperature constraints as described in the device properties
 431 table. In addition, the optimization for minimizing the bounding box had a head loss constraint of 1.5 m.
 432 Resulting layouts are shown in Fig. 12, and some corresponding values from the final layouts are listed
 433 in Table 6. The objective function and first order optimality value history for the two optimizations are
 434 shown in Figs. 13 and 14. In the head loss optimization, device 1 and 2 temperature constraints were
 435 active. In the bounding box optimization, the head loss constraint and the device 1 temperature constraint
 436 were active. As expected, using sharp angles at the elbows enables designs with smaller bounding boxes.
 437 The head loss objective layout has a higher total piping length, but lower head loss. This suggests that
 438 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

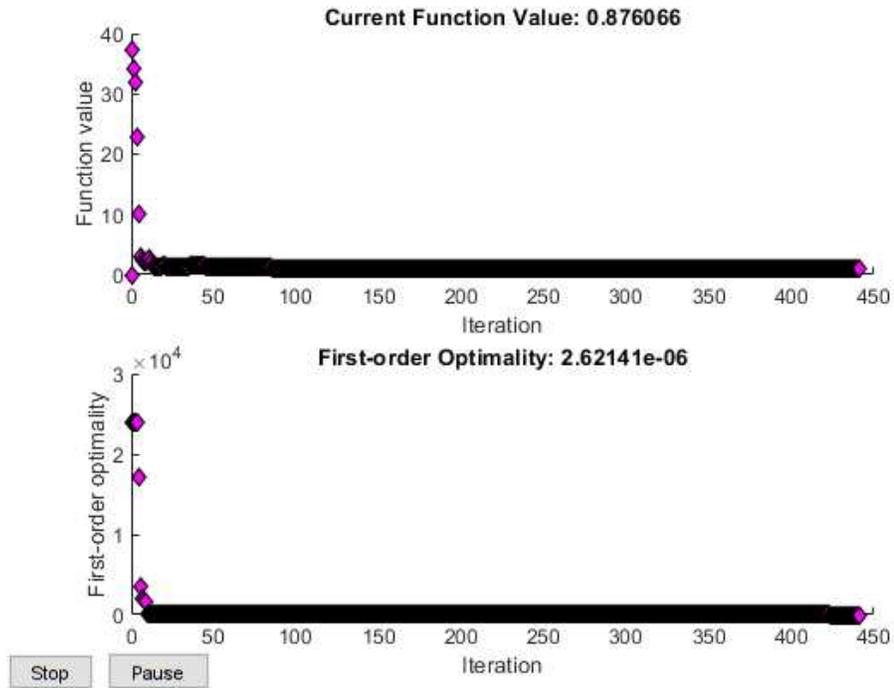


FIGURE 13: Objective function value and first order optimality condition value for pressure objective

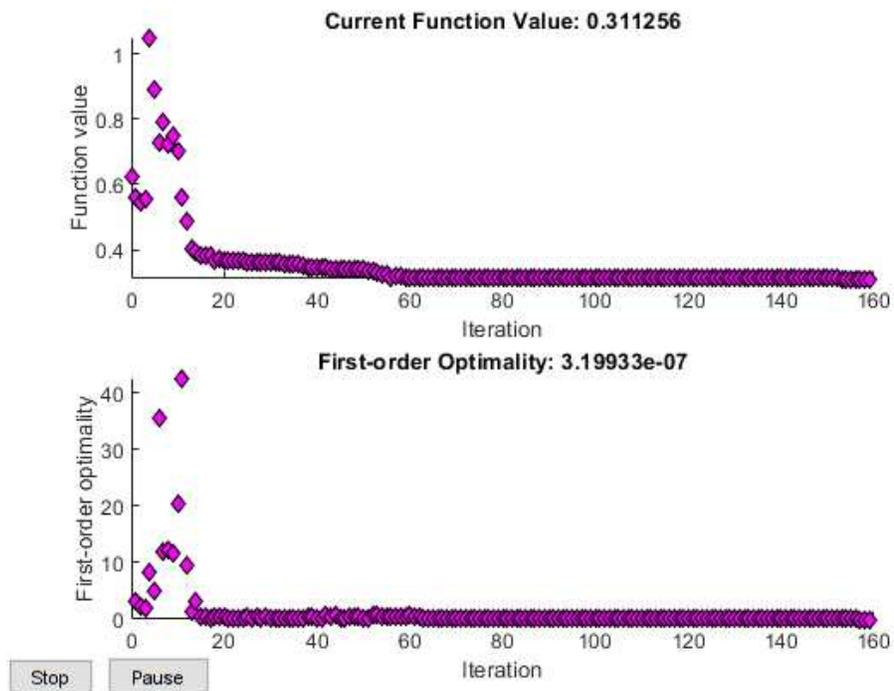


FIGURE 14: Objective function value and first order optimality condition value for bounding box objective

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.

