Abstract

Enterprise-level business decisions are linked with engineering product decisions by integrating enterprise utility optimization and engineering design optimization under a hierarchical, multilevel, decision-based design framework. The enterprise problem sets attribute targets (i.e., specifications) for engineering product development, which then optimizes product performance within the feasible design space to match the targets with minimum deviations. When the feasible domain imposed by engineering product development is disconnected in the space of attribute targets, an engineering design with the minimum deviation from the targets may not correspond to the design with the maximum utility value, even though the design is a converged solution from the multilevel optimization. To address this issue, a new algorithm is developed, which systematically explores the target space to lead the engineering product development to a feasible and optimal design in the enterprise context. Analytical examples and an automotive suspension design case study are presented to demonstrate the effectiveness of the proposed methodology.

Key words: multilevel, multidisciplinary design optimization, enterprise-driven design, analytical target cascading, discrete choice analysis, multinomial logit demand model, disconnected feasible domain.

1 Introduction

Department of General Engineering, University of Illinois, Urbana, IL, 61801, USA

Department of Mechanical Engineering, Northwestern University, Evanston, IL, 60208, USA

Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI, 48109, USA
Willcox. They have demonstrated the interaction between marketing and engineering in an enterprise. In this article the enterprise is defined as the organization that designs and produces an artifact to maximize its utility (e.g., profit). For simplicity, marketing, production planning, and other enterprise-level activities are referred to as enterprise-level product planning; engineering-related design activities are referred to as engineering product development.

Emphasis on enterprise-driven design models has led to incorporating demand and profit models that capture both producer and consumer needs into product design. For example, Wassenaar and Chen developed demand models utilizing Discrete Choice Analysis in a Decision-Based Design (DBD) framework. The demand models capture the choices that customers make as functions of customer-product-selection attributes, i.e., product attributes that are of interest to customers. Along with the cost models, Wassenaar’s DBD approach has been used to obtain the optimal settings of product attributes at the enterprise level to maximize the net revenue of a firm, considering engineering needs, socioeconomic and demographic background of customers, and time. In Wassenaar’s approach, the focus was on identifying the optimal product attributes, rather than linking the product attributes with engineering design process. In this article, following the DBD framework, a multi-level optimization formulation is proposed to link enterprise-level decision making with engineering-level product design, and to treat the engineering product development as a process of meeting performance targets identified from the enterprise decision making. Further, a new algorithm is developed to systematically explore the potentially disjoint performance target space, and lead the engineering product development process to a feasible and optimal design for the enterprise.

Under the existing DBD framework, all product design decisions, engineering or non-
are made simultaneously (all-in-one) to optimize the enterprise level design objective, i.e., to maximize the expected utility, expressed as a function of net revenue (profit).

This all-in-one (AIO) approach, shown in Figure 1 (a), is desirable, but often practically infeasible due to the computational and organizational complexity. From the viewpoint of organizational infrastructure of a company, decomposition and multilevel distributed approaches must be introduced. Figure 1 (b) illustrates a decomposed view of the interaction between enterprise-level product planning and engineering product development in a hierarchical, multilevel optimization framework. The presented framework shares similar ideas as in Cooper, et al. 6 and Michalek, et al. 8 that treat engineering design as a hierarchical process of meeting design targets at different levels. The difference is that this work follows the DBD practice in formulating the enterprise-level optimization problem and uses the Discrete Choice Analysis for demand modeling. As detailed in Section 2, the demand model provides a link between engineering product development and enterprise-planning. Following Figure 1 (b), the enterprise-level product planning problem maximizes the utility $V$ with respect to the target performance $U_T$ and the rest of the enterprise decision variables $x_{ent}$, subject to enterprise-level design capability $0h_0g_{ent} \leq 0$. When the utility is maximized, the corresponding target performance is defined as the utopia target. Following current industrial practice, we view the engineering product development process as a process where targets set at the enterprise level are met through product realization. At the engineering level, the objective is to minimize the deviation between the performance target and the achievable product performance response $R$, while satisfying the engineering feasibility constraints $g, h$. 

