Product platform design and customization:
Status and promise

TIMOTHY W. SIMPSON
Departments of Mechanical and Nuclear and Industrial and Manufacturing Engineering, Pennsylvania State University, University Park, Pennsylvania 16802, USA
(Received February 13, 2003; Accepted August 27, 2003)

Abstract
In an effort to improve customization for today’s highly competitive global marketplace, many companies are utilizing product families and platform-based product development to increase variety, shorten lead times, and reduce costs. The key to a successful product family is the product platform from which it is derived either by adding, removing, or substituting one or more modules to the platform or by scaling the platform in one or more dimensions to target specific market niches. This nascent field of engineering design has matured rapidly in the past decade, and this paper provides a comprehensive review of the flurry of research activity that has occurred during that time to facilitate product family design and platform-based product development for mass customization. Techniques for identifying platform leveraging strategies within a product family are reviewed along with metrics for assessing the effectiveness of product platforms and product families. Special emphasis is placed on optimization approaches and artificial intelligence techniques to assist in the process of product family design and platform-based product development. Web-based systems for product platform customization are also discussed. Examples from both industry and academia are presented throughout the paper to highlight the benefits of product families and product platforms. The paper concludes with a discussion of potential areas of research to help bridge the gap between planning and managing families of products and designing and manufacturing them.

Keywords: Mass Customization; Product Family; Product Platform; Product Variety

1. INTRODUCTION
Today’s highly competitive global marketplace is redefining the way many companies do business. The new form of competitive advantage is mass customization, and is, as Pine (1993a, p. xiii) says, “a new way of viewing business competition, one that makes the identification and fulfillment of the wants and needs of individual customers paramount without sacrificing efficiency, effectiveness, and low costs.” In his seminal text on mass customization, Pine (1993a, p. 6) argues that “customers can no longer be lumped together in a huge homogeneous market, but are individuals whose individual wants and needs can be ascertained and fulfilled.” He attributes the increasing attention on product variety and customer demand to the saturation of the market and the need to improve customer satisfaction: new products must be different from what is already in the market and must meet customer needs more completely. Sanderson and Uzumeri (1997, p. 3) add that “the emergence of global markets has fundamentally altered competition as many firms have known it” with the resulting market dynamics “forcing the compression of product development times and expansion of product variety.” Findings from studies of the automotive industry (Womack et al., 1990; MacDuffie et al., 1996; Alford et al., 2000) and empirical surveys of manufacturing firms (Chinnaiah et al., 1998; Duray et al., 2000) confirm these trends. Similar themes pervade the text by Wortmann et al. (1997), who examine industry’s response in Europe to the “customer-driven” market.

Because many companies typically design new products one at a time, Meyer and Lehnerd (1997, p. 2) have found that the focus on individual customers and products results in “a failure to embrace commonality, compatibility, standardization, or modularization among different products or product lines.” Mather (1995, p. 378) finds that “rarely does
the full spectrum of product offerings get reviewed at one
time to ensure it is optimal for the business.” The end result
is a “mushrooming” or diversification of products and parts
that can overwhelm customers (Stalk & Webber, 1993;
Mather, 1995; Huffman & Kahn, 1998). Nissan, for exam-
ple, reportedly had 87 different varieties of steering wheels
for one of their cars (Chandler & Williams, 1993). Although
offering a wide variety of products has both positive and
negative effects (cf., Galsworth, 1994; Anderson & Pine,
1997; Ho & Tang, 1998), the proliferation of product vari-
ety can incur substantial costs within a company (Lan-
caster, 1990; Child et al., 1991; Ishii, Juengel, & Eubanks,
(1997, p. 3), “is to understand and fulfill each individual
customer’s increasingly diverse wants and needs—while
meeting the coequal imperative for achieving low cost.”

Many companies are using product families and platform-
based product development to provide sufficient variety for
the market while maintaining economies of scale and scope
within their manufacturing processes. In general terms, a
product family is a group of related products that is derived
from a product platform to satisfy a variety of market niches.
As Robertson and Ulrich (1998, p. 20) point out, “by shar-
ing components and production processes across a platform
of products, companies can develop differentiated products
efficiently, increase the flexibility and responsiveness of
their manufacturing processes, and take market share away
from competitors that develop only one product at a time.”

Platform-based product development offers a multitude
of benefits including reduced development time and system
complexity, reduced development and production costs, and
improved ability to upgrade products. Platforms also pro-
 mote better learning across products and can reduce testing
and certification of complex products such as aircraft (Sab-
bagh, 1996), spacecraft (Caffrey et al., 2002c), and aircraft
engines (Rothwell & Gardiner, 1999). In the automotive
industry, platforms enable greater flexibility between plants
and can increase plant usage (sharing underbodies between
models can yield a 50% reduction in capital investment,
especially in welding equipment) and can reduce product
lead times by as much as 30% (Muffatto, 1999). Firms
using a platform-based product development approach in
the automotive industry recently gained a 5.1% market share
per year whereas firms that did not lost 2.2% (Cusumano &

Methods for platform-based product development have
progressed remarkably in the past decade, and this paper
provides a comprehensive review of the flurry of research
activity that has occurred during that time to facilitate prod-
uct family design and platform-based product development
for mass customization. In the next section, definitions and
examples of product families and product platforms are given
to set the stage for the discussions that follow. Sections 3
and 4 describe strategies for leveraging product platforms
across different market segments and metrics for assessing
product platforms, respectively. Sections 5 and 6 review
optimization approaches and artificial intelligence (AI) tech-
niques, respectively, for designing and configuring families
of products. The relationship between product platforms
and mass customization is examined further in Section 7,
and Web-based systems for product platform customization
are also discussed. Finally, closing remarks and avenues of
future research are outlined in Section 8.

