Artificial Intelligence for Engineering Design, Analysis and Manufacturing (1999), *13*, 253–275. Printed in the USA.

Influencing generative design through continuous evaluation: Associating costs with the coffeemaker shape grammar

MANISH AGARWAL, JONATHAN CAGAN, AND KATHERINE G. CONSTANTINE
Computational Design Laboratory, Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.

(RECEIVED June 15, 1998; FINAL REVISION March 30, 1999; ACCEPTED April 1, 1999)

Abstract

A grammatical approach to product design is demonstrated. In particular, shape grammars are shown to be especially useful for products that are differentiated primarily on the basis of form yet driven by function; they allow products to be designed as a sequence of well-defined steps. However, it is not always clear how to choose the sequence of rules that should be applied to generate the final shape. In this paper we demonstrate that at each stage during the process, partial designs of the final product can be used to provide feedback to the designer based on specific design objectives and thus suggest possible rule choices. We take advantage of the shape grammar for the generation of coffeemakers introduced by Agarwal and Cagan, and associate with the grammar rules expressions that model manufacturing costs. With each application of a shape grammar rule, an understanding of the overall cost of manufacturing the product is incrementally improved. Thus, at each stage of the design process the designer has an indication of what the overall cost of the product will be and how the selection of one grammar rule over another influences the final cost. Once the complete product is generated, an appraisal of its manufacturing cost is given to the designer. This evaluation methodology helps the designer understand the implications of decisions made early on in the design process. We have also verified the accuracy of this approach through the costs of some commercially available coffeemakers, generated by this method, which are comparable to the costs for those designs listed in the literature.

Keywords: Coffeemakers; Shape Grammars; Manufacturing Cost

1. INTRODUCTION

Shape grammar-based systems have been successfully used for generative design in architecture and recently for product design. However, there are no formal techniques that help the designer in selecting which rule to apply at any given stage. In this work, we argue that using performance metrics along with a grammar-based generative system will create a powerful feedback mechanism for the designer during the design generation process. Additionally, in the generative design of products, those designs that fare the best in terms of performance metrics are often the ones that are the most successful in the marketplace. However, the association of such metrics within shape grammars for engineering applications has received little attention. One approach to the evaluation of designs created by grammars is to use external analysis after the generation sequence is complete. The drawback to such an approach is that since the evaluation is carried out after the design is completed, no information can be provided to the designer during the design generation phase. In contrast, our approach is to associate performance evaluation directly with the grammar rules themselves. We illustrate the power of such an approach by associating manufacturing costs with the coffeemaker shape grammar of Agarwal and Cagan (1998).

Product cost is one of the most important design constraints during the design and redesign of products; a product will not succeed in the marketplace if it is not properly

Reprint requests to: Jonathan Cagan, Computational Design Laboratory, Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A. Phone: 412-268-3713; Fax: 412-268-3348; E-mail: cagan@cmu.edu

priced for its intended market. Studies of product design and product manufacture suggest that a significant portion of a product's cost is determined by the decisions made early in the design process (e.g., Nevins & Whitney, 1989; Ulrich & Pearson, 1996). Thus, the designers need to be made aware of not only how their later design decisions affect the cost of a product, but also (and perhaps more critically) how their early decisions affect the manufacturing cost. Further, it is important that this information be made available as soon as a design change is made so that a more extensive redesign iteration can potentially be avoided. To be able to make this information available, it is necessary that the manufacturing costs are correlated with the product configurations that exist at any given stage of the design creation cycle and not just with the final finished product. Such a technique will enable the designer to gauge the effect of a change when it is made as well as ensure that an accurate appraisal of the final manufacturing cost is provided to the designer as soon as the design is completed. An immediate feedback of this nature will enable the designer to explore a variety of design alternatives that may otherwise have been too time-intensive to consider. However, to achieve this, it is important that the design and costing methodology allow for rapid generation of results and, thereby, enable the designer to create designs with a small turnaround time.

We propose that a shape grammar-based design paradigm will meet all the requirements of such a system. Grammars not only allow the rapid generation of a wide variety of feasible designs by the application of different rules in a rule set, but they also maintain representations of the partial designs at each stage. We claim that based on the rules applied to reach a particular design stage, it is possible to estimate the manufacturing cost at that stage. In this paper we argue that by associating manufacturing cost with the grammar rules, one can identify design changes that have positive and negative effects on the product cost. The designer can then receive immediate feedback on how the various changes will affect the cost of the product, thereby allowing informed decisions to be made about whether to accept or reject those changes. In addition, because partial design costs are available at each stage, the final cost is obtained as soon as the design is completed. As an example of the usability of such a strategy, we apply this technique to the generation and costing of coffeemakers based on a grammar developed by Agarwal and Cagan (1998) by associating manufacturing cost expressions with the shape rules in the grammar. This then provides the designer with an integrated methodology for the design and costing of coffeemakers and allows various trade-offs to be studied from a manufacturing cost perspective. It is important to note that, while this paper discusses the strategy for associating expressions modeling cost with the coffeemaker grammar rules, the underlying technique is general and can be applied to a variety of grammars and performance metrics.