$$g_{ent} \leq 0 \ h_{ent} = 0$$
When solving the integrated enterprise and engineering problem using multi-level optimization, it is essential to ensure preference consistency: smaller deviation from the targets in engineering product development corresponds to a higher enterprise-level utility, so that the solution from the multi-level optimization procedure (a decomposed problem, Figure 1 (b)) matches the AIO solution of the integrated enterprise and engineering problem (Figure 1 (a)). Ensuring preference consistency is not very straightforward with a decomposed formulation. Under a multi-level design framework, an ideal product development scenario is when the targets corresponding to the best enterprise utility (i.e., utopia target) would lead to an engineering design matching the targets perfectly. A perfect match may be rare due to engineering constraints introduced at the product development level. Additionally, as further detailed in Section 3, if the feasible domain imposed by the engineering product development is disconnected (i.e., disjoint) in the space of...
attribute targets (engineering performance domain), the engineering design with the minimum deviation from the targets may not correspond to the design with the maximum utility value, even though the design is a converged solution from the multilevel optimization. Disconnected feasible performance domains often occur in complex systems design when there are multiple engineering disciplines involved in engineering development, and when each discipline can seek distinctly different design alternatives in engineering development. In the vehicle suspension design case study of Section 4, a vehicle manufacturer maximizes the enterprise utility based on two disconnected feasible target performance domains imposed by suppliers of suspension components. The suppliers (or engineering design teams) try to achieve the utopia targets (suspension stiffnesses) as closely as possible, but it is assumed that a perfect match is not possible.

To overcome this difficulty, one of the main contributions of this work is to propose an algorithm that can systematically explore attribute targets in the disconnected feasible domain to lead the engineering product design process to finding a feasible and optimal design in the enterprise context. The proposed algorithm in Section 3 guides the enterprise-level decision maker to assign alternative targets so that the enterprise maximizes the net revenue and the suppliers achieve the targets as closely as possible. The adjustment of targets set at the enterprise level guides the exploration within a disconnected engineering performance feasible domain. This adjustment may shift the enterprise utility value away from its original utopia value. In return, however, a better (i.e., higher utility) feasible design can be obtained satisfying the engineering constraints.

The proposed formulation uses a bi-level hierarchical structure that does not entail competition...
in their interaction. The interactive nature between enterprise product planning and engineering product development is similar in concept to the leader/follower model, except that existing applications of game theory emphasize interactions between different engineering groups, while our work focuses on the interaction between upper enterprise needs and lower engineering capabilities (Figure 1 (b)). Also, the traditional disjunctive programming literature focuses on modeling a problem with disjunctive feasible domains utilizing mixed integer nonlinear programming problem formulations and conventional search algorithms, such as branch and bound, are applied to solve them. Considering disconnected (or disjoint) feasible space is conceptually similar to the work here, but the proposed algorithm focuses on locally “exploring” the neighboring region of the performance target, rather than globally searching for the solution in an enumerative way.

In this work, the engineering product development problem at the bottom of the hierarchy in Figure 1 is viewed as a hierarchical process by itself. To achieve the targets identified from enterprise product planning, the analytical target cascading (ATC) approach is adopted for the hierarchical product development process. ATC is a multilevel, multidisciplinary design methodology to find an optimal system design, ensuring consistency among subsystems or disciplines and achieving the overall product targets assigned at the top of the hierarchy. In the field of MDO, several design architectures have been developed to support collaborative, multidisciplinary design environment using distributed design optimization, e.g., concurrent subspace optimization (CSSO), Bi-Level Integrated System Synthesis (BLISS), and collaborative optimization (CO). A comprehensive review of MDO architectures is provided by Sobieszczanski-Sobieski and Haftka. It should be noted that the engineering product
2 Enterprise Product Planning and Engineering Product Development Model
As shown in Figure 2, the demand $Q$ plays a critical role in assessing both the revenue and expenditure, and ultimately the profit (i.e., net revenue $V$). To use the enterprise model to guide engineering product development, $Q$ is expressed here as a function of the customer-product-selection attributes $A$, socioeconomic attributes $S$ of the market population, price $P$ and time $t$; the expenditure (i.e., lifecycle cost including manufacturing cost and others) is a function of the customer-product-selection attributes $A$, the engineering attributes $E$, exogenous variables $x$, demand $Q$ and time $t$. The customer-product-selection attributes $A$ are product features (next to brand, price, and warranty) that a customer typically considers when purchasing the product and the engineering attributes $E$ are quantifiable product properties that can be directly optimized by design engineers, but indirectly affect purchasing behavior and manufacturing cost. For example, a customer prefers better ride quality of a vehicle that is influenced by the stiffness of the suspension springs. Here the stiffness of spring is an example of the engineering attribute that influences demand and net revenue, but indirectly affects purchasing behavior of a customer. At the enterprise level, targets $T$ are set for both $A$ and $E$. To assist the selection of design alternatives in engineering development, the relationships of $A$ and $E$ with design options need to be established through engineering analyses.
Demand modeling using Discrete Choice Analysis (DCA) and Conjoint Analysis (CA) has been widely used in the marketing and transportation communities [Green and Srinivasan, Ben-Akiva and Lerman, Koppelman and Sethi]. Researchers in the design community [Besharati, Li and Azarm, Wassenaar and Chen, Michalek, et al.] have also had success in integrating such models in their work for product design. In this article, the DCA approach is adopted for demand modeling and the multinomial logit (MNL) model is used because it has a closed form and uses the Gumbel error distribution that closely approximates the normal distribution, a more realistic assumption for the error distribution when estimating choice behavior. The choice probability MNL model is shown in (Eq. 1), where $Pr_{n,i}$ is the probability that respondent $n$ chooses alternative $i$, $J$ is the choice set that is available to individual $n$, and $W$ is the observable/deterministic part of the utility function of customer:

$$Pr_{n,i} = \frac{e^{W_{n,i}}}{\sum_{j \in J} e^{W_{n,j}}}$$

This formulation implies that the probability of choosing an alternative increases monotonically with an increase in deterministic utility of that alternative and decreases with the increase of deterministic utility of any or all of the other alternatives. Using the MNL model imposes the Independence of Irrelevant Alternatives (IIA) property, which makes the calculation of choice probabilities much easier. However, there are limitations associated with the IIA property, and the explanation of them is beyond the scope of the current work.
Figure 3. Decomposition example with three-level hierarchy

\[ P_{ij} \]

\[ R_y = \begin{bmatrix} R_y \ R_y \end{bmatrix} = r_y \begin{pmatrix} R_{i+k} \ldots \ R_{i+k} \ x_y \ y_y \end{pmatrix} \]

\[ x = \begin{pmatrix} x_y \ y_y \ y_{i+k} \ldots \ y_{i+k} \ R_{i+k} \ldots \ R_{i+k} \ e_{ij} \ e_{ij}^T \end{pmatrix} \]

\[ R_{ij} \quad R_y \]

\[ e_{ij}^R \quad e_{ij}^y \]

\[ \gamma^U \quad \gamma^L \]

\[ T \quad g_y \quad h_y \]
1. **Constraints.** Weights \( w \) are assigned to the deviation term \( s \) in the objective and the convergence behavior of the ATC process can be affected by the way weights are set (or updated). In this work, the weights are set to reflect preferences after normalizing the individual deviation terms. For this, the reader can refer to weight update schemes and their effects on convergence\(^ {33, 34, 35} \).

2. **Mathematical Formulation**

   \[
   P_{ij} = w^R \| R_{ij} - T_{ij} \| + w^L \| R_{ij} - R^U_{ij} \| + w^R \| y_{ij} - y^U_{ij} \| + e^R_{ij} + e^Y_{ij}
   \]

   \[
   \sum_{k \in C_i} \| R_{i+k} - R^L_{i+k} \| \leq e^R_{ij}
   \]

   \[
   \sum_{k \in C_i} \| y_{i+k} - y^L_{i+k} \| \leq e^Y_{ij}
   \]

   \[
   g_{ij}(R_{ij}, x_{ij}, y_{ij}) \leq 0
   \]

   \[
   h_{ij}(R_{ij}, x_{ij}, y_{ij}) = 0
   \]

3. **Multilevel Enterprise and Engineering Problem Formulation and Solution Algorithm**

   \[
   P_{ent}^{T \mathbf{X}_{ent}} V T \mathbf{X}_{ent}
   \]

   \[
   V T
   \]
3.1 Introducing a new constraint in the enterprise-level problem
The idea of adding the constraint in (Eq. 4) is to explore targets in the new domain that may potentially lead to feasible designs with better enterprise utility. The points inside the circular constraint in (Eq. 4) are ruled out because they are infeasible (otherwise they should be identified as a solution in the previous iteration as their deviations from the utopia target are less). For example, in Figure 4 the engineering problem returns point A to the enterprise problem with the minimum deviation from the utopia target $T^*$. A circular inequality constraint is imposed on the utopia target at the center, with the distance between the utopia target and the engineering response as its radius. The modified enterprise problem $E_{\text{ent}}'$ (Eq. 5) generates a new target $T'$ for the engineering problem (see Figure 5). Based on the new target the engineering problem finds point B as the optimum with the minimum deviation from the new target $T'$. Point B is farther...
Figure 5. A circular constraint imposes a new target \( T' \) for the engineering problem.

Figure 6. Updating the radius \( \Delta \) in the additional constraint in the modified enterprise problem: the slope is considered to set the radius of the constraint (Eq. 4)
3.2 Utilization of the slope information

