DESIGN DECISION TEMPLATES AND THEIR IMPLEMENTATION FOR DISTRIBUTED DESIGN AND SOLID FREEFORM FABRICATION

David Rosen, Yong Chen, Jonathan Gerhard, Janet Allen, Farrokh Mistree

The George W. Woodruff School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA  30332-0405
404-894-9668
david.rosen@me.gatech.edu

ABSTRACT

In this paper, we describe issues arising in distributed design and fabrication, where fabrication of prototypes is performed by Solid Freeform Fabrication (SFF) technologies. The focus is on a design-manufacture collaboration problem, namely the transfer of design information from the design organization to a manufacturing organization. In other research, this has been called the need for a “clean digital interface.” In the context of designing and fabricating a prototype, we explore two factors: the types of design and product information to be transferred from design to manufacturing, and which types of decisions can be made by the manufacturer regarding the prototype. As a representative product design process, we focus on three main activities, including material and SFF process selection, geometric tailoring (detailed design for manufacture modifications), and process planning. Further we investigate three interfaces between design and manufacturing organizations, one interface before each main activity. We demonstrate that a series of design decision templates can be used to transfer design information between the design and manufacturing organizations in support of the main activities. We propose design decision templates for transferring design information from the design organization to a manufacturing organization, in the context of fabricating a prototype using Solid Freeform Fabrication (SFF) technologies. In the area of SFF, the objective of a “Standard Interchange Format” (SIF) has been proposed as a “clean digital interface” between design and manufacturing, where SIF contains part geometry and possibly surface finish, tolerance, material, and perhaps other related information types. Although this information is a significant extension beyond STL, IGES, or even STEP file formats, it is still insufficient to support design for manufacturing or, perhaps more importantly, guidance in case the available manufacturing processes are incapable of meeting requirements.

The purpose of this paper is to present our approach to providing a clean digital interface. We propose design decision
templates for a series of decisions, of increasing scope, as an effective digital interface. Two broad types of decisions are supported, selection and compromise, along with several combinations of these. Our approach is to transfer enough design information to enable the manufacturer to explore alternative materials and fabrication processes in the event that design requirements cannot be achieved. Effectively, this transfers the burden of design for manufacturing to the manufacturer. For example, if the designer has specified a part strength requirement, but the available manufacturing technologies cannot process appropriate materials, then it becomes feasible for the manufacturer to evaluate the suitability of alternative materials if this additional information is available.

In Figure 1, three candidate points are shown where this design-to-manufacture transfer could occur, indicated by the numbered circles (D-M Interfaces in the Figure key). The Geometric Tailoring activity indicates any design operations required to ensure manufacturability, such as adding rounds and draft to an injection molded part, or widening ribs to increase their strength. Should the design or manufacturing organization be responsible to geometric tailoring? Referring back to Figure 1, it is clear that a manufacturing process and a part material must be selected prior to fabrication. Again, which organization should be responsible for doing so?

Three different sequences of activities are shown in Figure 1, labeled by letters in squares. In some cases, no geometric tailoring may be necessary, indicated by flow B. For example, when a company orders a stereolithography part from a service bureau, they probably did not change the part design to facilitate the SLA process. Sometimes a geometric tailoring event is required, such as in injection molding; this is flow A. Flow C indicates iteration between material & process selection and geometric tailoring. Such an iterative flow is necessary if selection depends upon the extent of part redesigns to facilitate fabrication. In the STL baseline case, transfer point 1 is used, no geometric tailoring would be performed, and the designer may or may not have specified the material or process to be used. In this paper, we will not investigate flow B.

In Table 1, the combinations of flows and transfer points are analyzed for the implied allocation of responsibility among the design and manufacturing organization (abbreviated Design Org and Manufacturing Org, respectively). In order to perform the activities and make the decisions in Figure 1 and Table 1, five design decision templates are introduced to embody the relevant design information, designer preferences, and manufacturing capabilities. The decision templates are:

- **MPS** – Material & Process Selection, a coupled selection decision in which a material and a fabrication process are selected to construct a prototype.
- **MGT** – Material Geometric Tailoring, a compromise decision in which the component’s dimensions are modified to provide similar functional performance when a prototype material replaces the production material.
- **MPGT** – Material-Process Geometric Tailoring, a compromise decision in which the component’s dimensions are modified to suit a prototype material and fabrication process.
- **MPS-MGT** – Material & Process Selection, Material Geometric Tailoring, a coupling of the selection and compromise decisions.

![Figure 1](image-url)
DFM is often difficult for mechanical parts since significant manufacturing knowledge is required to adjust part designs to aid manufacturability by a specific process. Small design changes can cause large changes in the manufacturing process or may render that process infeasible. However, if the manufacturer understands the purpose of a design and its functional requirements, then the manufacturer may be in a better position to adjust the design to facilitate manufacturing without compromising functionality. We do not claim that the manufacturing organization can always perform better DFM, but that for some manufacturing organizations, their insights into their manufacturing processes enable them to achieve better, faster DFM in many cases. In any event, the well-defined interfaces between design and manufacture can always enhance product development, particularly when the development team is distributed.