Some commercially available coffeemakers are designed and costed using our method and the results compared to those reported by Ulrich and Pearson (1993) using a disassembly-based costing approach. In spite of the completely different costing strategy used in the two studies, the costs estimated by both methods turn out to be similar, validating our approach. Before we discuss the details of our method, it is worthwhile to examine the traditional costing approaches and contrast them with our technique.

Most of the current costing methodologies are either based on parametric estimation models or require a bottom-up cost calculation. Bottom-up cost estimation techniques require either a complete computer aided design (CAD) model of the product [e.g., complexity theory (Hoult & Meador, 1997) or commercial software like Cost Advantage from Cognition Corporation] or a detailed step-by-step manufacturing breakdown of the product (e.g., commercial software like KAPES from PS Industries). Neither of these two bottom-up techniques can provide information about estimated cost early on in the design phase because they work with complete product designs. Also, to determine how a design change will affect the cost, the cost has to be reestimated for the new product and compared with the original cost. These techniques also require a significant amount of set-up time before actual estimation can be carried out. Parametric estimation techniques (e.g., commercial software like SEER H from GA SEER and PRICE H from Lockheed Martin PriceSystems), on the other hand, require little information about the product design. They use statistical methods to relate the product weight, volume, manufacturing process, and a few other parameters to the final product cost. These techniques require extensive calibration based on existing products before they can be used to estimate cost and they do not provide any information about the influence of design decisions on cost. Because they do not identify the key cost drivers, these techniques are unable to provide feedback regarding possible redesign directions. They also usually require an estimate of the final weight and volume of the product (which may not be readily available) before cost can be estimated. In summary, most traditional costing techniques are unable to provide information about the product cost until after the design is completed and therefore may require the products to be completely recosted if a design change is made. We believe that the technique proposed in this paper will address these concerns.

Next, we will briefly discuss the coffeemaker grammar. We will then define a manufacturing cost structure for injection molded parts, metal stamped parts, and product assembly. By associating various elements of this cost structure with the coffeemaker grammar rules, a methodology for estimating costs of the designs generated by this grammar will be obtained. We will demonstrate the technique by discussing an example and verify the results by comparing them to those in the literature. We will demonstrate how this method can be used to study design trade-offs and guide the design generation process. Finally, we will conclude with a brief discussion on how this technique can be applied within other shape grammars.

2. COFFEEMAKER SHAPE GRAMMAR

2.1. Shape grammars

A shape grammar (Stiny, 1980*a*, 1980*b*) derives designs in the language it specifies by successive application of shape transformation rules to some evolving shape, starting with an initial shape. It can be used to describe how complex shapes are built from simple entities and how a complex shape can be decomposed into simpler subshapes. Shape grammars have been successfully used for spatial design in the field of architecture including villas in the style of Palladio (Stiny & Mitchell, 1978), Mughul gardens (Stiny & Mitchell, 1980), prairie houses in the style of Frank Lloyd Wright (Koning & Eizenberg, 1981), Greek meander patterns (Knight, 1986), suburban Queen Anne Houses (Flemming, 1987), and windows in the style of Frank Lloyd Wright (Rollo, 1995).

Examples illustrating the ideas behind shape grammars can be found in Stiny (1980*a*, 1980*b*). While there has been a limited application of shape grammars to engineering design, they had not been used for the generation of individual products until Agarwal and Cagan (1997, 1998) presented the coffeemaker grammar. Fitzhorn (1990) and Longenecker and Fitzhorn (1991) have presented shape grammars specifying the languages of constructive solid geometry and boundary representations (i.e., realizable solids). Brown, McMahon, and Sims Williams (1993) presented a manufacturing-oriented shape grammar that specifies the language of all axi-symmetric objects manufacturable on a given lathe. That work is particularly relevant here because, although they used a completely different strategy, they presented a technique for estimating the manufacturing time (which is an important component of manufacturing cost) for the various parts machined by the lathe. Reddy and Cagan (1995*a*, 1995*b*); Shea, Cagan, and Fenves (1997); and Shea and Cagan (1997) presented parametric shape grammars for the design of planar and geodesic dome truss structures that used the shape annealing technique of Cagan and Mitchell (1993) to generate optimal structures.

Stiny (1981) presented a general design description methodology that relied on associating description rules with the grammar rules much like we do here. However, the description rules themselves as well as their association with the grammar rules vary vastly based on the application domain. No formal techniques exist for creating these description rules or for associating them with the grammar rules. This work uses cost expressions along with the shape rules and thus applies grammars to the concurrent design and costing of a class of individual products.

2.2. Coffeemaker grammar

The coffeemaker grammar is a parametric, labeled 2D shape grammar consisting of 100 rules and can recreate a number of existing coffeemakers as well as create an infinite number of new designs. The rules in the shape grammar manipulate one or more of the three views of the product—top, side, front—to create a final 3D shape. The coffeemaker is considered to be made up of three main parts: the filter unit, the water storage unit, and the base unit. These three units are arranged around the space for the coffee pot, which acts as the initial shape for the grammar. The grammar creates a complete coffeemaker by first designing the base and the filter units and then blending them together using the water storage unit. Due to the similar functional breakdown of coffeemakers, the function drives the form in the product and in the application of the grammar rules; function labels are used to maintain the proper function-to-form sequence.