The goal of solving the modified enterprise problem (Eq. 5) is to lead to an other disconnected feasible domain by assigning a new target. However, depending on the slope of the enterprise utility curve, the initial new target for the engineering problem may not sufficiently lead to another disconnected feasible domain, thus setting the new target may need to be repeated. To visualize this situation, in Figure 6, a unimodal utility function is plotted with the engineering feasible domain overlapped. The shaded region denotes the infeasible engineering domain, while a disconnected feasible domain exists, on both sides of the infeasible one. At the first iteration the engineering problem returns a response \( \Delta - T \). A constraint, as shown in (Eq. 4), is added to the enterprise problem and the new target is identified at \( \Delta + T \). Take a case when the other feasible domain is farther from the new target \( \Delta + T \) than the previous feasible domain, as shown in Figure 6. The response with the minimum deviation from the target does not change from the previous response \( \Delta - T \), i.e., the algorithm returns to the same response as previously found. To avoid terminating the search without exploring farther, additional slope information needs to be utilized to adjust the radius of the restricted feasible domain in the enterprise problem. After finding the minimum deviation engineering design at \( \Delta - T \), the gradient of the utility function, denoted as \( \alpha \), is obtained analytically or numerically. When the next target for the engineering problem \( \Delta + T \) is obtained by solving the modified enterprise problem, the gradient of the utility function, denoted as \( \phi \), is also obtained. If the engineering problem returns the same response \( \Delta - T \) and \( \phi \leq |\alpha| \), then the constraint radius may be increased up to \( \Delta' \) with the gradient ratio:
An updated target \( \Delta' + \gamma \Delta' \) is assigned to the engineering problem and the engineering problem returns \( R' \) as response, where \( \gamma \) is the step size for updating the new radius of the constraint that takes a value \( \frac{1}{\alpha} < \gamma \leq \frac{\phi}{\alpha} \). When \( \alpha \phi \gamma = \frac{\phi}{\alpha} \), the radius is identical to the original radius \( \Delta \), and when \( \frac{1}{\alpha} = \gamma \), the radius becomes \( \Delta' \) in (Eq. 6). As shown in Figure 7, \( \gamma \) can take any value in \( \frac{1}{\alpha} < \gamma \leq \frac{\phi}{\alpha} \) as long as the newly assigned target corresponds to a better utility value. Hence the enterprise problem repeats assigning a new target by updating \( \gamma \) if the engineering problem returns the same response as far as the utility improvement is expected. The upper limit for increasing the radius of the constraint is provided in (Eq. 6), based on a linear approximation of the utility function as shown in Figure 6.
Figure 7. Solution algorithm

...
Algorithm
1. Start with \( x \).
2. Solve the original enterprise problem \( P_{ent} \) and find the utopia target \( T \).
3. Solve the engineering problem \( P_{eng} \) and obtain the response \( R^i \) with the minimum deviation from the target \( T \).
4. Add an additional constraint \( C_{aux} \) in the enterprise problem and find a new target \( T^i \) by solving the modified enterprise problem \( P'_{ent} \).
5. Solve the engineering problem and obtain the response \( R^{iL} \) with the minimum deviation from the target \( T^i \).
6. If \( R^{iL} \approx R^{iL} \quad \frac{\nabla V^{iL}}{\nabla V^i} < \frac{\nabla V^i}{\nabla V^{iL}} \), increase \( \Delta \) in \( C_{aux} \) in \( P'_{ent} \) to \( \Delta^i = \gamma \frac{\nabla V^{iL}}{\nabla V^i} \Delta^{iL} \) where \( \frac{\nabla V^i}{\nabla V^{iL}} \leq \gamma \leq \).
   
   6.1. If \( V T - \Delta^{iL} < V T + \Delta^i \), go to step 4 and solve \( P'_{ent} \).
   
   6.2. Otherwise, decrease \( \gamma \) until it satisfies \( V T - \Delta^{iL} < V T + \Delta^i \).
7. If \( R^{iL} \approx R^{L^{iL}} \quad \frac{\nabla V^{iL}}{\nabla V^i} \geq \frac{\nabla V^i}{\nabla V^{iL}} \), compare the current enterprise utility value to the previous one and accept the current design if improved, or accept the previous design.
8. End.

4 Demonstration and Verifications
4.1 Analytical Examples

4.1.1 Utility with Single Optimum
The iterative ATC process between $sP$ and $21$, sub super PP converges, the overall response with respect to the targets $21$, $TT$ is passed up to the top (enterprise) level problem and the utility is adjusted completing each iteration in Table 1.