The context in which we are investigating the clean digital interface is the Rapid Tooling Testbed (RTTB), in which we have implemented aspects of our clean digital interface. Material and process selection, part tailoring for trade-off resolution, and process planning can be performed to address product realization distribution problems. The RTTB is built on a distributed computing environment and has design, selection, compromise, and process planning tools integrated with databases of process information.

1.2 Research Context

The “clean digital interface” idea was raised several years ago in a NSF-sponsored workshop entitled “NSF Workshop on Design Methodologies for Solid Freeform Fabrication” (NSF, 1995). A good survey of the state-of-the-art in data formats for sending CAD models to SFF technologies is provided by Kumar and Dutta (1997). An updated document is available from NIST (Kumar et al., 1999). Most approaches center on developing a standard declarative language with which to specify part geometry and materials. Suggestions have been offered to develop a STEP standard for exchanging CAD information for the purposes of fabricating designs using SFF technologies. An interesting approach to CAD data exchange was suggested by Storti et al., (1999): encapsulate the CAD model in a computational object and provide a dynamically-linked library (DLL) with methods for extracting information from the object. XML could be used to communicate method protocols. This avoids the need for CAD object recipients to understand the standard declarative language.

From another perspective, the context of this research is may be considered as design-for-prototyping, where the objective is to develop production-representative prototypes. In this context, it is important for the prototype to behave similarly to the production design, for the engineering characteristics of interest. Typically, however, the prototype will be fabricated in a SFF machine in a different material than the production part, yielding different behaviors. A new empirical similarity method has been developed (Cho et al., 1999) to construct error measures that apply to prototypes, with confidence considerations. Our approach as articulated in this paper has similarities to that work, but we believe it is more comprehensive in scope.

In the next section, we outline scenarios of RTTB usage and motivate the need for information technology solutions for RP applications. We also describe our research approach. In Section 3, we present the RTTB architecture and discuss its implementation. Design decision templates are presented in Section 4. In Section 5, we present an example problem on gear train design that involves material and process selection, part tailoring (redesign including trade-off resolution), and process planning using the RTTB. We outline information flows and describe multiple scenarios of RTTB usage. Conclusions are offered in the final section.

2 RAPID TOOLING TESTBED

2.1 Overview

We imagine a product realization environment in which geographically distributed engineers collaboratively develop, build, and test solutions to design-manufacture problems encountered in product realization processes. Progress toward this vision is presented in this paper, highlighting the solution of design-manufacture decisions in the context of our Rapid Tooling Testbed (RTTB).

The RTTB is intended to be an experimental testbed that supports exploration of design and design-for-manufacture issues related to rapid prototyping and rapid tooling for injection molding (Allen & Rosen, 1997; Rosen, 2000). The RTTB supports the design of parts and molds, the selection of prototyping technologies and vendors, and the fabrication of those parts and molds. For the purposes of this paper, we focus exclusively on rapid prototyping issues. Given a product design problem and a candidate design, the RTTB is intended to assist the designer in obtaining useful prototypes, where “useful” means that the prototype mimics some desired production characteristics.

2.2 Usage Scenarios

To explore distributed design and manufacture in the context of RP, we developed several example usage scenarios. For this paper, the spur gears in a single layer of a cordless drill power transmission are designed and prototyped (Conner, et al., 1999). We chose a simple design problem that is well understood to allow us to focus on supporting the solution of the problem in a distributed computing environment. The problem is relevant to commonly available engineering software analysis tools.

The design-manufacture context is also important to specify: we assume that a design organization is responsible for designing the power transmission and that they want a prototype transmission constructed. They will contract with a single organization that constructs prototypes (call this the manufacturing organization) for this transmission prototype. We explore the interaction between the two organizations. In
particular, we investigate the extreme case where the design organization requests a prototype and provides a functionally complete transmission design. Further the design organization specifies their functional requirements and requests the manufacturing organization to construct a prototype that best matches these functional requirements. In this usage scenario in this paper, we focus on the design of a single gear in this transmission.

The solution to the design-manufacture problem forms a process as described below.
1. First, the design organization specifies dimensions and material properties of the production gear.
2. A compromise DSP is formulated using a design decision template to define the design freedom that the manufacturing organization can explore. The design organization communicates their production gear solution, a CAD model of the gear with parameters, and the partially instantiated decision template to the prototyping organization.
3. The prototyping organization selects a suitable prototyping technology and material that appears to be a promising match for the design requirements.
4. A model of gear performance as a function of dimensions and material properties is constructed by parametrically generating CAD models of different parameter sets and simulating their performance with FEA. In this work, the gear performance model is a response surface.
5. The prototyping organization searches in the design space specified by the response surface for gear dimensions that enable the prototype gear to match as well as possible the performance of production gears.