The designs generated by the grammar can satisfy a wide variety of functional requirements. For example, the design can be a single-heater or double-heater unit, have a conical or a flat filter (with or without a flow rate control mechanism), and can use a lid or a grating to cover the water storage unit. It should be pointed out, though, that the designs generated by the grammar do not incorporate all of the design details. For example, the number and position of screws, the power cord, the color of the product, and the form of the switch are not designed by the existing grammar rules. Thus, in this work, the cost corresponding to these components is added separately and is not obtained directly through the shape grammar.

3. ASSOCIATING COST EQUATIONS WITH THE GRAMMAR RULES

The first step in associating the manufacturing costs with the rules of the coffeemaker grammar involves breaking the cost into its components in a manner compatible with the shape rules, that is, such that each of the components can be associated with the rules. In this work we assume the cost of a coffeemaker to be made up of three main components—the cost of manufactured parts, which will form the focus of this work; the cost of purchased parts; and the cost of assembling all of the parts into a functional product. The cost of the manufactured parts can be further broken into five components—material cost, equipment operating cost, tooling cost, burden, and labor cost. Expressions for each of these components (for plastic and metal parts) are given in Appendix A. As mentioned in the appendix, each of these five cost components depends primarily on the part configuration and geometry and that is precisely the information obtained from the shapes representing the designs. This fact is crucial to the success of our methodology because it allows us to develop general parametric expressions from the shape rules that are then instantiated as the shape rules are applied. More specifically, we develop expressions for the areas and volumes of the shapes generated by the shape rules and then use them together with Eqs. (A1)–(A9) to determine the manufacturing costs. If the cost components involved an attribute that could not be determined from the

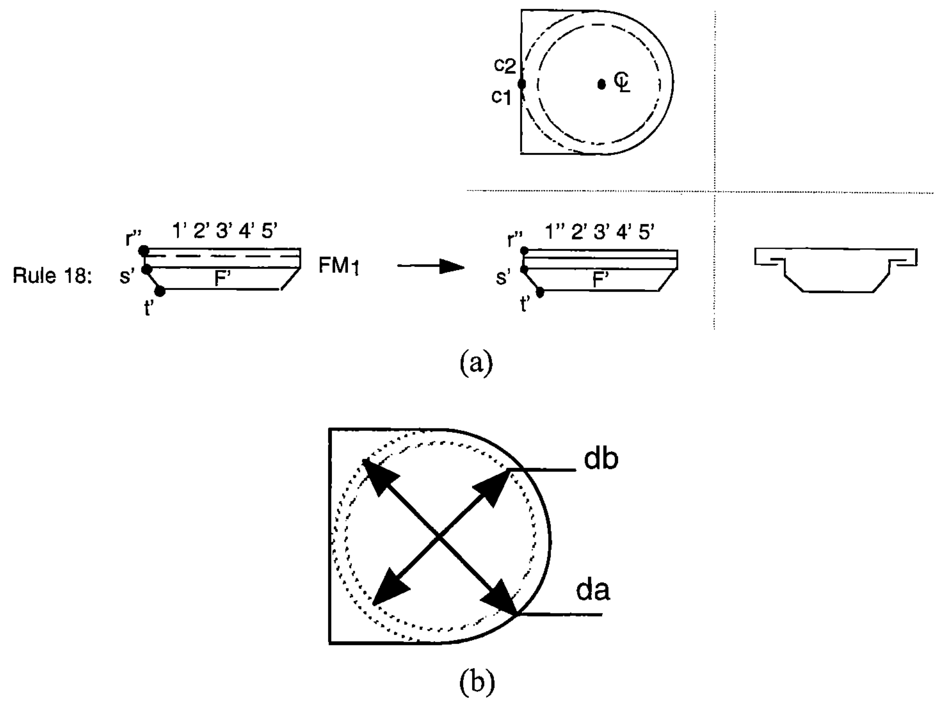


Fig. 1. (a) Shape rule creating a sliding filter unit; (b) top view created by the shape.

shape rules, then this technique could not have been directly applied.

