Proceedings of the ASME 2010 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2010
August 15-18, 2010, Montreal, Quebec, Canada

DETC2010-28788

CONSTRAINT MANAGEMENT OF REDUCED REPRESENTATION VARIABLES IN DECOMPOSITION-BASED DESIGN OPTIMIZATION

Michael J. Alexander*
Optimal Design Laboratory
Mechanical Engineering
University of Michigan
Ann Arbor, Michigan 48104
Email: malexanz@umich.edu

James T. Allison
The Mathworks, Inc.
Natick, Massachusetts 01760
Email: james.allison@mathworks.com

Panos Y. Papalambros
Optimal Design Laboratory
Mechanical Engineering
University of Michigan
Ann Arbor, Michigan 48104
Email: pyp@umich.edu

David J. Gorsich
U.S. Army TARDEC
Warren, MI 48397
Email: david.gorsich@us.army.mil

ABSTRACT

In decomposition-based design optimization strategies, such as Analytical Target Cascading (ATC), it is sometimes necessary to use reduced dimensionality representations to approximate functions of large dimensionality whose values need to be exchanged among subproblems. The reduced representation variables may not be physically meaningful, and it can become challenging to constrain them properly and define the model validity region. For example, in coordination strategies like ATC, representing vector-valued coupling variables with improperly constrained reduced representation variables can lead to poor performance or convergence failure. This paper examines two approaches for constraining effectively the model validity region of reduced representation variables based on proper orthogonal decomposition: a penalty value-based heuristic and a support vector domain description. An ATC application on electric vehicle design helps to illustrate the concepts discussed.

KEYWORDS

Reduced representation, decomposition, design optimization, analytical target cascading, proper orthogonal decomposition, support vector machines.

1 INTRODUCTION

Complex design problems are often addressed through decomposition. Consider, for example, an electric vehicle (EV) powertrain, consisting of a battery, electric traction motors and belt drives. In a design optimization formulation, we can partition the design problem into a system-level problem to design the battery, belt-drive ratios, and motors for fuel economy, performance, and packaging; and a subsystem-level problem to design the motors so that their performance curves match those desired by the system-level problem. Therefore, partitioning a system requires also coordination of the partitioned problems to provide the overall system solution. Partitioning and coordination are the two main elements of a decomposition-based solution strategy.

One such strategy is Analytical Target Cascading (ATC) [1, 2]. In ATC, coupling quantities exchanged between subproblems are treated as decision variables. As the number of coupling variables becomes larger, ATC convergence becomes more difficult to achieve. When the coupling variables are vector-valued, ATC may converge slowly or not at all due to the large number of variables resulting from discretization [3-5]. In the EV problem, motor performance curves are represented as discretized function outputs to high-fidelity, "blackbox" simulations, in the form:

* Address all correspondence to this author.

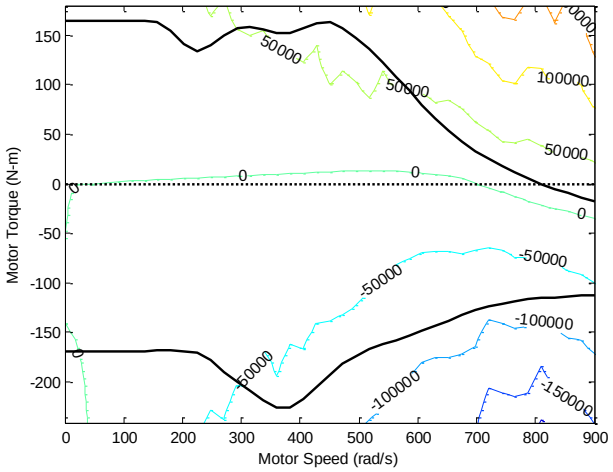

Figure 1. POD-APPROXIMATED MOTOR MAP AT FAILED DESIGN POINT [5]

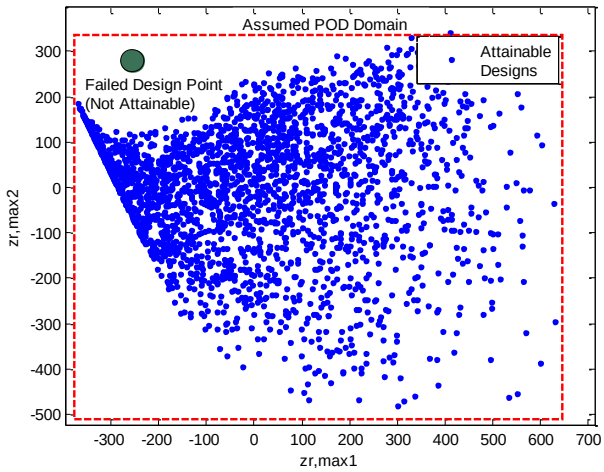

Figure 2. POD MODEL VALIDITY REGION FOR TWO COMPONENTS [5]

$$z = f(y) \approx F([z_1, z_2, \cdots, z_q]^T, [y_1, y_2, \cdots, y_q]^T, y) \qquad (1)$$

Here, y is the independent variable, z is the dependent variable, q is the number of discretized points and the dimensionality of the vector-valued coupling variable (VVCV), and F is an interpolation function, such as a lookup table. Note that Eq. (1) implies that $z_i = f(y_i)$, where f is the blackbox simulation. To use ATC effectively, one may choose a reduced representation method [4, 5], such as proper orthogonal decomposition (POD), to construct low-dimension VVCV approximations that improve optimization performance while maintaining accuracy [3-5]. However, reduced representation variables may not be physically meaningful, leading to difficulties in constraining the design space appropriately. This, in turn, can inhibit ATC performance as the subsystem optimization algorithm may select values incompatible with the underlying analysis models [5].

This paper examines two approaches that can be used to constrain the model validity region for reduced representation variables, based on a penalty value heuristic and support vector domain description (SVDD), respectively. Section 2 provides a brief background on reduced representations in related studies; Section 3 describes POD and its implementation for motor performance curves; Section 4 gives a brief review of ATC and describes the problem formulation associated with the EV powertrain; Section 5 discusses the methods for managing the POD model validity region; Section 6 highlights the results from implementing one of the constraint management techniques in ATC; and Section 7 offers some conclusions.