**Figure 8.** Decomposed problem hierarchy following analytical target cascading.

**Figure 9.** Disconnected feasible domains are mapped over the utility space. The arrows represent the gradient.
Note that the engineering problem (Eq. 8) has a disconnected feasible domain with respect to $21$, $TT$. This domain is plotted over the utility domain in Figure 9 (a). For the unconstrained enterprise level problem, the optimal target, i.e., utopia target, is found at $(0.79, 0.2)^*$. Based on the utopia target, the engineering problem finds an optimal response with the minimum deviation at $(60.14, 0.0064)^* sysR$, which is not the best available solution in the enterprise utility sense. With the proposed algorithm, based on the engineering response $(0 sysR)$, a constraint in (Eq. 4) is added to the enterprise level problem and the enterprise level problem is solved again to assign a new target $T'$ to the engineering problem. An updated engineering level solution $(1 sysR)$ is found near the previous solution $(0 sysR)$, i.e., the engineering level response is found in the same feasible domain. The radius for the additional constraint in (Eq. 4) and (Eq. 5) is increased to $\Delta V = \Delta' - \gamma_i V_i$, where $\gamma_i V_i \leq \Delta V - \gamma_i V_i$. A newly assigned target $T'$ based on the slope information (Eq. 6) from the modified enterprise level problem guides the engineering level problem to find the optimal response $(2 sysR)$ in other feasible domain. The corresponding utility value for $(2 sysR)$ is better and the lower level response is accepted as a solution and the algorithm is terminated. At the optimum, the solution is found at $(71.84, 2.84, 2.5, 2.58, -0.7, 0.28, 2.57, 0.30, 2.59, 0.21, 4.75, 0.09, -0.06, 1.5, 15.30, 0, 0, 0, 0)$. The iteration process is summarized in Table 1. Note that the responses are getting closer to the all-in-one (AIO) solution that is obtained when the decomposed problems in (Eq. 7) and (Eq. 8) are solved in a single integrated problem.
4.1.2 Utility with Multiple Local Optima

\[
P_{\text{ent}} = V = +T -T + -T + -T + T T T T
\]
The engineering problem is decomposed and solved as in the first example. The contour plot of the utility function with the overlapped engineering level constraints is presented in Figure 9 (b). For the unconstrained enterprise level problem, the optimal (utopia) target is found at \((695.4, 837.3)\). Based on this target, the engineering level problem finds an optimal response with the minimum deviation at \((216.5, 259.4)\). Based on response \(T_{sys}\) a constraint, as shown in (Eq. 4), is added to the enterprise level problem and the enterprise level problem is solved again to assign a new target \(T\) to the engineering level problem. The additional constraint guides the search process to find another local optimum in the utility space and assigns a new target for the engineering level problem. The engineering level problems find an optimal solution \(1_{sys}\) in the other region of the feasible domain. After checking that the corresponding utility value for \(1_{sys}\) is better than the previous one, the engineering level response is accepted as a solution. At the optimum, the solution is found at \((-0.96, 1.40, 3.38, 2.19, 63.2(), (), (), 1421\) = \(K\). The iteration process is summarized in Table 2.
Table 2 Iteration History: Utility with Multiple Optima

<table>
<thead>
<tr>
<th>Target 1</th>
<th>Target 2</th>
<th>Desired utility</th>
<th>Response</th>
<th>Actual utility</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.837</td>
<td>4.695</td>
<td>1.903</td>
<td>4.259</td>
<td>5.216</td>
</tr>
<tr>
<td>2.470</td>
<td>3.356</td>
<td>1.365</td>
<td>2.634</td>
<td>3.383</td>
</tr>
</tbody>
</table>

The preceding two analytical examples demonstrated that the proposed approach successfully explores the enterprise utility space to meet the enterprise objective with a consistent feasible engineering design. Note that the final solutions in Table 1 and Table 2 have deviations from the reference AIO solution, mainly contributed by the way target values are set. The AIO reference solution is based on the utopia target and the current solution is based on the updated new targets, satisfying the geometric constraint. In other words, the focus of the proposed algorithm is not on matching the AIO solution exactly, rather it focuses on a systematic exploration of the feasible space to generate the final solution near the AIO solution. Additionally, in case there exists no additional disconnected feasible domain, the algorithm can still find the best feasible design.

4.2 Enterprise–Driven Multilevel Vehicle Suspension Design

- The algorithm is now applied to an enterprise-driven vehicle design problem with emphasis on vehicle suspension design at the product development level. As shown in Figure 9, at the enterprise level, the vehicle profit model is created based on the customers' demographic attributes and all critical vehicle system attributes. Since the product development level focuses only on the suspension for vehicle ride quality improvement, only the targets for front and rear suspension stiffness parameters are treated as variables at the enterprise level. Modeling details of the vehicle chassis composed of front/rear double A-arm suspension and vertical/ cornering tire models can be found in Kim, et al. ATC successfully simulated the vehicle chassis design process based on the targets for handling and ride quality of a sport-utility vehicle. Here a simplified chassis
4.2.1 Medium Size Vehicle Demand Model

\[ U_{in} = W_{in} + \varepsilon_{in} \]
The deterministic part of the utility can be parameterized as a function of observable independent variables (customer-product-selection attributes $A$, engineering attributes $E$, socioeconomic and demographic attributes $S$, and price $P$) and unknown coefficients $\beta$, which can be estimated by observing the choices respondents make and thus represent the respondents' preference. The $\beta$ coefficients and utility functions are indicated with the subscript $n$, representing the $n^{th}$ respondent, while the index $i$ refers to the $i^{th}$ choice alternative. There is no functional form imposed on the utility function $W$. For the purpose of the current study, an additive form of the utility function, linear in the $\beta$ coefficients is used.