Developing expressions for areas and volumes that are associated with the shape rules involves examining the geometric forms created by a rule. Note that the most general form of the grammar rule must be used, though frequently approximations need to be made when calculating areas and volumes to keep the expressions tractable. As an example consider the shape rule shown in Figure 1(a). This figure represents Rule 18 of the shape grammar of Agarwal and Cagan and is used to generate a sliding filter unit for the coffeemaker (the three views on the right-hand side of the shape rule—at the top, at the bottom left, and at the bottom right—correspond to the three views—top, side, and front—of the coffeemaker)

To determine the projected area of the filter unit created by this shape rule, the top view of the shape [Fig. 1(b)] is examined. The area of the top view is given as [d_a and d_b are defined by Eqs. (B3) and (B4)]:

$$A_{\text{top}} = \frac{d_a^2}{2} + \frac{\pi d_a^2}{8}. \tag{1}$$

The first term in the expression corresponds to the area of the half square (in the left part of the top view) and the second term corresponds to the semicircle (in the right part of the top view). Next, the volume of the filter unit at this stage is determined. Note that the volume at this stage corresponds to only the top part of the filter unit because only that part of the unit is designed by this rule. The volume is given as:

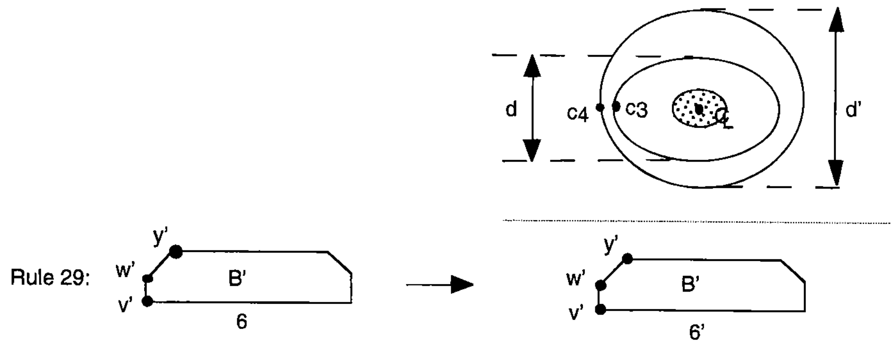
$$V_{\text{top}} = \left(\frac{d_a^2}{2} + \frac{\pi d_a^2}{8} - \frac{\pi d_b^2}{4}\right) \times 2. \tag{2}$$

The first two terms in the parentheses correspond to the total area of the filter unit. The third term in the parentheses corresponds to the hollow area of the filter unit (based on a 2-mm wall thickness). Thus, the quantity in the parentheses corresponds to the total solid area of the filter unit. This quantity multiplied by the height (again equal to 2 mm) is the volume of the top part of the filter unit at this stage. These two equations are the same as Eqs. (B15) and (B16).

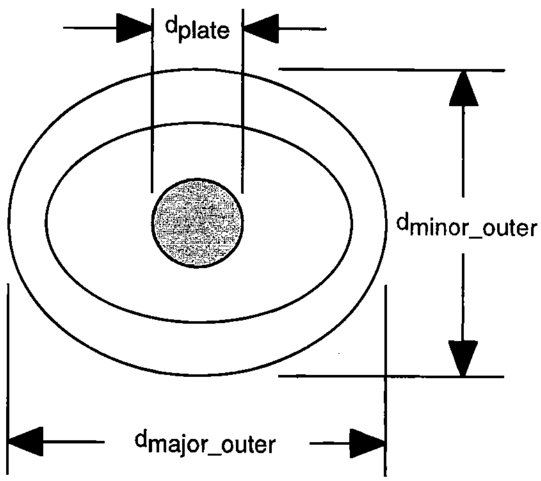
As another example, consider the shape rule shown in Figure 2(a). This rule corresponds to Rule 29 of the coffeemaker grammar and is used to design an elliptical base unit (as opposed to a polygonal unit). To determine the projected area of the base unit generated by this rule, the top view shape (Fig. 2b) is examined. The projected area is given by [the diameters are defined by Eqs. (B37)–(B39)]:[1]

$$\text{Area} = \pi(d_{\text{major\_outer}} \times d_{\text{minor\_outer}} - d_{\text{plate}}^2). \tag{3}$$

[1] These expressions have been derived by assuming that the heater plate is circular. This assumption, while not necessary, was found to be valid in all commercially available designs.



(a)



(b)

Fig. 2. (a) Rule generating a base unit; (b) top view of the unit generated by the rule.

This equation is the same as Eq. (B40). Equations corresponding to the other shape rules can similarly be derived and are listed in Appendix B.

4. USING COST EXPRESSIONS TO GUIDE THE DESIGN GENERATION PROCESS

This section demonstrates how the expressions derived in the previous section can be used to guide the design generation process for coffeemakers. The methodology will be illustrated by following an example generation sequence. The first set of rules that are applied to the initial shape shown in Figure 3(a) distinguish between the two main classes of coffeemakers, those with one heating element and those with two. They also break apart the space around the initial shape into three regions, corresponding to the filter, base, and water storage units, and design the basic cross-sectional shapes of the filter and the base units. The rule designing a one-heater unit (signified by the square label), for example, is shown in Figure 3(b); a similar rule designs a two-heater unit. Next the filter, the base, and the water storage units are designed separately. For each of these three units, the various shape rules ensure that form design is carried out within the context of function design, that is, only forms that do not violate any functional specifications can be created. This is done by first applying the function design rules that add labels to the shape based on the required functional specifications and then using the various form design rules, based on the labels, to create the actual shapes. Because the function design rules do not directly create 3D shapes, they add no cost to the design. Note that the decisions made during the function design will strongly influence the cost of the product; they just do not directly add cost to the design.