Several design technologies that are required to perform this problem-solving process are presented in the next subsection.

2.3 Design Technologies

Within the RTTB, several design technologies, methods, and tools are utilized to support making trade-off decisions. In particular, the trade-off decisions of most interest are called part tailoring decisions, where DFM operations are performed. Typically, product dimensions are modified such that the prototype is easier to manufacture and the prototype meets performance requirements, even though it may be of a different material than the production product and be fabricated using a different process.

The design technologies of interest here include the Decision Support Problems (DSP), Design-Of-Experiments (DOE), and Response Surface Methodology (RSM). Decision Support Problems provide a means for modeling decisions encountered in design and the domain specific mathematical models so built are called templates. Multiple objectives, quantified using analysis-based “hard” and insight-based “soft” information, can be modeled in the DSPs. For real-world, practical systems, all of the information for modeling systems comprehensively and accurately in the early stages of the project may not be available. However, solutions to DSPs can be used to support a designer's quest for a superior solution. Formulation and solution of DSPs provide a means for making the following types of decisions:

- **Selection** - the indication of a preference, based on multiple attributes, for one among several alternatives (Bascaran et al., 1989).
- **Compromise** - the improvement of an alternative through modification (Mistree et al., 1990; Mistree, et al., 1993).

Word formulations of the Decision Support Problems are shown in Table 2. Note that each is described by a set of keywords (Given, Find, Identify, etc.) and descriptors (alternatives, attributes, importances, etc.) that form a language for describing decision problems. The math form of the compromise DSP is shown in Figure 2. There is a 1-to-1 correspondence between word and math forms of this DSP; furthermore, specific forms of bounds, constraints, and goals and deviation functions ensure that all compromise DSP’s are structured in the same manner. It is important to note that the compromise DSP is an extensive of goal programming methods, where the purpose is to satisfy “rigid” objectives, called constraints, and to meet “soft” objectives, called goals, as well as possible, provided significant flexibility in solving problems. Hence, the compromise DSP is not a traditional multiobjective optimization formulation.

In addition to selection and compromise decisions, combinations of decisions in coupled, hierarchical or non-hierarchal forms are possible; selection/selection, compromise/compromise and selection/compromise decisions may be coupled (Bascaran et al., 1989; Herrmann & Allen, 1999).

The next design technology, Response Surface Methodology (RSM), comprises mathematical and statistical techniques to enable the construction of approximation models (Myers & Montgomery, 1995). RSM allows for a better understanding of the relationships between the inputs and the response, in this case between gear dimensions and material properties and maximum stress, that can be written in the form of a polynomial function describing a surface, such as Equation

---

### Table 2. Word Formulations for Selection and Compromise Decision Support Problems.

<table>
<thead>
<tr>
<th>Selection DSP</th>
<th>Compromise DSP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Given:</strong> Alternatives from which to select.</td>
<td><strong>Given:</strong> Alternative to be improved through modification.</td>
</tr>
<tr>
<td><strong>Identify:</strong> Key attributes that influence the selection of alternatives.</td>
<td><strong>Find:</strong> Values of System Variables.</td>
</tr>
<tr>
<td>Relative importances of attributes.</td>
<td>Values of Deviation Variables.</td>
</tr>
<tr>
<td><strong>Rate:</strong> Alternatives with respect to each attribute.</td>
<td><strong>Satisfy:</strong> Goals, Constraints, Bounds</td>
</tr>
<tr>
<td><strong>Rank:</strong> Order the alternatives in terms of preference.</td>
<td><strong>Minimize:</strong> Deviation of solution from goals.</td>
</tr>
</tbody>
</table>
1. We use second order response surfaces in this work; \( k = 2 \).

\[
y = b_0 + \sum_{i=1}^{k} b_i x_i + \sum_{i=1}^{k} b_i x_i^2 + \sum_{i=1}^{k} \sum_{j=1, i \neq j}^{k} b_{ij} x_i x_j \quad (1)
\]

### Compromise DSP

**GIVEN:**
- An alternative to be improved
- Target value for goals, \( G_i, i = 1 \ldots n \)
- Relative importances, \( W_i \)

**FIND:**
- System Variables: \( X_j, j = 1 \ldots m \)
- Deviation Variables: \( d_i^-, d_i^+, i = 1 \ldots n \)

**SATISFY:**
- Goals:
  \[
  A_i(X) + d_i^- - d_i^+ = G_i
  \]
- Constraints:
  \[
  C_k(X) \leq 0, k = 1 \ldots p
  \]
  \[
  d_i^- \cdot d_i^+ = 0, d_i^-, d_i^+ \geq 0
  \]
- Bounds:
  \[
  l_{bj} \leq X_j \leq u_{bj}
  \]

**MINIMIZE:**
- Deviation Function:
  \[
  Z = \sum W_i \cdot d_i^+
  \]

Figure 2 Math Formulation of Compromise DSP.