\[
W_{in} = f(A_i, E_i, P, S_n, \beta_n)
\]

Based on the market data from J.D. Power and Associates and the vehicle attribute descriptions from Ward's automotive yearbook, the demand model $Q$ is created as a function of demographic and product attributes such as income, age, retail price, resale value, vehicle dependability index (VDI: a quality measure, expressed in terms of defects per 100 parts), annual percentage rate (APR) of loan, fuel economy, vehicle length, front suspension stiffness, and rear suspension stiffness (Table 3). The model makes predictions on the change in market share, i.e., change in the number of vehicles sold of a particular make. Twelve vehicles (seven models and twelve trim levels) are considered in the demand model representing the midsize car segment. Considering other segments, for example, sports models or pick-up trucks, would run the risk of yielding models that have a heavy demographic bias and that are not very sensitive to changes in product attributes. The assumption is that customers only consider vehicles from the midsize car segment, and specifically the twelve vehicle trims, when making their decision. Note that the demographic attributes can only be included as alternative specific variables due to the nature
of the MNL. In the current example, there are 12 vehicle alternatives, considering the various midsize car trims. If a demographic attribute, like age or income of the respondent, should be included in the model, the coefficients for a maximum eleven variables can be estimated for each demographic attribute. Each of these eleven variables corresponds to one of the alternatives or car trims. The demographic variable corresponding to at least one of the alternatives, e.g., alternative 1, has to be fixed. Usually, the coefficient for the alternative that is fixed is set to be zero. The coefficients for the rest of the alternatives, alternatives 2 to 12, are then estimated with respect to the reference alternative 1. In the demand model used here, eleven income variables corresponding to alternatives 2 to 12 are estimated and they are evaluated with alternative 1 as reference. Similarly, the age variable is assumed to be equal for all domestic cars and is evaluated with respect to imported cars. The number of survey correspondents was 3881. Front and rear suspension spring stiffnesses are used to model suspension characteristics.

Starting with a baseline specification that includes alternative specific constants (ASC), i.e., essential product attributes and demographic variables, the demand model is improved incrementally by adding additional attributes to the model. Alternative specific constants, such as fuel economy and suspension stiffnesses, are added to represent the average preference of individuals for an alternative relative to a reference alternative and also to account for the average effect of all explanatory variables. The estimated models are then evaluated on several criteria including behavioral realism or their ability to model customer behavior in line with the analyst's expectations and goodness of empirical fit to the data. The model estimation is carried out by maximizing the log likelihood using STATA37.

Table 3 includes results of the model estimation. Several observations can be made. Negative
Signs of retail price, VDI, APR and vehicle length mean that customers prefer lower values for these variables, i.e., customers prefer cheaper cars, lower interest rates, fewer defects and cars that facilitate easy parking. Positive sign for fuel economy means that customers prefer higher gas mileage and positive signs for the suspension stiffnesses mean that stiffer suspensions are preferred. A stiffer suspension generally translates to better handling and load-carrying abilities but also results in a harsher ride. In this context, the current choice model indicates that customers value handling characteristics more than ride quality. Also, since we are dealing with variables normalized with respect to their extreme values, the magnitude of the coefficients should reflect their relative importance.

<table>
<thead>
<tr>
<th>Variable Type</th>
<th>Description</th>
<th>$\beta$</th>
<th>Coefficient</th>
<th>t-value</th>
<th>95% Confidence Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demographic Age</td>
<td></td>
<td>0.13</td>
<td></td>
<td>12.37</td>
<td>(0.11, 0.15)</td>
</tr>
<tr>
<td>Product Retail Price</td>
<td></td>
<td>-1.57</td>
<td></td>
<td>-4.14</td>
<td>(-2.31, -0.82)</td>
</tr>
<tr>
<td>Resale Value</td>
<td></td>
<td>2.15</td>
<td></td>
<td>2.54</td>
<td>(0.49, 3.80)</td>
</tr>
<tr>
<td>Vehicle Dependability</td>
<td></td>
<td>-1.69</td>
<td></td>
<td>-1.49</td>
<td>(-3.92, 0.53)</td>
</tr>
<tr>
<td>Index APR</td>
<td></td>
<td>-1.05</td>
<td></td>
<td>-1.34</td>
<td>(-2.58, 0.49)</td>
</tr>
<tr>
<td>Product Fuel Economy</td>
<td></td>
<td>0.64</td>
<td></td>
<td>1.51</td>
<td>(-0.19, 1.46)</td>
</tr>
<tr>
<td>Vehicle Length</td>
<td></td>
<td>-0.60</td>
<td></td>
<td>-0.5</td>
<td>(-2.95, 1.74)</td>
</tr>
<tr>
<td>Front Suspension Stiff</td>
<td></td>
<td>1.75</td>
<td></td>
<td>3.11</td>
<td>(0.65, 2.85)</td>
</tr>
<tr>
<td>Rear Suspension Stiff</td>
<td></td>
<td>0.88</td>
<td></td>
<td>1.28</td>
<td>(-0.47, 2.24)</td>
</tr>
</tbody>
</table>

Enterprise level utility is defined as the change of total profit (Eq. 13), which leads to (Eq. 14), a function of demand and cost. Price of a vehicle $P$ is assumed as a constant, $C_{\text{susp}}$ and $C$. 