474 6 Conclusion

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

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

491 7 Acknowledgements

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

497 References

- 498 [1] Schafer, M., and Lengauer, T., 1999. “Automated Layout Generation and Wiring Area Estimation
499 for 3D Electronic Modules”. *Journal of Mechanical Design*, **123**(3), 05, pp. 330–336.
- 500 [2] Dong, H., Guarneri, P., and Fadel, G., 2011. “Bi-level Approach to Vehicle Component Layout
501 With Shape Morphing”. *Journal of Mechanical Design*, **133**(4), 05, 041008.
- 502 [3] Panesar, A., Brackett, D., Ashcroft, I., Wildman, R., and Hague, R., 2015. “Design Framework for
503 Multifunctional Additive Manufacturing: Placement and Routing of Three-Dimensional Printed
504 Circuit Volumes”. *Journal of Mechanical Design*, **137**(11), 10, 111414.
- 505 [4] Yin, S., Cagan, J., and Hodges, P., 2004. “Layout Optimization of Shapeable Components With
506 Extended Pattern Search Applied to Transmission Design”. *Journal of Mechanical Design*, **126**(1),
507 03, pp. 188–191.
- 508 [5] Landon, M. D., and Balling, R. J., 1994. “Optimal Packaging of Complex Parametric Solids
509 According to Mass Property Criteria”. *Journal of Mechanical Design*, **116**(2), 06, pp. 375–381.
- 510 [6] Szykman, S., Cagan, J., and Weisser, P., 1998. “An Integrated Approach to Optimal Three
511 Dimensional Layout and Routing”. *Journal of Mechanical Design*, **120**(3), 09, pp. 510–512.
- 512 [7] Szykman, S., and Cagan, J., 1997. “Constrained Three-Dimensional Component Layout Using
513 Simulated Annealing”. *Journal of Mechanical Design*, **119**(1), 03, pp. 28–35.
- 514 [8] Aladahalli, C., Cagan, J., and Shimada, K., 2006. “Objective Function Effect Based Pattern
515 Search—Theoretical Framework Inspired by 3D Component Layout”. *Journal of Mechanical
516 Design*, **129**(3), 03, pp. 243–254.

- 517 [9] Yin, S., and Cagan, J., 2000. “An Extended Pattern Search Algorithm for Three-Dimensional
518 Component Layout”. *Journal of Mechanical Design*, **122**(1), 01, pp. 102–108.
- 519 [10] Szykman, S., and Cagan, J., 1996. “Synthesis of Optimal Nonorthogonal Routes”. *Journal of*
520 *Mechanical Design*, **118**(3), 09, pp. 419–424.
- 521 [11] Park, J.-H., and Storch, R., 2002. “Pipe-routing algorithm development: case study of a ship engine
522 room design”. *Expert Syst. Appl. (UK)*, **23**(3), pp. 299 – 309.
- 523 [12] Ren, T., Zhu, Z.-L., Dimirovski, G., Gao, Z.-H., Sun, X.-H., and Yu, H., 2014. “A new pipe routing
524 method for aero-engines based on genetic algorithm”. *Proceedings of the Institution of Mechanical*
525 *Engineers, Part G (Journal of Aerospace Engineering)*, **228**(3), pp. 424 – 34.
- 526 [13] Qu, Y., Jiang, D., Gao, G., and Huo, Y., 2016. “Pipe routing approach for aircraft engines based on
527 ant colony optimization”. *Journal of Aerospace Engineering*, **29**(3), p. 04015057.
- 528 [14] Van der Velden, C., Bil, C., Yu, X., and Smith, A., 2007. “An intelligent system for automatic layout
529 routing in aerospace design”. *Innovations in Systems and Software Engineering*, **3**(2), pp. 117 –
530 128.
- 531 [15] Liu, C., 2018. “Optimal design of high-rise building wiring based on ant colony optimization”.
532 *Cluster Computing*, pp. 1 – 8.
- 533 [16] Wu, L., Tian, X., Wang, H., Liu, Q., and Xiao, W., 2019. “Improved ant colony optimization
534 algorithm and its application to solve pipe routing design”. *Assembly Automation*, **39**(1), pp. 45 –
535 57.
- 536 [17] Sigmund, O., 2001. “A 99 line topology optimization code written in matlab”. *Structural and*
537 *Multidisciplinary Optimization*, **21**(2), 04, pp. 120–127.
- 538 [18] Kazemi, H., Vaziri, A., and Norato, J. A., 2018. “Topology Optimization of Structures Made
539 of Discrete Geometric Components With Different Materials”. *Journal of Mechanical Design*,
540 **140**(11), 09. 111401.
- 541 [19] Iga, A., Nishiwaki, S., Izui, K., and Yoshimura, M., 2009. “Topology optimization for thermal
542 conductors considering design-dependent effects, including heat conduction and convection”.
543 *International Journal of Heat and Mass Transfer*, **52**(11-12), pp. 2721 – 2732.
- 544 [20] Dirker, J., and Meyer, J. P., 2013. “Topology optimization for an internal heat-conduction cooling
545 scheme in a square domain for high heat flux applications”. *Journal of Heat Transfer*, **135**(11).
- 546 [21] de Kruijff, N., Zhou, S., Li, Q., and Mai, Y.-W., 2007. “Topological design of structures and
547 composite materials with multiobjectives”. *International Journal of Solids and Structures*, **44**(22-
548 23), pp. 7092 – 109.
- 549 [22] Takezawa, A., Yoon, G. H., Jeong, S. H., Kobashi, M., and Kitamura, M., 2014. “Structural
550 topology optimization with strength and heat conduction constraints”. *Computer Methods in*
551 *Applied Mechanics and Engineering*, **276**, pp. 341 – 61.
- 552 [23] Kang, Z., and James, K. A., 2019. “Multimaterial topology design for optimal elastic and thermal
553 response with material-specific temperature constraints”. *International Journal for Numerical*
554 *Methods in Engineering*, **117**(10), pp. 1019–1037.
- 555 [24] James, K., Kennedy, G., and Martins, J., 2014. “Concurrent aerostructural topology optimization
556 of a wing box”. *Computers & Structures*, **134**, pp. 1 – 17.
- 557 [25] Dunning, P., Stanford, B., and Kim, H., 2015. “Coupled aerostructural topology optimization using
558 a level set method for 3d aircraft wings”. *Structural and Multidisciplinary Optimization*, **51**(5),
559 pp. 1113 – 32.
- 560 [26] Oktay, E., Akay, H., and Merttopcuoglu, O., 2011. “Parallelized structural topology optimization
561 and cfd coupling for design of aircraft wing structures”. *Comput. Fluids (UK)*, **49**(1), pp. 141 – 5.
- 562 [27] Zhu, J., Zhang, W., Beckers, P., Chen, Y., and Guo, Z., 2008. “Simultaneous design of compo-
563 nents layout and supporting structures using coupled shape and topology optimization technique”.
564 *Structural and Multidisciplinary Optimization*, **36**(1), pp. 29 – 41.