2 BACKGROUND

A previous study broadly defined reduced representations as methods to decrease the dimension of VVCVs while maintaining a sufficient level of accuracy [5]. These methods include Fourier coefficients [6], weights of various basis functions [7, 8], and low-dimension metamodel inputs [9]. In general, low-dimension metamodels violate the necessary condition of additive-separability in ATC because they typically use physically meaningful design variables that may also appear elsewhere in the optimization structure [4, 5]. The violation of this condition would imply that implementing ATC as an optimization strategy is not necessary and that a single all-in-one formulation would suffice. Methods using coefficients or weights as reduced representation variables work better because they use physically meaningless design variables that are unlikely to be used elsewhere. Among these latter methods, POD is attractive as it generates a functional form without prior user assumptions, makes limited assumptions regarding the number of fitting parameters, and uses a relatively small number of fitting parameters for the VVCV approximations based on data samples [5].

A critical issue in implementing POD in this context is that the POD coefficients serving as reduced representation variables may not be well constrained. In an earlier study [5], using simple bound constraints led to failure of model computations and of the overall process, because the model validity region was actually nonlinear. Simulation failure was a direct result of poorly approximated VVCVs, which in turn was a result of the optimizer selecting POD coefficients outside their model validity region. Figures 1 and 2 illustrate this problem from the previous study. The first figure shows the VVCV approximations of the electric motor performance curves (where the bold lines represent the maximum/minimum torque curves and the numbered lines are power loss isocontours), and the second figure shows the infeasible design point relative to the POD model validity region (where the small dots represent sample designs within the POD model validity region).

Explicit constraints on reduced representation variables with no physical meaning are very difficult to concoct. A penalty value-based heuristic that assigned arbitrarily large objective and constraint function values when the simulation failed was successful. A more formal approach along this direction would be desirable.

This paper reexamines the penalty value-based heuristic and compares it to another approach, SVDD, in an effort to identify a more appropriate method for constraining the model validity region of reduced representation variables. SVDD may offer some advantages as it can represent data set boundaries that are nonlinear, non-convex, and even disconnected, without adding much complexity or computational burden.

3 PROPER ORTHOGONAL DECOMPOSITION

POD, also known as Karhunen-Loeve expansion [10, 11] and principal component analysis [12], has been used to simplify the analysis, design, and optimization of dynamic systems. Specifically, it reduces the state-space representation of dynamic systems using the following transformation [13]:

$$\mathbf{z}(t) \approx \mathbf{\Phi}_p \mathbf{z}_r(t) + \bar{\mathbf{z}}(t) . \tag{2}$$

Here, $\mathbf{z}(t)$ is the original state vector of dimension q, $\mathbf{z}_r(t)$ is the reduced state vector of dimension $p << q$, and $\mathbf{\Phi}_p$ is a matrix of the p most energetic, orthogonal basis functions $\boldsymbol{\varphi}$ used to construct the approximation of the original state vector. The final term $\bar{\mathbf{z}}(t)$ is known as the sample mean vector of dimension q and is used to center the data for the approximation. Without any loss of generality, the VVCVs in this study can be thought of as state vectors, and therefore the transformation in Eq. (2) can be modified as

$$\mathbf{z} \approx \mathbf{\Phi}_p \mathbf{z}_r + \bar{\mathbf{z}} , \tag{3}$$

where $\mathbf{z}$ is the original VVCV of dimension q, $\mathbf{z}_r$ is the reduced representation of $p << q$, and $\mathbf{\Phi}_p$ and $\bar{\mathbf{z}}$ have the same meaning as in the state vector context but applied to VVCVs. The matrix $\mathbf{\Phi}$ containing the full set of orthogonal basis functions is constructed with m samples $\mathbf{z}_i = [z_1, z_2, \ldots, z_q]^T$ using either the direct method or the method of snapshots [14].

The direct method is more efficient when $q \leq m$ [15] and begins by forming the covariance matrix $\mathbf{R}$:

$$\mathbf{R} = \frac{(\mathbf{Z} - \bar{\mathbf{Z}})(\mathbf{Z} - \bar{\mathbf{Z}})^T}{m-1} . \tag{4}$$

Here, $\mathbf{Z}$ is a (q x m) matrix containing all the samples of the original VVCV and $\bar{\mathbf{Z}}$ is a (q x m) matrix of the sample mean vector repeated m times. Next, $\mathbf{\Phi}$ is determined through a (q x q) eigenvalue problem associated with the covariance matrix,

$$\mathbf{R\Phi} = \mathbf{\Phi\Lambda} , \tag{5}$$

where $\mathbf{\Lambda}$ represents the diagonal matrix of eigenvalues. It is assumed that the orthogonal basis functions in $\mathbf{\Phi}$ are arranged based on the magnitude of their associated eigenvalues:

$$\mathbf{\Phi} = \left[\boldsymbol{\varphi}_1 \quad \boldsymbol{\varphi}_2 \quad \cdots \quad \boldsymbol{\varphi}_q\right]^T, \lambda_1 > \lambda_2 > \cdots > \lambda_q . \tag{6}$$

Finally, the number of orthogonal basis functions in $\mathbf{\Phi}$ is truncated to form $\mathbf{\Phi}_p$ based on the cumulative percentage variation (CPV), which is a measure of the relative importance of each orthogonal basis function in $\mathbf{\Phi}$ [16]:

$$\left(\sum_{i=1}^{p} \lambda_i \Big/ \sum_{i=1}^{q} \lambda_i \right) \times 100 \geq CPV_{goal} \tag{7}$$

In the above, CPV_{goal} is set arbitrarily based on the desired amount of information to be captured, which is usually 99.99% [17].

The method of snapshots [14] is more efficient when $q > m$ [15, 18] and begins by forming the correlation matrix $\mathbf{R}$:

$$\mathbf{R} = \frac{(\mathbf{Z} - \bar{\mathbf{Z}})^T (\mathbf{Z} - \bar{\mathbf{Z}})}{m} . \tag{8}$$

The next step is to solve the (m x m) eigenvalue problem associated with the correlation matrix,

$$\mathbf{RV} = \mathbf{V\Lambda} , \tag{9}$$

where $\mathbf{V}$ represents the matrix of eigenvectors. The orthogonal basis functions are then determined from

$$\mathbf{\Phi} = \mathbf{ZV}_n, \; v_{n,ij} = \left(1 \Big/ \sqrt{m\lambda_{ii}}\right) v_{ij} \tag{10}$$

where $\mathbf{\Phi}$ is of dimension (q x m). This captures the essence of the method and states that each orthogonal basis function is a linear combination of the m sample vectors [14]. Finally, $\mathbf{\Phi}_p$ is determined according to the same procedures outlined in Eqs. (6)-(7) with q replaced by m.

