

# A Novel Framework for Simultaneous Topology and Sizing Optimization of Complex, Multi-Domain Systems-of-Systems

**Donald J. Docimo<sup>1</sup>**

Department of Mechanical Engineering,  
Texas Tech University,  
Lubbock, TX 79409  
e-mail: donald.docimo@ttu.edu

**Ziliang Kang**

Department of Aerospace Engineering,  
University of Illinois at Urbana-Champaign,  
Urbana, IL 61820  
e-mail: kang134@illinois.edu

**Kai A. James**

Department of Aerospace Engineering,  
University of Illinois at Urbana-Champaign,  
Urbana, IL 61801  
e-mail: kaijames@illinois.edu

**Andrew G. Alleyne**

Department of Mechanical Science and  
Engineering,  
University of Illinois at Urbana-Champaign,  
Urbana, IL 61801  
e-mail: alleyne@illinois.edu

*This article presents a novel design framework for topology and component sizing optimization of multi-domain dynamic systems described by conservation laws. Multidisciplinary design optimization (MDO) is a powerful tool for minimizing metrics such as inefficiency and cost for these systems-of-systems (SoS). However, quality of the designs identified from the optimization procedure depends on model accuracy and ability to capture inter-system interactions. This work utilizes a conservation-based, graphical modeling approach to capture physical system dynamics and interactions, and expands it to be used in MDO techniques. This yields three contributions to the literature. First, an augmented graph-based model is provided, expressing continuous and discrete design variable values as changes to vertex size, edge size, and edge connections of the dynamic system graph. Second, a sizing and topology optimization framework is developed using the augmented graph-based model as a basis. Third, analytical and numerical sensitivity functions are derived for a cooling system design problem, stemming from application of the design framework. The design framework is applied to two case studies for cooling subsystem design and electric vehicle (EV) powertrain design, with the goal of optimizing thermal and electrical component sizes, as well as discrete choices in the topology of the system being designed. These case studies provide examples for how the design framework enables analysis of alternatives (AoA) during early design stages.*

[DOI: 10.1115/1.4046066]

**Keywords:** energy systems design, multidisciplinary design and optimization

## 1 Introduction

This article presents a novel framework for the design optimization of conservation-based, multi-domain, dynamic systems-of-systems (SoS). Conservation laws (e.g., mass, energy) define the dynamics of a number of modern systems, including mobile power systems [1], water distribution networks [2], and chemical process networks [3]. Many of these application areas seek to optimize these systems to improve energy efficiency [4], minimize cost [2,5], or reduce weight [6,7]. However, the complexity, size, and multi-domain nature of these SoS require multidisciplinary design optimization (MDO) to identify optimal architectures [8].

To develop the framework for MDO as applied to conservation-based SoS, this article builds upon two distinct bodies of literature: MDO of dynamic systems and physics-based modeling of dynamic systems.

**MDO of dynamic systems:** The concept of MDO as applied to dynamic systems is well established in certain application areas, including flexible structures [9], submarine hulls [10], and electric aircraft characteristics [11,12]. Furthermore, generalized MDO frameworks that provide the ability to express preference directly over traditional weight-based methods have been developed [13]. Applied properly, MDO is a powerful tool capable of producing superior plant designs. However, there are several recognized open challenges for MDO of dynamic systems [8]. (i) Black box

and static models often used in MDO underemphasize system dynamic behavior. (ii) Incomplete representation of system-to-system interactions leads to suboptimal results or failures requiring redesign [8,14]. (iii) MDO operates best with accurate models that capture multi-domain dynamics, retain computational efficiency, and permit modular changes in the design of the system [8].

**Physics-based modeling of dynamic systems:** Dynamic systems can be classified into two broad categories. Non-conservative systems include economic models [15,16] and wireless sensor networks [17], whose properties do not obey physical laws of conservation. Conservative systems, the focus of this work, describe dynamic systems such as drug-cell interactions [18], water networks [2], and insect flight systems [19] that follow physical laws such as conservation of energy or mass. Approaches for modeling the dynamics of these physical systems include state space models [20], partial differential equations [21–23], bond graphs [24], and block diagram modeling techniques [25]. These modeling approaches vary in modular capabilities, computational complexity, and the ability to describe interactions between subsystems. Impeding the use of a subset of physics-based models in optimization is the barrier of integration. Models of different disciplines and domains do not always pass information through the same signals, making communication and simultaneous simulation of the models difficult [8].

These two bodies of literature complement one another, with the gaps of one potentially addressable through techniques and insights from the other. In particular, MDO of dynamic systems can be improved by modular physics-based models that explicitly account for system behavior, system-to-system interactions, and multiple domain signals. A physics-based model that is inclusive to different domains reduces communication challenges, making the model

<sup>1</sup>Corresponding author.

Contributed by the Design Automation Committee of ASME for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received January 31, 2019; final manuscript received December 19, 2019; published online January 21, 2020. Assoc. Editor: Samy Missoum.

more easily utilized in MDO. Some studies begin to bridge the gap between these bodies of literature for certain classes of SoS. This includes bond graphs, which capture flow-effort dynamics, for design generation and optimization of electromechanical systems [26,27], and a standardized method to unify separate system models for MDO of aircraft [28]. However, for a variety of system classes, there is still room to address MDO challenges, which can be done by exploiting different tools. For example, computational challenges of MDO can be potentially mitigated by using graph theory to perform agglomerative model reduction [29].

This article is centered around MDO for a class of dynamic SoS described by conservation laws. Using the insights from the literature on MDO of dynamic systems, as well as a conservation-based modeling approach previously developed by the authors [30–33], a design optimization framework is developed. The models from this approach (i) are physics-based, (ii) capture dynamic interactions between multi-domain elements, and (iii) are modular [34]. Furthermore, the graphical nature of the modeling approach permits the use of established graph theory techniques to perform agglomerative model reduction [32] and system decomposition [33]. All of these features are useful in the context of MDO.

While these graph-based models have been used for ranking enumerated architectures with optimized control [35], the models were not capable of being used in MDO for continuous and discrete variable design optimization. This article establishes this capability through the introduction of design matrices representative of the impact of design choices on the SoS. These matrices describe how continuous and discrete (e.g., sizing and topology) design variable changes impact a graph's vertex/edge sizes and edge addition/removal, and translate these changes into the mathematical representation of the graph's dynamics. From this, the article introduces two contributions to the literature:

- The augmented graph-based model, expressed by augmenting the design matrices to the original graph-based model's matrix representation.
  - A design optimization framework for multi-domain SoS with dynamics based on conservation laws. This framework permits simultaneous topology and sizing optimization.
- Within the scope of using a graphical representation of conservation-based models, the proposed framework provides an effective solution technique. Application of the framework to a case study problem yields a third contribution:
- Derivations of analytical and numerical sensitivity formulations for a cooling system design problem.

In this work, monolithic approaches [36] for a full-order, dynamic SoS model with open-loop control is the focus, leaving model reduction, system decomposition, and optimal control within the context of design optimization for future studies. The framework does not demand use of a specific optimization algorithm, but rather utilizes the design engineer's experience to guide objective function formulation, constraints, and solver choice. This permits expert knowledge, which is often privy to unmodeled

factors (for example, consumer preference or serviceability in mobile power systems), to be utilized in the design optimization process.

The rest of this paper is outlined as follows: Sec. 2 reviews the class of conservation-based systems considered and the basics of the graphical modeling technique. Section 3 presents the design optimization framework, starting from the graphical model and ending with the optimization problem formulation. This framework is applied to two case studies of interest. Section 4 applies the framework to the case study of cooling system design. Section 5 applies the framework to the second case study of electric vehicle (EV) powertrain design with multi-domain power flows. The paper ends with a summary of the contributions and future work.

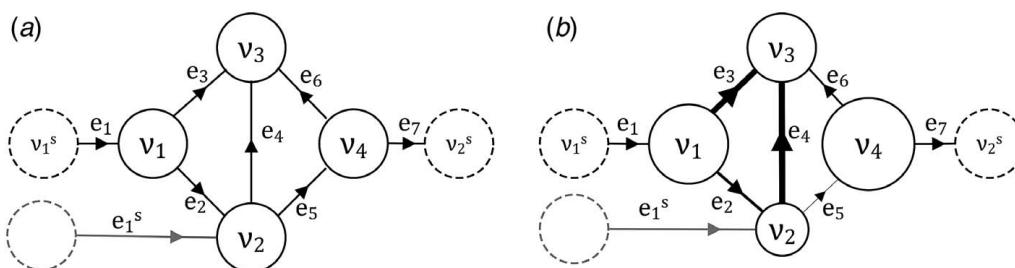
## 2 Graph-Based Modeling

This section presents the basics of the graph-based modeling technique, with the full description provided in Refs. [30,31,33]. The class of conservation-based dynamic systems that are the focus in this article is stated. The concept of design optimization in the context of these graphical models is outlined. In this study, the size of system elements is represented by vertex and edge sizes in the graph, and topology selection is achieved by edge addition and removal.

**2.1 Graph-Based Modeling Basics.** The graph-based modeling technique is structured for modular modeling of systems through conservation-based principles. These graphs are capable of capturing dynamics for mass conservation-based systems [32] or energy conservation-based systems containing different power flow domains [30,31,33]. Figure 1(a) presents an example graph-based model that contains the four main features of the graph. Each vertex  $v_i$  ( $i = 1, \dots, N_v$ ) is analogous to a control volume, and describes a storage capacitance  $C_i$  and a state  $x_i$ . Each edge  $e_j$  ( $j = 1, \dots, N_e$ ) relates to a transfer rate  $y_j$  (e.g., mass flow rate, power flow) that is a function of the edge's head vertex ( $x_j^{head}$ ), the edge's tail vertex ( $x_j^{tail}$ ), input  $u_j$ , and switching input  $s_j$ . Equation (1) shows that this transfer rate can relate to these variables through a nonlinear function  $f_j$ .

$$y_j = f_j(x_j^{head}, x_j^{tail}, u_j, s_j) \quad (1)$$

Each sink vertex  $v_l^s$  ( $l = 1, \dots, N_s$ ) is associated with state  $x_l^s$ , and is considered a disturbance to the system through the edges connecting the sink vertex to the graph. Similarly, each source edge  $e_n^s$  ( $n = 1, \dots, N_s$ ) is also considered a disturbance to the system. For both sink vertices and source edges, reversal of the connecting edge's sign convention can make any sink look like a source (and vice versa). For simplicity, any sink or source vertex is referred to as a sink vertex, and any sink or source edge is referred to as a source edge.



**Fig. 1 (a)** A graph-based model with the four main features: vertices (solid circles), edges (solid lines), sink vertices (dashed circles), and source edges (gray lines). **(b)** Graphical representation of component sizing through vertex/edge sizing.

The incidence matrix  $\mathbf{M} \in \mathbb{R}^{(N_v+N_e) \times N_e}$  describes the relationship between edge transfer rates and vertex dynamics in the system. An entry  $M_{i,j}$  is +1 if  $v_i$  is the tail of  $e_j$ , -1 if  $v_i$  is the head of  $e_j$ , and 0 in all other cases. This matrix can be partitioned as:

$$\mathbf{M} = \begin{bmatrix} \bar{\mathbf{M}} \\ \underline{\mathbf{M}} \end{bmatrix} \quad (2)$$

In Eq. (2),  $\bar{\mathbf{M}} \in \mathbb{R}^{N_v \times N_e}$  is the mapping of transfer rates to state vector  $\mathbf{x} \in \mathbb{R}^{N_v}$  and  $\underline{\mathbf{M}} \in \mathbb{R}^{N_e \times N_e}$  is the mapping of transfer rates to sink state vector  $\mathbf{x}^s \in \mathbb{R}^{N_e}$ . Similarly, the matrix  $\mathbf{D} \in \mathbb{R}^{N_v \times N_s}$  describes the relationship between source transfer rates and vertex dynamics in the system. An entry  $D_{i,n}$  is +1 if  $v_i$  is the head of source  $e_n^s$  and 0 in all other cases. With storage capacitance matrix  $\mathbf{C} = \text{diag}(C_1, \dots, C_{N_v}) \in \mathbb{R}^{N_v \times N_v}$ , the dynamics of the graph are described by:

$$\mathbf{C}\dot{\mathbf{x}} = -\bar{\mathbf{M}}\mathbf{y} + \mathbf{D}\mathbf{y}^s \quad (3)$$

In Eq. (3),  $\mathbf{y} = [y_1, \dots, y_{N_e}]^T \in \mathbb{R}^{N_e}$  and  $\mathbf{y}^s = [y_1^s, \dots, y_{N_s}^s]^T \in \mathbb{R}^{N_s}$ . Equation (3) is the matrix representation of the graph-based model, and describes the class of conservation-based systems that is the focus on this work.

**2.2 Vertex and Edge Design.** The graph-based modeling technique enables easy understanding of the connection between design variables and the model. This paper focuses on (i) component size optimization and (ii) topology optimization. Component (or element) size optimization is translated into vertex and edge size optimization, as represented in Fig. 1(b). Vertex size relates to capacitance of the vertex. For example, an increase in voltage bus capacitor size is represented by a larger vertex and larger capacitance  $C_{bus}$ . Edge size relates to transfer rate. For example, consider a motor with current  $I$  and resistance  $R$ . A smaller value of  $R$  indicates the motor is more efficient, producing less heat  $RI^2$ . This is represented by a smaller or thinner edge.

Topology (or architecture) optimization is translated to edge addition or removal, as represented in Fig. 2. The variable  $z$  is the architecture selection parameter, where  $z=1$  is the topology that includes element  $v_2$  (Fig. 2(a)) and  $z=0$  is the topology that excludes element  $v_2$  (Fig. 2(b)). This approach of relating design variables to the graph-based model is used as a basis to the design optimization framework.

### 3 Framework for Design Optimization of Conservation-Based Systems

This section presents the four step design optimization framework. Starting from the graph-based model described in Sec. 2, the framework augments the model through the use of design matrices, relating sizing, and topology options to the model. After defining constraints and objective functions, the design optimization problem is formulated.

**3.1 Overview of the Framework.** The four main steps of the framework are as follows:

- (1) *Augment the matrix representation of the graph-based model with design matrices.* Sizing and topology design options, represented by node/edge sizing and edge addition/removal, are related to the mathematical representation of the graph's dynamics by augmenting Eq. (3) with design matrices.
- (2) *Define the objective function.* The objective function is defined to represent desired dynamic characteristics of the system. The objective function may capture a variety of features, such as performance or the maximum temperature of a component.
- (3) *Define design constraints.* Nonlinear constraints and bounds on the sizing and topology design variables are defined. Design variable bounds are determined based on problem-specific limitations such as cost, weight, and size (e.g., a battery pack for an EV is bounded in size based on weight limitations).
- (4) *Formulate and solve the optimization problem.* Using the first three steps, the optimization problem is set up, allowing the solution to be determined via the algorithm most appropriate to solve the problem as determined by the user.