- 565 [28] Zhu, J.-H., Guo, W.-J., Zhang, W.-H., and Liu, T., 2017. “Integrated layout and topology opti-
566 mization design of multi-frame and multi-component fuselage structure systems”. *Structural and*
567 *Multidisciplinary Optimization*, **56**(1), pp. 21 – 45.
- 568 [29] Zegard, T., and Paulino, G. H., 2016. “Bridging topology optimization and additive manufacturing”.
569 *Structural and Multidisciplinary Optimization*, **53**(1), pp. 175 – 192.
- 570 [30] Tejani, G. G., Savsani, V. J., Patel, V. K., and Savsani, P. V., 2018. “Size, shape, and topology
571 optimization of planar and space trusses using mutation-based improved metaheuristics”. *Journal*
572 *of Computational Design and Engineering*, **5**(2), pp. 198 – 214.
- 573 [31] Zhang, X. S., Paulino, G. H., and Ramos, A. S., 2018. “Multi-material topology optimization
574 with multiple volume constraints: a general approach applied to ground structures with material
575 nonlinearity”. *Structural and Multidisciplinary Optimization*, **57**(1), pp. 161 – 182.
- 576 [32] Norato, J., Bell, B., and Tortorelli, D., 2015. “A geometry projection method for continuum-based
577 topology optimization with discrete elements”. *Computer Methods in Applied Mechanics and*
578 *Engineering*, **293**, pp. 306 – 27.
- 579 [33] Zhang, S., Norato, J. A., Gain, A. L., and Lyu, N., 2016. “A geometry projection method for the
580 topology optimization of plate structures”. *Structural and Multidisciplinary Optimization*, **54**(5),
581 pp. 1173 – 1190.
- 582 [34] Bendsøe, M., 1989. “Optimal shape design as a material distribution problem”. *Structural*
583 *optimization*, **1**(4), p. 193–202.
- 584 [35] Hughes, T. J. R., 2000. *The Finite Element Method*. Dover Publications.
- 585 [36] Hormann, K., and Agathos, A., 2001. “The point in polygon problem for arbitrary polygons”.
586 *Computational Geometry: Theory and Applications*, **20**(3), pp. 131 – 44.
- 587 [37] Rennels, D. C., and Hudson, H. M., 2012. *Pipe Flow: A Practical and Comprehensive Guide*. "".
- 588 [38] Tarquin, A. J., and Dowdy, J., 1989. “Optimal pump operation in water distribution”. *Journal of*
589 *Hydraulic Engineering*, **115**(2), pp. 158–168.
- 590 [39] Nakayama, Y., 2018. “Chapter 7 - flow in pipes”. In *Introduction to Fluid Mechanics (Second*
591 *Edition)*, Y. Nakayama, ed., second edition ed. Butterworth-Heinemann, pp. 135 – 161.
- 592 [40] Martins, J., and Hwang, J., 2013. “Review and unification of methods for computing derivatives of
593 multidisciplinary computational models”. *AIAA Journal*, **51**(11), pp. 2582 – 99.
- 594 [41] Sunday, D., 2012. Distance between 3d lines & segments.