For the current study, three POD representations were designed to approximate VVCVs associated with maximum and minimum motor torque curves and power loss maps:

$$\begin{aligned} \mathbf{z}_{\max} &\approx \mathbf{\Phi}_{\mathrm{p,max}} \mathbf{z}_{\mathrm{r,max}} + \bar{\mathbf{z}}_{\max} \\ \mathbf{z}_{\min} &\approx \mathbf{\Phi}_{\mathrm{p,min}} \mathbf{z}_{\mathrm{r,min}} + \bar{\mathbf{z}}_{\min} \\ \mathbf{z}_{\mathrm{pLoss}} &\approx \mathbf{\Phi}_{\mathrm{p,pLoss}} \mathbf{z}_{\mathrm{r,pLoss}} + \bar{\mathbf{z}}_{\mathrm{pLoss}} \end{aligned} \tag{11}$$

Each VVCV associated with the torque curves contained $q_{max} = q_{min} = 41$ values, whereas the VVCV associated with the power loss map contained $q_{pLoss} = 3321$ values. The sample vectors in $\mathbf{Z}_{max}$, $\mathbf{Z}_{min}$, and $\mathbf{Z}_{pLoss}$ that were used to construct $\mathbf{\Phi}_{p,max}$, $\mathbf{\Phi}_{p,min}$,

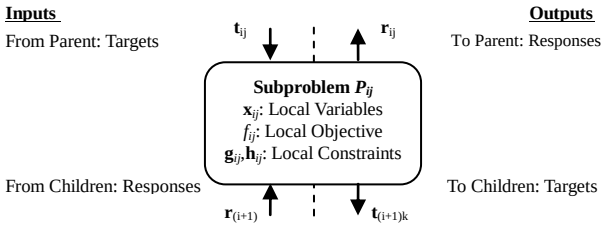

Figure 3. ATC INFORMATION FLOW [19]

and $\mathbf{\Phi}_{p,pLoss,}$ respectively, were generated through a Latin hypercube sample (LHS) design of experiments of m = 2500 samples each. Because $q_{max} = q_{min} << m$, the direct method was used to develop $\mathbf{\Phi}_{p,max}$ and $\mathbf{\Phi}_{p,min}$, whereas the method of snapshots was used to develop $\mathbf{\Phi}_{p,pLoss}$ since $q_{pLoss} >> m$. Performing POD, it was found that the dimensionality of $\mathbf{z}_{r,max}$, $\mathbf{z}_{r,min}$, and $\mathbf{z}_{r,pLoss}$ was reduced to p_{max} = 14, p_{min} = 13, and p_{pLoss} = 96. Consequently, the combined dimensionality of the VVCVs was reduced from $Q = q_{max} + q_{min} + q_{pLoss}$ = 3403 to $Q = p_{max} + p_{min} + p_{pLoss}$ = 123.

4 ANALYTICAL TARGET CASCADING

ATC [1, 2] is a decomposition-based optimization strategy applied to large-scale systems that uses a hierarchical structure to enable design targets determined at upper levels to be cascaded down to lower levels. This technique then attempts to minimize deviations between design targets and subsystem responses to achieve an optimal and consistent system design solution.

4.1 Review and General Problem Formulation

The system is first decomposed into subproblems hierarchically. In this configuration, the top level is the system level and the lower levels are the subsystem levels. A subproblem linked above (below) any given element of interest is known as a parent (child). The general ATC subproblem P_{ij} for the *ith* level and the *jth* element is defined as [19]:

$$\begin{aligned} &\min_{\bar{\mathbf{x}}_{ij}} f_{ij}(\bar{\mathbf{x}}_{ij}) + \pi(\mathbf{c}(\bar{\mathbf{x}}_{11},\cdots,\bar{\mathbf{x}}_{NM})) \\ &\text{subject to } \mathbf{g}_{ij}(\bar{\mathbf{x}}_{ij}) \le \mathbf{0},\ \mathbf{h}_{ij}(\bar{\mathbf{x}}_{ij}) = \mathbf{0} \\ &\text{where } \quad \bar{\mathbf{x}}_{ij} = [\mathbf{x}_{ij}, \mathbf{r}_{ij}, \mathbf{t}_{(i+1)k_1},\cdots,\mathbf{t}_{(i+1)k_{c_{ij}}}],\ \mathbf{c} = [\mathbf{c}_{22},\cdots,\mathbf{c}_{NM}] \end{aligned} \tag{12}$$

In the above, $\mathbf{x}_{ij}$ is the vector of local design variables, $\mathbf{t}_{ij}$ is the vector of target linking variables passed from the element's parent at level (i - 1), $\mathbf{r}_{ij}$ is the vector of response linking variables passed to the element's parent at level (i - 1), $\mathbf{c}_{ij} = \mathbf{t}_{ij} - \mathbf{r}_{ij}$ is the vector of consistency constraints between target and response linking variables, f_{ij} is the local objective function, π is the penalty function, $\mathbf{g}_{ij}$ is the vector of inequality constraints, $\mathbf{h}_{ij}$ is the vector of equality constraints, N is the number of levels, and M is the total number of elements. In general, the linking variables in $\mathbf{t}_{ij}$ and $\mathbf{r}_{ij}$ consist of both coupling and shared variables, but in this study, only coupling variables are considered. Ideally, the consistency constraints on these variables should evaluate to zero for an exact system solution; however, this may not happen due to non-differentiability at the solution and unknown minimal parameter values [19]. Therefore, the consistency constraints are relaxed and inserted into some penalty function $\pi(\mathbf{c})$ that is also minimized in Eq. (12). The coordination strategy used here requires $\|\mathbf{c}^{(K)} - \mathbf{c}^{(K-1)}\|_\infty$ to be within some small tolerance before the algorithm is terminated, where K denotes the iteration number.