2. PRODUCT FAMILIES AND PLATFORMS:
DEFINITIONS AND EXAMPLES

There are two basic approaches to product family design
(Simpson, Maier, & Mistree, 2001). The first is a top-down
(proactive platform) approach wherein a company strategi-
cally manages and develops a family of products based on
a product platform and its derivatives. For instance, Sony has
strategically managed the development of its Walkman®
products using carefully designed product platforms and
derivatives (Sanderson & Uzumeri, 1997). Similarly,
Kodak’s product platform-based response to Fuji’s intro-
duction of the QuickSnap® single-use camera in 1987 enabled
them to develop products faster and more cheaply, allowing
them to regain market share and eventually overtake Fuji
(Wheelwright & Clark, 1995). The second is a bottom-up
(reactive redesign) approach, wherein a company redesigns
or consolidates a group of distinct products to standardize
components to improve economies of scale. For example,
after working with individual customers to develop 100+
lighting control products, Lutron redesigns its product line
around 15–20 standard components that can be configured
into the same 100+ models from which customers could
initially choose (Pessina & Renner, 1998). Black & Decker
(Lehnerd, 1987) and John Deere (Shirley, 1990) have ben-
efited from similar redesign efforts to reduce variety in
their motor and valve lines, respectively.

The key to success in either approach is the product plat-
from from which the product family is derived. A product
platform can be either narrowly or broadly defined as

- “a set of common components, modules, or parts from
  which a stream of derivative products can be effi-
  ciently developed and launched” (Meyer & Lehnerd,
  1997, p. 7);
- “a collection of the common elements, especially the
  underlying core technology, implemented across a range
  of products” (McGrath, 1995, p. 39); and
- “the collection of assets [i.e., components, processes,
  knowledge, people and relationships] that are shared
  by a set of products” (Robertson & Ulrich, 1998, p. 20).

As an example, a platform at Volkswagen consists of the
floor group, drive system, and running gear, along with the
unseen part of the cockpit as shown in Figure 1. This plat-
form is shared across several models as well as all of its
brands (i.e., Volkswagen, Audi, Seat, and Skoda). According
to Bremmer (1999), Volkswagen owned three of the six auto-
motive platforms that successfully achieved production volumes over 1 million in 1999. The number of million-unit platforms is expected to reach 16 by 2004, with Volkswagen leading the way with its A04 and A4/A5 platforms.

The prominent approach to platform-based product development, be it top-down or bottom-up, is through the development of a module-based product family, wherein product family members are instantiated by adding, substituting, and/or removing one or more functional modules from the platform. An alternative approach is through the development of a scale-based product family, wherein one or more scaling variables are used to “stretch” or “shrink” the platform in one or more dimensions to satisfy a variety of market niches. Examples and methods for module-based product family design are described in the next section, followed by examples of scale-based product families in Section 2.2.

2.1. Module-based product families

There are numerous examples of module-based product families in the literature; some of the more frequently quoted examples follow.

- **Sony** builds all of its Walkmans around key modules and platforms and uses modular design and flexible manufacturing to produce a variety of quality products at low cost, which allowed them to introduce 250+ models in the United States in the 1980s (Sanderson & Uzumeri, 1997).
- **Nippondenso Co. Ltd.** makes an array of automotive components for a variety of automotive manufacturers using a combinatoric strategy that involves several different modules with standardized interfaces; for instance, 288 different types of panel meters can be assembled from 17 standardized subassemblies (Whitney, 1993).
- **Hewlett Packard** successfully developed several of their ink jet and laser jet printers around modular components to gain the benefits of postponing the point of differentiation in their manufacturing and assembly processes (Feitzinger & Lee, 1997).
- **Bally Engineering Structures** offers an almost infinite variety of environmentally controlled structures that can be readily assembled from one basic modular component—the pre-engineered panel—that can be produced in a variety of shapes and sizes and customized with options, attachments, and finishes to fit into any size structure (Pine, 1993).

These successful examples resulted from careful attention to customer needs and the underlying product architecture in the family. Ulrich (1995, p. 420) defines the product architecture as “(1) the arrangement of functional elements; (2) the mapping from functional elements to physical components; (3) the specification of the interfaces among interacting physical components.” A product architecture is classified as either modular, if there is a one to one or many to one mapping of functional elements to physical structures, or integral, if a complex or coupled mapping of functional elements to physical structures and/or interfaces exists. For example, personal computers (PCs) are highly modular. Baldwin and Clark (2000) trace the development of the IBM’s System/360, the first modular computer family. Automotive architectures, on the other hand, are predominantly integral (cf. Siddique et al., 1998; Muffatto, 1999), but modularity has become a major strategic focus for future
product development within many automotive companies (Kobe, 1997; Shimokawa et al., 1997; Cusumano & Nobeoka, 1998). For instance, Volkswagen’s Golf II comprises several modules to facilitate assembly (see Fig. 2a), and the rolling chassis module produced by the Dana Corporation (Fig. 2b) saved DaimlerChrysler nearly $700 million when developing their new Dodge Dakota facility (Kimberly, 1999). The rolling chassis module consists of brake, fuel, steering, and exhaust systems; suspension; and drive-line assembled to the frame. It is the largest, most complex module provided by a supplier, accounting for 25% of the vehicle content.

Modularity is an important topic in many product design textbooks (see, e.g., Pahl & Beitz, 1996; Ulrich & Eppinger, 2000; Otto & Wood, 2001) and is the sole focus in several texts (Ericsson & Erixon, 1999; O’Grady, 1999; Baldwin & Clark, 2000). Approaches for developing modular product architectures and module-based product families abound in the engineering design literature. For instance, Mattsson and Magleby (2001) discuss concept selection techniques for managing modular product development in the early stages of design. Wood and his coauthors (McAdams et al., 1999; Stone et al., 2000b; McAdams & Wood, 2002) present a methodology for representing a functional model of a product in a quantitative manner to assist in developing product architectures and facilitate the identification of a core set of modules for a product family. As part of their work, Stone et al. (2000a) present a heuristic method to identify modules for these product architectures; heuristics to identify functional and variational modules within a product family are introduced by Zamirowski and Otto (1999). Their work is foundational to the methods for developing modular product architectures developed by Otto and his coauthors (Dahmus et al., 2001; Otto, 2001; Sudijanto & Otto, 2001).