First, the form design of the filter unit is carried out based on the functional specifications. This step creates new shapes and modifies existing ones and thus cost equations are associated with these form design rules. The rule shown in Figure 4, for example, generates a rotating filter. This rule

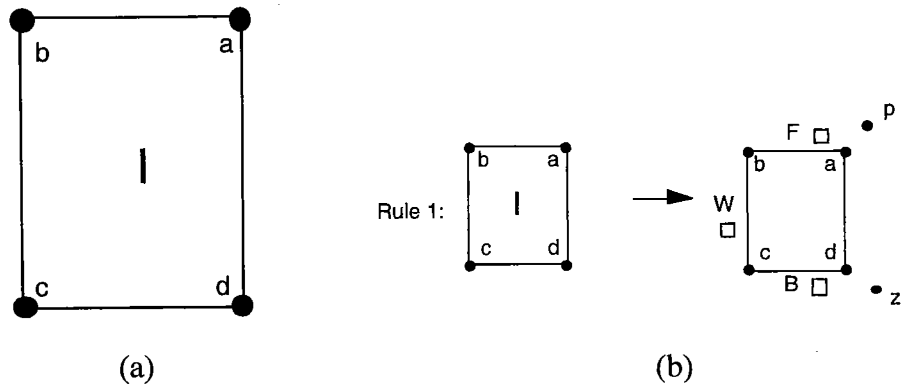


Fig. 3. (a) Initial shape for the example generation sequence; (b) rule designing a one-heater unit.

also creates a top view for the filter unit and thus converts it into a 3D object. The shape around the filter unit obtained after this rule is applied to the evolving shape is shown in Figure 5. Note that the shape generated has a top view and a side view (a single view is defaulted to be a side view; multiple views separated by a hairline are the top and the side views). The labels correspond to the various functional attributes of the filter unit: FT_1 signifies a conical filter, FI corresponds to the inlet tube, FF_1 means that the coffee flow rate cannot be changed, and FS_1 signifies that a flow stop mechanism is present in the coffeemaker.

The first step in associating the manufacturing cost with this rule is to calculate the volume and projected area of the created shape. The area and volume calculated are then used along with Eqs. (A1), (A3), and (A4) to calculate the material and equipment operating cost. Note that the area and volume are calculated only for the top part of the filter unit (based on a 2-mm wall thickness mentioned earlier) because the rest of the unit has not been completely designed yet. The expressions for the area (A) and the volume (V_{top}) are given by Eqs. (B17) and (B18). It must be reiterated that these equations have been determined solely based on the geometry of the shapes created by this shape rule and are independent of the rest of the design. The incremental totals for the cost of the filter unit for the example coffeemaker are now determined. In addition to the material and equipment operating costs, the cost at this stage also includes tooling, burden, and labor costs. From the geometry of the shape, $d_a = 120$ mm and $d_b = 116$ mm. Using Eq. (B17), the area of the top of the filter is given by $A = 12081$ mm^2. The equipment size that is required is now calculated from Eq. (A3) and is equal to 3200 kN. The tooling cost based on an annual volume of 1,000,000 parts [Eq. (A2)] is thus \$0.015. The operating cost, calculated from Eq. (A4) based on a cycle time of 30 s, is \$0.165. The labor and burden costs [Eqs. (A5) and (A6), respectively] are \$0.015 and \$0.018. To calculate the material cost, first the volume of material used needs to be calculated. Using Eq. (B18), $V_{top} = 3027.96$ mm^3. The cost of the material required can now be calculated using Eq. (A1), and is equal to \$0.002 for this particular design. These costs can be used by the designer to get some indication of the final cost of the filter unit even before it is designed.

The design of the filter unit is completed by applying the other filter form design rules based on the function labels previously associated with the design. The filter design sequence is shown in Figure 6, where labels are omitted for

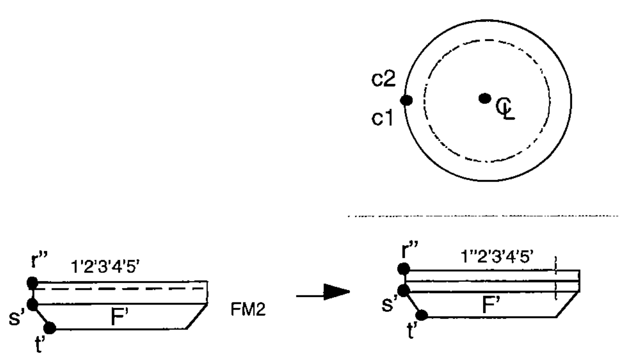


Fig. 4. Shape rule designing a rotating filter.