For this study, an augmented-Lagrangian (AL) penalty function was chosen, which resulted in the following general ATC-AL subproblem formulation for the *ith* level and *jth* element [19]:

$$\begin{aligned} &\min_{\bar{\mathbf{x}}_{ij}} f_{ij}(\bar{\mathbf{x}}_{ij}) - \mathbf{v}_{ij}^T\mathbf{r}_{ij} + \sum_{k\in C_{ij}} \mathbf{v}_{(i+1)k}^T\mathbf{t}_{(i+1)k} + \left\|\mathbf{w}_{ij} \circ (\mathbf{t}_{ij} - \mathbf{r}_{ij})\right\|_2^2 \\ &\qquad + \sum_{k\in C_{ij}} \left\|\mathbf{w}_{ij} \circ (\mathbf{t}_{(i+1)k} - \mathbf{r}_{(i+1)k})\right\|_2^2 \\ &\text{subject to } \mathbf{g}_{ij}(\bar{\mathbf{x}}_{ij}) \le \mathbf{0},\ \mathbf{h}_{ij}(\bar{\mathbf{x}}_{ij}) = \mathbf{0} \\ &\text{where } \quad \bar{\mathbf{x}}_{ij} = [\mathbf{x}_{ij}, \mathbf{r}_{ij}, \mathbf{t}_{(i+1)k_1},\cdots,\mathbf{t}_{(i+1)k_{c_{ij}}}] \end{aligned} \tag{13}$$

Here, the vectors $\mathbf{v}$ and $\mathbf{w}$ are weights corresponding to the linear and quadratic terms in the AL penalty function, respectively. These decomposed problems are solved in an inner loop strategy where the weights remain constant. After inner loop convergence, termination conditions are evaluated in the outer loop, and if another inner loop execution is required the penalty weights are updated according to the following scheme:

$$\begin{aligned} \mathbf{v}^{(K+1)} &= \mathbf{v}^{(K)} + 2\mathbf{w}^{(K)} \circ \mathbf{w}^{(K)} \circ \mathbf{c}^{(K)} \\ \mathbf{w}^{(K+1)} &= \beta\mathbf{w}^{(K)}, \text{ where } \beta \ge 1 \end{aligned} \tag{14}$$

The information flow for the general ATC-AL subproblem is illustrated in Fig. 3.

4.2 Problem-Specific Formulation

The formulation for the EV powertrain system [5, 20] consists of a two-level hierarchical decomposition based on Eq. (13). The vehicle system and the motor subsystem described in Section 1 are the top-level and bottom-level subproblems, respectively. In this study, the vehicle system objective is to maximize gasoline-equivalent fuel economy while minimizing the AL penalty function, whereas the motor subsystem objective is to minimize the AL penalty function exclusively. Although both subproblems are subject to decision variable bound constraints, only the top-level contains additional constraints based on packaging, performance, motor feasibility, power consumption, and battery capacity.

Applying Eq. (13) directly, the vehicle subproblem P_{11}, excluding decision variable bound constraints, is:

$$\begin{aligned}
&\min_{\bar{\mathbf{x}}_{11}} -mpg_e(\bar{\mathbf{x}}_{11}) + \mathbf{v}_{22}^T(\mathbf{t}_{22} - \mathbf{r}_{22}) + \left\|\mathbf{w}_{22} \circ (\mathbf{t}_{22} - \mathbf{r}_{22})\right\|_2^2 \\
&\text{s. t. } g_1(\bar{\mathbf{x}}_{11}) = b_w(\bar{\mathbf{x}}_{11}) - b_{w\max} \le 0 \\
&\quad g_2(\bar{\mathbf{x}}_{11}) = b_\ell(\bar{\mathbf{x}}_{11}) + x_b - b_{\ell\max} \le 0 \\
&\quad g_3(\bar{\mathbf{x}}_{11}) = t_{60}(\bar{\mathbf{x}}_{11}) - t_{60\max} \le 0 \\
&\quad g_4(\bar{\mathbf{x}}_{11}) = \tau_V(\bar{\mathbf{x}}_{11}) \le 0 \\
&\quad g_5(\bar{\mathbf{x}}_{11}) = \omega_V(\bar{\mathbf{x}}_{11}) \le 0 \\
&\quad g_6(\bar{\mathbf{x}}_{11}) = R_{\min} - R(\bar{\mathbf{x}}_{11}) \le 0 \\
&\quad g_7(\bar{\mathbf{x}}_{11}) = P_V(\bar{\mathbf{x}}_{11}) \le 0 \\
&\quad g_8(\bar{\mathbf{x}}_{11}) = C_b(\bar{\mathbf{x}}_{11}) - C_{b\max}(\bar{\mathbf{x}}_{11}) \le 0 \\
&\text{where} \quad \bar{\mathbf{x}}_{11} = [B_I, B_W, B_L, x_b, p_r, \mathbf{z}_{comb,r}^T, J_r^T, \omega_{\max}^T], \\
&\qquad \mathbf{t}_{22} = [\mathbf{z}_{comb}^T, J_r^T, \omega_{\max}^T],\ \mathbf{z}_{comb}^T = f(\mathbf{z}_{comb,r}^T), \\
&\qquad \mathbf{r}_{22} = [\mathbf{z}_{comb}^R, J_r^R, \omega_{\max}^R]
\end{aligned} \tag{15}$$

In the above, g_1 and g_2 are battery packaging constraints, g_3 is a performance (0-60 mph time) constraint, g_4 and g_5 are motor feasibility constraints, g_6 is a vehicle range constraint, g_7 is a power violation constraint, and g_8 is a battery capacity constraint [20]. The vectors $\mathbf{z}_{comb}$ and $\mathbf{z}_{comb,r}$ refer to the original vector of combined VVCVs and the combined vector of reduced representation variables, respectively. Additionally, the vectors $\mathbf{t}_{22}$ and $\mathbf{r}_{22}$ include two scalar-valued coupling variables J_r and $\omega_{\max}$. Finally, the superscripts T and R denote target and response versions of the same coupling variable, respectively.

Similarly, the motor subproblem P_{22}, excluding decision variable bound constraints, is stated as:

$$\begin{aligned}
&\min_{\bar{\mathbf{x}}_{22}} \mathbf{v}_{11}^T(\mathbf{t}_{11} - \mathbf{r}_{11}) + \left\|\mathbf{w}_{11} \circ (\mathbf{t}_{11} - \mathbf{r}_{11})\right\|_2^2 \\
&\text{where} \quad \bar{\mathbf{x}}_{22} = [\ell_s, r_m, R_r, n_c],\ \mathbf{t}_{11} = [\mathbf{z}_{comb}^T, J_r^T, \omega_{\max}^T], \\
&\qquad \mathbf{r}_{11} = [\mathbf{z}_{comb}^R, J_r^R, \omega_{\max}^R] = f(\bar{\mathbf{x}}_{22})
\end{aligned} \tag{16}$$

Table 1 provides definitions for the input/output quantities to the objective function and constraint functions for both subproblems.