A method for incorporating customer demand into the development of the modular product architecture is discussed in Yu et al. (1999), and a method for assessing value in a module-based product family using real options concepts in the presence of uncertainty has also been developed (Gonzalez–Zugasti et al., 2001). Sundgren (1999) proposes a method for managing interfaces between modules within a product family after studying several product family development projects in the Swedish manufacturing industry over a period of 3 years; a method for developing robust interfaces for modular products is also introduced in Blackenfelt and Sellgren (2000). Comparisons of methods for modularizing product architectures can be found in Guo and Gershenson (2003) and Holtta and Salonen (2003), and researchers have investigated modular design approaches specifically for electronic products (Tseng & Jiao, 1997b), digital circuits (Kusiak & Huang, 1997), and mechatronics products (Huang & Kusiak, 1999). Finally, Schilling (2000) is developing a general theory of modular systems based on causal models developed from studying systems research in many engineering and nonengineering disciplines.

Modularity also plays a key role in product evolution, upgradeability, and retirement (Ishii, Lee, & Eubanks, 1995). Zhang et al. (2001) study the impact of modularity on product retirement costs, and life cycle cost issues associated with modular product architectures are discussed in Ulrich (1995), Riitahuta and Andreasen (1999), and Dahmus and Otto (2001). Meanwhile, Newcomb et al. (1998) present a decomposition algorithm to partition architectures into modules based on different life cycle viewpoints, whereas both Coulter et al. (1998) and Umeda et al. (1999) have proposed methodologies for designing modules for evolving families of products subject to life cycle concerns. The impact of modularity on component reuse is discussed in Kimura et al. (2001), and Allen and Carlson–Skalak (1998) develop a methodology for designing modular products that involves identifying and reusing modules from previous generations of products. Similarly, Martin and Ishii (2002) consider multiple generations of products when presenting their approach for designing modular product platform architectures. Their approach is one of several that uses Quality Function Deployment (QFD) to help identify modules within a product family (Cohen, 1995; Erixon, 1996; Ericsson & Erixon, 1999; Sand et al., 2002). Erixon (1996) has extended QFD into Modular Function Deployment™, a five-step process that utilizes the Module Identification Matrix™ to help

![Fig. 2.](https://www.cambridge.org/core)
generate module concepts (see also, Ericsson & Erixon, 1999; Huang and Kusiak, 1998) introduce a similar modularity matrix. Techniques for clustering modules based on functional requirements using design structure matrices are discussed in Suh (1990), Pimmler and Eppinger (1994), Blackenfelt (2000a) Stake and Blackenfelt (2000), Kusiak (2002), and Sharman et al. (2002). Optimization-based approaches and AI techniques for configuring and sizing modules are discussed in Sections 5 and 6, respectively.

2.2. Scale-based product families

As stated previously, scale-based product families are developed by scaling one or more variables to “stretch” or “shrink” the platform and create products whose performance varies accordingly to satisfy a variety of market niches. Although some consider scale-based product families to be a subset of module-based product families (see, e.g., Fujita & Yoshida, 2001), platform scaling is a common strategy employed in many industries. For example:

- **Honda** developed an automobile platform that can be stretched in both width and length to realize a “world car,” which was developed after failing to satisfy the Japanese and American markets with a single platform (Naughton et al., 1997).
- **Boeing** developed many of its commercial airplanes by “stretching” the aircraft to accommodate more passengers, carry more cargo, or increase flight range (Sabbagh, 1996).
- **Rolls Royce** scaled its RTM322 aircraft engine by a factor of 1.8, as shown in Figure 3, to realize a family of engines with different shaft horsepower and thrust (Rothwell & Gardiner, 1990).

A frequently quoted example of a successful scale-based product platform is Black & Decker’s universal electric motor. According to Lehnerd (1987), in the 1970s, Black & Decker developed a family of universal motors for its power tools in response to a new safety regulation: double insulation. Prior to that, they used different motors in each of their 122 basic tools with hundreds of variations. Through redesign and standardization of the product line, they were able to produce all of their power tools using a line of motors that varied only in the stack length and the amount of copper wrapped within the motors. As a result, all of the motors could be produced on a single machine with stack lengths varying from 0.8 in to 1.75 in, and power output ranging from 60 to 650 W. By paying attention to standardization and exploiting platform scaling around the motor stack length, material costs dropped from $0.77 to $0.42 per motor while labor costs fell from $0.248 to $0.045 per motor, yielding an annual savings of $1.82M per year. Tool costs decreased by as much as 62%, boosting sales, increasing production volumes, and further improving savings. Furthermore, new designs were developed using standardized components such as the redesigned motor, which allowed products to be introduced, exploited and retired with minimal expense related to product development. This electric motor example has served as a test problem for several optimization-based approaches for product family design as noted in Section 5. Meanwhile, the idea of sharing components across different market segments (e.g., power tools, lawn tools) leads us to the topic of platform leveraging.

3. PLATFORM LEVERAGING STRATEGIES

Regardless of whether the platform is modular or scalable, the basic development strategy within any product family is to leverage the product platform across multiple market segments or niches. Early attempts at mapping the evolution of a product family based on extensions and upgrades to a product platform can be found in Wheelwright and Sasser (1989) and Meyer and Utterback (1993), but it was not until Meyer (1997) introduced the market segmentation
grid that platform leveraging strategies were clearly articulated. As shown in Figure 4, market segments are plotted horizontally in the grid while price/performance tiers are plotted vertically; each intersection of a market segment with a price/performance tier constitutes a market niche that is served by one or more of a company’s products. Three platform leveraging strategies can be identified within the grid as shown in Figure 4: horizontal leveraging, vertical leveraging, and the beachhead approach, which combines both. Meyer and Lehnerd (1997) discuss the advantages and drawbacks of each leveraging approach, and examples of market segmentation grids can be found in Caffrey et al. (2002b) for spacecraft and avionics systems and in Meyer and Lehnerd (1997) for computers, data storage systems, power tools, and office furniture.

The market segmentation grid is useful for both platform development (i.e., as part of a top-down approach to product family design), as well as product family consolidation (i.e., as part of a bottom-up approach). For instance, Farrell and Simpson (2003) use the market segmentation grid to identify potential platform leveraging strategies for a line of flow control valves using historical sales data. Although most horizontal leveraging strategies take advantage of modular platforms, the relationship between vertical leveraging strategies and scalable platforms is discussed in (Simpson, Maier, & Mistree, 2001). Finally, Meyer describes adaptations of the market segmentation grid for platform-based development approaches to nonassembled products (Meyer & Dalal, 2002) and the design and renewal of services (Meyer & DeTore, 2001). Metrics for measuring the success of platforms and platform leveraging strategies are discussed next.