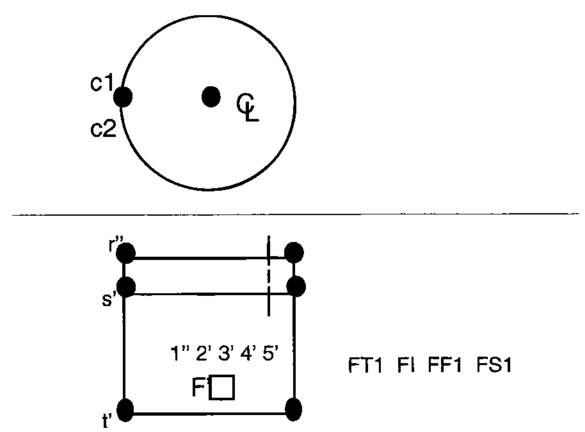


Fig. 5. Shape after the design of a rotating filter unit.

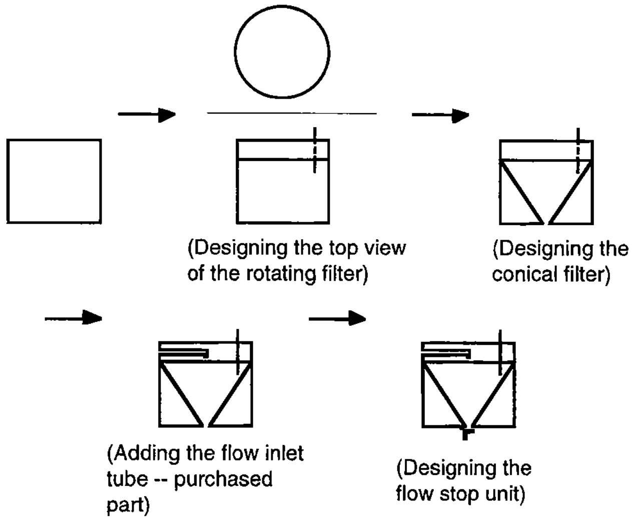


Fig. 6. Filter design sequence for the example coffeemaker.

Table 1. *Incremental cost (in $) of the filter unit for the example coffeemaker at each stage during the design sequence*

Design Step	Material Cost	Equipment Operating Cost	Tooling Cost	Burden	Labor Cost
Top view	0.003	0.165	0.015	0.018	0.015
Conical filter	0.115	0.033	0.013	0	0.003
Flow stop	0.027	0.045	0.009	0.003	0.008

clarity. The manufacturing costs associated with each design step are shown in Table 1. The first row corresponds to the design of the top of the filter (discussed above), whereas the second row corresponds to the design of the conical filter. Because the same machine is used both for manufacturing the top of the filter and the conical cup, the burden cost is added only once. The third row corresponds to the design of the flow stop mechanism. The costs in the three rows of Table 1 must be added together to determine the final cost of the filter unit and it is equal to $ 0.472. Note that because the inlet tube is a purchased part, no cost is added for that step.

The cost of $ 0.472 for the filter unit of the coffeemaker is derived independent of the rest of the design (which, in fact, has not even been completed at this stage). This provides useful feedback to the designer about the cost of the design and can be used to guide further design decisions and suggest directions for redesign. For example, if the designer decides (based on the feedback provided by the technique) that the cost of the unit is too high, one possible change might be to remove the flow stop mechanism. This would then bring down the cost of the filter unit to $ 0.38 (the first two rows of Table 1). Note that this change in the design (and therefore the cost) of the filter unit can be made at this stage itself, rather than at the end of the design cycle when redesign might be more expensive. While removing the flow stop mechanism from the filter unit is a rather obvious choice for reducing the cost of the unit, the same procedure can be applied to study more complex trade-offs.

Continuing with the design process, the next step is the design of the base unit, which follows the same procedure as the design of the filter unit. The base design sequence for the example coffeemaker is demonstrated in Figure 7. A smooth blend is assumed between the top and bottom planes of the base unit. The manufacturing costs are calculated using Eqs. (B37)–(B41), (B43), and (B69). The costs obtained are shown in Table 2.

Again, suppose the designer wants to make changes that will bring down the cost of the base unit. There are various possible changes that can be explored to determine their effect on the cost of the base unit (and, therefore, the cost of the coffeemaker). One alternative is to use a cylindrical base

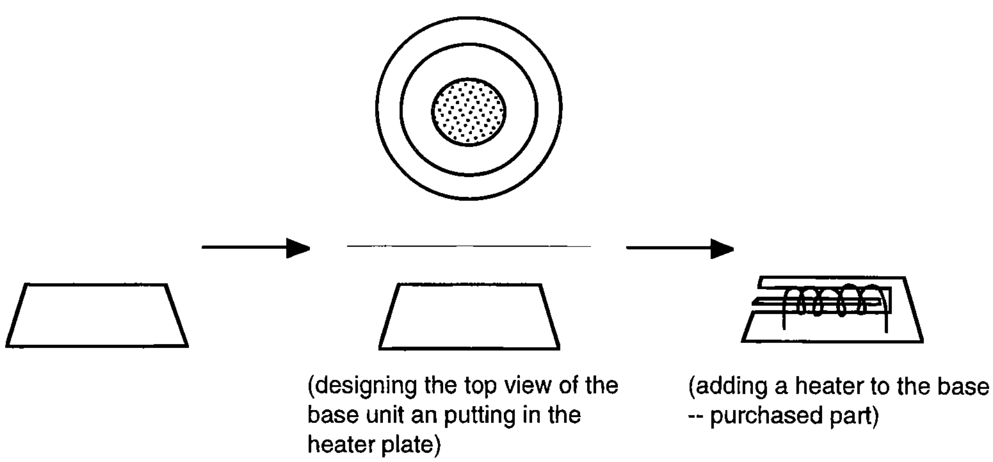


Fig. 7. Base design sequence for the example coffeemaker.