Table 1. DEFINITION OF INPUT/OUTPUT QUANTITIES TO OBJECTIVE/CONSTRAINT FUNCTIONS

Quantity	Definition
B_I	Battery electrode thickness scale
B_W	Battery cell width scale
B_L	Number of cell windings
x_b	Battery compartment clearance (m)
p_r	Belt drive ratio
J_r	Rotor moment of inertia (kg-m^2)
$\omega_{\max}$	Maximum motor speed
mpg_e	Gasoline-equivalent fuel economy (mpg)
b_w	Battery width (m)
b_ℓ	Battery length (m)
t_{60}	0-60 mph time (s)
τ_V	Torque violation constraint (N-m)
ω_V	Speed violation constraint (rad/s)
R	Vehicle range (mi)
P_V	Power violation constraint (W)
C_b	Battery capacity (A-h)
ℓ_s	Motor stack length (m)
r_m	Rotor radius (m)
R_r	Rotor resistance (Ω)
n_c	Number of turns per stator coil

5 CONSTRAINT MANAGEMENT METHODS FOR POD MODEL VALIDITY REGION

As mentioned in Section 2, the approximations of the motor performance curves via POD are valid only within the sampling domain of the original representations. This is true not only for POD, but for the majority of data approximation applications; that is, approximations to data can be reasonably interpolated, but rarely, if ever, successfully extrapolated. In the context of design optimization, one can ensure that such data extrapolation, and hence ill-behaved analysis and optimization, do not occur, by introducing appropriate constraints on the approximation models. In some cases, simply identifying the maximum and minimum attainable values for the parameters in these models is sufficient; in general, however, one cannot assume that the parameter space is a hypercube, constrained by simple upper and lower bounds. Rather, the parameter space can, in many cases, be highly-nonlinear, resulting in a general hypersurface. The fact that these parameters often lack physical meaning further compounds the difficulty, since it is often impossible to construct constraints manually that effectively define the validity domain of high-dimensional, non-convex, abstract quantities. This is evident in the current ATC problem, where the parameters are POD coefficients serving as reduced representation variables. In this section, two approaches, a penalty value-based heuristic and SVDD, are introduced as a means to constrain the POD model validity region effectively.

5.1 Penalty Value-based Heuristic

The premise behind this approach is to constrain the model validity region indirectly by assigning large penalty values to objective function and constraint function outputs that depend on the reduced representation variables. This would, in turn, force the optimizer to select reduced representation variables

```
try
   % If sim works, set err msg to zero
   sim('pt_060',[],options)
   pt060err = 0;
   varargout{1} = pt060err;
catch
   % Set sim outputs to penalty vals
   mpge = -inf;
   t60 = inf;
   R = -inf;
   PV = inf;
   return
end
```

Figure 4. PENALTY VALUE-BASED HEURISTIC: MATLAB® TRY-CATCH STATEMENT

that lie within the parameter space or model validity region. A key assumption for the successful implementation of this approach is that non-gradient-based optimizers be used instead of gradient-based optimizers. This is because penalizing quantities with large values, such as the objective function, in gradient-based optimizers can result in ill-conditioned optimization problems due to large gradients. The EV problem in Section 4.2 is solved using the non-gradient-based optimizer NOMADm [21].

One possibility for the execution of this heuristic is to program some type of conditional statement that attempts to compute all quantities that are functions of the reduced representation variables and, if it cannot perform the computation, returns penalty values for the appropriate quantities and continues solving the optimization problem. A reasonable approach would be to use a "try-catch" statement when programming in the MATLAB® environment [22]. In this technique, MATLAB attempts to run the block of code between the keywords "try" and "catch", and in most cases will return the results between these keywords. However, if the block of code between "try" and "catch" fails and produces an error, then MATLAB can run an alternative block of code between the keywords "catch" and "end" [22].

Therefore, in the context of the penalty value-based heuristic, one can attempt to compute all quantities that are functions of the reduced representation variables between "try" and "catch" and, if the computations cannot be performed, assign penalty values to the said quantities between "catch" and "end". Figure 4 shows an excerpt from the MATLAB code for the EV study, where the program attempts to run the 0-60 mph powertrain simulation and, upon failing, returns infinite values as appropriate for mpg_e, t_{60}, R, and P_V.

5.2 Support Vector Domain Description

SVDD [23-26] is a classification method that uses a machine learning algorithm to approximate the boundary of a set of data points and to identify whether new data points lie inside the boundary description. In particular, SVDD can be used to represent data set boundaries that are nonlinear, non-convex, and even disconnected without adding much complexity or computational burden. It is also distinct from other machine learning algorithms in that it requires only one class of data for classification as it aims to identify the minimum radius hypersphere enclosing the class. This feature is advantageous for classification problems in which a second class of data is unknown or difficult to generate, as is the case for reduced representation variables.

We assume that the data space can be effectively characterized by a hypersphere, and so we pose the following primal optimization problem [23-26]:

$$\begin{aligned} &\min_{R_{hyp},\mathbf{a},\xi_i} R_{hyp}^2 + C_p \sum_i \xi_i \\ &\text{subject to } \left\|\mathbf{z}_{r,i} - \mathbf{a}\right\|^2 \le R_{hyp}^2 + \xi_i, \quad i = 1 \cdots m \end{aligned} \tag{17}$$

Here, R_{hyp} denotes the hypersphere radius, ξ denotes a hypersphere radius slack variable, C_p denotes the slack variable penalty constant, $\mathbf{z}_r$ denotes a data sample (which is a p-dimensional vector of reduced representation variables in this application), and $\mathbf{a}$ denotes the hypersphere center. The second term in the objective function of Eq. (17) relaxes the optimization problem and permits the inclusion of outliers. In practice, this optimization problem is never solved for reasons given in [27]; instead, its dual is formulated by constructing the Lagrangian

$$\begin{aligned} L(R_{hyp},\mathbf{a},B_i,\xi_i,\mu_i) = R_{hyp}^2 + C_p \sum_i \xi_i \\ - \sum_i B_i \left(R_{hyp}^2 + \xi_i - \left\|\mathbf{z}_{r,i} - \mathbf{a}\right\|^2 \right) - \sum_i \mu_i \xi_i \end{aligned} \tag{18}$$

with nonnegative Lagrange multipliers B_i and μ_i and applying the Karush-Kuhn-Tucker (KKT) conditions to obtain the following constraints [25]:

$$\sum_i B_i = 1, \qquad \mathbf{a} = \frac{\sum_i B_i \mathbf{z}_{r,i}}{\sum_i B_i} = \sum_i B_i \mathbf{z}_{r,i},$$
$$C_p - B_i - \mu_i = 0, \quad i = 1 \cdots m \tag{19)-(21}$$

The new Wolfe dual optimization problem is then stated as follows [25]:

$$\begin{aligned} &\max_{B_i} \sum_i B_i (\mathbf{z}_{r,i}^T \mathbf{z}_{r,i}) - \sum_i \sum_j B_i B_j (\mathbf{z}_{r,i}^T \mathbf{z}_{r,j}) \\ &\text{subject to } \ 0 \le B_i \le C_p, \quad i = 1 \cdots m \\ &\sum_i B_i = 1 \end{aligned} \tag{22}$$