4. METRICS FOR PRODUCT PLATFORMS AND PRODUCT FAMILIES

An important measure of success of a product platform is how quickly and cheaply new products can be developed from it. To help determine when to renew or refocus product platform efforts, Meyer et al. (1997) introduced metrics for platform efficiency and effectiveness. Platform efficiency assesses how much it costs to develop derivative products relative to how much it costs to develop the product platform within the product family. Platform effectiveness measures the ratio of the revenue a product platform and its derivatives create to the cost required to develop them. A similar approach is taken in Schellhammer and Karandikar (2001), wherein a project ranking index, which combines an investment index and a revenue index, is introduced to assist in project planning. Their approach extends the aggregate project planning concepts of Wheelwright and Clark (1992) for managing platform projects, derivative projects, breakthrough projects, advanced research and development projects, and partnership projects and alliances.

During product platform design, much of the focus revolves around the trade-off between commonality and distinctiveness: designers must balance the commonality of the products in the family with the individual performance (i.e., distinctiveness) of each product in the family. For instance, Airbus has enjoyed a competitive advantage over Boeing due to improved commonality, particularly in the cockpit. The A330 cockpit is common to all other Airbus types while Boeing’s 767-400 cockpit is common only with the 757. This has enabled the A330-200, a less efficient “shrink” of a larger aircraft, to outsell Boeing’s 767-400ER, a more efficient “stretch” design of a smaller aircraft (Aboulafia, 2000). Commonality can also adversely impact a company’s reputation: in the late 1980s, engineers at Chrysler were accused of having “fallen asleep at the typewriter with our finger stuck on the K key” (Lutz, 1998, p. 17) due to overusage of the K-car platform and lack of distinctive new products.

Despite the potential drawbacks of commonality, numerous indices have been developed to measure commonality in the management science and operations research community (see, e.g., Rutenberg, 1969; Collier, 1981; Baker et al., 1986; Trelevan & Wacker, 1987; Thomas, 1992; Lee & Billington, 1994; McDermott & Stock, 1994; Vakharia et al., 1996; Kim & Chhajed, 2000). The degree of commonality index proposed by Collier (1981) was one of the first such indices that uses information contained in the company’s bills of materials to assess commonality for a single end item, a product family, or an entire product line. Jiao and Tseng (2000) extend Collier’s commonality index to create indices for component part commonality and process com-

![Fig. 4. Platform leveraging strategies. Adapted from Meyer (1997).](https://www.cambridge.org/core)
monality, overcoming the limitations of his index (cf. Wacker & Treleven, 1986). Siddique et al. (1998) propose separate indices for measuring component commonality and connection commonality, applying them to automotive underbodies, which are predominantly integral architectures. Finally, Kota and coauthors (2000) introduce a product line commonality index to capture the level of commonality within a product family based on size and shape, materials and manufacturing processes, and assembly and fastening schemes. Comparisons of these various commonality metrics are lacking in the literature.

Martin and Ishii (1996, 1997) also introduced a commonality index similar to Collier’s, along with indices for measuring setup costs and the point of product differentiation, which correlate with many of the indirect costs of providing variety. Martin and Ishii (2002) most recently proposed a generational variety index to help identify which components are likely to change over time to meet future market requirements and a coupling index to measure the coupling between these components. The importance of minimizing the coupling in a product architecture has been studied extensively by Suh (1990). A functional similarity index was introduced by McAdams et al. (1999; McAdams & Wood, 2002) to assist in concept development and modular product design. Finally, indices for measuring the degree of variation within a scale-based product family have also been proposed (Simpson, Seepersad, & Mistree, 2001; Nayak et al., 2002; Messac et al., 2002b); these indices are useful for product family optimization as discussed next.

5. OPTIMIZATION-BASED APPROACHES FOR PRODUCT FAMILY AND PRODUCT PLATFORM DESIGN

Several optimization approaches have been developed within the engineering design community to help determine the best design variable settings for the product platform and individual products within the family; a summary of these approaches is given in Table 1. In looking at the table, the approaches are split evenly between module-based and scale-based product families, while the work by Fujita and Yoshida (2001) specifically targets both. Almost two-thirds of the approaches require specifying the platform a priori to the optimization to reduce the design space and make the problem more tractable. This is not ideal, however, because designers would like to use optimization to explore varying levels of platform commonality to help identify which variables to make common and unique within the family (cf. Simpson & D’Souza, 2002). More than half of the approaches use multiobjective optimization to accomplish this. Three assumptions are often made when using multiobjective optimization to design a product family:

1. maximizing each product’s performance maximizes its demand,
2. maximizing commonality among products minimizes production costs, and
3. resolving the trade-off between assumptions 1 and 2 yields the most profitable product family.

However, without explicitly modeling the market demand for the products in the family and their associated manufacturing costs, these assumptions may lead to suboptimal product families. The universal electric motor example from Lehnerd (1987) employed in Simpson, Maier, and Mistree (2001), Messac et al. (2002b), and Nayak et al. (2002) provides a realistic case of when this can occur. The objective is to design a family of 10 motors based on a scalable platform. The initial formulation scaled the motors around the stack length of the motor (Simpson, Maier, & Mistree, 2001), but maximizing commonality in the family using two different approaches revealed that the motor platform should be scaled by the radius to maximize performance (Messac et al., 2002b; Nayak et al., 2002). According to Lehnerd (1987), the best choice is stack length, and through discussions with experienced motor designers, production costs, not performance, drive the use of stack length as the scaling variable (Simpson, Maier, & Mistree, 2001). In the table, note that only about half the approaches integrate manufacturing costs directly within the formulation while less than one-third incorporate market demand/sales. Also, note that the majority of approaches that include costs or sales in their formulation use single objective optimization, rather than multiobjective, where the objective is to either maximize profit or minimize cost.