Section 3.2 goes over each step in detail.

**3.2 The Design Optimization Framework.** The design optimization framework starts with a graph-based model mathematically described by Eq. (3). This graph-based model can either represent the full system or a subsystem determined through decomposition of the graphical model [33].

*Step 1: Augment the matrix representation of the graph-based model to include design matrices.* The first step takes the matrix representation of a graph-based model of the form of Eq. (3) and augments it to include the impact of design choices on the system dynamics. For  $\sigma=1, \dots, N_c$ , let  $\theta_{c,\sigma} \in \mathbb{R}$  be a continuous sizing design variable that impacts vertex/edge sizes,  $\boldsymbol{\theta}_c^* = \{\theta_{c,1}, \dots, \theta_{c,N_c}\}$  as the set of continuous design variables, and  $\boldsymbol{\theta}_c \in \mathbb{R}^{N_c}$  as the vector of continuous design variables. Similarly, for  $q=1, \dots, N_z$ , let  $z_q \in \mathbb{Z}$  be a discrete topology design variable that describes edge addition/removal,  $\mathbf{z}^* = \{z_1, \dots, z_{N_z}\}$  as the set of discrete design variables, and  $\mathbf{z} \in \mathbb{Z}^{N_z}$  as the vector of discrete design variables. The total set of design variables is described by  $\boldsymbol{\theta} = [\boldsymbol{\theta}_c, \mathbf{z}]$ , making the total number of design variables  $N_c + N_z$ .

The graph-based model mathematically described by Eq. (3) is not suitable for design optimization as it is, and requires six factors to be included in its representation for optimization. These six factors are the mathematical representation of the impact of sizing on (i) vertex size, (ii) edge size, and (iii) source edge size, and the impact of different topologies on (iv) vertex size, (v) edge addition or removal, and (vi) source edge addition or removal. This is accomplished by augmenting the matrix representation from Eq. (3) with six design matrices. Equation (4) presents the matrix representation of the augmented graph-based model:

$$\boldsymbol{\Psi}_c \boldsymbol{\Phi}_c \mathbf{C}\dot{\mathbf{x}} = -\bar{\mathbf{M}}\boldsymbol{\Psi}\boldsymbol{\Phi}\mathbf{y} + \mathbf{D}\boldsymbol{\Psi}^*\boldsymbol{\Phi}^*\mathbf{y}^s \quad (4)$$

The six design matrices of Eq. (4) are  $\boldsymbol{\Psi}_c$ ,  $\boldsymbol{\Phi}_c$ ,  $\boldsymbol{\Psi}$ ,  $\boldsymbol{\Phi}$ ,  $\boldsymbol{\Psi}^*$ , and  $\boldsymbol{\Phi}^*$ . These matrices translate the scaling of component sizes and alternative architectures to the mathematical representation of the model. The first two design matrices,  $\boldsymbol{\Psi}_c$  and  $\boldsymbol{\Phi}_c$ , translate the impact of

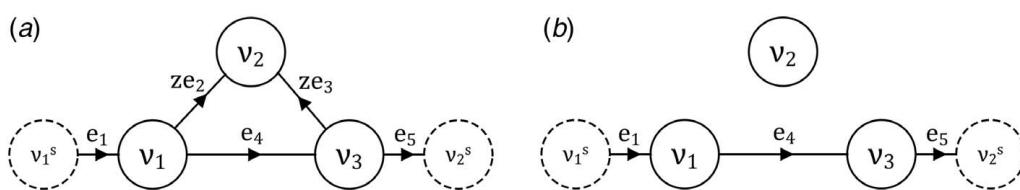


Fig. 2 Graph-based representation of topology selection through (a)–(b)

sizing and topology choices to vertex size. The second two matrices,  $\Psi$  and  $\Phi$ , translate sizing and topology choices to edge size and removal. The last two matrices,  $\Psi^s$  and  $\Phi^s$ , do the same for the source edges of the graph-based model. The six design matrices of Eq. (4) are defined as follows:

- The diagonal matrix  $\Psi_c \in \mathbb{R}^{N_v \times N_v}$  describes the relationship between sizing design variables and vertex size. The capacitance matrix  $C$  is augmented by  $\Psi_c$ , reflecting how sizing design variables impact capacitance sizes. The design matrix  $\Psi_c$  is defined as:

$$\Psi_c(\theta_c) = \begin{bmatrix} f_{\psi_{c,1}}(\theta_c) & & 0 \\ & \ddots & \\ 0 & & f_{\psi_{c,N_v}}(\theta_c) \end{bmatrix} \quad (5)$$

In Eq. (5), each  $f_{\psi_{c,i}}$  is a function that describes the impact of  $\theta_c$  on the capacitance of  $v_i$ . Note that  $f_{\psi_{c,i}}$  may not explicitly depend on each  $\theta_{c,\sigma}$ .

- The diagonal matrix  $\Psi \in \mathbb{R}^{N_e \times N_e}$  describes the relationship between sizing design variables and edge size. The vector of transfer rates,  $y$ , is augmented by  $\Psi$ , capturing how sizing design variables impact the sizes of edges. The design matrix  $\Psi$  is defined as:

$$\Psi(\theta_c) = \begin{bmatrix} f_{\psi_1}(\theta_c) & & 0 \\ & \ddots & \\ 0 & & f_{\psi_{N_e}}(\theta_c) \end{bmatrix} \quad (6)$$

In Eq. (6), each  $f_{\psi_j}$  is a function that describes the impact of  $\theta_c$  on the transfer rate of edge  $e_j$ . Again,  $f_{\psi_j}$  may not explicitly depend on each  $\theta_{c,\sigma}$ .

- The diagonal matrix  $\Psi^s \in \mathbb{R}^{N_s \times N_s}$  describes the relationship between sizing design variables and source edge size. The vector of source transfer rates,  $y^s$ , is augmented by  $\Psi^s$ , capturing the impact of sizing design variables on the size of source edges. Equation (7) defines the design matrix  $\Psi^s$ .

$$\Psi^s(\theta_c) = \begin{bmatrix} f_{\psi_1^s}(\theta_c) & & 0 \\ & \ddots & \\ 0 & & f_{\psi_{N_s}^s}(\theta_c) \end{bmatrix} \quad (7)$$

In Eq. (7), each  $f_{\psi_n^s}$  is a function that describes the impact of  $\theta_c$  on the transfer rate of source edge  $e_n^s$ . Again,  $f_{\psi_n^s}$  may not explicitly depend on each  $\theta_{c,\sigma}$ .

- The diagonal matrix  $\Phi_c \in \mathbb{R}^{N_v \times N_v}$  describes the relationship between topology design variables and vertex size. The capacitance matrix  $C$  is augmented by  $\Phi_c$ , reflecting how topology design variables impact capacitance sizes. The design matrix  $\Phi_c$  is defined as:

$$\Phi_c(z) = \begin{bmatrix} f_{\phi_{c,1}}(z) & & 0 \\ & \ddots & \\ 0 & & f_{\phi_{c,N_v}}(z) \end{bmatrix} \quad (8)$$

In Eq. (8), each  $f_{\phi_{c,i}}$  is a function that describes the impact of  $z$  on the capacitance of  $v_i$ . Note that  $f_{\phi_{c,i}}$  may not explicitly depend on each  $z_q$ .

- The diagonal matrix  $\Phi \in \mathbb{R}^{N_e \times N_e}$  describes the relationship between topology design variables and edge addition/removal. The transfer rate vector  $y$  is augmented by  $\Phi$ , translating how topology design variables impact edge sizes.

Equation (9) defines  $\Phi$ .

$$\Phi(z) = \begin{bmatrix} f_{\phi_1}(z) & & 0 \\ & \ddots & \\ 0 & & f_{\phi_{N_e}}(z) \end{bmatrix} \quad (9)$$

To define  $f_{\phi_j}$  in Eq. (9), let  $z^{**} \subseteq z^*$  of size  $n_z \leq N_z$  and  $z' \in \mathbb{R}^{n_z}$ . Then:

$$f_{\phi_j}(z) = \hat{f}_{\phi_j}(z') = \hat{f}_{\phi_j}(z'_1, \dots, z'_{n_z}) = \prod_{l=1}^{n_z} z'_l \quad (10)$$

Note that each  $z'_l$  belongs to the subset  $z^{**}$ . To provide an example, consider  $N_z = 3$ , with  $z_1$  and  $z_3$  impacting edge  $e_5$  of a graph ( $n_z = 2$ ). The function  $f_{\phi_5}$  is defined as follows, with  $z'_1 = z_1$  and  $z'_2 = z_3$ :

$$f_{\phi_5}(z_1, z_3) = \hat{f}_{\phi_5}(z'_1, \dots, z'_{n_z}) = \prod_{l=1}^2 z'_l = z_1 z_3 \quad (11)$$

In the case of  $z_3 = 0$  representing edge addition, it is permissible to define  $z'_2 = 1 - z_3$ .

- The diagonal matrix  $\Phi^s \in \mathbb{R}^{N_s \times N_s}$  describes the relationship between sizing design variables and source edge size. The transfer rate vector  $y^s$  is augmented by  $\Phi^s$ , reflecting how topology design variables impact source edge sizes. The matrix  $\Phi^s$  is defined as:

$$\Phi^s(z) = \begin{bmatrix} f_{\phi_1^s}(z) & & 0 \\ & \ddots & \\ 0 & & f_{\phi_{N_s}^s}(z) \end{bmatrix} \quad (12)$$

The function  $f_{\phi_n^s}$  in Eq. (12) is defined using the same process used to define  $\hat{f}_{\phi_j}$  previously in Eq. (10).

The addition of these design matrices allows the design engineer to quickly and intuitively relate the design variables to the dynamics of the graph. By starting with a baseline graphical model that can be represented by Eq. (3), the design engineer can identify which components should be sized and which topologies should be explored. This is translated to vertex/edge sizing and edge addition/removal, and thus modifies the dynamics of the system represented through Eq. (4).

*Step 2: Define the objective function.* The framework gives the design engineer freedom in terms of defining objective functions. Using the expertise of the design engineer allows for consideration of unmodeled factors, such as consumer product preference or serviceability of a component, to be factored into the decision-making process. Unique types of objective functions can be defined that relate the conservation basis of the graph to desired system capabilities and features. A transfer rate-based objective function relates transfer rates along the edges to desired system characteristics. Equation (13) defines a potential transfer rate-based objective function  $J_{tr}$  over a test period from times  $t_0$  to  $t_f$

$$J_{tr} = \sum_{j=1}^{N_e} w_{tr,j} \int_{t_0}^{t_f} y_j(t, \theta) dt \quad (13)$$

In Eq. (13), each  $w_{tr,j}$  is a weighting term internal to  $J_{tr}$  which allows one to tune the importance of different edges for the objective calculation. This type of objective function can be used to determine total energy losses within the system during the test period.

A storage-based objective function correlates the importance of changes in quantity stored in the system to desired system characteristics. Equation (14) defines a typical storage-based objective

function  $J_{st}$  over a test period from times  $t_0$  to  $t_f$

$$J_{st} = \sum_{i=1}^{N_v} w_{st,i} \int_{x_i(t_0)}^{x_i(t_f)} C_i(x, \theta) dx \quad (14)$$

The weights  $w_{st,i}$  are internal to  $J_{st}$ , and allows one to tune the importance of different vertices' storage levels. This type of objective function can be used to quantify the total change in energy stored in all energy storage components in the system over the test period.

The objective functions are not limited to these forms in the framework, and can encompass other performance metrics as well. The objective function can be augmented with a penalty term,  $J_{pen}$ , to implicitly enforce design constraints, thereby ensuring that the system satisfies necessary requirements. Other desired objectives, such as system performance (e.g., 0–60 mph time for a car), desired state limits, and state tracking, are all permitted in the framework. Note that problems with multiple objectives can be defined using an aggregate objective function  $J_{total}$ . Within the framework, this aggregate objective function can be defined using a linear, weighted combination of individual objectives, or using a definition suitable for techniques such as compromise programming [37].

*Step 3: Define design constraints.* Equation (15a) defines the design variable bounds, which limit the variables between minimum  $\underline{\theta}$  and maximum  $\bar{\theta}$  values. These represent any problem-specific limitations design variables may have. For example, a heat exchanger for an aircraft may be limited in size due to limited space, or a motor may be limited in maximum power due to excessive cost. Equation (15b) presents the set of nonlinear constraints with nonlinear function  $g$  for the design problem.

$$\underline{\theta} \leq \theta \leq \bar{\theta} \quad (15a)$$

$$g(\theta) \leq 0 \quad (15b)$$

*Step 4: Formulate and solve the optimization problem.* Using the previous three steps of the framework, the optimization problem can be set up:

$$\min \quad J_{total}(\theta) \quad (16a)$$

$$\text{subj. to: } \underline{\theta} \leq \theta \leq \bar{\theta} \\ g(\theta) \leq 0 \quad (16b)$$

In this article, the shooting method is used, by which the graph-based model is simulated for the current set of design variables and the objective function is calculated from those trajectories. This is one option, and direct transcription or other methods can also be used to formulate the optimization problem. Note that the solver is purposefully left as a choice for the design engineer to make. This leverages the design engineer's experience and knowledge of the design problem to yield results for their needs.

The Secs. 4 and 5 apply the design framework to two different case studies. The first explores the design of a cooling system, and shows how gradient-based solution methods with analytical and numerical sensitivity functions can be used in the framework. The second explores the electrical, thermal, and mechanical design of an EV powertrain using a genetic algorithm (GA). This shows the applicability of the framework for design problems containing different dynamic domains and different objectives.

#### 4 Case Study 1: Cooling System Design

This section applies the design optimization framework to the case study of configuration and sizing of a cooling system. The baseline configuration with design options is outlined and translated into a graph-based model. By relaxing the discrete topology design

problem, analytical and numerical sensitivity functions are determined that relate the objective function to the design variables. Using a gradient-based solution algorithm, the optimal design of the cooling system is identified.

**4.1 Problem Statement.** The first case study explores the design of a liquid cooling system. The system, a pipe with water being pumped through it, absorbs heat from a resistive load and a second thermal load, represented by a battery pack in this case study. There are two design parameters: size of the cross-sectional area of the pipe, and placement of the battery pack with respect to the pipe. The objective is to maximize the heat drawn from the thermal loads to the pipe fluid for the purpose of maximizing cooling capability. Figure 3 presents an illustration of the cooling system setup.