Table 3 Results of Demand Model Estimation
\[
\Pi = Q \cdot A, E, S \times P - C_{\text{susp}} - C
\]

\[
P = C = a_f \quad a_r \quad k_{sf} = k_{sr} =
\]

\[
\Delta \Pi = Q \cdot k_{sf} \cdot k_{sr} \times (a_f \cdot k_{sf} - a_r \cdot k_{sr}) \cdot (P - C) \\
- Q \cdot k_{sf} \cdot k_{sr} \times (a_f \cdot k_{sf} - a_r \cdot k_{sr}) \cdot (P - C)
\]

Figure 11. Vehicle profit model: profit change with respect to suspension stiffness changes. The shaded
areas represent feasible suspension design domain. The baseline vehicle suspension stiffnesses are 
\( k_{sf} = 5.250 \text{ [kN/m]} \) and \( k_{sr} = 5.190 \text{ [kN/m]} \). The solid curve connecting A and T’ indicates the geometric distance constraint.

### 4.2.2 Implementation of Proposed Algorithm

Table 4 Iteration History: Maximizing Profit with Vehicle Suspension Design Change

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Target ( k_{sf} )</th>
<th>Target ( k_{sr} )</th>
<th>Desired utility ($)</th>
<th>Response ( k_{sf} )</th>
<th>Response ( k_{sr} )</th>
<th>Response utility ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>30.2</td>
<td>19.5</td>
<td>80,460</td>
<td>25.0</td>
<td>19.5</td>
<td>-8,935</td>
</tr>
<tr>
<td>2</td>
<td>28.6</td>
<td>24.5</td>
<td>33,701</td>
<td>29.4</td>
<td>25.0</td>
<td>27,234</td>
</tr>
</tbody>
</table>

Table 4 summarizes the iteration process for the algorithm. The utopia target for suspension design is given at \( [5.19, 2.30] \) \( \ast \) \( T \) (kN/m) with profit $80,460 corresponding to the peak point on the utility surface in Figure 11. Due to engineering feasibility the design with the minimum deviation from the utopia target is found at point \( [5.19, 0.25] \) \( \ast \) \( A \) (kN/m) with -$8,935 profit (i.e., loss). Based on this design, the algorithm imposes a limiting constraint (Eq. 4) to the enterprise problem. After adding the constraint in Figure 11 (solid curve), the modified enterprise problem finds a new optimal target with the maximum utility at \( [5.24, 6.28] \) \( \ast \) \( T' \) (kN/m) with $33,701 profit. This new target guides the ATC process to reach design \( [0.25, 4.29] \) \( \ast \) \( B \) (kN/m) with the improved profit $27,234. Note that the utopia targets assigned to the ATC problem are not
5 Conclusions
that the proposed algorithm works effectively for problems with disconnected target space; it also has the capability to escape infeasible local utility optima.

The vehicle design example captures the disjoint nature of the product performance targets and manufacturing limitations. With the current algorithm, the final optimal design is a better design in maximizing the profit of a firm as well as meeting the feasibility requirements imposed in the vehicle suspension design specifications. Multinomial logit model was successfully incorporated in developing the demand model that was part of the enterprise level objective for the vehicle case study. Also analytical target cascading was successfully incorporated to achieve the performance targets in the hierarchical engineering product development process.

The proposed algorithm is applicable to a general multilevel, multidisciplinary design case where the lower level performance (i.e., specification) space is imposed in a disconnected feasible domain at the higher level. Note that if all design variables are continuous with the exception of the target space, traditional optimization algorithms such as sequential quadratic programming (SQP) can still be applied in both enterprise and engineering level problems. The results of the two analytical examples in Section 4 were obtained by SQP in Matlab [38]. The number of disconnected feasible domains in the examples was two; however, the method is not limited to just two domain cases. After updating the Euclidean distance radius in the additional constraint at the enterprise level, a global search algorithm can be incorporated at the engineering design level. Currently the algorithm terminates as soon as it finds a better utility or confirms that the current utility is the best after increasing the radius to the upper limit. However, this algorithm can be applied repeatedly in combination with global search to handle the case where the utility function at the enterprise level has multiple optima. Future work also involves...
introducing uncertainty in multilevel optimization and adopting the nested multinomial logit model for demand modeling, which can be more effectively incorporated in multilevel decision making scenarios as well as in heterogeneous market segments (e.g., considering sedans with sport utility vehicles). Further, the sensitivity of multiple product attributes to profit from the enterprise model can be utilized to refine the objective in the engineering product development model.