In much of our work, response surfaces are used to replace computationally complex, but high fidelity analyses for usage during design synthesis. Response surfaces approximate the actual design space and are based on the high fidelity analyses, but enable much faster syntheses. After synthesis, a check is performed to ensure that the performance indicated by the synthesis result is not too far off. This type of synthesis is particularly useful during concept exploration, rather than later-stage detailed design. We call this method the Robust Concept Exploration Method (RCEM) (Chen et al., 1996).

In order to create the response surfaces for the system goals, a number of experiments must be run to gather data for an empirical model. When the system goals are dependent on two or more factors (system variables), Design of Experiments (DOE) techniques, the third design technology, can be practiced to determine the experiment sequence for the empirical model. Factorial experiment designs involve testing a number of variables, or factors, at different values, or levels. The experiments used to construct response surfaces (model building experiments) were fractional factorial experiment designs with a face centered central composite design. Three levels of each factor were considered.

### 3 RTTB SYSTEM

#### 3.1 RTTB Architecture

Our long term vision for the RTTB is to create a distributed computing environment that supports the design and manufacture of prototype parts. The environment should be able to reconfigure the organization of its software modules in response to the nature of a specific design-manufacture problem. Additionally, it should adjust itself to changes in computing resources. Both commercial and research SFF technologies are being supported, as well as select rapid tooling technologies to produce multiple parts through injection molding.

The RTTB is being structured to integrate design process and fabrication process models with product models to enable a designer to instantiate an approved design process with the appropriate tasks and resources (Allen and Rosen, 1997). Currently, selection, CAD, synthesis, analysis, and DOE modules are integrated into the RTTB. The RTTB is implemented on an extension of the distributed computing environment called PRE from Sandia National Laboratory (Gerhard et al., 1999). The latest version of the distributed computing environment is called PRE-RMI, where RMI denotes Remote Method Invocation, a term originating with Java developers.

To focus the presentation, only the subset of RTTB that is relevant to this paper is presented in Figure 3. Legacy codes such as ProEngineer and ANSYS have been wrapped so that they can be integrated into the RTTB. The combination of the two is called the GearAgent, which will become clear in Section 5 when the example is presented. Additional software tools have been developed to perform other tasks and have been integrated as well. To initiate RTTB operation, the designer provides a CAD model of the part, tolerances and surface finish specifications, and preferences for prototype part characteristics (accuracy, finish, build time). Additionally, the designer may specify performance specifications, desired part characteristics, preferences for materials or fabrication processes, and preferences on other specifications and characteristics.

A web-based tool has been written in Java for coupled material and process selection (Herrmann & Allen, 1999), called the RPAgent in Figure 3. Material property and RP process capability descriptions reside in a relational database and provide this information to the selection tool. A process planning tool, called the ProcessPlanAgent, was written in C++ for the stereolithography process (West & Rosen, 1999). Input to process planning consists of an ACIS solid part model with tolerances and surface finishes specified.

Within the ClientAgent, two main modules tend to be the focus of user interaction and control of other agents: Design of Experiments and Synthesis. The Design of Experiments module is implemented by commercial codes Optimus and/or Minitab. They are used to construct and execute designs of experiments that typically construct and analyze CAD and FEA models. The synthesis module performs geometric tailoring. Two multiobjective optimization codes are currently integrated, DSIDES (Mistree et al., 1993) and LINGO (Nemhauser & Wolsey, 1988), as well as an exhaustive search code. At present, we have developed an initial geometric tailoring problem formulation in the form of a Compromise DSP template. This template may be modified by hand; see Section 4. Analysis codes must be written and integrated into one of the optimization codes.
Three main constructs form the core of the distributed computing environment underlying the RTTB: event, channels, and agents. Communication among the RTTB modules occurs in two main event channels, one for design communications and the other for fabrication communications. This is a novel approach to distributed computing, and is similar to the idea of a "listener" in the Java AWT 1.1. Agents subscribe to a channel and receive all events that are passed into the channel, as well as sending all self-generated events to the channel. An agent can send and receive events to and from multiple channels. Any information that must be passed between agents is packaged into an event. Events are typed, with types chosen by the application developer to model actions that may need to occur in support of the application. For example, a PrintEvent may be required to send documents to the printer, while a DataBaseEvent may enable a database to be queried to retrieve information for an agent. In our implementation, we deal with handled and unhandled event, where the handling status indicates the state of the event.

High-level system behavior consists of the interaction between agents, as specified by the connectivity among channels and by which events are sent through the channels. Combinations of event, channels, and agents can represent any process. The use of two event channels to integrate five different distributed services is illustrated in Figure 3. Events are passed along the event channels and cause the agents that manage the services to execute their software tools with the process information carried in the events. More information on the design and implementation of the distributed computing environment is given in (Gerhard et al., 1999; Gerhard et al., 2000).