Figure 4 presents the graph-based model of the cooling system, determined through the conservation of energy. There are two vertices that describe temperature of the pipe fluid (states  $x_1$  and  $x_2$ ), with a capacitance matrix:

$$C = \begin{bmatrix} mC_p & 0 \\ 0 & mC_p \end{bmatrix} \quad (17)$$

In Eq. (17),  $C_p = 4185 \text{ J/kg K}$  is the specific heat of the fluid, and the fluid mass in each control volume is  $m = \rho L A_{nom}$ , with  $\rho = 1000 \text{ kg/m}^3$  as the fluid density,  $L = 2.86 \text{ m}$  as the control volume length, and  $A_{nom} = 0.0034 \text{ m}^2$  as the nominal cross-sectional area. This area is to be scaled when applying the design optimization framework. There are four sink vertices with constant temperatures, one each for the battery ( $x_1^s = 308.15 \text{ K}$ ), the resistive load ( $x_2^s = 308.15 \text{ K}$ ), the upstream fluid temperature ( $x_3^s = 283.15 \text{ K}$ ), and the downstream fluid temperature ( $x_4^s$ ). Equation (18) defines

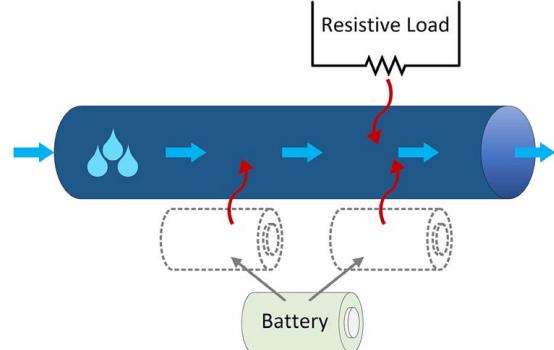


Fig. 3 Cooling system for case study 1, with potential locations of the battery pack outlined

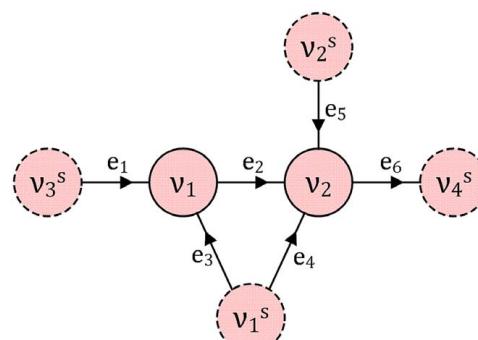


Fig. 4 Graph-based model of the cooling system for case study 1

$\bar{M}$ , developed from Fig. 4.

$$\bar{M} = \begin{bmatrix} -1 & 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & -1 & 1 \end{bmatrix} \quad (18)$$

Edges  $e_1$ ,  $e_2$ , and  $e_6$  describe thermal transport between the control volumes of the pipe, and the power flows are defined as

$$\begin{aligned} y_1 &= \dot{m}C_p x_3^s \\ y_2 &= \dot{m}C_p x_1 \\ y_6 &= \dot{m}C_p x_2 \end{aligned} \quad (19)$$

In Eq. (19), the mass flowrate is  $\dot{m} = \rho v A_{nom}$ , with fluid velocity  $v = 0.15 \text{ m/s}$ . Edges  $e_3$  and  $e_4$  describe heat conduction from the battery to the fluid:

$$\begin{aligned} y_3 &= \frac{1}{R_c}(x_1^s - x_1) \\ y_4 &= \frac{1}{R_c}(x_1^s - x_2) \end{aligned} \quad (20)$$

with  $R_c = 0.0358 \text{ K/W}$  as the thermal resistance. Only one of these edges is to be selected for the final design. Edge  $e_5$  describes heat conduction from the resistive load to the fluid:

$$y_5 = \frac{1}{R_{c,load}}(x_2^s - x_2) \quad (21)$$

with  $R_{c,load} = 0.0717 \text{ K/W}$ . For this system,  $y^s = \mathbf{0}$  and  $D = \mathbf{0}$ .

**4.2 Application of the Framework.** With a graph-based model's representation in the form of Eq. (3), the design framework can be applied. There are two design variables: (i) a continuous sizing option that scales pipe area,  $\theta_c$ , and (ii) a topology option that selects battery pack placement,  $z$ . Applying the first step of the framework, the size design matrices are defined. Resizing of the pipe area scales mass  $m$  in each control volume, and thus scales the capacitances:

$$\Psi_c(\theta_c) = \begin{bmatrix} \theta_c & 0 \\ 0 & \theta_c \end{bmatrix} \quad (22)$$

This also scales  $\dot{m}$ , which impacts edges  $e_1$ ,  $e_2$ , and  $e_6$ :

$$\Psi(\theta_c) = \text{diag}([\theta_c, \theta_c, 1, 1, 1, \theta_c]) \quad (23)$$

The battery can be placed in one of two locations: either it is connected to the graph through  $e_3$  or it is connected through  $e_4$ . Assigning  $z = 0$  for downstream placement and  $z = 1$  for upstream placement, the matrix  $\Phi$  becomes:

$$\Phi(z) = \text{diag}([1, 1, z, 1-z, 1, 1]) \quad (24)$$

The remaining design matrices are equal to the identity matrix.

Before applying the remaining steps, the topology design variable is smoothed using a RAMP function [38]:

$$z = \frac{\theta_z}{1 + Q(1 - \theta_z)} \quad (25)$$

with  $Q = 1000$ , and  $\theta_z$  as the underlying, continuous design variable. While smoothing of the topology design variable is not necessary, it permits the use of gradient-based solvers to be applied to the optimization problem. This is utilized in Sec. 4.3 to derive analytical sensitivity functions, which are then used in the gradient-based optimization procedure. The vector of design variables is  $\boldsymbol{\theta} = [\theta_c, \theta_z]$ .

Step 2 of the framework is to define the objective function  $J_{total}$ . The objective is to maximize the heat drawn from the loads at a final time  $t_f$ . This is equivalent to minimizing the negative of the power flows along edges  $e_3$ ,  $e_4$ , and  $e_5$  at:

$$J_{total} = -z(\theta_z)y_3(t_f, \boldsymbol{\theta}) - (1-z)y_4(t_f, \boldsymbol{\theta}) - y_5(t_f, \boldsymbol{\theta}) \quad (26)$$

Step 3 of the framework requires the definition of the constraints for the design variables  $\boldsymbol{\theta}$ . The design variable constraints are:

$$\left\{ \begin{array}{l} 0.75 \\ 0 \end{array} \right\} \leq \boldsymbol{\theta}^T \leq \left\{ \begin{array}{l} 1.25 \\ 1 \end{array} \right\} \quad (27)$$

Note that in this case study, there are no nonlinear design constraints enforced.

The value of the objective is calculated using the shooting method with the MATLAB variable step solver "ode15s." With the first three steps completed, the optimization problem is defined in the form of Eq. (16) as stated in Step 4. The code is run on a desktop computer with a 3.4 GHz Intel Core i7-6700 processor and 16 GB RAM.

**4.3 Optimization Strategy and Sensitivity Functions.** A Hessian-free, gradient-based algorithm is used to solve the optimization problem. The MATLAB library function "fmincon" is used with sequential quadratic programming (SQP) as the solver, and default tolerances. To provide the objective function gradients to the algorithm, two methods to analyze sensitivities are used. The first is derived from an analytical solution of the state and power flow trajectories. Equation (28) presents the analytical solution of the temperature trajectories, with  $\alpha_1$  and  $\alpha_2$  as the eigenvalues of the system and  $\beta_1$  and  $\beta_2$  as the steady-state temperature values.

$$\begin{aligned} x_1(t_f) &= \gamma_1 e^{-\alpha_1 t_f} + \beta_1 \\ x_2(t_f) &= \frac{v}{L} \frac{\gamma_1}{\alpha_2 - \alpha_1} e^{-\alpha_1 t_f} + \gamma_2 e^{-\alpha_2 t_f} + \beta_2 \end{aligned} \quad (28a)$$

$$\begin{aligned} \alpha_1 &= \frac{v}{L} + \frac{z}{\rho L \theta_c A_{nom} C_p R_c} \\ \alpha_2 &= \frac{v}{L} + \frac{1-z}{\rho L \theta_c A_{nom} C_p R_c} + \frac{1}{\rho L \theta_c A_{nom} C_p R_{c,load}} \end{aligned} \quad (28b)$$

$$\begin{aligned} \beta_1 &= \alpha_1^{-1} \left( \frac{v}{L} x_3^s + \frac{z}{\rho L \theta_c A_{nom} C_p R_c} x_1^s \right) \\ \beta_2 &= \alpha_2^{-1} \left( \frac{v}{L} \beta_1 + \frac{1-z}{\rho L \theta_c A_{nom} C_p R_c} x_1^s + \frac{1}{\rho L \theta_c A_{nom} C_p R_{c,load}} x_2^s \right) \end{aligned} \quad (28c)$$

$$\begin{aligned} \gamma_1 &= x_3^s - \beta_1 \\ \gamma_2 &= x_3^s - \frac{v}{L} \frac{\gamma_1}{\alpha_2 - \alpha_1} - \beta_2 \end{aligned} \quad (28d)$$

Equation (29) differentiates the above solution to analytically derive the design sensitivities.

$$\begin{aligned} \frac{\partial J_{total}}{\partial \theta_c} &= \frac{z}{R_c} \frac{\partial x_1}{\partial \theta_c} + \left( \frac{1-z}{R_c} + \frac{1}{R_{c,load}} \right) \frac{\partial x_2}{\partial \theta_c} \\ \frac{\partial J_{total}}{\partial \theta_z} &= \frac{1}{R_c} (x_1 - x_2) \frac{\partial z}{\partial \theta_z} + \frac{z}{R_c} \frac{\partial x_1}{\partial \theta_z} + \left( \frac{1-z}{R_c} + \frac{1}{R_{c,load}} \right) \frac{\partial x_2}{\partial \theta_z} \end{aligned} \quad (29a)$$

$$\begin{aligned} \frac{\partial x_1}{\partial \theta_c} &= \frac{\partial x_1}{\partial \alpha_1} \frac{\partial \alpha_1}{\partial \theta_c} + \frac{\partial x_1}{\partial \beta_1} \frac{\partial \beta_1}{\partial \theta_c} + \frac{\partial x_1}{\partial \gamma_1} \frac{\partial \gamma_1}{\partial \theta_c} \frac{\partial \beta_1}{\partial \theta_c} \\ \frac{\partial x_1}{\partial \theta_z} &= \frac{\partial x_1}{\partial \alpha_1} \frac{\partial \alpha_1}{\partial \theta_z} + \frac{\partial x_1}{\partial \beta_1} \frac{\partial \beta_1}{\partial \theta_z} + \frac{\partial x_1}{\partial \gamma_1} \frac{\partial \gamma_1}{\partial \theta_z} \frac{\partial \beta_1}{\partial \theta_z} \end{aligned} \quad (29b)$$

$$\begin{aligned} \frac{\partial x_2}{\partial \theta_c} &= \frac{\partial x_2}{\partial \alpha_1} \frac{\partial \alpha_1}{\partial \theta_c} + \frac{\partial x_2}{\partial \alpha_2} \frac{\partial \alpha_2}{\partial \theta_c} + \frac{\partial x_2}{\partial \beta_2} \frac{\partial \beta_2}{\partial \theta_c} + \frac{\partial x_2}{\partial \gamma_1} \frac{\partial \gamma_1}{\partial \beta_1} \frac{\partial \beta_1}{\partial \theta_c} \\ &+ \frac{\partial x_2}{\partial \gamma_2} \left( \frac{\partial \gamma_2}{\partial \alpha_1} \frac{\partial \alpha_1}{\partial \theta_c} + \frac{\partial \gamma_2}{\partial \alpha_2} \frac{\partial \alpha_2}{\partial \theta_c} + \frac{\partial \gamma_2}{\partial \gamma_1} \frac{\partial \gamma_1}{\partial \beta_1} \frac{\partial \beta_1}{\partial \theta_c} + \frac{\partial \gamma_2}{\partial \beta_2} \frac{\partial \beta_2}{\partial \theta_c} \right) \quad (29c) \\ \frac{\partial x_2}{\partial \theta_z} &= \frac{\partial x_2}{\partial \alpha_1} \frac{\partial \alpha_1}{\partial \theta_z} + \frac{\partial x_2}{\partial \alpha_2} \frac{\partial \alpha_2}{\partial \theta_z} + \frac{\partial x_2}{\partial \beta_2} \frac{\partial \beta_2}{\partial \theta_z} + \frac{\partial x_2}{\partial \gamma_1} \frac{\partial \gamma_1}{\partial \beta_1} \frac{\partial \beta_1}{\partial \theta_z} \\ &+ \frac{\partial x_2}{\partial \gamma_2} \left( \frac{\partial \gamma_2}{\partial \alpha_1} \frac{\partial \alpha_1}{\partial \theta_z} + \frac{\partial \gamma_2}{\partial \alpha_2} \frac{\partial \alpha_2}{\partial \theta_z} + \frac{\partial \gamma_2}{\partial \gamma_1} \frac{\partial \gamma_1}{\partial \beta_1} \frac{\partial \beta_1}{\partial \theta_z} + \frac{\partial \gamma_2}{\partial \beta_2} \frac{\partial \beta_2}{\partial \theta_z} \right) \end{aligned}$$

The second sensitivity analysis approach is derived from an approximate numerical solution of the trajectories, using fourth-order Runge–Kutta integration. Equation (30) presents the numerical solution of the temperature trajectories, with  $\Delta t$  as the time-step.

$$\mathbf{g} = \begin{cases} -\alpha_1 x_1 + \alpha_1 \beta_1 \\ \frac{v}{L} x_1 - \alpha_2 x_2 + \alpha_2 \beta_2 - \frac{v}{L} \beta_1 \end{cases} \quad (30a)$$

$$\begin{aligned} \mathbf{K}_1 &= \Delta t \cdot \mathbf{g}(t_n, \mathbf{x}_n) \\ \mathbf{K}_2 &= \Delta t \cdot \mathbf{g}\left(t_n + \frac{\Delta t}{2}, \mathbf{x}_n + \frac{\mathbf{K}_1}{2}\right) \\ \mathbf{K}_3 &= \Delta t \cdot \mathbf{g}\left(t_n + \frac{\Delta t}{2}, \mathbf{x}_n + \frac{\mathbf{K}_2}{2}\right) \\ \mathbf{K}_4 &= \Delta t \cdot \mathbf{g}(t_n + \Delta t, \mathbf{x}_n + \mathbf{K}_3) \end{aligned} \quad (30b)$$

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \frac{1}{6}(\mathbf{K}_1 + \mathbf{K}_2 + \mathbf{K}_3 + \mathbf{K}_4) \quad (30c)$$

We then use the adjoint method to derive the sensitivity formulation corresponding to the Runge–Kutta solution. Equation (30c) is rewritten in residual form as:

$$\mathbf{R}_{n+1} = \mathbf{x}_n - \mathbf{x}_{n+1} + \frac{1}{6}(\mathbf{K}_1 + \mathbf{K}_2 + \mathbf{K}_3 + \mathbf{K}_4) \quad (31)$$