Although not specifically noted in the table, most of the approaches that incorporate uncertainty in the formulation model it in the market demand and future sales of the products in the family (Seepersad et al., 2000; Gonzalez–Zugasti et al., 2001; Jiang & Allada, 2001; Allada & Jiang, 2002; Li & Azarm, 2002). Uncertainty in customer requirements has also been used to develop robust product platforms. Chang and Ward (1995) were among the first to use robust design techniques to develop a family of products that were insensitive to design changes. Simpson and coauthors use robust design techniques to develop scale-based platforms for General Aviation Aircraft (Simpson et al., 1999), electric motors (Simpson, Maier, & Mistree, 2001), and absorption chillers (Hernandez et al., 2001). Blackenfelt (2000b) uses robust design techniques to maximize profit and balance commonality and variety within a family of lift tables.

The number of stages in the optimization approach is another interesting statistic. Single-stage approaches seek to optimize the product platform and corresponding family of products simultaneously while two-stage approaches optimize the platform first and then instantiate the individual products within the family during the second stage; multi-stage approaches are those that involve more than two stages. Single-stage and two-stage approaches are employed almost equally in the literature, and the reader is referred to the
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Allada &amp; Jiang, 2002</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>DP</td>
<td>Generic modular products (3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Blackenfelt, 2006</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>OA</td>
<td>Lift tables (4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cetin &amp; Saitou, 2003</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>OA</td>
<td>Welded automotive structures (2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chang &amp; Ward, 1995</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>OA</td>
<td>Automatic A/C units (6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D’Souza &amp; Simpson, 2003</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>OA</td>
<td>General Aviation Aircraft (3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Farrell &amp; Simpson, 2003</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>GRG</td>
<td>Flow control valves (16)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fellini et al., 2000</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>B&amp;B</td>
<td>Automotive powertrain (3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fellini, Kokkolaras, Michelena, et al., 2002</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Pressure vessels (16)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fellini, Kokkolaras, Papalambros, et al., 2002</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Pressure vessels (16)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fujita et al., 1998</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>B&amp;B</td>
<td>TV receiver circuits (6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fujita et al., 1999</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>TV receiver circuits (6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fujita &amp; Yoshida, 2001</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>TV receiver circuits (6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gonzalez–Zugasti &amp; Otto, 2000</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>B&amp;B</td>
<td>Interplanetary spacecraft (3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gonzalez–Zugasti et al., 2000</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>B&amp;B</td>
<td>Interplanetary spacecraft (3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gonzalez–Zugasti et al., 2001</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>B&amp;B</td>
<td>Interplanetary spacecraft (3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hernandez et al., 2001</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>B&amp;B</td>
<td>Absorption chillers (8)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hernandez et al., 2002</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hernandez et al., 2003</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jiang &amp; Allada, 2001</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kokkolaras et al., 2002</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Li &amp; Azarm, 2002</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Messac et al., 2002b</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Messac et al., 2002a</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nayak et al., 2002</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nelson et al., 2001</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ortega et al., 1999</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rai &amp; Allada, 2002</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seepersad et al., 2000</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seepersad et al., 2002</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simpson et al., 1999</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simpson, Maier, &amp; Mistree, 2001</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>Y</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simpson &amp; D’Souza, 2002</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>ExS</td>
<td>Universal electric motor (10)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: SLP, sequential linear programming; SQP, sequential quadratic programming; NLP, nonlinear programming; GA, genetic algorithm; SA, simulated annealing; DP, dynamic programming; OA, orthogonal array; GRG, generalized reduced gradient; B&B, branch and bound; PatS, pattern search; ExS, exhaustive search.
table for examples. Although both approaches are effective at determining the best design variable settings for the product platform and product family, single-stage approaches will yield the best overall performance of the product family because the optimization is not partitioned into two or more stages (cf. Messac et al., 2002a). The dimensionality of single-stage optimization problems, however, is considerably higher than in two-stage approaches, which often leads to many computational challenges (cf. Messac et al., 2002b). It is also worth noting that a modification to the two-stage approach is introduced by Nelson et al. (2001) and used by Fellini et al. (2000), Fellini, Kokkolaras, Michelena, et al. (2002), and Fellini, Kokkolaras, Papalambros, et al. (2002): the first stage involves individually optimizing each product while the second stage involves optimizing the product family with constraints on performance losses due to commonality. Meanwhile, only two multistage approaches have been developed. First, Hernandez et al. (2002, 2003) develop a multistage optimization approach by viewing the product platform design problem as a problem of access in a geometric space. Second, Allada and Jiang (2002) introduce a dynamic programming (DP) model for configuring module instances within an evolving family of products. Their DP-based approach is used to plan module introduction (i.e., which modules to introduce when) to maximize the total profit in a given planning horizon. An alternative classification of optimization approaches based on the extent of the optimization (i.e., module attributes, module combinations, or both) is discussed in Fujita (2002).

Based on the variety of optimization algorithms listed in the table, there does not appear to be a preferred algorithm for product family design. Linear and nonlinear programming algorithms (e.g., sequential linear programming, sequential quadratic programming, nonlinear programming, generalized reduced gradient) are employed by many researchers, as are derivative-free methods such as genetic algorithms (GAs), simulated annealing, pattern search, and branch and bound. When the design space is small enough, exhaustive search techniques (Hernandez et al., 2003) or orthogonal arrays (Chang & Ward, 1995; Blackenfelt, 2000b) can be used to enumerate different combinations of parameter settings and modules. However, very few problems involve so few options that such an approach can be taken, and many researchers advocate the use of GAs for product platform design due to the combinatorial nature of the product family design problems (Gonzalez–Zugasti & Otto, 2000; Fujita & Yoshida, 2001; Li & Azarm, 2002; D’Souza & Simpson, 2003). Finally, algorithm choice is often mandated by the selected framework, for example, Decision-Based Design (DBD, Li & Azarm, 2002), Target Cascading (Kokkolaras et al., 2002), 0–1 integer programming (Fujita et al., 1999), Physical Programming (Messac et al., 2002a), and the Compromise Decision Support Problem (Simpson et al., 1999).