### 3.2 RTTB System

#### Implementation

The implementation of the RTTB using PRE-RMI is presented by describing the implementation and behavior of each major service in Figure 3.

**ClientAgent:** The ClientAgent is called to submit an experiment or final model request to the system from a command line interface or from a batch text file. In the command line interface, a single set of gear transmission parameters can be entered and submitted. In the batch text file interface, the text file output from MiniTab is parsed by the ClientAgent into a set of experiments. For each experiment or final model request, the ClientAgent simply generates an unhandled GearDesignEvent and broadcasts it to the DesignEventChannel. When a handled GearDesignEvent is broadcast to the DesignEventChannel, the ClientAgent writes the stress results to the screen.

**GearAgent:** When an unhandled GearDesignEvent is broadcast to the DesignEventChannel, the GearAgent generates an experimental model with Pro/Engineer, analyzes it with AnSys, and retrieves the stress results. If a final model is requested, the GearAgent produces VRML and STL files of the sun and planetary gears as well as solving for the stresses in the experimental model. All results are stored in the GearDesignEvent and it is re-sent to the DesignEventChannel as a handled event.

**RPAgent:** The RPAgent acts on a handled GearDesignEvent on the DesignEventChannel that contains a final model. It invokes the process/material selection code with which the user interacts to select an appropriate RP technology and material for the required prototype. When these selection results are entered into the RPAgent, the RPAgent generates a new RPEvent, storing the STL models and the RP process/material selection information. This RPEvent is sent to the RPEventChannel.

**ProcessPlanAgent:** The ProcessPlanAgent acts on an unhandled RPEvent on the RPEventChannel. It extracts the STL gear models and stores them in the appropriate locations for the RP process planning software tool. It then invokes this process planning tool to set up stereolithography builds (West & Rosen, 1999). The process plan tool orients and slices part models, then recommends SLA machine settings that best meet designer requirements on tolerances, surface finishes, and build times. Multiple solutions are found and reported. Once these results are computed, the ProcessPlanAgent re-sends the RPEvent as handled to the RPEventChannel.


**DisplayAgent:** The DisplayAgent has an application GUI that displays the results of the stress analysis, the RP machine and material selection, and the process planning analysis for final model runs. It extracts the stress results and the VRML models from a handled GearDesignEvent on the DesignEventChannel. It also extracts the RP machine/material selection results and process planning build times from a handled RPEvent on the RPEventChannel. This information is displayed when the events arrive from the event channels.

Agent behaviors are summarized in Table 3.

Table 3 Agent Behaviors.

<table>
<thead>
<tr>
<th>Agent</th>
<th>Generates Messages</th>
<th>Acts upon Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClientAgent</td>
<td>GearDesignEvent (unhandled),</td>
<td>GearDesignEvent (handled),</td>
</tr>
<tr>
<td></td>
<td>GearSelectEvent (unhandled),</td>
<td>GearSelectEvent (handled),</td>
</tr>
<tr>
<td>GearAgent</td>
<td>GearDesignEvent (handled)</td>
<td>GearDesignEvent (unhandled)</td>
</tr>
<tr>
<td>RPAgent</td>
<td>RPEvent (unhandled)</td>
<td>RPEvent (unhandled)</td>
</tr>
<tr>
<td>ProcessPlanAgent</td>
<td>RPEvent (handled)</td>
<td></td>
</tr>
<tr>
<td>DisplayAgent</td>
<td>GearDesignEvent (handled), RPEvent (handled)</td>
<td>none</td>
</tr>
</tbody>
</table>

Figure 4 MPS - Coupled Material-Process Selection Template.

### 4 DESIGN DECISION TEMPLATES

In this section, the decision templates that were introduced in Section 1 are presented and some are instantiated for the gear design problem. The templates are based on the Selection and Compromise DSP’s mentioned earlier, as well as coupled combinations of these DSP’s.

In the RPAgent, a process/material selection code is invoked. The decision template for this code is shown in Figure 4. Three types of decision attributes are evident: material, process, and coupling. Ratings of coupling attributes depend upon both the material and process. As examples, typical material attributes are tensile strength, elastic modulus, and freezing temperature, while process attributes are cost per hour and accuracy. Coupling attributes include surface finish and fabrication speed. This template utilizes a target-matching strategy in that target values for material and process attributes are specified and the objective of the coupled decisions is to minimize the weighted sum of deviations from these targets (Herrmann, 1999). Weights are specified as relative importances in Figure 4. This template is denoted MPS, for Material-Process Selection.

The second template considered is for a geometric tailoring decision that involves the material choice, but not the process choice. This template is denoted MGT for Material Geometric Tailoring, and is based on the Compromise DSP. With decision problems formulated with this template, designs can be tailored such that their performance matches as well as possible that of a production design, even though the material of the prototype design is different than that of the production design. The MGT template is shown in Figure 5. MGT decisions are implemented in the RTTB system in the ClientAgent.