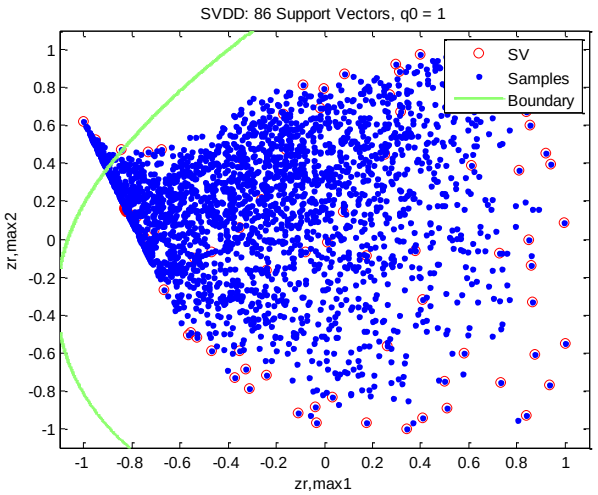

Figure 5. PARTIAL SVDD FOR MAX-TORQUE POD MODEL VALIDITY REGION

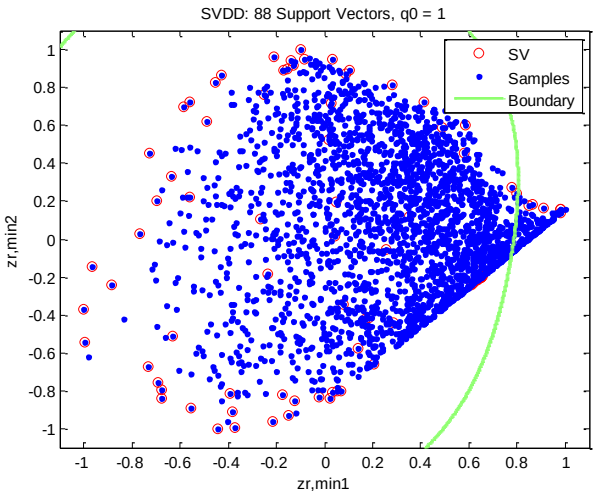

Figure 6. PARTIAL SVDD FOR MIN-TORQUE POD MODEL VALIDITY REGION

In the above, the μ_i's were eliminated using the bound constraints on B_i. The dual solutions are categorized according to three conditions: $B_i = 0$, $0 < B_i < C_p$, and $B_i = C_p$. The first condition ($B_i = 0$) is satisfied by the majority of the dual variables for large m [25] and implies that the associated data sample $\mathbf{z}_{r,i}$ lies within the hypersphere. The second condition ($0 < B_i < C_p$) implies that the associated data sample $\mathbf{z}_{r,i}$ lies at the boundary of hypersphere and is essential to its description; these vectors are termed support vectors [23-26]. The third condition ($B_i = C_p$) implies that the associated data sample $\mathbf{z}_{r,i}$ lies outside the hypersphere and is an outlier.

Using the dual variables and Eq. (20), the squared distance R^2_{dist} from $\mathbf{a}$ to any arbitrary data point $\mathbf{y}$ is calculated as

$$\begin{aligned} R^2_{dist}(\mathbf{y}) &= \|\mathbf{y}-\mathbf{a}\|^2 \\ &= \mathbf{y}^T\mathbf{y} - 2\sum_i B_i(\mathbf{y}^T\mathbf{z}_{r,i}) + \sum_i\sum_j B_iB_j(\mathbf{z}_{r,i}^T\mathbf{z}_{r,j}) \end{aligned} \tag{23}$$

where the indices i and j run over the support vectors and their associated Lagrange multipliers. With this definition, R_{hyp} can be calculated by setting $\mathbf{y} = \mathbf{z}_{r,i}$ for any data sample that is a support vector, and in turn this information can be used to determine whether an arbitrary data point lies inside the boundary description:

$$R^2_{dist}(\mathbf{y}) \le R^2_{hyp} \tag{24}$$

Such a condition can be added to the ATC problem formulation in Eq. (15) to constrain the POD model directly.

A key limitation in the SVDD formulation is that it assumes a hypersphere data space. Since this is rarely the case, one must usually map the data into some higher-dimensional "feature space" through a nonlinear transformation where the hypersphere assumption is more appropriate [25]. Because these nonlinear transformations can be difficult to develop explicitly, Mercer kernel functions [2z] are used to represent the dot product between any two nonlinear transformations. Although several kernel functions exist, the most preferred in the literature is the Gaussian kernel function [28]

$$K_G(\mathbf{z}_{r,i},\mathbf{z}_{r,j}) = e^{-q_0\|\mathbf{z}_{r,i}-\mathbf{z}_{r,j}\|^2} \tag{25}$$

where q_0 is the kernel width parameter. Equation (25) can then be substituted for the dot product terms in Eqs. (22)-(23), yielding the following dual optimization problem and squared distance formulations:

$$\begin{aligned} &\max_{B_i} \sum_i B_iK_G(\mathbf{z}_{r,i},\mathbf{z}_{r,i}) - \sum_i\sum_j B_iB_jK_G(\mathbf{z}_{r,i},\mathbf{z}_{r,j}) \\ &\text{subject to} \quad 0 \le B_i \le C_p, \quad i = 1\cdots m \\ &\qquad\qquad \sum_i B_i = 1 \end{aligned} \tag{26}$$

$$\begin{aligned} R^2_{dist}(\mathbf{y}) &= K_G(\mathbf{y},\mathbf{y}) - 2\sum_i B_iK_G(\mathbf{y},\mathbf{z}_{r,i}) \\ &\quad + \sum_i\sum_j B_iB_jK_G(\mathbf{z}_{r,i},\mathbf{z}_{r,j}) \end{aligned} \tag{27}$$

Two parameters, q_0 and C_p, in Eqs. (26)-(27) must be tuned to construct an appropriate SVDD. In practice, modifications to C_p have minimal impact on the solution [24, 25], leaving only q_0 to be tuned. For $p \le 3$, this tuning can be done visually by assessing contour plots for over-fitting characteristics. For higher dimensions of p, a more formal approach is to assess over-fitting characteristics by using a leave-one-out method [27]