Finally, these optimization approaches have been tested on a variety of product families as noted in the last column of the table. These product families range from 2 to 16 products, and include consumer products such as drills (see, e.g., Li & Azarm, 2002), vacuum cleaners (Jiang & Allada, 2001) and automobiles (Fellini, Kokkolaras, Michelena, et al., 2002; Kokkolaras et al., 2002); industrial products such as chillers (Hernandez et al., 2001) and flow control valves (Farrell & Simpson, 2003); and complex systems such as aircraft (see, e.g., Fujita & Yoshida, 2001; Simpson & D’Souza, 2002) and spacecraft (see, e.g., Gonzalez–Zugasti et al., 2000). Detailed analyses for the universal electric motor problem can be found in Simpson, Maier, and Mistree, (2001); it has been used to benchmark a variety of optimization approaches, as noted in the table. The commercial aircraft problem found in Fujita et al. (1998) and Fujita and Yoshida (2001) uses aircraft analyses available in the literature in combination with their own models for design and development, facility, and production costs and a profit model for the manufacturer. The nail gun (Nelson et al., 2001), vacuum cleaner (Jiang & Allada, 2001), and power screwdriver and electric knife (Rai & Allada, 2002) examples are comprehensive as well. The automotive example used in Fellini, Kokkolaras, Michelena, et al. (2002) and Kokkolaras et al. (2002) is based on a detailed vehicle body structural model that is currently unavailable to the public; simpler models of the automotive vehicle frame can be in Fellini, Kokkolaras, Papalambros, et al. (2002) and Cetin and Saitou (2003). The analyses for the absorption chiller problem (Seepersad et al., 2000; Hernandez et al., 2001) are not publicly available either.

6. AI IN PRODUCT PLATFORM DESIGN AND CUSTOMIZATION

AI techniques for product platform design and customization lag behind optimization-based approaches; however, they have been successfully employed and shown great promise for automatic product configuration and automatic computer-aided design (CAD) modeling and geometry generation. Sabin and Weigel (1998) recently reviewed rule-based and model-based techniques for automatic product configuration; they state that the acquisition of the rules or constraints on which the reasoning depends is one of the major challenges in knowledge-based configuration systems. One such system is the Product Module Reasoning System developed by Rosen (1996), which reasons about sets of product architectures, translates design requirements into constraints on these sets, compares architecture modules from different viewpoints (e.g., material, connections, covers), and directly enumerates all feasible module combinations. His approach uses discrete mathematics (combinatorics and set theory) and provides the foundation for the Product Family Reasoning System developed by Siddique and Rosen (2000, 2001), which reasons about families of products in addition to individual product architectures. A configuration framework for mass customization of products that employs the Unified Modeling Language is
introduced by Felfernig, Friedrich, and Jannach (2001) and Felfernig, Friedrich, et al. (2001). Claesson et al. (2001) use function–means–trees and a chromosome model to create configurable components that represent a parameterized set of design solutions; the concept is currently being deployed at Saab automobile to help control product variety. Finally, agent-based systems offer many advantages for concurrent product and process design and configuration (Shen et al., 2001), and agent-based approaches for module-based product family design are being explored by some researchers (Allada & Rai, 2002; Liang & Huang, 2002; Rai & Allada, 2002). However, these approaches are not yet as mature as the aforementioned optimization-based approaches.

Sabin and Weigel (1998) also discuss several case-based reasoning techniques for automatic product configuration. Such techniques generally involve eliciting customer requirements, retrieving a configuration from a pool of stored cases, and adapting the case to satisfy the new situation. A case-based approach for mass customization of goods was proposed by Tseng and Jiao (1997a). Their approach organizes information around the common features and platform within a product family architecture to facilitate case retrieval; adaptation of cases occurs either by reinstantiating the case if it is similar enough or by providing specific adaptation knowledge (configurational or topological) to modify aspects of the case. In Tseng and Jiao (1997c), a two-phase methodology is presented for recognizing patterns of functional requirements in similar existing products to help define new products.

To reduce data redundancy when modeling families of products, the Generic Bill-of-Material (GBOM) concept developed at the Eindhoven University of Technology (Hegge & Wortmann, 1991; Erens et al., 1992; van Veen, 1992; Erens & Hegge, 1994) allows all variants of a product family to be specified only once. Within a GBOM, the Primary Generic Product (PGP) represents the set of all variants of a particular primary product and Generic Sub-assembly Products (GSPs) describe the sets of subassemblies. Parameter values of the PGP are passed through the levels of the GBOM and are inherited by lower level GSPs. McKay et al. (1996) combine the GBOM concept with product modeling concepts and software to reduce data redundancy when considering multiple views (e.g., sales, manufacturing, assembly). Jiao and coauthors (2000) have extended the GBOM concept to include operations, and De Lit et al. (2001) use GBOMs to support assembly planning for families of products. Cheng et al. (2002) extended the GBOM concept for product family development within an extended enterprise.

Finally, grammar-based approaches have been developed for automating the generation of products within a family as well as automating the generation of CAD models. Agarwal and Cagan (1997, 2000) first popularized the approach, developing shape grammars to create a variety of coffeemakers. Siddique and Rosen (1999) use the same coffeemaker example to demonstrate a graph grammar-based approach for product platform design and instantiation. Within their graph grammar, graphs are used to represent the core function and structure (i.e., the platform), and grammars are used to specify the relationships between the core and the options (i.e., product variants). Meanwhile, Du et al. (2001b) have also developed a graph grammar-based approach for modeling product families. They use a Programmed Attribute Graph Grammar to specify the design space of the product family, which is then customized by varying modules according to a control diagram that captures the complex relationships and configuration constraints between modules. Their approach is extended in Du et al. (2002) by implementing a graph rewriting process to transform product family graphs into product variant graphs; a family of office chairs is used to illustrate their approach.

Graph grammars are being used by Siddique and Shao (2001) to develop a Web-based system for product family reasoning and Siddique and Yanjiang (2002) describe a template-based approach that automatically generates CAD models for each member in the product family. The approach was expanded to include parametric design, mating relationships, and modularity (Martinez–Larrosa & Siddique, 2002), and is one of many Web-based approaches for customization that is discussed next.