The Lagrangian form of the objective function is defined based on the residual, with  $N_f$  representing the number of time-steps in the numerical simulation, and  $\xi_n^T$  representing the set of adjoint vectors:

$$\Pi = J_{total} + \sum_{n=1}^{N_f} \xi_n^T \mathbf{R}_n \quad (32)$$

The total sensitivity of the objective function with respect to design variables  $\theta$  is:

$$\begin{aligned} \frac{dJ_{total}}{d\theta} &= \frac{d\Pi}{d\theta} = \frac{\partial J_{total}}{\partial \theta} + \sum_{n=1}^{N_f} \frac{\partial J_{total}}{\partial \mathbf{x}_n} \frac{d\mathbf{x}_n}{d\theta} \\ &+ \sum_{n=1}^{N_f} \xi_n^T \left( \frac{\partial \mathbf{R}_n}{\partial \theta} + \frac{\partial \mathbf{R}_n}{\partial \mathbf{x}_n} \frac{d\mathbf{x}_n}{d\theta} + \frac{\partial \mathbf{R}_n}{\partial \mathbf{x}_{n-1}} \frac{d\mathbf{x}_{n-1}}{d\theta} \right) \quad (33) \end{aligned}$$

Equation (34) is determined by setting the sum of all implicit terms to zero, which can be used to solve for the adjoint vectors as shown in Eq. (35).

$$\begin{aligned} \left( \xi_n^T \frac{\partial \mathbf{R}_n}{\partial \mathbf{x}_n} + \xi_{n+1}^T \frac{\partial \mathbf{R}_n}{\partial \mathbf{x}_n} \right) \frac{d\mathbf{x}_n}{d\theta} &= 0, \quad n < N_f \\ \left( \frac{\partial J_{total}}{\partial \mathbf{x}_n} + \xi_n^T \frac{\partial \mathbf{R}_n}{\partial \mathbf{x}_n} \right) \frac{d\mathbf{x}_n}{d\theta} &= 0, \quad n = N_f \quad (34) \end{aligned}$$

$$\begin{aligned} \xi_n &= -\left( \frac{\partial \mathbf{R}_n}{\partial \mathbf{x}_n} \right)^{-T} \left( \frac{\partial \mathbf{R}_{n+1}}{\partial \mathbf{x}_n} \right)^T \xi_{n+1}^T, \quad n < N_f \\ \xi_n &= -\left( \frac{\partial \mathbf{R}_n}{\partial \mathbf{x}_n} \right)^{-T} \left( \frac{\partial J_{total}}{\partial \mathbf{x}_n} \right), \quad n = N_f \end{aligned} \quad (35)$$

Using this adjoint solution, which causes all implicit terms to vanish, Eq. (33) reduces to the final form shown in Eq. (36). This provides an exact derivative of the numerically computed objective function.

$$\frac{dJ_{total}}{d\theta} = \frac{\partial J_{total}}{\partial \theta} + \sum_{n=1}^{N_f} \xi_n^T \frac{\partial \mathbf{R}_n}{\partial \theta} \quad (36)$$

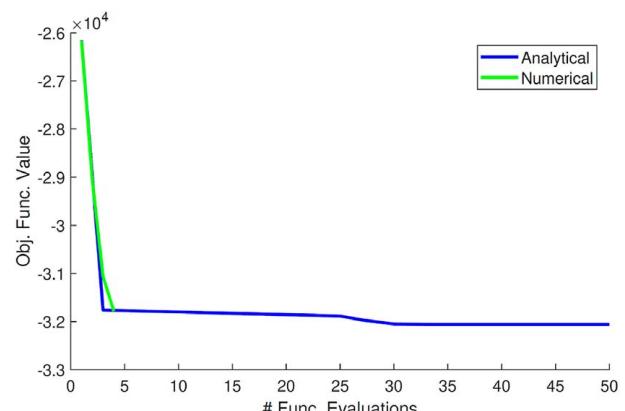
Both of these sensitivity functions are used to guide the gradient-based optimizer in the Sec. 4.4.

**4.4 Results.** Table 1 provides the results of using SQP to optimize the system with both the analytical and Runge–Kutta approaches, and Fig. 5 presents the convergence plot for both functions. Note that the true optimal set, determined by a grid search, is  $\theta = [1.25, 1]$ . This matches the intuition that the best cooling would come from cooling the battery first with the largest pipe possible. Comparing the results of Table 1 against this true set shows that both solution methods converge to the optimal design. This shows that the design optimization framework produces accurate results for this case study.

Table 1 shows that the analytical method reaches the optimum an order of magnitude faster than the Runge–Kutta approach, as expected. However, the benefit of the numerical approach is its applicability to more systems and the reduced number of function evaluations as compared to the analytical method. For larger or more dynamically nonlinear systems, analytically solving the system response and its accompanying design sensitivities becomes more challenging, which is addressed using the Runge–Kutta approach.

**Table 1 Results for the first case study**

	Initial	Analytical	Numerical
$\theta_c$	1	1.25	1.25
$\theta_z$	0	1	1
Number of function evaluations	—	50	4
Convergence time (s)	—	0.013	0.22



**Fig. 5 Convergence plot for the first case study using analytically and numerically based sensitivity functions**

## 5 Case Study 2: Design of an Electric Vehicle Powertrain

This section applies the design optimization framework to the case study of EV powertrain configuration that considers electrical, mechanical, and thermal power flows. The baseline configuration with sizing and architecture options is outlined and translated into a graph-based model. Configuration options are translated into an optimization problem through the framework. Powertrain configurations are determined for two different sets of optimization function weight values, highlighting the ability of the design framework to identify different designs based on the design engineer's requirements.

### 5.1 Baseline Electric Vehicle Powertrain Configuration.

Mobile power systems such as aircraft and on- and off-highway vehicles are becoming increasingly electrified to reduce fuel costs [39], greenhouse gas emissions [40], and weight [7,39]. However, electrification comes with the price of rising thermal management challenges, leading to reduced component lifespans [41], limited performance [42], and power electronic defects [43]. Design optimization is a powerful tool to mitigate these thermal management issues through improved early-stage architecture and component selection. This case study uses the design framework to optimize an EV powertrain, with consideration of the multi-domain interactions found within such a system.

Figure 6 presents the baseline EV powertrain configuration, and Figs. 7–9 show the graphical models of the three sections of the powertrain. The powertrain consists of a battery as the main power source, connected in parallel to a high-voltage direct current (DC) bus. This voltage bus splits power into two bidirectional inverters that power AC motors which drive the vehicle inertia, and a bidirectional DC/DC converter. In addition, the voltage bus connects to a dissipative resistor through a switch to prevent overvoltage. The DC/DC converter steps voltage down to a low voltage DC bus. This low voltage bus powers five DC/DC converters. The first powers extraneous loads such as the radio and power steering. The second powers the vapor compression system (VCS), which cools the liquid cooling loop fluid and dissipates heat into the environment. The third, fourth, and fifth converters power motors that run the fan for the cabin air cycle, the fan for the air cooling path, and the pump for the liquid cooling loop, respectively. Inputs to the system include each inverter and converter duty cycle (eight total) plus the braking command. In this article, the VCS converter's duty cycle uses proportional control to track a desired liquid loop temperature, while all other inputs

are held constant. Switch input commands include the directionality command for the eight converters (buck or boost) plus the command to connect the dissipative resistor to the high-voltage bus. The switch commands are all held to buck mode for the case study. Sink states (disturbances) include ambient air temperature, current demanded by external loads, radiative and convective heat into the cabin, and heat generated in the cabin. For clarity, the definitions of  $\mathbf{x}$ ,  $\mathbf{C}$ , and  $\mathbf{y}$ , as well as nominal parameter values, are presented in Appendix A. Note that these notional parameter values and conditions permit the validation of the design framework without focus on a specific powertrain. This system does not have any source edges, and as such  $\mathbf{D} = \mathbf{0}$  and  $\mathbf{y}^* = \mathbf{0}$ .

This case study explores six sizing design options ( $N_c = 6$ ) and five discrete topology options ( $N_z = 5$ ). The sizing design options include size of the battery ( $\theta_{c,1}$ ), the pair of motor #1 and inverter #1 ( $\theta_{c,2}$ ), the pair of motor #2 and inverter #2 ( $\theta_{c,3}$ ), the VCS ( $\theta_{c,4}$ ), the air cooling path fan ( $\theta_{c,5}$ ), and the liquid cooling loop pump ( $\theta_{c,6}$ ). The topology options include the decision to keep or remove the pair of motor #2 and inverter #2 ( $z_1$ ), as well as to choose air or liquid cooling for the battery ( $z_2$ ), the power electronics ( $z_3$ ), and the motors ( $z_4$  and  $z_5$ ). Section 5.2 relates these design options to the graph-based model through the framework.

### 5.2 Applying the Framework to the Electric Vehicle Powertrain.

The steps of the design optimization framework are applied to the case study system. Step 1 requires definitions for the design matrices described in Eq. (4), which are presented in Appendix B.

In Step 2 of the framework, we define the objective function,  $J_{total}$ , which is given by the following linear weighted combination:

$$\begin{aligned} J_{total} &= w_1 J_{perf} + w_2 J_{maxT} + w_3 J_{tr} + w_4 J_{st} \\ J_{maxT} &= w_a J_{maxT,batt} + w_b J_{maxT,m1} + w_c J_{maxT,powElec} \end{aligned} \quad (37)$$

with the coefficients  $w_1, \dots, w_4$  determining the relative weighting of each term. The performance time objective function,  $J_{perf}$ , is the amount of time the vehicle takes to reach a desired velocity of 45 mph when starting from rest. The function  $J_{maxT}$  is a weighted summation of the maximum temperatures of the battery ( $J_{maxT,batt}$ ), first motor ( $J_{maxT,m1}$ ), and the first inverter and the six DC/DC converters ( $J_{maxT,powElec}$ ). The internal weights  $w_a = 0.162$ ,  $w_b = 0.142$ , and  $w_c = 0.0202$  are selected to place equal importance on each device relative to the component's maximum operating temperature.

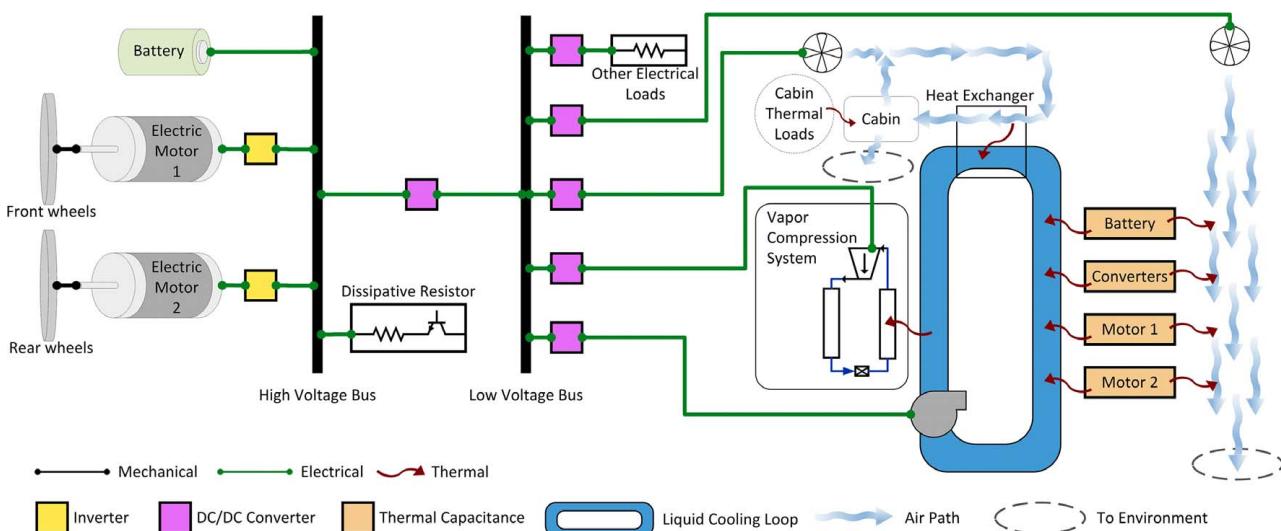


Fig. 6 Electric vehicle baseline powertrain configuration

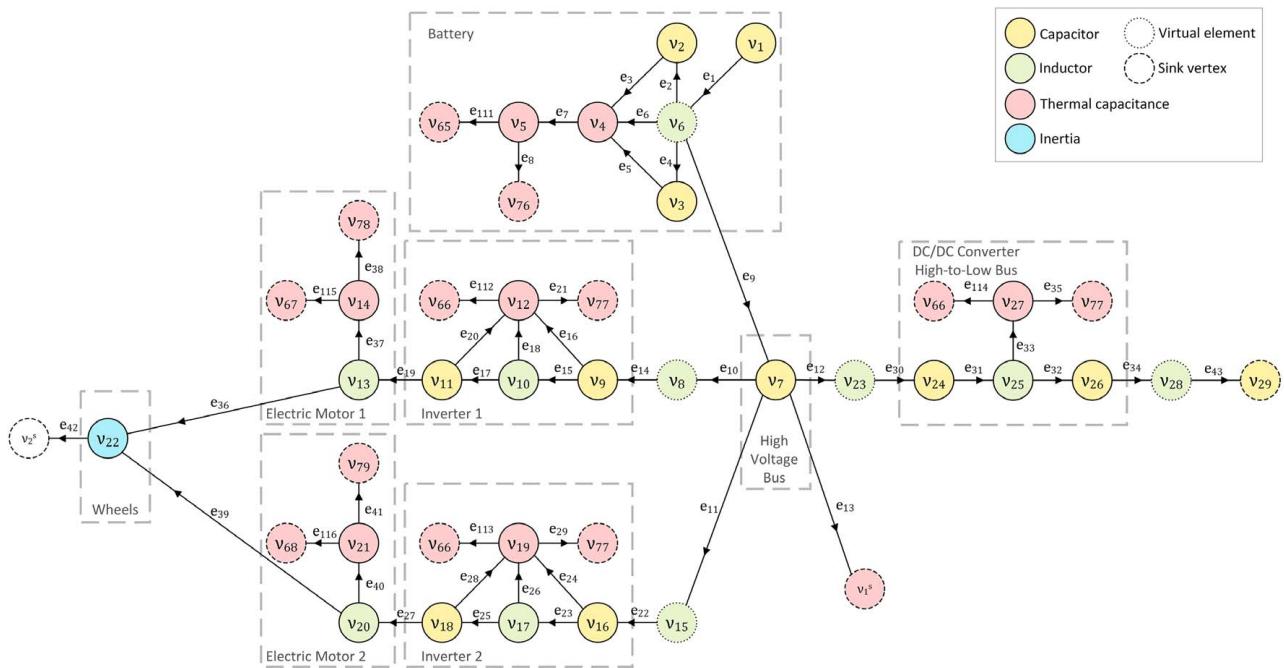


Fig. 7 Graph-based model of the battery and motors (Color version online.)