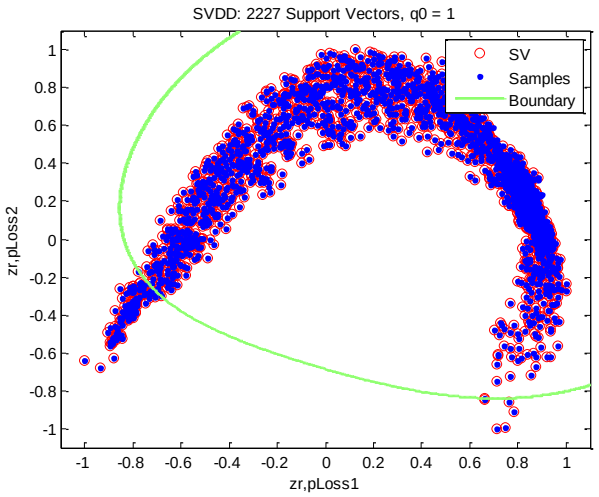

Figure 7. PARTIAL SVDD FOR POWER LOSS POD MODEL VALIDITY REGION

or by testing a small, independent validation data set with the condition in Eq. (24) [24]. The latter method is used in the EV study since m is too large to perform the leave-one-out method efficiently and a validation set is readily available. Note that none of these approaches addresses under-fitting error, which in general violates the original assumptions for SVDD because it requires a significant number of test data points outside the domain.

In the EV study, three SVDDs were constructed to approximate the POD model validity regions associated with the motor torque curves and power loss map, leading to the following additional constraints in Eq. (15):

$$g_9(\bar{\mathbf{x}}_{11}) = R^2_{dist,\max}(\bar{\mathbf{x}}_{11}) - R^2_{hyp,\max} \le 0$$
$$g_{10}(\bar{\mathbf{x}}_{11}) = R^2_{dist,\min}(\bar{\mathbf{x}}_{11}) - R^2_{hyp,\min} \le 0 \qquad (28)\text{-}(30)$$
$$g_{11}(\bar{\mathbf{x}}_{11}) = R^2_{dist,pLoss}(\bar{\mathbf{x}}_{11}) - R^2_{hyp,pLoss} \le 0$$

The sample vectors used to design the SVDDs were identical to those used in the first POD study, with the exception that they were appropriately mapped into the POD space. Figures 5-7 illustrate portions of the SVDDs for two dimensions of the POD model validity regions associated with the torque curves and power loss map.

6 IMPLEMENTATION

The ATC problem formulation, shown in Eqs. (15)-(16), was solved initially by implementing the penalty value-based heuristic discussed in Section 5.1. NOMADm [21], a derivative-free optimization software package based on mesh-adaptive search algorithms, was used. In the P_{11} subproblem, the settings for this MATLAB-based optimizer were modified such that only a Latin hypercube search was performed and only 4,000 function evaluations were permitted. This was necessary

Table 2. OPTIMAL DECISION VECTOR FOR VEHICLE SUBPROBLEM

Vehicle Subproblem, P_{11}							
Variable	B_I	B_W	B_L	x_b	p_r	J_r^T	ω_{max}^T
Value	1.04	1.08	16.30	0.43	2.12	0.42	676

Table 3. OPTIMAL DECISION VECTOR FOR MOTOR SUBPROBLEM

Motor Subproblem, P_{22}				
Variable	ℓ_s^R	r_m^R	n_c^R	R_r^R
Value	0.15	0.12	17.17	0.20

Table 4. OPTIMAL CONSISTENCY CONSTRAINT VECTOR AND PENALTY WEIGHTS

Consistency Constraint	c_{opt}	v_{opt}	w_{opt}
$c_{z,\max}$	5.38	9.83×10^{10}	1.92×10^{5}
$c_{z,\min}$	3.11	5.60×10^{10}	1.92×10^{5}
$c_{z,pLoss}$	1.09	2.06×10^{10}	1.92×10^{5}
c_{Jr}	0.02	2.77×10^{8}	1.92×10^{5}
$c_{\omega max}$	1.31	2.44×10^{10}	1.92×10^{5}

to alleviate computational issues associated with memory availability. However, in the P_{22} subproblem, the default settings were appropriate. Finally, the tolerance on $\|\mathbf{c}^{(K)}\mathbf{c}^{(K-1)}\|_\infty$ for outer loop convergence was set to 10^{-2}.

The optimization results using the penalty value-based heuristic for constraint management are shown in Tables 2-4. The algorithm converged after 15 ATC iterations and resulted in a system solution that was reasonably consistent between both subproblems. The only constraint activity in the subproblems was the upper bound on R_r in the motor subproblem. The optimal values of the reduced representation variables are not listed here as they are not physically meaningful; however, the optimal motor performance curves computed by these reduced representation variables are shown in Figure 8. Under these design conditions, the EV could achieve a gasoline-equivalent fuel economy of approximately 201 mpg.

An optimization problem implementing SVDD for constraint management can be formulated using Eqs. (15)-(16) and Eqs. (28)-(30). Computational results associated with this formulation are left for future work.

7 CONCLUSIONS

Although no complete comparison can be made as of yet between the penalty value-based heuristic and SVDD as constraint management methods, there is a number of advantages and limitations that can be highlighted in both approaches. The penalty value-based heuristic is fairly simple to implement and only requires the user to make a decision regarding the penalty value. However, this approach is limited to slower, non-gradient-based optimizers, as large penalty values for the objective function could lead to ill-conditioned optimization problems due to large gradients. Furthermore, the optimization runtime may be prolonged by the fact that the

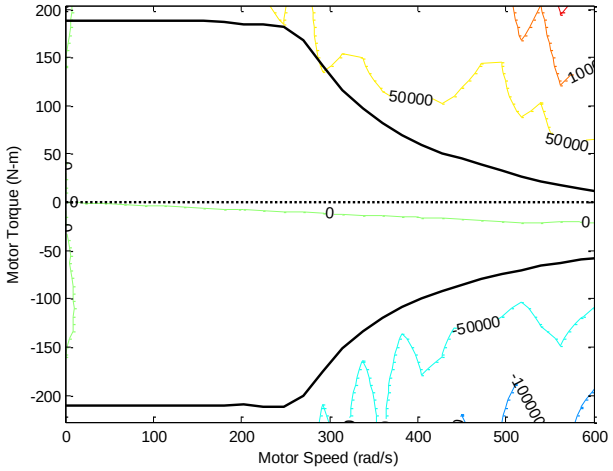

Figure 8. OPTIMAL MOTOR PERFORMANCE CURVES/MAP FOR ATC-POD USING "TRY-CATCH"

optimizer can spend significant amounts of time outside the model validity region because the reduced representation variables are not constrained properly.