7. PRODUCT PLATFORMS AND (WEB-BASED) CUSTOMIZATION

In addition to improving economies of scale and scope, a product platform can facilitate customization by enabling a variety of products to be quickly and easily developed to satisfy the needs and requirements of distinct market niches (Pine, 1993a). Although flooding the market with a variety of products derived from a platform may satisfy some customers by providing a substitute for customization, variety is not customization. Variety provides choices for customers but does not enable the customer to specify the product. A customized product, on the other hand, is designed to meet the specific needs of a particular customer; therefore, customers must be involved at one or more points in the product realization process for the product to be truly customized (see Fig. 5). This distinction is overlooked in much of the mass customization literature as noted by Duray et al. (2000), who study and classify companies that customize products based on the point of customer involvement. Customized products can be either made to order, tailored to order, assembled to order, or made to stock, each of which has different implications for product platform development and the associated information technologies needed to deliver that product (cf. Duray & Milligan, 1999). Regardless of the stage of customer involvement, product platforms play an integral role in facilitating the product customization process. Tseng, Jiao, and coauthors have extensively studied the relationship between product family architecture and mass customization (Tseng et al., 1996; Tseng...

Choi and Whinston (1999) assert that the “computer-mediated market will accelerate the process of customization through its technologies.” Efficient information technology for product customization is one of several important research directions facing many of today’s firms (Da Silveira et al., 2001; Zipkin, 2001). For instance, Web-based customization opens new paradigms for one to one marketing (Gilmore & Pine, 1999), and it can provide valuable information for companies to improve customer demand estimates and determine whether to contract or expand product variety (Kotha, 1995).

Some prototypical Web-based systems for platform customization have been developed. For example, Simpson et al. (2003) created a Web-based system for customizing refiner plates for pulp and paper processing based on pre-defined platforms; the system exploits a parametric and feature-based modeling scheme for refiner plates developed in earlier work (Kulvatunyou et al., 2000). Flores et al. (2002) present a Web-based system for customizing coated steel belt sheaves, extending early parametric modeling capability developed in their lab (Rohm et al., 2000). A Web-based system that incorporates fuzzy geometric customization and fuzzy reasoning is presented in Chen et al. (2001) and demonstrated for customizing wineglasses and furniture. A Web-based knowledge system to support product family design has been developed by Zha and Lu (2002) and tested with a power supply product family. Researchers now recognize that suppliers are playing an increasingly important role in new product development, particularly in the automotive industry (MacDuffie et al., 1996; Gupta & Krishnan, 1998a; Fisher et al., 1999), and Web-based systems are being developed to include suppliers in the design process (Huang, Huang, & Mak, 2000; Huang & Mak, 2000). Huang and his colleagues have also been developing systems to support collaborative product development and Design for X capabilities over the Internet (Huang & Mak, 1999; Huang, Ski, & Mak, 2000; Huang, Lee, & Mak, 2001; Huang, Shen, & Mak, 2001). Finally, commercial software developers are staking their claims in this rapidly growing area of research. For instance, Windchill® DynamicDesign-Link™ enables dynamic, collaborative, Web-based product customization for design to order products (Parametric Technologies Corporation, 2002). The software allows customers to create custom products via the Web by providing guided product selection and configuration, automated product and process selection and generation, and integration with enterprise business systems.

8. CLOSING REMARKS AND FUTURE RESEARCH DIRECTIONS

As evidenced by this comprehensive review of product family and product platform design research, there has been considerable progress in planning, modeling, designing, and assessing product platforms and the families of products derived from them. Be it a module-based or a scale-based product family, there are now a variety of optimization-based and AI-based techniques to support the design and decision making process. Although optimization-based approaches are more prevalent in the literature, AI-based techniques show great promise for product family design and customization. Web-based systems are also becoming more prominent to promote customer involvement in the product realization process and facilitate individual product customization via a well-defined product platform. Thanks to this flurry of research activity, this nascent field of study has matured rapidly in the past decade; however, considerable research is still needed to help bridge the gap between planning and managing families of products and designing and manufacturing them. Toward this end, several research thrusts are identified in the remainder of this section to help guide future efforts in this burgeoning field of research.

8.1. Product family planning and platform development

In an empirical study of 108 new product development projects, Tatikonda (1999) found that platform and deriva-
tive projects are executed in a similar manner and do not differ in terms of project success even though they do differ in task characteristics (i.e., complexity and amount of new technology introduced). This suggests that some firms can manage single product and platform projects in similar ways, but system-level designers must still “address the problem of what product architecture should be used to deliver the different products while sharing parts and production steps across the products” (Robertson & Ulrich, 1998, p. 21). Product family maps and the market segmentation grid support product family planning from a management perspective but offer little support to product and process engineers who must determine each product’s architecture and the appropriate levels of platform commonality and component sharing within the family (cf. Maier & Fadel, 2001). Information management systems to support product family development and foster component/module reuse are needed, and platform planning processes such as given in Robertson and Ulrich (1998) need to be developed, tested, and refined to help support engineering decision making during platform-based product development. Lessons learned from the set-based approaches employed successfully for years at Toyota (Ward et al., 1995) also show promise as models for product platform planning and development. Approaches for determining the extent and number of platforms to offer within a family (Seepersad et al., 2000, 2002; de Weck et al., 2003) also need to be further investigated.

8.2. Quantifying the benefits and drawbacks of platform-based product development

The metrics discussed in Section 4 provide surrogates for measuring the financial impact of platform development; however, quantifying the economic benefit of platform-based product development is important for strategic decision making (cf., Meyer et al., 1997; Schellhammer & Karandikar, 2001). Some preliminary work in this area includes the use of activity-based costing to estimate the cost savings (Siddique & Repphun, 2001) and reduction in development time (Siddique, 2001) for developing a hard disk drive spindle motor platform for a family of hard disks. Although many researchers espouse the benefits of platforms, platform-based approaches can impose additional costs on product development. The fixed costs of developing a product platform can be enormous, as evidenced by Ulrich and Eppinger (2000), who note that developing a product platform can cost 2–10 times more than a single product, and sharing components across low-end and high-end products can increase unit variable costs due to over-designed low-end products (Gupta & Krishnan, 1998a; Fisher et al., 1999). In the automotive industry, Muffato (1999) found that up to 80% of total vehicle development cost is spent on platform development (including engine and transmission); others argue that platform development accounts for only 60% of these costs (Sundgren, 1999). Krishnan and Gupta (2001) develop a mathematical model to examine some of the costs of platform-based product development and find that platforms are not appropriate for extreme levels of market diversity or high levels of non-platform scale economies.