The transfer rate function,  $J_{tr}$ , represents energy losses during the performance period through the battery, motors, and power electronics. This function has the same form as Eq. (13). The internal weights  $w_{tr,j}$  are assigned a value of 1 for the following values of  $j$ :

$$j = 3, 5, 6, 16, 18, 20, 24, 26, 28, 33, 37, 40, 52, 58, 64, 70, 76$$

The internal weights are set to zero for all other values of  $j$ . The storage objective function,  $J_{st}$ , represents the change in energy stored in the battery during the performance period, and has the same form as Eq. (14). The internal weights  $w_{st,i}$  are set to 1 for  $i = 1, 2, 3$ , and zero for all other values of  $i$ . The objective function is defined in a manner to identify one or multiple potential

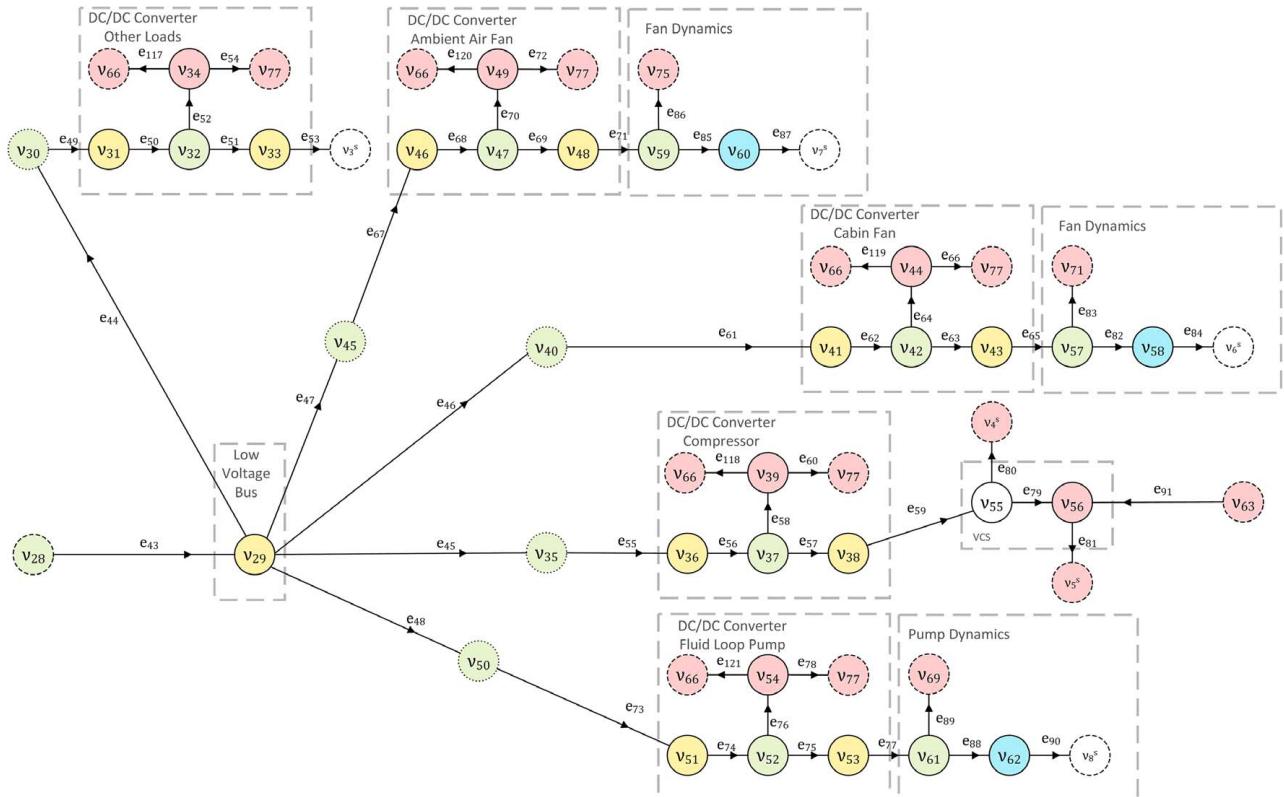
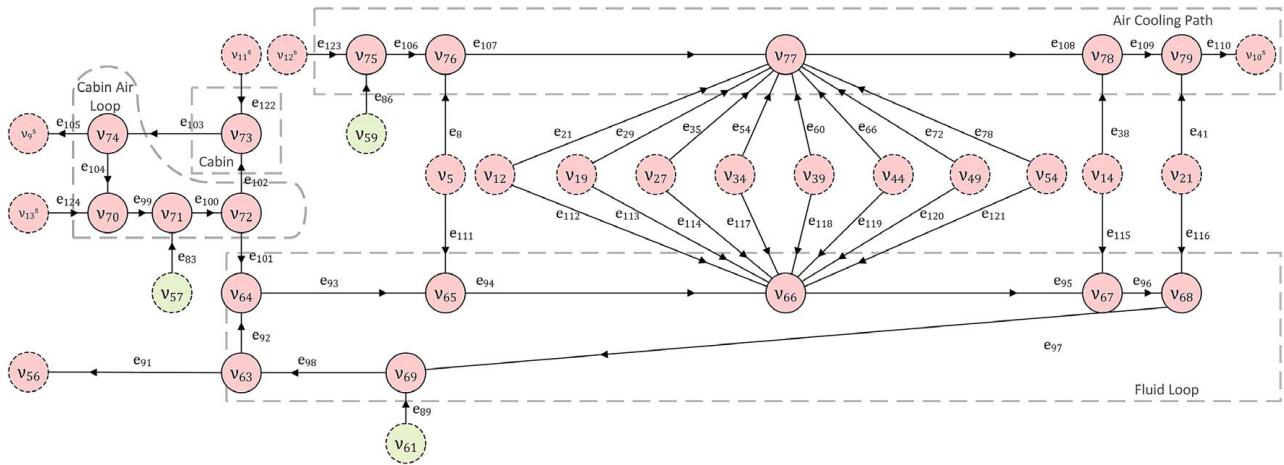


Fig. 8 Graph-based model of the low voltage bus and DC/DC converters (Color version online.)



**Fig. 9 Graph-based model of the cabin, air cooling path, and liquid cooling paths (Color version online.)**

**Table 2 Weights for the second case study**

Weight	Test #1	Test #2
$w_1$	0.05	0.05
$w_2$	0	1
$w_3$	0.025	0
$w_4$	-0.033	0

designs, based on the weights selected by the design engineer. This can be useful for designing products for slightly different markets, such as designing a sedan versus a minivan, without searching the design space for all optimized designs for different weights.

In *Step 3*, we set the design variable bounds. As mentioned in the description of step 1 above, the design variables have been normal-

**Table 3 Results for the second case study, test 1**

	Best initial set (global optimum trial)	Final (global optimum trial)	Final (median trial)	Standard deviation between trials
$\theta_{c,1}$	0.063	$2.7 \times 10^{-5}$	$9.5 \times 10^{-5}$	$2.6 \times 10^{-5}$
$\theta_{c,2}$	1.0	1.0	1.0	$2.0 \times 10^{-5}$
$\theta_{c,3}$	0.88	0.50	0.51	$9.1 \times 10^{-3}$
$\theta_{c,4}$	0.14	0.037	$5.1 \times 10^{-4}$	0.015
$\theta_{c,5}$	0.72	0.94	0.81	0.18
$\theta_{c,6}$	0.94	0.92	0.87	0.048
$z_1$	1	1	1	0
$z_2$	1	1	1	0
$z_3$	0	1	1	0
$z_4$	1	1	1	0
$z_5$	1	1	1	0
$J_{total}$	$5.49 \times 10^4$	$5.11 \times 10^4$	$5.11 \times 10^4$	0.83
Convergence time (s)	—	$4.6 \times 10^4$	$4.6 \times 10^4$	$1.3 \times 10^3$
Number of function evaluations	—	6000	6000	179

**Table 4 Results for the second case study, test 2**

	Best initial set (global optimum trial)	Final (global optimum trial)	Final (median trial)	Standard deviation between trials
$\theta_{c,1}$	0.15	$2.0 \times 10^{-6}$	$2.8 \times 10^{-6}$	$2.8 \times 10^{-5}$
$\theta_{c,2}$	0.94	1.0	1.0	$3.1 \times 10^{-5}$
$\theta_{c,3}$	0.54	$1.6 \times 10^{-3}$	$1.5 \times 10^{-3}$	$1.8 \times 10^{-3}$
$\theta_{c,4}$	0.77	0.069	0.011	0.37
$\theta_{c,5}$	0.027	$5.4 \times 10^{-3}$	0.018	0.1
$\theta_{c,6}$	0.90	$2.0 \times 10^{-3}$	0.016	$7.7 \times 10^{-3}$
$z_1$	1	1	1	0
$z_2$	1	1	1	0
$z_3$	0	0	0	0
$z_4$	1	1	1	0.45
$z_5$	1	1	1	0
$J_{total}$	142.08	141.16	141.16	$6.4 \times 10^{-3}$
Convergence time (s)	—	$3.0 \times 10^4$	$3.0 \times 10^4$	$1.6 \times 10^3$
Number of function evaluations	—	5300	5200	89

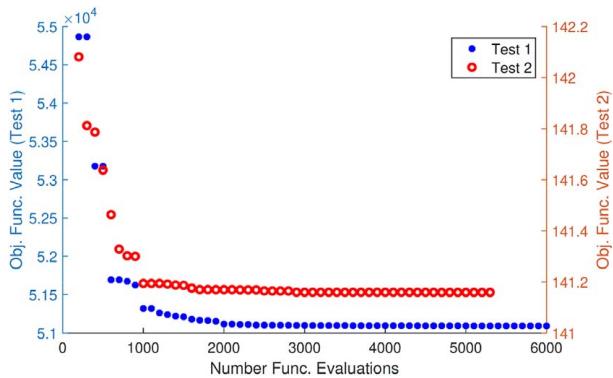


Fig. 10 Convergence plot for the two tests in the second case study

ized for this case study. For  $\theta = [\theta_{c,1}, \dots, \theta_{c,6}, z_1, \dots, z_5] = [\theta_1, \dots, \theta_{11}]$ , the bounds are:

$$0 \leq \theta \leq 1 \quad (38)$$

Again, for this case study, no nonlinear design constraints are explicitly enforced.

Step 4 of the framework can now be applied, with the full problem having the same form as Eq. (16). For the dynamic

simulation, the vehicle is considered initially at rest, with a battery state of charge (SOC) of 80% and all temperatures equal to the ambient temperature. The shooting method is used again, with the dynamics simulated using MATLAB's "ode15s" function. The MATLAB genetic algorithm function "ga" is utilized to optimize the system, with a population of 100 and default initial population settings, and function convergence criteria of  $10^{-4}$ . The code is run on a desktop computer with a 3.4 GHz Intel Core i7-6700 processor and 16 GB RAM. Due to the stochastic nature of the algorithm, five trials are run for each set of objective function weights to better understand the convergence of the designs.

**5.3 Results.** For this second case study, the design framework is evaluated using two different sets of objective function weights. Test #1 emphasizes minimizing performance time ( $J_{perf}$ ), energy losses through the battery, motors, and power electronics ( $J_{tr}$ ), and the change in energy stored in the battery ( $J_{st}$ ). Test #2 places focus on minimizing performance time and minimizing the maximum temperature of the battery, first motor, and power electronics ( $J_{maxT}$ ). Table 2 presents the weights for each of these tests.

Tables 3 and 4 present the design parameter values, convergence times, and function evaluations for the two tests. Figure 10 presents the convergence plots for the trials with the global optimal cases (referred to as the global optimum trials), showing that the GA converges to a superior design as compared to the initial configurations. The low standard deviation values of the parameters and objective

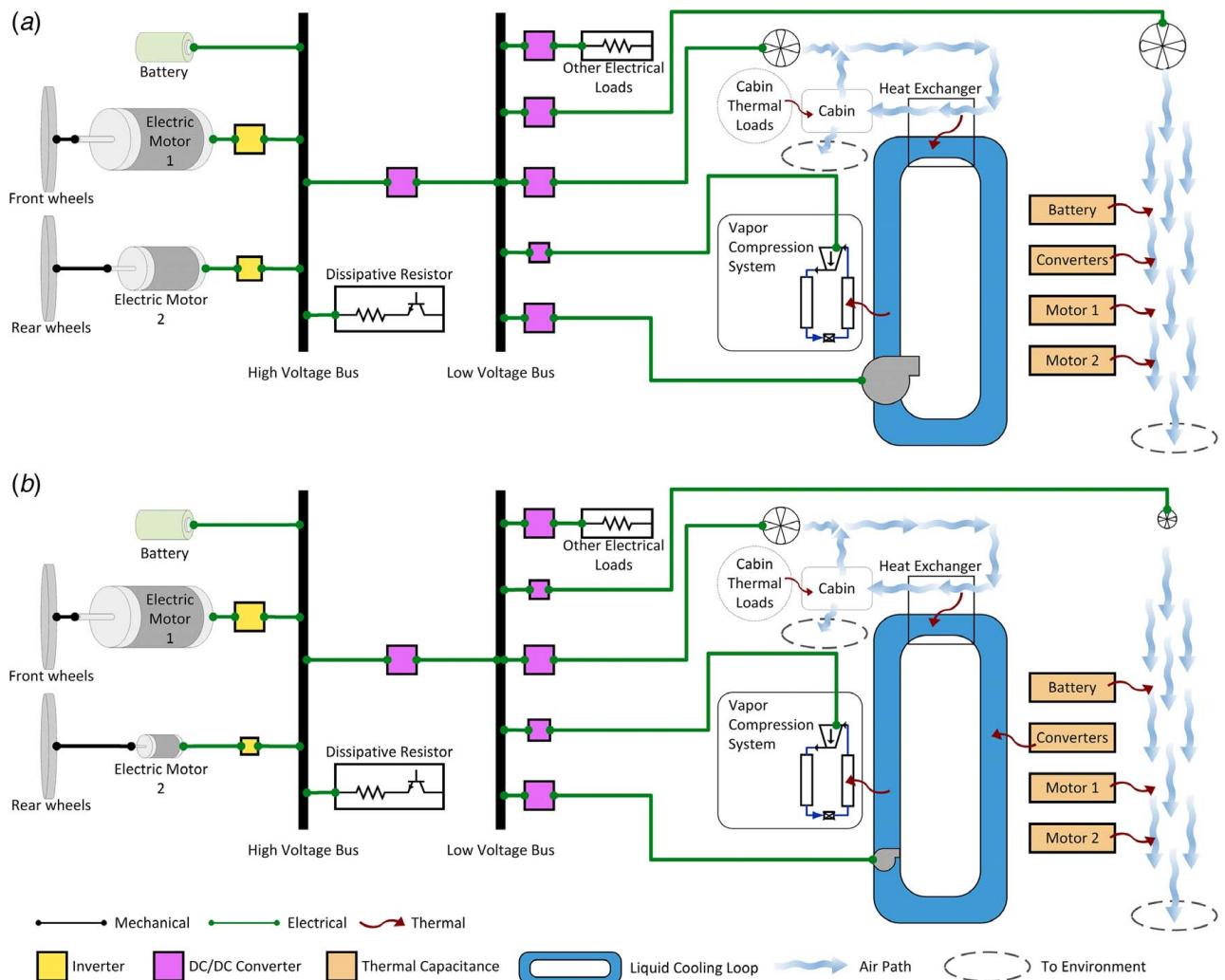
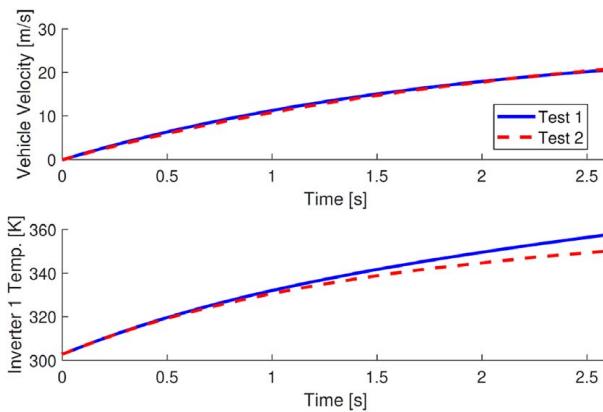
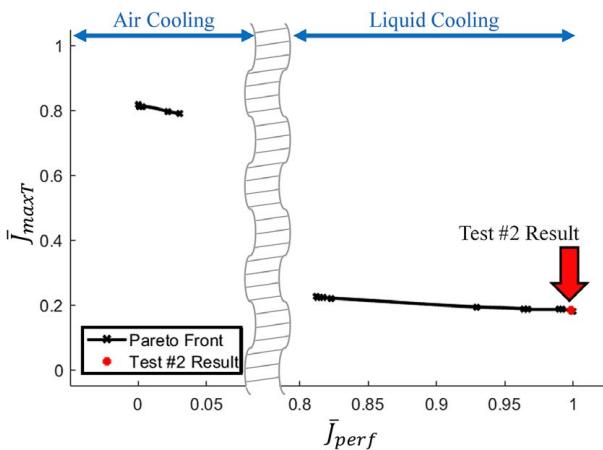