A related decision template is for tailoring relative to both the material AND process of the prototype. The template is denoted MPGT for Material-Process Geometric Tailoring and is also based on the Compromise DSP. The difference between MGT and MPGT is the addition of process variables, constraints, and goals. MPGT is shown in Figure 6. MPGT decisions would also be implemented in the ClientAgent, but would require integration with the ProcessPlanAgent. No MPGT decisions have been instantiated in our system, as this is an area currently under investigation.

The final template is for coupled material selection and geometric tailoring decisions. Essentially, these decisions enable the simultaneous selection of a prototype material and the geometric tailoring of the part. The template, shown in Figure 7, is a combination of the MPS and MGT templates and is denoted MPS-MGT. The mathematical formulation of this template is shown in Figure 8.

---

### 4.1 MPS-MGT Template

<table>
<thead>
<tr>
<th>GIVEN:</th>
<th>Material Selection</th>
<th>Process Selection</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Set of Feasible Material Alternatives</td>
<td>Set of Feasible Process Alternatives</td>
</tr>
<tr>
<td>IDENTIFY:</td>
<td>Set of Material Attributes</td>
<td>Set of Process Attributes</td>
</tr>
<tr>
<td>SPECIFY:</td>
<td>Coupling Attributes for Materials and Technologies</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Target values for each material attribute</td>
<td>Target values for each process attribute</td>
</tr>
<tr>
<td></td>
<td>Relative importances of material attributes</td>
<td>Relative importances of process attributes</td>
</tr>
<tr>
<td>RATE:</td>
<td>Relative importances of coupling attributes</td>
<td>Each alternative with respect to each attribute.</td>
</tr>
</tbody>
</table>

Figure 5 MGT - Material Geometric Tailoring Template.

---

### 4.2 MPGT Template

| GIVEN: | Parametric CAD model of design | Material Properties |
|        | Target values for variables | Goal preferences as weights |
|        | Target value for weight goal |

| FIND: | Deviation Variables: |
|       | deviation of goals from targets |

| SATISFY: | System Variables: |
|          | CAD model parameters |
|          | (none built into template) |

| MINIMIZE: | Deviation Function: |
|           | Weighted sum of Goal Deviations |

Figure 6 MPGT - Material-Process Geometric Tailoring Template.

---

Copyright © 2000 by ASME
Given:
- Parametric CAD model of design
- Target values for variables
- Target value for weight goal
- Material Properties
- Process Properties
- Goal preferences as weights

Find:
- System Variables:
  - CAD model parameters
  - Process variables
- Deviation Variables:
  - deviation of goals from targets

Satisfy:
- Goals: Meet target weight
- Meet targets of CAD model parameters
- Meet targets of process goals
- Constraints: (none built into template)

Minimize:
- Deviation Function: Weighted sum of Goal Deviations

Figure 6 MPGT - Material-Process Geometric Tailoring Template.

Figure 7 MPS-MGT - Coupled Material-Process Selection, Material Geometric Tailoring Template.

Figure 8 Carefully, the mathematical formulations of MPS and MGT can be surmised. The MPS-MGT template has been implemented in the RTTB by integrating the RPAgent and the ClientAgent. In Figure 8, the remaining undefined terms are defined here: \( d_i^* \) = deviation variables for compromise goals, \( e_{ij}^* \) = deviation variables for selection goals, \( c_{e_i}^* \) = deviation variables for coupled selection goals.

For completeness, we mention the coupled material-process selection, material-process geometric tailoring (MPS-MPGT) decision template. This template is a simple extension of MPS-MGT that incorporates process-related geometric tailoring (see Figures 4, 5, 7, and 8). This template has not been instantiated in the RTTB, but is under development.

5 Example Problem

The design problem under consideration here is the design of ring gears for a speed reducer in a family of cordless drills. These speed reducers are planetary gear trains, typically with three stages, as shown in Figure 9. Typically, ring gears for layers 1 and 2 are made of injection molded, glass-filled nylon. For our purposes, we assume that a gear train has been designed and a prototype gear train is required for functional testing. Rather than wait for production injection mold tooling for the ring gear to be machined, the designer wants to fabricate a prototype ring gear using an SFF process. Further assume that the designer desires two types of testing to be performed on the prototype, assemblability into the drill housing, and stress/strength characteristics during operation.

The scenario we describe is Flow A with Design-Manufacturing Interface 3 from Table 1, where the manufacturing organization is responsible for material and process selection, geometric tailoring, and process planning. For simplicity, we assume that the manufacturing organization performs a coupled material-process selection problem, then formulates and solves a MGT problem. We focus on the MGT problem formulation and solution process in the RTTB.

Figure 9 Speed Reducer for Drill Family.
Table 4  Design Factors and Their Ranges.