595 **A Distance between segments MATLAB code**

```
596
597 function [sqDist, dd_dx] = sqDistance(xa0, xaf, xb0, xbf)
598 %segmentDistance calculates minimum squared distance between
599 % two line segments
600 % xa0, xaf - end points of first segment - row vector
601 % xb0, xbf - end points of second segment - row vector
602 % returns dd_dx in order [xa0, xaf, xb0, xbf]
603
604 dim = size(xa0,2);
605 I = eye(dim, dim);
606 z = zeros(dim, dim);
607
608 u = xaf - xa0;
609 dudx = [-I I z z];
610 v = xbf - xb0;
611 dvdx = [z z -I I];
612 w = xa0 - xb0;
613 dwdx = [I z -I z];
614 a = u*u';
615 b = u*v';
616 c = v*v';
617 d = u*w';
618 e = v*w';
619 dadx = 2*u*dudx;
620 dbdx = v*dudx + u*dvdx;
621 dcdx = 2*v*dvdx;
622 dddx = w*dudx + u*dwdx;
623 dedx = w*dvdx + v*dwdx;
624
625 D = a*c-b^2;
626 dDdx = a*dcdx + c*dadx - 2*b*dbdx;
627
628 sD = D;
629 tD = D;
630 dsddx = dDdx;
631 dtddx = dDdx;
632
633 eps = 1e-5; % tolerance for nearly parallel
634 if D < eps
635 % segments are nearly parallel
636 sN = 0;
637 sD = 1;
638 tN = e;
639 tD = c;
640 dsndx = zeros(1, 4*dim);
```

```

641     dsddx = zeros(1,4*dim);
642     dtndx = dedx;
643     dtddx = dcdx;
644 else
645     sN = (b*e-c*d);
646     tN = (a*e-b*d);
647     dsndx = e*dbdx + b*dbdx - d*dcdx - c*dddx;
648     dtndx = a*dedx + e*dadx - d*dbdx - b*dddx;
649     if sN < 0
650         sN = 0;
651         tN = e;
652         tD = c;
653         dsndx = zeros(1,4*dim);
654         dtndx = dedx;
655         dtddx = dcdx;
656     elseif sN > sD
657         sN = sD;
658         tN = e+b;
659         tD = c;
660         dsndx = dsddx;
661         dtndx = dedx + dbdx;
662         dtddx = dcdx;
663     end
664 end
665
666 if tN < 0
667     tN = 0;
668     dtndx = zeros(1,4*dim);
669     if -d < 0
670         sN = 0;
671         dsndx = zeros(1,4*dim);
672     elseif -d > a
673         sN = sD;
674         dsndx = dsddx;
675     else
676         sN = -d;
677         sD = a;
678         dsndx = -dddx;
679         dsddx = dadx;
680     end
681 elseif tN > tD
682     tN = tD;
683     dtndx = dtddx;
684     if (-d+b) < 0
685         sN = 0;
686         dsndx = zeros(1,4*dim);
687     elseif (-d+b) > a
688         sN = sD;

```

```

689     dsndx = dsddx;
690     else
691         sN = -d + b;
692         sD = a;
693         dsndx = -ddd + dbdx;
694         dsddx = dadx;
695     end
696 end
697
698 if abs(sN) < eps
699     sC = 0;
700     dscdx = zeros(1,4*dim);
701 else
702     sC = sN/sD;
703     dscdx = (1/sD)*dsndx - (sN/(sD^2))*dsddx;
704 end
705
706 if abs(tN) < eps
707     tC = 0;
708     dtcdx = zeros(1,4*dim);
709 else
710     tC = tN/tD;
711     dtcdx = (1/tD)*dtndx - (tN/(tD^2))*dtddx;
712 end
713
714 dP = w+(sC*u)-(tC*v);
715 dPdw = I;
716 dPdu = sC*I;
717 dPdv = -tC*I;
718 dPdsc = u';
719 dPdte = -v';
720 ddP_dx = [dPdw dPdu dPdv dPdsc dPdte]*[dwdx; dudx; dvdx; dscdx; dtcdx];
721 sqDist = dP*dP';
722 dd_dx = 2*dP*ddP_dx;
723 end

```