Fig. 11 Case study 2 optimized powertrain designs from the global optimum trials of the (a) first and (b) second tests



**Fig. 12 Vehicle velocity and first inverter temperature for the two optimized powertrain configurations**



**Fig. 13 The Pareto front for  $J_{perf}$  versus  $J_{maxT}$ , normalized as represented by the overbar**

functions for each test show the similarity of the final designs of each trial. This indicates that the trials for each test converge to a similar final design. Relatively higher values of standard deviation are found for variables sized to be near zero (e.g.,  $\theta_{c,1}$  in both tests) due to the variable value approaching zero.

Figure 11(a) presents the global optimum powertrain design configuration for test 1. The battery is at the minimum allowable size, the first motor and inverter are at the maximum allowable size, and the second motor and inverter are at an intermediate size. The air cooling path's fan is at a large size, and air cooling is used to cool all components. Figure 11(b) presents the global optimum powertrain design configuration for test 2. In this case, the battery and second motor are sized as small as allowable. The air cooling path's fan and liquid cooling loop's pump, as well as the associated converters, are also sized as small as possible. Air cooling is used for all components except for the power electronics, which are cooled by the liquid loop. The only discrete variable within a nonzero standard deviation is  $z_4$  in test 2, due to the low impact this motor cooling option has on the objective function.

Figure 12 plots the velocity of the vehicle for both global optimum design configurations during the performance test period, and the temperature of the first inverter during this test period. The configuration from test 1 has a slightly slower performance time (2.48 s) as compared to the configuration from test 2 (2.45 s). Furthermore, the first configuration has a much higher inverter temperature as compared to the second configuration. This matches the selection of liquid cooling by the optimizer in the second configuration. The first configuration, however, has

$J_{tr} = 2.58 \times 10^5$  J, as compared to  $J_{tr} = 2.74 \times 10^5$  J for the second configuration. The first configuration is sized in such a manner to minimize these energy losses. Summarizing these results for the designed systems:

- The first configuration reduces energy losses during an acceleration phase of the vehicle as compared to the second configuration.
- The second configuration provides a slightly better performance time, as well as lower temperature power electronic components, due to the liquid cooling for these components.

Beyond identifying specific designs, the steps of the framework are also capable of identifying Pareto optimal solutions. As an example, the weights  $w_1$  and  $w_2$  are varied to generate a 2D Pareto front, with  $w_3=0$  and  $w_4=0$ . Figure 13 presents the Pareto front with the objective functions normalized, determined by the compromise programming approach with  $n=4$  [37]. The GA is run iteratively with the same equipment and solver settings as used with the earlier EV design tests. Note that due to discontinuities in the design space caused by the presence of discrete variables, particularly  $z_3$ , there is a discontinuity in the Pareto front. This discontinuity is characterized by a region within objective space (represented by a vertical band) that is unpopulated due to the large discontinuous jump in performance ( $J_{perf}$ ) that occurs when transitioning from air cooling ( $z_3 = 1$ ) to liquid cooling ( $z_3 = 0$ ) of the power electronics. The arrow in Fig. 13 points to the placement of the global optimum trial's design for Test #2 using the linear weighted combination method. This shows that the design identified by the global optimum trial lies on the Pareto front.

## 6 Conclusions

This article presents a novel design framework to accomplish MDO for dynamic, conservation-based SoS. The framework utilizes a graphical modeling tool to capture the physical dynamics and interactions found among these systems. By augmenting this representation of the system, design options, including component sizing and topologies, are represented by the vertices and edges of the graph, easing the formulation of the design problem. The design framework allows the design engineer to select the solver, and permits simultaneous optimization of both component sizes and configurations. Two case studies are explored to provide examples of the framework's performance for conservation-based systems. The first case study explores the design of a cooling subsystem, which is optimized using a gradient-based solver. In this case study, the system response and design sensitivities are modeled using both numerical and analytical approaches. The second case study explores the design space of an EV powertrain, sizing electrical, mechanical, and thermal components, as well as making discrete architecture choices. Future work will include the expansion of the design framework to consider topology generation and improved solution speed using derived sensitivity functions specific to the graphical nature of the conservation-based modeling approach.

## Acknowledgment

This material is based upon work supported by the National Science Foundation Engineering Research Center for Power Optimization of Electro Thermal Systems (POETS) with cooperative agreement EEC-1449548.

## Appendix A

This appendix summarizes the dynamics of the graph-based model for the baseline powertrain configuration (Figs. 7–9), and Table 5 presents the parameter values. There is a total of  $N_v=79$  vertices,  $N_e=124$  edges, and  $N_t=13$  sink vertices. In Figs. 7–9, there are four vertex types which identify if a vertex's state

**Table 5 Parameter values for case study 2**

Symbol	Value
$Q$	$N_{ser} \cdot 2.27 \cdot 3600 \text{ C}$
$C_1$	$2500 \cdot N_{ser} \text{ F}$
$C_2$	$7.5 \times 10^4 \cdot N_{ser} \text{ F}$
$\mathbb{C}_{b,c}$	$62.7 \cdot N_{ser} \text{ J/K}$
$\mathbb{C}_{b,s}$	$4.5 \cdot N_{ser} \text{ J/K}$
$N_{ser}$	108
$R_1$	$\frac{0.01}{N_{ser}} \Omega$
$R_2$	$\frac{0.02}{N_{ser}} \Omega$
$R_s$	$0.01 \cdot N_{ser} \Omega$
$R_{p,c}$	$\frac{1.94}{N_{ser}} \text{ K/W}$
$C_{bus}$	0.0272 F
$R_{bus}$	12 Ω
$\mathbb{C}_{inv}$	2800 J/K
$R_{inv,C1}$	11 Ω
$R_{inv,C2}$	11 Ω
$R_{inv,L}$	0.03 Ω
$L_m$	0.0012 H
$R_m$	0.15 Ω
$\mathbb{C}_m$	14400 J/K
$k_m$	0.45 N m/A
GR	10
$R_w$	0.5 m
$\mathbb{C}_c$	2800 J/K
$R_c$	0.12 Ω
$R_{VCS}$	0.61 Ω
$\eta_{VCS}$	0.85
$COP_{nom}$	4.16
$k_p$	0.25 N m/A
$R_p$	0.2 Ω
$b_p$	0.0027 N m s
$\mathbb{C}_i$ ( $i = 63, \dots, 79$ )	123 J/K
$C_{p,liq}$	4186 J/kg K
$C_{p,air}$	1005 J/kg K
$\mathbb{C}_{cab}$	3500.42 J/K
$R_{c,cab}$	12 K/W
$A_{inv}$	0.1 m <sup>2</sup>
$A_c$	0.1 m <sup>2</sup>
$A_m$	0.1 m <sup>2</sup>

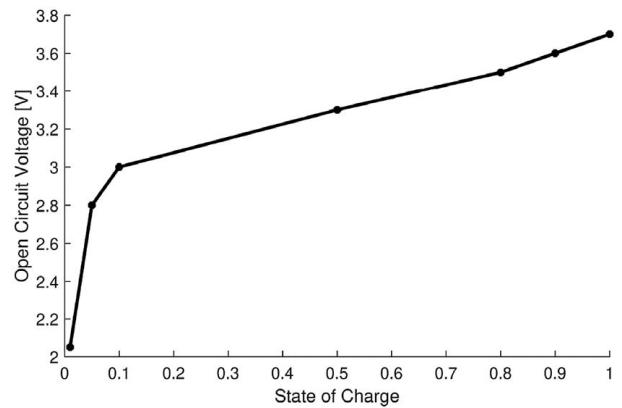
describes (i) a voltage or charge, (ii) a current, (iii) a temperature, or (iv) a velocity. The sink states represent ambient air temperature ( $v_1^s, v_2^s, v_4^s, \dots, v_{10}^s, v_{12}^s, v_{13}^s$ ), current demanded by external loads ( $v_3^s$ ), and radiative and convective heat into the cabin with heat generated in the cabin ( $v_{11}^s$ ). The matrix  $\bar{\mathbf{M}}$  is determined from Figs. 7–9, and  $\mathbf{P}^s = 0$  and  $\mathbf{D} = 0$ .

The battery, with capacity  $Q$ , is modeled using third-order electrical circuit dynamics with second-order temperature dynamics [44], described by vertices [ $v_1, \dots, v_5$ ] and edges [ $e_1, \dots, e_8, e_{11}$ ] [34]. Equation (A1) defines the capacitance matrix of the battery ( $\mathbf{C}_{batt}$ ), with  $C_1$  and  $C_2$  as the electrical capacitances of the first and second relaxation terms, respectively. The variables  $\mathbb{C}_{b,c}$  and  $\mathbb{C}_{b,s}$  are the thermal capacitances of the core and surface of the battery, respectively. The nonlinear open-circuit voltage,  $V_{oc}$ , is a function of battery state of charge  $x_1$ , as defined in Fig. 14.

$$\mathbf{C}_{batt} = \text{diag}([QV_{oc}(x_1), C_1x_2, C_2x_3, \mathbb{C}_{b,c}, \mathbb{C}_{b,s}]) \quad (\text{A1})$$

The electrical battery elements and the high-voltage bus ( $v_7$ ) all share the same current, represented by the virtual vertex  $v_6$ . The expressions for these power flows are:

$$\begin{aligned} y_1 &= N_{ser} V_{oc}(x_1)x_6 \\ y_2 &= N_{ser}x_2x_6 \\ y_4 &= N_{ser}x_3x_6 \\ y_9 &= x_7x_6 \end{aligned} \quad (\text{A2})$$

**Fig. 14 Open-circuit voltage for the battery**

Note that all edges that connect to a virtual element that are not defined take the form of  $y_j = x_j^{head}x_j^{tail}$ . Heat is dissipated to the core of the battery due to inefficiencies represented by battery resistances  $R_1$ ,  $R_2$ , and  $R_s$ :

$$\begin{aligned} y_3 &= \frac{1}{R_1}x_2^2 \\ y_5 &= \frac{1}{R_2}x_3^2 \\ y_6 &= R_s x_6^2 \end{aligned} \quad (\text{A3})$$

Equation (A4) represents conduction from the battery core to the battery surface, with thermal resistance  $R_{b,c}$ :

$$y_7 = \frac{1}{R_{b,c}}(x_4 - x_5) \quad (\text{A4})$$

Power dissipated through air cooling ( $y_8$ ) and liquid cooling ( $y_{111}$ ) is described by:

$$\begin{aligned} y_8 &= h_{batt,air}(\dot{m}_{air})(x_5 - x_{76}) \\ y_{111} &= h_{batt,liq}(\dot{m}_{liq})(x_5 - x_{65}) \end{aligned} \quad (\text{A5})$$

In Eq. (A5),  $h_{batt,air}$  and  $h_{batt,liq}$  represent the effective convection coefficient terms from air or liquid cooling. These are linearly related to the mass flowrates of air ( $\dot{m}_{air}$ , related to fan speed  $x_{60}$ ) or the liquid ( $\dot{m}_{liq}$ , related to pump speed  $x_{62}$ ), respectively. The area of the battery, which also impacts these edge terms, is defined through the design matrices in Appendix B.

The high-voltage bus has a storage capacitance of  $C_{bus}x_7$ , and connects to surrounding virtual vertices using an expression of the same form as  $y_9$ . The power flow through the dissipative resistance  $R_{bus}$  is described by:

$$y_{13} = \frac{s_9}{R_{bus}}x_7^2 \quad (\text{A6})$$

where  $s_9$  is a discrete switching input that connects or disconnects the resistor from the voltage bus.