<table>
<thead>
<tr>
<th>Design Factor</th>
<th>Low End of Range</th>
<th>High End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gear Face Width (W) (in.)</td>
<td>0.1</td>
<td>0.4</td>
</tr>
<tr>
<td>Diametral Pitch (P) (teeth/in.)</td>
<td>39</td>
<td>46</td>
</tr>
<tr>
<td>Number of Teeth (N) (-)</td>
<td>51</td>
<td>57</td>
</tr>
<tr>
<td>Torque in Ring Gear (in-lb)</td>
<td>2.3</td>
<td>4.1</td>
</tr>
</tbody>
</table>

5.2 Manufacturer Activities

The manufacturer interacts with the ClientAgent in the RTTB system to initiate the project. All the information received from the designer is entered into the ClientAgent. A material-process selection must be performed, so the manufacturer generates a GearSelectEvent from the ClientAgent, sending it to the DesignEventChannel as an unhandled event. The RPAgent acts on the event, formulating and solving a MPS problem with manufacturer interaction. The
selected material and rapid prototyping process and material are the stereolithography process: SLA-3500 and SL 7510 resin. These results are sent to the DesignEventChannel as a handled GearSelectEvent, which is then acted upon by the ClientAgent.

With material and process selected, the manufacturer initiates the formulation of a MGT problem to find dimensions of the prototype ring gear that provide the same performance as a production ring gear. To formulate the MGT problem, three pieces of information are required: prototype material properties, the maximum stress to which the prototype gear should be subjected, and the stress behavior as a function of gear dimensions (response surface model). With this information, the empty entries in Figure 10 can be completed.

Material properties of SL-7510 resin are: elastic modulus is 400 kpsi, flexural strength is 11.7 kpsi, and tensile strength is 8300 psi. Given the anisotropic properties, we estimate that the yield strength for a gear application is 10.5 kpsi since gear teeth are subjected to a combination of bending and compressive loads. With these data, we can establish the condition for a corresponding stress state in the prototype gear using Equation 2. The maximum stress in the prototype gear should be 10500 * 5171 / 13000 = 4176.6 psi. Bounds on the maximum stress are also specified to define a feasibility range; 5 percent fluctuation is allowed, providing the range 3967.8 ≤ MS ≤ 4385.4 psi.

The response surface model of ring gear stresses, as a function of dimension values, is required in order to avoid performing finite element analyses during design synthesis. By interacting with the ClientAgent in the RTTB, the manufacturer requests a response surface to be constructed that provides a model (Equation 1) to relate the design factors to the stress results within a prototype ring gear. Using the Optimus commercial software, a design of experiments was set up to construct the desired response surface. The ClientAgent coordinates with the GearAgent to construct a CAD model, generate a finite-element model, and conduct an analysis for each experiment. A Central Composite Face-centered design was utilized for the four factors in Table 4 for a three-level experiment. This results in 25 experiments (implying 25 CAD models and 25 finite element models and analyses) that yield estimates of maximum stress in the ring gear teeth, as measured by von Mises stress. By running these experiments, Optimus constructed a quadratic response surface that provides approximate maximum stress values, given as Equation 3.

Using the response surface model, the ClientAgent formulates and solves the MGT problem from Figure 10 using an exhaustive search algorithm. Three scenarios of goal preferences were investigated, one of which corresponded to the designer’s specified preferences among stress and the dimensional variables. The two other scenarios weighted the stress goal more highly, in scenario 2, or weighted the dimensional variables. The two other scenarios weighted the stress goal more highly, in scenario 2, or weighted the dimension goals more highly, as in scenario 3. Weights for the three scenarios are shown in Table 5.

![Figure 10 MGT Ring Gear Problem Formulation.](image-url)

### Table 5 Scenarios of Goal Weights

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Stress Goal</th>
<th>Face Width Goal</th>
<th>Pitch Goal</th>
<th># Teeth Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.1</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Copyright © 2000 by ASME
Results of the MGT problem are shown in Table 6 for each of the three scenarios. As can be seen, the results across all scenarios are almost identical, indicating a solution that is stable with respect to preferences. Based on these results, the manufacturer selects the following ring gear dimensions:

Face Width = 0.22 in., Diametral Pitch = 40 teeth/in., # Teeth = 55

<table>
<thead>
<tr>
<th>Scenario</th>
<th>W</th>
<th>P</th>
<th>N</th>
<th>Max Stress</th>
<th>Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[inch]</td>
<td>[teeth/in]</td>
<td>[-]</td>
<td>[psi]</td>
<td>[%]</td>
</tr>
<tr>
<td>1</td>
<td>0.22</td>
<td>40</td>
<td>55</td>
<td>4275.6</td>
<td>1.65</td>
</tr>
<tr>
<td>2</td>
<td>0.225</td>
<td>40</td>
<td>55</td>
<td>4156.7</td>
<td>0.99</td>
</tr>
<tr>
<td>3</td>
<td>0.22</td>
<td>40</td>
<td>55</td>
<td>4275.6</td>
<td>1.5</td>
</tr>
</tbody>
</table>

Once again, the ClientAgent sends an unhandled GearDesignEvent to the DesignEventChannel to request the GearAgent to create a CAD model of the ring gear, this time using the results of the MPS-MGT problem shown above. Additionally, STEP, STL and VRML files are requested for subsequent rapid prototyping and visualization purposes.