Table 2. *Incremental cost (in $) of the coffeemaker at each stage during the base design sequence*

Design Step	Material Cost	Equipment Operating Cost	Tooling Cost	Burden	Labor Cost
Top view	0.102	0.264	0.022	0.018	0.024
Heater plate	0.019	0.039	0.015	0.004	0.018

Table 3. *Incremental costs (in $) of the modified base unit*

Design Step	Material Cost	Equipment Operating Cost	Tooling Cost	Burden	Labor Cost
Top view	0.073	0.166	0.022	0.007	0.024
Heater plate	0.019	0.039	0.015	0.004	0.018

unit similar to that shown in the base design sequence of Figure 8. It is not immediately clear whether such a change will reduce the cost of the base unit or increase it. However, the equations discussed above can be used to determine the costs shown in Table 3. By examining the table, the designer can determine not only that the cost of the base unit will decrease due to the proposed change, but the exact magnitude of that change is also known ($0.387 instead of $0.525). If the designer feels that the reduction in cost is significant, then the change can be accepted; otherwise the original designer preference of a tapered base unit can be preserved. Note, however, that these decisions can be made at this stage, rather than waiting until the design of the entire coffeemaker is complete to determine their effect. It is information like this that helps the designer in making the appropriate choices with respect to rule selection, highlighting the value of this approach.

The last stage in the design of a coffeemaker is the creation of a water storage unit that satisfies all functional requirements and blends the three units together into a final product. To do this, the top view cross sections of the water storage unit are generated on four horizontal planes (at the top and bottom of the base and the filter units). The cross sections on these four planes are then blended together in the vertical direction to create the final 3D shape of the water storage unit, which also integrates together all the units of the product. The cross section on each plane is generated by merging together shapes created by sweeping a desired number of squares and circles in a designer-specified manner. One such rule that sweeps a square about the center of the filter is shown in Figure 9. The dimensions of the square as well as the distance from the center are specified by the designer and can change as a function of the sweep angle. This process imparts the grammar with an ability to generate shapes not commonly seen in commercial products as well as the more traditional ones.

For the example, the sweep and merge sequence generating the water storage unit cross sections is shown in Figure 10. The manufacturing costs at the various stages of the sweep sequence (on a plane) are shown in Table 4. It should be noted that each row corresponds to the cost of the water storage unit after the respective design step, and the costs at the end of step 2 must not be added to those at the end of design step 1 to obtain a total cost. This is because the shape at the end of step 1 is just an intermediate shape and the costs corresponding to that shape are relevant only at that step. Once a new shape is created by step 2, the costs must be updated. This is different than the design of the other units of the coffeemaker where, once a part is created, it is not modified and thus the costs calculated after the application of each rule remain valid throughout the process.

The 3D shape of the final product is shown in Figure 11. It is similar to a Rowenta FK26-S coffeemaker shown in Figure 12. The final cost of the coffeemaker estimated by using the method discussed above is $7.32 (cost of manu-

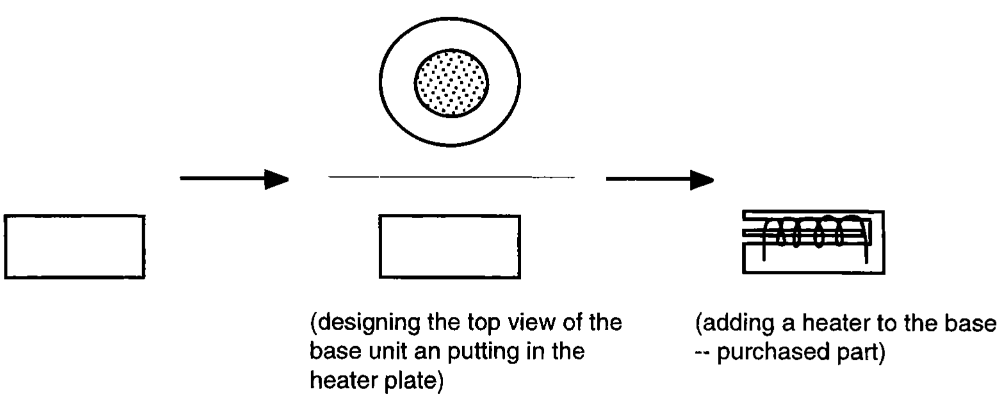


Fig. 8. A modified base design sequence.