The first inverter is represented as a converter, with dynamics described by using the small ripple approximation [45,46]. There are three electrical vertices ( $v_9, v_{10}, v_{11}$ ) of zero storage capacitance, and a thermal vertex with thermal capacitance  $\mathbb{C}_{inv}$ . Power flowing into the converter, through it, and to the motor is expressed as:

$$\begin{aligned} y_{14} &= x_9x_8 \\ y_{15} &= (s_1u_1 + (1 - s_1)(1 - u_1))x_{10}x_9 \\ y_{17} &= x_{11}x_{10} \\ y_{19} &= x_{13}x_{11} \end{aligned} \quad (\text{A7})$$

In Eq. (A7),  $s_1$  is a switching input that determines buck versus boost mode, and  $u_1$  is the duty cycle for the inverter. Inefficiencies in the first inverter are expressed using resistances  $R_{inv,C1}$ ,  $R_{inv,L}$ , and  $R_{inv,C2}$ :

$$\begin{aligned} y_{16} &= \frac{s_1 u_1 + (1 - s_1)(1 - u_1)}{R_{inv,C1}} x_9^2 \\ y_{18} &= R_{inv,L} x_{10}^2 \\ y_{20} &= \frac{s_1 u_1 + (1 - s_1)(1 - u_1)}{R_{inv,C2}} x_{11}^2 \end{aligned} \quad (\text{A8})$$

Power dissipated through air cooling ( $y_{21}$ ) and liquid cooling ( $y_{112}$ ) is expressed in a manner similar to the battery's cooling terms, with  $A_{inv}$  as the inverter's surface area for cooling:

$$\begin{aligned} y_8 &= h_{inv,1,air}(\dot{m}_{air}) A_{inv}(x_{12} - x_{77}) \\ y_{111} &= h_{inv,1,liq}(\dot{m}_{liq}) A_{inv}(x_{12} - x_{66}) \end{aligned} \quad (\text{A9})$$

The first motor's electrical dynamics are described by an electrical storage capacitance of  $L_m x_{13}$  for  $v_{13}$ , with  $L_m$  as the motor inductance. Motor resistance  $R_m$  means that some power is transferred to the thermal capacitance  $C_m$  of the motor:

$$y_{37} = R_m x_{13}^2 \quad (\text{A10})$$

Power dissipated through air cooling ( $y_{38}$ ) and liquid cooling ( $y_{115}$ ) is expressed in a manner similar to the battery's cooling terms, with  $A_m$  as the inverter's surface area for cooling:

$$\begin{aligned} y_{38} &= h_{m,1,air}(\dot{m}_{air}) A_m(x_{14} - x_{78}) \\ y_{115} &= h_{m,1,liq}(\dot{m}_{liq}) A_m(x_{14} - x_{67}) \end{aligned} \quad (\text{A11})$$

The motor transfers the remaining power to the vehicle's mass ( $v_{22}$ , nominal capacitance of 1) to propel the linear velocity  $x_{22}$  of the vehicle:

$$y_{36} = k_m \frac{\text{GR}}{R_w} x_{22} x_{13} \quad (\text{A12})$$

In Eq. (A12), GR is the gear ratio of the transmission and final drive, and  $R_w$  is the radius of the vehicle's wheels. Edge  $e_{42}$ 's power flow is:

$$y_{42} = u_9 x_{22} \quad (\text{A13})$$

where  $u_9$  is an input that controls the braking force. The models for inverter 2 and motor 2 follow the same procedure as with inverter 1 and motor 1, with inputs  $u_2$  and  $s_2$ .

The model of the DC/DC converter from the high-to-low voltage bus is very similar to the inverter model, only with the removal of the resistances connected to the electrical capacitor terms. With thermal capacitance  $C_c$  ( $v_{27}$ ), the power flows of the converter are:

$$\begin{aligned} y_{30} &= x_{24} x_{23} \\ y_{31} &= (s_3 u_3 + (1 - s_3)(1 - u_3)) x_{25} x_{24} \\ y_{32} &= x_{26} x_{25} \\ y_{34} &= x_{28} x_{26} \end{aligned} \quad (\text{A14})$$

$$\begin{aligned} y_{33} &= R_c x_{25}^2 \\ y_{35} &= h_{c,1,air}(\dot{m}_{air}) A_c(x_{27} - x_{77}) \\ y_{114} &= h_{c,1,liq}(\dot{m}_{liq}) A_c(x_{27} - x_{66}) \end{aligned} \quad (\text{A15})$$

Again,  $s_3$  and  $u_3$  are the switching input and duty cycle for that converter, and the cooling terms are defined in a similar manner as that of the inverter. This same set of equations can be used to define the other converters, with inputs  $u_4/s_4, \dots, u_8/s_8$ .

The VCS is described by two vertices of zero capacitance,  $v_{55}$  and  $v_{56}$  [47]. Electrical power flowing from the converter output voltage

state ( $x_{38}$ ) into the VCS is represented by a resistive ( $R_{VCS}$ ) load:

$$y_{59} = x_{38}^2 / R_{VCS} \quad (\text{A16})$$

An efficiency of  $\eta_{VCS}$  describes the power used for cooling ( $y_{79}$ ) and power dissipated ( $y_{80}$ ):

$$\begin{aligned} y_{79} &= \eta_{VCS} x_{55} \\ y_{80} &= (1 - \eta_{VCS}) x_{55} \end{aligned} \quad (\text{A17})$$

Equation (A18) describes the thermal power removed from the liquid loop's fluid ( $y_{81}$ ) and the heat rejected to the environment ( $y_{91}$ ). Note that these power flows are multiplied by functions relating to the coefficient of performance in Appendix B.

$$y_{81} = y_{91} = x_{56} \quad (\text{A18})$$

The dynamics of the pump within the liquid cooling path are similar to a DC motor, with motor constant  $k_p$ , resistance  $R_p$ , and negligible inductance. Electrical power into the pump is described by the converter's output voltage ( $x_{53}$ ) and the pump's current ( $x_{61}$ ):

$$y_{77} = x_{53} x_{61} \quad (\text{A19})$$

The majority of the electrical power goes into vertex  $v_{62}$ , which has a rotational velocity state ( $x_{62}$ ) and negligible storage capacitance. This power relates to motor constant  $k_p$ :

$$y_{88} = k_p x_{62} x_{61} \quad (\text{A20})$$

The remaining electrical power is dissipated through resistor  $R_p$ :

$$y_{89} = R_p x_{61}^2 \quad (\text{A21})$$

The pump faces friction slowing it down, with friction coefficient  $b_p$ :

$$y_{90} = b_p x_{62}^2 \quad (\text{A22})$$

The pump's velocity state ultimately relates to the mass flow-rate through the liquid cooling loop, which impacts convection coefficients as stated earlier. The cabin and air cooling fans operate identically to the pump, and are represented by the same model.

The liquid cooling loop is described by several vertices ( $v_{63}, \dots, v_i, \dots, v_{69}$ ), each with thermal capacitance  $C_i$ . Heat flowing from vertex to vertex is described using the mass flowrate, the specific heat of the liquid ( $C_{p,liq}$ ), and tail vertex state, and applies to  $e_{92}, \dots, e_{98}$ . Equation (A23) provides an example for edge  $e_{92}$ .

$$y_{92} = \dot{m}_{liq} C_{p,liq} x_{63} \quad (\text{A23})$$

The same is true for the cabin cooling ( $v_{70}, \dots, v_{74}$ ) and air cooling ( $v_{75}, \dots, v_{79}$ ) paths, with specific heat  $C_{p,air}$ . This applies to edges  $e_{99}, e_{100}, e_{102}, \dots, e_{105}, e_{124}$  and  $e_{106}, \dots, e_{110}, e_{123}$ , respectively. Equation (A24) provides an example for edge  $e_{100}$ , with  $\dot{m}_{cab}$  as the mass flowrate of air through the cabin, which relates to cabin fan velocity. Note that the cabin has 50% air recirculation.

$$y_{92} = \dot{m}_{cab} C_{p,air} x_{71} \quad (\text{A24})$$

The cabin ( $v_{73}$ ) has a thermal capacitance of  $C_{cab}$ , with heating coming into it described by:

$$y_{122} = Q_{rad} + Q_{conv} + Q_{gen} \quad (\text{A25})$$

with  $Q_{rad}$  as radiative heat in,  $Q_{conv}$  as convective heat in, and  $Q_{gen}$  as heat generated in the cabin. The power flow  $y_{101}$  describes how the cabin air is cooled by the liquid cooling loop, with  $R_{c,cab}$  as the heat conduction resistance:

$$y_{101} = \frac{1}{R_{c,cab}} (x_{72} - x_{64}) \quad (\text{A26})$$

## Appendix B

This appendix summarizes the terms of the design matrices' entries for case study 2, with Table 6 defining parameter values. Equation (B1) defines a linear relationship between a design variable  $\theta_{c,\sigma}$  and an output variable  $f$ , with  $\mu$  representing the percentage difference the minimum and maximum values of  $f$  have from the nominal value of  $f$ .

$$f(\theta_{c,\sigma}, \mu) = \left(1 - \frac{\mu}{100}\right) + 2 \frac{\mu}{100} \theta_{c,\sigma} \quad (\text{B1})$$

Sizing of the battery is considered analogous to adding fractions of cells in parallel ( $N_{par}$ ), which increases battery parameters such as capacity and mass, with  $\theta_{c,1} \in [0, 1]$  and  $N_{par} = N_{par,nom} \cdot f(\theta_{c,1}, \mu_{N_{par}})$ . As an example, consider the storage capacitance of the first vertex,  $QV_{oc}(x_1)$ , with  $Q$  as the nominal battery capacity,  $V_{oc}$  as the nonlinear open-circuit voltage, and  $x_1$  as the battery SOC. As the number of cells in a battery pack increases, the capacity increases proportionally. To reflect this, the first entry of  $\Psi_c$  is defined as:

$$f_{\psi_{c,1}}(\theta_{c,1}) = N_{par}(\theta_{c,1}) \quad (\text{B2})$$

This same logic applies to the remaining electrical ( $v_2, v_3$ ) and thermal ( $v_4, v_5$ ) storage capacitances of the battery:

$$f_{\psi_{c,2}}(\theta_{c,1}) = \dots = f_{\psi_{c,5}}(\theta_{c,1}) = N_{par}(\theta_{c,1}) \quad (\text{B3})$$

Increasing the size of the battery also affects resistances and surface area  $A_{batt}$  of the battery, impacting the following edges:

$$f_{\psi_3}(\theta_{c,1}) = f_{\psi_5}(\theta_{c,1}) = f_{\psi_7}(\theta_{c,1}) = N_{par}(\theta_{c,1}) \quad (\text{B4})$$

$$f_{\psi_6}(\theta_{c,1}) = \frac{1}{N_{par}(\theta_{c,1})} \quad (\text{B5})$$

$$f_{\psi_8}(\theta_{c,1}) = f_{\psi_{111}} = A_{batt}(\theta_{c,1}) = 2h_{cell}d_{cell}N_{ser} + d_{cell}^2N_{ser}N_{par}(\theta_{c,1}) \quad (\text{B6})$$

In Eq. (B6),  $h_{cell}$  and  $d_{cell}$  are the height and diameter of a cell, respectively, and  $N_{ser}$  is the number of cell modules placed in series.

For sizing motor/inverter #1, a larger  $\theta_{c,2}$  indicates an increase in resistances of the inverter ( $e_{16}, e_{18}, e_{20}$ ), the motor constant  $k_m$  ( $e_{36}$ ), and the motor resistance  $R_m$  ( $e_{37}$ ). This can be used to define the

following design matrix terms:

$$f_{\psi_j}(\theta_{c,2}) = 1/f(\theta_{c,2}, \mu_{e_j}) \quad \text{for } j = 16, 20 \quad (\text{B7})$$

$$f_{\psi_j}(\theta_{c,2}) = f(\theta_{c,2}, \mu_{e_j}) \quad \text{for } j = 18, 36, 37 \quad (\text{B8})$$

Sizing of motor/inverter #2 relates to  $\theta_{c,3}$ , and is defined using the same process.

Sizing of the VCS and its converter is associated with  $\theta_{c,4}$ . Increasing  $\theta_{c,4}$  increases the resistance of the converter ( $e_{58}$ ) and the coefficient of performance ( $e_{81}, e_{91}$ ). The design matrix terms are defined as:

$$f_{\psi_{58}}(\theta_{c,4}) = f(\theta_{c,4}, \mu_{e_{58}}) \quad (\text{B9})$$

$$f_{\psi_{81}}(\theta_{c,4}) = 1 + COP_{nom} \cdot f(\theta_{c,4}, \mu_{e_{81}}) \quad (\text{B10})$$

$$f_{\psi_{91}}(\theta_{c,4}) = COP_{nom} \cdot f(\theta_{c,4}, \mu_{e_{91}}) \quad (\text{B11})$$

where  $COP_{nom}$  is the nominal coefficient of performance value.

The design variables  $\theta_{c,5}$  and  $\theta_{c,6}$  size the air cooling loop's fan and the liquid cooling loop's pump, respectively. The impact to these elements is similar to how the motors are sized:

$$f_{\psi_j}(\theta_{c,5}) = f(\theta_{c,5}, \mu_{e_j}) \quad \text{for } j = 70, 85, 86 \quad (\text{B12})$$

$$f_{\psi_j}(\theta_{c,6}) = f(\theta_{c,6}, \mu_{e_j}) \quad \text{for } j = 76, 88, 89 \quad (\text{B13})$$

The sizing design variables also impact the mass of the vehicle ( $v_{22}$ ), represented by the design matrix term in Eq. (B14). In Eq. (B14),  $m_0$  is the nominal mass of the vehicle,  $m_{cell}$  is the mass of a single battery cell,  $m_{m,nom}$  is the nominal mass of a motor, and  $m_{VCS,nom}$  is the nominal mass of the VCS.

$$\begin{aligned} f_{\psi_{c,22}}(\theta_{c,1}, \theta_{c,2}, \theta_{c,3}, \theta_{c,4}) &= m_0 + (N_{ser}m_{cell}N_{par}(\theta_{c,1})) \\ &+ m_{m,nom}f(\theta_{c,2}, \mu_{m,v_{22}}) \\ &+ m_{m,nom}f(\theta_{c,3}, \mu_{m,v_{22}}) \\ &+ m_{VCS,nom}f(\theta_{c,4}, \mu_{VCS,v_{22}}) \end{aligned} \quad (\text{B14})$$