A handled GearDesignEvent message is sent from the ClientAgent to both the Design and RP EventChannels to develop a process plan for stereolithography. The STEP file is converted to an ACIS model for processing by our process planning system (West & Rosen, 1999). The results of process planning are sent in a handled RPEvent, and received by the DisplayAgent where the VRML file, stress results, and process planning results are all displayed to the manufacturer.

5.3 Discussion

This example problem illustrates one scenario of design-manufacturing collaboration in the context of SFF technologies, a scenario that explores an extreme case of the manufacturer assuming responsibility for decisions that conventionally would be performed by the design organization. Specifically, the manufacturer was given a ring gear design for which a prototype was needed. The manufacturer was responsible for selecting a suitable material and fabrication process for the prototype and for fine-tuning the design to provide production-representative performance. In this case, “production-representative” meant that the prototype gear should be subjected to a similar stress state as the production gear. Delegating such design responsibility to the manufacturer places a communication burden on the designer: the designer must specify allowable design freedoms to the manufacturer, and must provide the structure of the “fine-tuning” decision. In our approach, these requirements are communicated using a problem formulation based on a compromise DSP template, the Material Geometric Tailoring (MGT) template in this case. This MGT template effectively provides a “clean digital interface” between the design and manufacturing organizations, enabling information to be packaged in a convenient manner and enabling selection and compromise (tailoring) decisions to be made by the manufacturing organization. Furthermore, we demonstrated that a distributed computing environment can be effectively utilized to support the information transfer and subsequent computing for aiding decision making.

Broadly, application of these templates to conventional manufacturing processes is feasible. However, the purposes of these templates change since matching production properties is not necessary. In molding, stamping, etc. there are many DFM rules that enable designers to predict and correct potential manufacturing problems during middle and late design stages. Some of these rules deal with dimensioning and positioning features. These are the rules that can be most readily incorporated into the templates. In summary, the templates can be applied for use with conventional processes, but will only incorporate a subset of applicable DFM rules.

Validity of templates derives from their application in certain design-manufacturing contexts and on problems within the scope of the templates. Templates are internally valid since they are derived from the selection and compromise DSP’s and their coupled brethren. That is, the mathematics underlying these DSP’s has been shown to be valid. The larger issue is the external validity of templates. Templates are meaningful only in certain design-manufacturing contexts. Within those contexts, the templates seem useful in that when they are used, they can produce reasonable results. Since this is a typical criterion used in design research, we believe that we can claim some external validity, but more research is needed for more conclusive findings.

6 CLOSURE

In this paper, we investigated issues arising in distributed design and fabrication, where fabrication is performed by Solid Freeform Fabrication (SFF) technologies. The main issue is the relationship between responsibility for decisions and the flow of information to support those decisions. Related issues include the extent to which design-for-manufacture can be transferred to the manufacturer, and the efficacy of supporting distributed design and fabrication with a distributed computing environment. Our purpose with this paper is to demonstrate that communicating design information with decision templates enables a “clean digital interface” between design and fabrication, effectively separating design activities from manufacture activities. Through an example scenario of design-manufacture collaboration in designing a prototype ring gear, we demonstrated that an extreme case of transferring design responsibility to the manufacturing organization can succeed, provided that the designer provides sufficient information.

This scenario provides evidence for the following theses:

- Transferring design-for-manufacture responsibility to the manufacturing organization is feasible, at least for fabricating prototypes.
- Design decision templates for material and process selection, and for geometric tailoring, provide a medium of communication (clean digital interface) that is sufficient
designing and fabricating prototypes, where the prototypes are to be as production-representative as possible.

- If the design and manufacturing organizations are collaborating using a distributed computing environment, computing resources can be shared, facilitating the separation of design and fabrication activities.

Future work is needed in several directions. First, more substantive examples are needed to more fully investigate all proposed decision templates in more rigorous engineering contexts. Second, the work needs to be extended to include the design and fabrication of tooling for injection molding of prototypes. It appears to be straightforward to deal with SFF technologies, but incorporating conventional manufacturing processes into our “clean digital interface” approach is challenging. Finally, the distributed computing environment must be more rigorously tested and expanded to include additional agents and database facilities.

ACKNOWLEDGEMENT

We gratefully acknowledge the support from NSF grant DMI-9618039, from Sandia National Laboratories, and from the Rapid Prototyping and Manufacturing Institute member companies.

REFERENCES