### Appendix

#### Table 5 Front Suspension Design

<table>
<thead>
<tr>
<th>Rear suspension subsystem design</th>
<th>Optimal value</th>
<th>Lower bound</th>
<th>Upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear coil spring stiffness (N/mm)</td>
<td>113.4</td>
<td>30</td>
<td>160</td>
</tr>
<tr>
<td>Spring free length (mm)</td>
<td>375.1</td>
<td>300</td>
<td>650</td>
</tr>
<tr>
<td>Torsional stiffness (N-m/deg)</td>
<td>30</td>
<td>20</td>
<td>85</td>
</tr>
<tr>
<td>Overall rear suspension stiffness (N/mm)</td>
<td>29.3</td>
<td>19</td>
<td>30.2</td>
</tr>
<tr>
<td>Suspension travel (m)</td>
<td>0.099</td>
<td>0.05</td>
<td>0.1</td>
</tr>
</tbody>
</table>

#### Table 6 Rear Suspension Design

<table>
<thead>
<tr>
<th>Rear suspension subsystem design</th>
<th>Optimal value</th>
<th>Lower bound</th>
<th>Upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear coil spring stiffness (N/mm)</td>
<td>70.5</td>
<td>30</td>
<td>160</td>
</tr>
<tr>
<td>Spring free length (mm)</td>
<td>472.4</td>
<td>300</td>
<td>650</td>
</tr>
<tr>
<td>Torsional stiffness (N-m/deg)</td>
<td>69.3</td>
<td>20</td>
<td>85</td>
</tr>
<tr>
<td>Overall rear suspension stiffness (N/mm)</td>
<td>25</td>
<td>19</td>
<td>30.2</td>
</tr>
<tr>
<td>Suspension travel (m)</td>
<td>0.099</td>
<td>0.05</td>
<td>0.1</td>
</tr>
</tbody>
</table>

#### Table 7 Front Coil Spring Design

<table>
<thead>
<tr>
<th>Front coil spring component design</th>
<th>Optimal value</th>
<th>Lower bound</th>
<th>Upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wire diameter (m)</td>
<td>0.015</td>
<td>0.005</td>
<td>0.03</td>
</tr>
<tr>
<td>Coil diameter (m)</td>
<td>0.077</td>
<td>0.05</td>
<td>0.2</td>
</tr>
<tr>
<td>Pitch</td>
<td>0.04</td>
<td>0.04</td>
<td>0.1</td>
</tr>
<tr>
<td>Linear coil spring stiffness (N/mm)</td>
<td>114.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spring bending stiffness (N-m/deg)</td>
<td>28.5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Table 8 Rear Coil Spring Design

<table>
<thead>
<tr>
<th>Rear coil spring component design</th>
<th>Optimal value</th>
<th>Lower bound</th>
<th>Upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wire diameter (m)</td>
<td>0.02</td>
<td>0.005</td>
<td>0.03</td>
</tr>
<tr>
<td>Coil diameter (m)</td>
<td>0.154</td>
<td>0.05</td>
<td>0.2</td>
</tr>
<tr>
<td>Pitch</td>
<td>0.05</td>
<td>0.04</td>
<td>0.1</td>
</tr>
<tr>
<td>Linear coil spring stiffness (N/mm)</td>
<td>73.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spring bending stiffness (N-m/deg)</td>
<td>59.3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Acknowledgment

References


Management: Special Issue on Design and Marketing in New Product Development

*AIAA Journal*

Discrete Choice Analysis

*Systems Engineering: An Approach to Information-Based Design*

*Journal of Mechanical Design*

2004 *ASME Design Engineering Technical Conference*

Concurrent Engineering: Research and Applications

*Advances in Design Automation*

*Engineering Optimization*

*AIAA Journal*

Engineering

*Annals of Discrete Mathematics*

*Engineering Chemistry Research*

Disjunctive approaches for solving general mixed-integer linear programs

*Proceedings of the 12th International Conference on Engineering Design*

Target Cascading in Optimal System Design

*Transactions of ASME: Journal of Mechanical Design*
Transactions of ASME: Journal of Mechanical Design

Collaborative optimization: An architecture for large-scale distributed design

Structural Optimization


Journal of Marketing
Handbook of Transport Modeling

Transactions of ASME:

Journal of Mechanical Design

Proceedings of IFORS

2005

Proceedings of 6th World Congress of Structural and Multidisciplinary Optimization

Ward’s automotive yearbook
Stata Programming Reference Manual