8.3. Modeling customer demand for product families

As discussed in Section 7, incorporating customers in the product realization process is critical to successful platform customization, and recent work in DBD has revealed the importance of formulating a proper objective function to reflect the interests of both consumers and producers (Hazelrigg, 1996, 1998; Chen et al., 2000). Georgiopoulos et al. (2002) integrate engineering and business models for evaluating portfolio decisions involving a premium-compact and sport utility vehicle, and a formal framework for DBD was proposed by Hazelrigg (1996, 1998) and implemented for product design (Gu et al., 2000; Li & Azarm, 2000; Wassenaar & Chen, 2001) and product line design (Li & Azarm, 2002). Within their framework, Li and Azarm (2000, 2002) use conjoint analysis (see, e.g., Louviere, 1988; Green & Srinivasan, 1990) to estimate customer demand within their DBD framework. Tseng and Du (1998) have also used conjoint analysis techniques for soliciting customer input for mass customized products that are based on a product family. Conjoint analysis has been widely used in marketing and management science for product platform design (Moore et al., 1999), product line design (see, e.g., Green & Krieger, 1985; McBride & Zufryden, 1988; Kohli & Sunku- mas, 1990; Dobson & Kalish, 1993), and product line redesign (Page & Rosenbaum, 1987); however, Chen and Hausman (2000) note that many of these conjoint based approaches are mathematically intractable or NP-hard. Consequently, alternatives to conjoint analysis such as choice-based conjoint analysis (Azarm et al., 2003) and discrete choice analysis (Wassenaar & Chen, 2001) are being investigated for use with DBD, but these techniques have not yet been extended for product family design.

8.4. Design for manufacturing and assembly (DFMA)

Most DFMA techniques have been developed for single products (cf. Bralla, 1999; van Vliet et al., 1999; Shah & Wright, 2000; Boothroyd et al., 2002), and they do not effectively support product family design. For instance, creating modular product architectures runs counter to the DFMA principle of reducing part count by integrating parts, which often makes the architecture more integral; however, modularity can facilitate assembly. Conversely, increasing commonality will reduce the overall part count within a product family, and it will have the added benefit of delaying the point of product differentiation within the manufacturing and assembly process (see, e.g., Lee & Tang, 1997; Gupta...
& Krishnan, 1998b; He et al., 1998; Desai et al., 2001; Ma et al., 2002). Approaches for concurrent design of product families and assembly systems have been developed (Stadzisz & Henrioud, 1995; Stadzisz et al., 1995; Martinez et al., 2000; Fouda et al., 2001). Kusiak (2000) discusses the importance of modular design in developing agile manufacturing systems, and an approach for optimizing modules for reconfigurable manufacturing systems was recently introduced by Yigit et al. (2002). These DFMA guidelines need to be formalized to support product family design and platform-based product development.

8.5. Support for small- and medium-size manufacturers

Although much of the research in product family design has focused on large corporations (e.g., Volkswagen, Boeing, Kodak, HP), product platforms and customization are becoming particularly important for small- and medium-size firms. Small manufacturers lack an “adequately trained technical workforce” and do not have the “deep pockets” and “financial float” that is available in larger companies (Maupin & Stauffer, 2000). Developing product platforms that can be leveraged across a variety of products is critical to remaining profitable, and tools such as Variant Mode and Effects Analysis, which have been tested at 10 medium-to-large firms, are useful in reducing product complexity and unwanted product variety (Schuh & Tanner, 1998). Preliminary efforts to reengineer and improve commonality within a product family at small manufacturing firms can also be found in Maupin and Stauffer (2000), Berti et al. (2001), and Farrell and Simpson (2003).

8.6. Overcoming organizational barriers to platform-based product development

Designing a product platform and corresponding family of products embodies all of the challenges of product design while adding the complexity of coordinating the design of multiple products in an effort to increase commonality across the set of products without compromising their distinctiveness. For instance, Sanchez and Mahoney (1996) and Pederson (1999) discuss the impact of modularity and platforms, respectively, on the organization. Despite the benefits and advantages of platform-based product development, many companies remain hesitant to embrace product families and product platforms. For instance, despite the success and growth of the PC industry due to open, modular product architectures and product families (cf. Baldwin & Clark, 1997; Christensen & Verlinden, 2002), many aerospace organizations are resistant to such an approach because they fear losing their competitive edge in a relatively small market (cf. Caffrey et al., 2002a, 2002b). Erens (1997, p. 2) notes that “if sales engineers and designers focus on individual customer requirements, they feel that sharing components compromises the quality of their products.”

Dismantling organizational complexity is the first step in reducing the negative effects of product variety (cf. Galsworth, 1994), and innovative approaches are needed to overcome the corporate inertia that develops within many companies. As an example, Chrysler’s dramatic reorganization in 1988 around five platform teams (small car, large car, Jeep, truck, and minivan) with crossfunctional “tech clubs” (e.g., chassis, engine, HVAC, etc.) achieved significant reductions in development costs, lead time, and variable costs (Lutz, 1998). Organizational structures at other automotive companies (e.g., GM, Ford, Toyota, Volkswagen, Nissan, Fiat, Mazda, Honda) to promote platform development are documented and discussed in (Shimokawa et al., 1997; Cusumano & Nobeoka, 1998). Meanwhile, Kotha (1995) examines an innovative approach used by Japan’s National Bicycle Industrial Company for combining mass production and mass customization. Due to the impressive gains resulting from such innovations, we must strive to understand the organizational impact of platform-based product development better.

ACKNOWLEDGMENTS

This work was supported by National Science Foundation Career Grant DMI-0133923. Any opinions, findings, and conclusions or recommendations presented in this paper are those of the author and do not necessarily reflect the views of the National Science Foundation.

REFERENCES


Platform design and customization


Timothy W. Simpson is an Associate Professor of Mechanical and Industrial Engineering at Pennsylvania State University. He received a BS degree in mechanical engineering from Cornell University in 1994 and MS and PhD degrees in 1995 and 1998, respectively, in mechanical engineering from the Georgia Institute of Technology. Timothy is a member of ASME, AI/AA, and ASEE and is Director of the Product Realization Minor in the College of Engineering at Penn State. Dr. Simpson’s research and teaching interests include product family and product platform design, engineering optimization, and Web-based customization.