The first discrete design variable,  $z_1$ , determines whether motor/inverter #2 is connected to the system, with  $z_1 = 1$  indicating it is included. Figure 7 shows that motor #2 connects to the rest of the powertrain through four edges ( $e_{11}, e_{39}, e_{41}, e_{116}$ ), and inverter #2 connects to the rest of the powertrain through two edges ( $e_{29}, e_{113}$ ). The choice of cooling for motor #2 also impacts edges  $e_{41}$  (air cooling,  $z_5 = 1$ ) and  $e_{116}$  (liquid cooling,  $z_5 = 0$ ). Using this knowledge, the design matrix terms relating to motor #2's topology choices are defined as:

$$f_{\phi_{11}} = f_{\phi_{39}} = z_1 \quad (\text{B15})$$

$$f_{\phi_{41}} = z_1 z_5 \quad (\text{B16})$$

$$f_{\phi_{116}} = z_1(1 - z_5) \quad (\text{B17})$$

The choice of cooling for inverter #2 also impacts edges  $e_{29}$  (air cooling,  $z_3 = 1$ ) and  $e_{113}$  (liquid cooling,  $z_3 = 0$ ), yielding:

$$f_{\phi_{29}} = z_1 z_3 \quad (\text{B18})$$

$$f_{\phi_{113}} = z_1(1 - z_3) \quad (\text{B19})$$

Regarding discrete variable design choices, the battery cooling design variable impacts edges  $e_8$  (air cooling,  $z_2 = 1$ ) and  $e_{111}$  (liquid cooling,  $z_2 = 0$ ):

$$f_{\phi_8} = z_2 \quad (\text{B20})$$

$$f_{\phi_{111}} = 1 - z_2 \quad (\text{B21})$$

Parameter	Value
$N_{par,nom}$	85
$h_{cell}$	$65.4 \times 10^{-3}$ m
$d_{cell}$	$26.5 \times 10^{-3}$ m
$m_0$	1300 kg
$m_{cell}$	0.076 kg
$m_{m,nom}$	40 kg
$m_{VCS,nom}$	20 kg
$\mu_{N_{par}}$	0.47
$\mu_j$ ( $j = 16, 20, 24, 28$ )	-0.05
$\mu_j$ ( $j = 18, 26, 37, 40$ )	0.05
$\mu_j$ ( $j = 36, 40$ )	0.25
$\mu_j$ ( $j = 58, 70, 76, 81, 85, 86, 88, 89, 91$ )	0.1
$\mu_{m,v_{22}}$	0.1
$\mu_{VCS,v_{22}}$	0.1
$\mu_{z,m_m}$	0.015
$\mu_{batt,cool}$	-0.01
$\mu_{powElec,cool}$	-0.01
$\mu_{m_1,cool}, \mu_{2,cool}$	-0.01

Motor #1's cooling design variable impacts edges  $e_{38}$  (air cooling,  $z_4 = 1$ ) and  $e_{115}$  (liquid cooling,  $z_4 = 0$ ):

$$f_{\phi_{38}} = z_4 \quad (B22)$$

$$f_{\phi_{115}} = 1 - z_4 \quad (B23)$$

The remaining terms relating to the power electronic cooling option are:

$$f_{\phi_{21}} = f_{\phi_{35}} = f_{\phi_{54}} = f_{\phi_{60}} = f_{\phi_{66}} = f_{\phi_{72}} = f_{\phi_{78}} = z_3 \quad (B24)$$

$$f_{\phi_{112}} = f_{\phi_{114}} = f_{\phi_{117}} = \dots = f_{\phi_{121}} = 1 - z_3 \quad (B25)$$

The mass of the vehicle is also scaled by each of the discrete choices (e.g., excluding motor #2 reduces mass, liquid cooling is heavier than air cooling):

$$\begin{aligned} f_{\phi_{c,22}} = & (1.08)(f(z_1, \mu_{z,m_m}) \cdot f(z_2, \mu_{batt,cool}) \\ & \cdot f(z_3, \mu_{powElec,cool}) \cdot f(z_4, \mu_{m_1,cool}) \cdot f(z_5, \mu_{m_2,cool})) \end{aligned} \quad (B26)$$

## References

- [1] Hutcheson, R. S., McAdams, D. A., Stone, R. B., and Turner, I. Y., 2007, "Function-Based Systems Engineering (Fuse)," 16th International Conference on Engineering Design, Paris, France, Aug. 28–31, pp. 1–12.
- [2] Eusuff, M. M., and Lansey, K. E., 2003, "Optimization of Water Distribution Network Design Using the Shuffled Frog Leaping Algorithm," *J. Water Resour. Plan. Manag.*, **129**(3), pp. 210–225.
- [3] Tippett, M. J., and Bao, J., 2013, "Distributed Model Predictive Control Based on Dissipativity," *AIChE J.*, **59**(3), pp. 787–804.
- [4] Ahn, J., Chung, D. H., and Cho, S., 2018, "Energy Cost Analysis of an Intelligent Building Network Adopting Heat Trading Concept in a District Heating Model," *Energy*, **151**, pp. 11–25.
- [5] Tong, S., Cheng, Z., Cong, F., Tong, Z., and Zhang, Y., 2018, "Developing a Grid-Connected Power Optimization Strategy for the Integration of Wind Power with Low-Temperature Adiabatic Compressed Air Energy Storage," *Renew. Energy*, **125**, pp. 73–86.
- [6] Wheeler, P., and Bozhko, S., 2014, "The More Electric Aircraft: Technology and Challenges," *IEEE Electrif. Mag.*, **44**(0), pp. 6–12.
- [7] Hoffman, A. C., Hansen, I. G., Beach, R. F., Plencner, R. M., Dengler, R. P., Jefferies, K. S., and Frye, R. J., 1985, "Advanced Secondary Power System for Transport Aircraft," NASA Tech. Pap.
- [8] Allison, J. T., and Herber, D. R., 2014, "Multidisciplinary Design Optimization of Dynamic Engineering Systems," *AIAA J.*, **52**(4), pp. 691–710.
- [9] Maraniello, S., and Palacios, R., 2016, "Optimal Vibration Control and Co-Design of Very Flexible Actuated Structures," *J. Sound Vib.*, **377**, pp. 1–21.
- [10] Vasudev, K. L., Sharma, R., and Bhattacharyya, S. K., 2019, "Multi-Objective Shape Optimization of Submarine Hull Using Genetic Algorithm Integrated with Computational Fluid Dynamics," *J. Eng. Marit. Environ.*, **233**(1), pp. 55–66.
- [11] Brejle, B. J., and Martins, J. R., 2018, "Development of a Conceptual Design Model for Aircraft Electric Propulsion with Efficient Gradients," 2018 AIAA/IEEE Electric Aircraft Technologies Symposium, Cincinnati, OH, July 9–11, pp. 1–25, AIAA.
- [12] Duffy, M. J., Wakayama, S. R., Hupp, R., Lacy, R., and Stauffer, M., 2017, "A Study in Reducing the Cost of Vertical Flight with Electric Propulsion," 17th AIAA Aviation Technology, Integration, and Operations Conference, Denver, CO, June 5–9, pp. 1–24.
- [13] Messac, A., 1996, "Physical Programming: Effective Optimization for Computational Design," *AIAA J.*, **34**(1), pp. 149–158.
- [14] Strogatz, S. H., Abrams, D. M., McRobie, A., Eckhardt, B., and Ott, E., 2005, "Theoretical Mechanics: Crowd Synchrony on the Millennium Bridge," *Nature*, **438**(7064), pp. 43–44.
- [15] Degond, P., Liu, J. G., and Ringhofer, C., 2014, "Evolution of Wealth in a Non-Conservative Economy Driven by Local Nash Equilibria," *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, **372**(2028), pp. 1–15.
- [16] Bhattacharya, K., Mukherjee, G., Saramäki, J., Kaski, K., and Manna, S. S., 2008, "The International Trade Network: Weighted Network Analysis and Modelling," *J. Stat. Mech. Theory Exp.*, **2008**(2), pp. 1–5.
- [17] Vasilevski, M., Pecheux, F., Beilleau, N., Aboushady, H., and Einwich, K., 2008, "Modeling Refining Heterogeneous Systems with SystemC-AMS: Application to WSN," 2008 Design, Automation and Test in Europe, Munich, Germany, Mar. 10–14, pp. 134–139.
- [18] McKenna, M. T., Weis, J. A., Barnes, S. L., Tyson, D. R., Miga, M. I., Quaranta, V., and Yankelev, T. E., 2017, "A Predictive Mathematical Modeling Approach for the Study of Doxorubicin Treatment in Triple Negative Breast Cancer," *Nat. Sci. Reports*, **7**(June), pp. 1–14.
- [19] Ghanaatpishe, M., Bayiz, Y. E., Cheng, B., and Fathy, H. K., 2018, "Rotary Versus Flapping Flight: An Application Study for Optimal Periodic Control Theory," ASME 2018 Dynamic Systems and Control Conference, Atlanta, GA, Sept. 30–Oct. 3, pp. 1–11.
- [20] Kumar, M., Kar, I. N., and Ray, A., 2008, "State Space Based Modeling and Performance Evaluation of an Air-Conditioning System," *HVAC R Res.*, **14**(5), pp. 797–816.
- [21] Xu, M., Zhang, Z., Wang, X., Jia, L., and Yang, L., 2015, "A Pseudo Three-Dimensional Electrochemical-Thermal Model of a Prismatic LiFePO4 Battery During Discharge Process," *Energy*, **80**, pp. 303–317.
- [22] Bizeray, A. M., Zhao, S., Duncan, S. R., and Howey, D. A., 2015, "Lithium-Ion Battery Thermal-Electrochemical Model-Based State Estimation Using Orthogonal Collocation and a Modified Extended Kalman Filter," *J. Power Sources*, **296**, pp. 400–412.
- [23] Lindborg, H., Eide, V., Unger, S., Henriksen, S. T., and Jakobsen, H. A., 2004, "Parallelization and Performance Optimization of a Dynamic PDE Fixed Bed Reactor Model for Practical Applications," *Comput. Chem. Eng.*, **28**(9), pp. 1585–1597.
- [24] Garcia, J., Dauphin-Tanguy, G., and Rombaut, C., 1996, "Modeling and Simulating the Dynamic Electrothermal Behavior of Power Electronic Circuits Using Bond Graphs," PESC Record. 27th Annual IEEE Power Electronics Specialists Conference, Baveno, Italy, June 23–27, pp. 1641–1647.
- [25] Rizzoni, G., Guzzella, L., and Baumann, B. M., 1999, "Unified Modeling of Hybrid Electric Vehicle Drivetrains," *IEEE/ASME Trans. Mechatronics*, **4**(3), pp. 246–257.
- [26] Bayrak, A. E., Ren, Y., and Papalambros, P. Y., 2016, "Topology Generation for Hybrid Electric Vehicle Architecture Design," *ASME J. Mech. Des.*, **138**(8), p. 081401.
- [27] Bayrak, A. E., Kang, N., and Papalambros, P. Y., 2016, "Decomposition-Based Design Optimization of Hybrid Electric Powertrain Architectures: Simultaneous Configuration and Sizing Design," *ASME J. Mech. Des.*, **138**(7), p. 071405.
- [28] Nagel, B., Böhnke, D., Göllnick, V., Schmolgruber, P., Rizzi, A., La Rocca, G., and Alonso, J. J., 2012, "Communication in Aircraft Design: Can We Establish a Common Language?," 28th International Congress on Aeronautical Science, Brisbane, Australia, Sept. 23–28, pp. 1–13.
- [29] Kun Deng, B., Mehta, P., and Meyn, P. G., and P. S., 2014, "Building Thermal Model Reduction via Aggregation of States," American Control Conference, Baltimore, MD, June 30–July 2, pp. 5118–5123.
- [30] Koeln, J. P., Williams, M. A., Pangborn, H. C., and Alleyne, A. G., 2016, "Experimental Validation of Graph-Based Modeling for Thermal Fluid Power Flow Systems," ASME Dynamic Systems and Control Conference, Minneapolis, Oct. 12–14, pp. 1–10.
- [31] Pangborn, H. C., Williams, M. A., Koeln, J. P., and Alleyne, A. G., 2017, "Graph-Based Hierarchical Control of Thermal-Fluid Power Flow Systems," American Control Conference, Seattle, WA, May 24–26, pp. 2099–2105.
- [32] Tannous, P. J., and Alleyne, A. G., 2019, "Fault Detection and Isolation for Complex Thermal Management Systems," *J. Dyn. Syst. Meas. Control*, **141**(6), p. 061008.
- [33] Docimo, D. J., and Alleyne, A. G., 2018, "Electro-Thermal Graph-Based Modeling for Hierarchical Control with Application to an Electric Vehicle," IEEE Conference on Control Technology and Applications, Copenhagen, Denmark, Aug. 21–24, pp. 812–819.
- [34] Aksland, C. T., Bixel, T. W., Raymond, L. C., Rottmayer, M. A., and Alleyne, A. G., 2019, "Graph-Based Electro-Mechanical Modeling of a Hybrid Unmanned Aerial Vehicle for Real-Time Applications," American Control Conference, Philadelphia, PA, July 10–12, pp. 4253–4259.
- [35] Peddada, S. R. T., Herber, D. R., Pangborn, H. C., Alleyne, A. G., and Allison, J. T., 2019, "Optimal Flow Control and Single Split Architecture Exploration for Fluid-Based Thermal Management," *ASME J. Mech. Des.*, **141**(August), p. 083401.
- [36] Martins, J. R. R. A., and Lambe, A. B., 2013, "Multidisciplinary Design Optimization: A Survey of Architectures," *AIAA J.*, **51**(9), pp. 2049–2070.
- [37] Messac, A., 2015, *Optimization in Practice with MATLAB for Engineering Students and Professionals*, Cambridge University Press, Cambridge.
- [38] Stolpe, M., and Svanberg, K., 2001, "An Alternative Interpolation Scheme for Minimum Compliance Topology Optimization," *Struct. Multidiscip. Optim.*, **22**(2), pp. 116–124.
- [39] Wheeler, P., and Bozhko, S., 2014, "The More Electric Aircraft: Technology and Challenges," *IEEE Electrif. Mag.*, **2**(4), pp. 6–12.
- [40] Offer, G. J., 2015, "Automated Vehicles and Electrification of Transport," *Energy Environ. Sci.*, **8**(1), pp. 26–30.
- [41] Allam, A., Onori, S., Marelli, S., and Taborelli, C., 2015, "Battery Health Management System for Automotive Applications: A Retroactivity-Based Aging Propagation Study," American Control Conference, Chicago, IL, July 1–3, pp. 703–715.
- [42] Bennion, K., 2017, *Electric Motor Thermal Management Research: Annual Report*.
- [43] Bailey, C., 2008, "Thermal Management Technologies for Electronic Packaging: Current Capabilities and Future Challenges for Modeling Tools," 10th Electronics Packaging Technology Conference, Singapore, Dec. 9–12, pp. 527–532.
- [44] Lin, X., Perez, H. E., Siegel, J. B., Stefanopoulou, A. G., Ding, Y., and Castanier, M. P., 2013, "Parameterization and Observability Analysis of Scalable Battery Clusters for Onboard Thermal Management," *Oil Gas Sci. Technol. IFP Energies Nouv.*, **68**(1), pp. 165–178.
- [45] Erickson, R. W., and Maksimovic, D., 2001, *Fundamentals of Power Electronics*, Kluwer Academic Publishers, Dordrecht.
- [46] Vural, B., Dusmez, S., and Uzunoglu, M., 2010, "An Improved Power Conditioning Unit for Energy Storage Systems in Hybrid Electrical Vehicles," 5th International Ege Energy Symposium and Exhibition, Denizli, Turkey, June 27–30.
- [47] Docimo, D. J., Pangborn, H. C., and Alleyne, A. G., 2018, "Hierarchical Control for Electro-Thermal Power Management of an Electric Vehicle Powertrain," ASME Dynamic Systems and Control Conference, Atlanta, GA, Sept. 30–Oct. 3, pp. 1–10.