SVDD offers an approach to overcome these limitations using hyperspherical boundary definitions to develop conditions that can be used to directly constrain the reduced representation variables. With this method, almost any type of domain, including nonlinear, non-convex, and disconnected ones, can be defined without adding much complexity and computational burden in most situations. Nevertheless, as the number of data samples used in SVDD increase, the computational expense can rise dramatically because of the corresponding increase in the number of decision variables (B_i's) in the dual optimization problem of Eq. (22). This is compounded by the fact that another optimization routine must be performed simultaneously that identifies the proper q_0 for the domain description. We see this situation in the current study, where the combined runtime for developing three SVDD models containing 2500 samples each was approximately 60 hours on a 3 GHz, 4 GB RAM, Intel® Core™ 2 Duo CPU. Therefore, in order for SVDD to be a competitive alternative to the penalty value-based heuristic when using non-gradient-based optimizers, the ATC optimization run time would have to be significantly lower than that of the penalty value-based heuristic. This point is unresolved and left for future work.

ACKNOWLEDGMENTS

This research has been partially supported by the Automotive Research Center, a US Army RDECOM Center of Excellence headquartered at the University of Michigan. This support is gratefully acknowledged.

REFERENCES

[1] Kim, H. M., 2001. *Target Cascading in Optimal System Design*. PhD Dissertation, Dept. of Mechanical Engineering, University of Michigan, Ann Arbor, MI.

[2] Kim, H. M., Michelena, N. M., Papalambros, P. Y., Jiang, T., 2003. "Target cascading in optimal system design". *Journal of Mechanical Design*, **125**(3), pp. 474-480.

[3] Alexander, M. J., 2008. *Analytical Target Cascading Optimization of an Electric Vehicle Powertrain System.* MS Thesis, Dept. of Mechanical Engineering, University of Michigan, Ann Arbor, MI.

[4] Alexander, M. J., Allison, J. T., Papalambros, P. Y., 2009. "Reduced representations of vector-valued coupling variables in decomposition-based design optimization". In Proccedings of the 8th World Congress on Structural and Multidisciplinary Optimization.

[5] Alexander, M. J., Allison, J. T., Papalambros, P. Y., 2010. "Reduced representations of vector-valued coupling variables in decomposition-based design optimization". [In submission].

[6] Sobieski, I., Kroo, I., 1996. "Aircraft design using collaborative optimization". In Proceedings of the AIAA 34th Aerospace Sciences Meeting and Exhibit.

[7] Meade, A. J., Kokkolaras, M., 1996. "Enhancement of a viscous-inviscid-interaction airfoil analysis code using the parallel direct search algorithm". Technical Report CRPC TR96711-S.

[8] LeGresley, P. A., Alonso, J. J., 2004. "Improving the performance of design decomposition methods with POD. In Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA 2004-4465.

[9] Kokkolaras, M., Louca, L. S., Delagrammatikas, G. J., Michelena, N. F., Filipi, Z. S., Papalambros, P. Y., Stein, J. L., Assanis, D. N., 2004. "Simulation-based optimal design of heavy trucks by model-based decomposition: an extensive analytical target cascading case study". *International Journal of Heavy Vehicle Systems*, **11**(3/4), pp. 402-431.

[10] Karhunen, K., 1946. "Zur spektral theorie stochastischer prozesse". *Ann of Acad Sci Fennicae Ser*.

[11] Loeve, M., 1945. "Functions aleatoire de second ordre". C R Academie des Sciences.

[12] Ahmed, N., Goldstein, M. H., 1975. *Orthogonal Transforms for Digital Signal Processing*. Springer, Berlin.

[13] Wilcox, K., 2005. "An introduction to model reduction for large scale applications". *Aerospace Computational Design Laboratory Seminar*. Massachusetts Institute of Technology, http://web.mit.edu/mor/papers/ADCL_Sept05.pdf.

[14] Sirovich, L., 1987. "Turbulence and the dynamics of coherent structures. I – coherent structures. II – symmetries and transformations. III – dynamics and

scaling." *Quarterly of Applied Mathematics*, **43**, pp. 561-571, 573-590.
[15] Burkhardt, J., Du, Q., Gunzburger, M., Lee, H. C., 2003. "Reduced order modeling of complex systems". In Proceedings of the 20th Biennial Conference on Numerical Analysis.
[16] Toal, D. J. J., Bressloff, N.W., Keane, A. J., 2008. "Geometric filtration using POD for aerodynamic design optimization". In Proceedings of the 26th AIAA Applied Aerodyamics Conference, AIAA 2008-6584.
[17] Bui-Thanh, T., Damodaran, M., Wilcox, K., 2004. "Aerodynamic reconstruction and inverse design using proper orthogonal decomposition". *AIAA Journal*, **42**(8), pp. 1505-1516.
[18] Lucia, D. J., Beran, P. S., Silva, W. A., 2003. "Reduced order modeling: new approaches for computational physics". *Progress in Aerospace Sciences*, **40**, pp. 51-117.
[19] Tosserams, S., Etman, L. F. P., Papalambros, P. Y., Rooda, J. E., 2006. "An augmented-Lagrangian relaxation for analytical target cascading using the alternating direction method of multipliers". *Structural and Multidisciplinary Optimization*, **31**, pp. 176-189.
[20] Allison, J. T., 2008. *Optimal Partitioning and Coordination Decisions in Decomposition-based Design Optimization*. PhD Dissertation, Dept. of Mechanical Engineering, University of Michigan, Ann Arbor, MI.
[21] Abramson, M. A., 2007. *NOMADm Version 4.5 User's Guide*. Air Force Institute of Technology, Wright Patterson AFB.
[22] *MATLAB® Function Reference*. The MathWorks, Inc., Natick, MA.
[23] Tax, D. M. J, Duin, R. P. W., 1999. "Data domain description using support vectors". In Proceedings of European Symposium on Artificial Neural Networks.
[24] Tax, D. M. J, Duin, R. P. W., 1999. "Support vector domain description". *Pattern Recognition Letters*, **20**, pp. 1191-1199.
[25] Malak, R. J., 2008. *Using Parameterized Efficient Sets to Model Alternatives for Systems Design Decisions*. PhD Dissertation, School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA.
[26] Malak, R. J., Paredis, C. J. J., 2009. "Using support vector machines to formalize the valid input domain of models in data-driven predictive modeling for systems design". In Proceedings of the ASME 2009 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference.
[27] Vapnik, V., 1995. *The Nature of Statistical Learning Theory*. Springer, New York.
[28] Scholkopf, B., Smola, J. A., 2002. *Learning with Kernels*. MIT Press.