Table 4. *Incremental cost (in $) of the water storage unit for the example coffeemaker at each stage during its generation*

Design Step	Material Cost	Equipment Operating Cost	Tooling Cost	Burden	Labor Cost
Step 1	0.123	0.199	0.035	0.010	0.024
Step 2 (Final design)	0.111	0.199	0.035	0.010	0.024

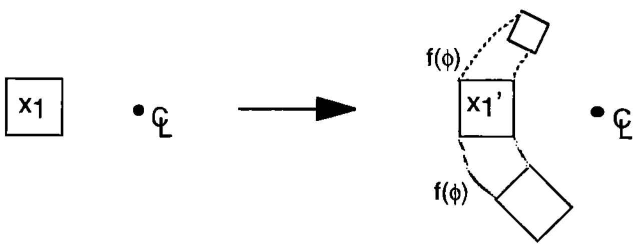


Fig. 9. Representative sweep rule.

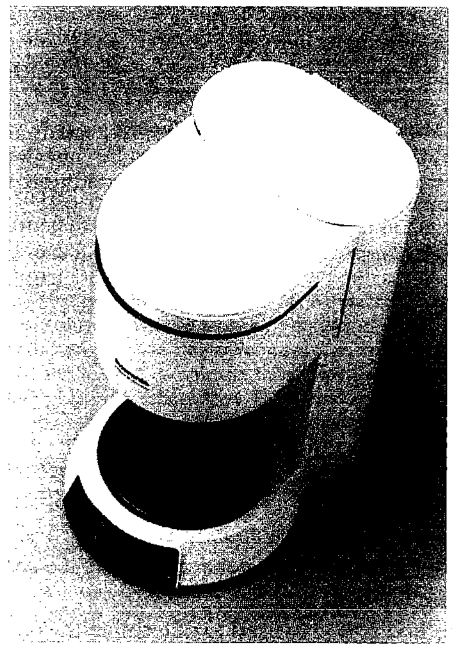


Fig. 12. Rowenta FK26-S coffeemaker.

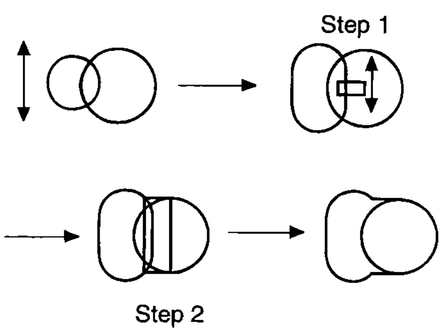


Fig. 10. Sweep sequence generating the water storage unit of the example coffeemaker.

factured and purchased parts and the cost of assembly). The cost of the Rowenta FK26-S estimated by Ulrich and Pearson (1993) is $7.09, which is within 3% of the cost generated by using the shape grammar. Further, and perhaps more critically, the shape grammar costing method also provides incremental costs at each stage of the design process in addition to estimating the cost of the final product.

Suppose, however, that the designer decides to use a new water storage sweep sequence shown in Figure 13. The costs resulting from this sequence are shown in Table 5 (again, only the last row must be used).

If the designer, based on the feedback received at each stage, chooses to accept all the design changes discussed above, then the design shown in Figure 14 results. It is similar to the Rowenta FG22-O coffeemaker shown in Fig-

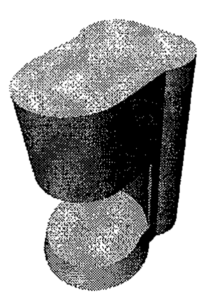


Fig. 11. Coffeemaker generated by the choice of shape rules.

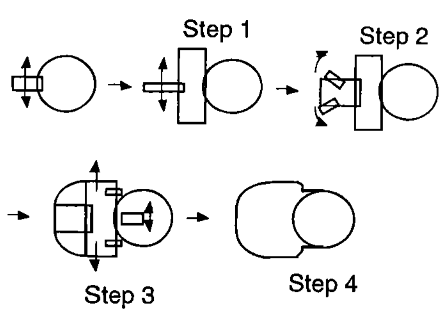


Fig. 13. Modified water storage design sequence.

}

if $\text{line}_{\text{top}} = 22$ and $\text{line}_{\text{bot}} = 12$

{

$$c_{\text{initial}} = \text{diagonal}_{\text{initial}} \quad \text{(B239)}$$

$$c_{\text{final}} = d_{\text{final}} \quad \text{(B240)}$$

$$\text{end\_area} = d_{\text{initial}} \cdot h_{\text{initial}}/2 \quad \text{(B241)}$$

}

if $\text{line}_{\text{top}} = 22$ and $\text{line}_{\text{bot}} = 22$

{

$$c_{\text{initial}} = d_{\text{initial}} \quad \text{(B242)}$$

$$c_{\text{final}} = d_{\text{final}} \quad \text{(B243)}$$

$$\text{end\_area} = d_{\text{initial}} \cdot h_{\text{initial}} \quad \text{(B244)}$$

}

$$s_{\text{initial}} = (a_{\text{initial}} + b_{\text{initial}} + c_{\text{initial}})/2 \quad \text{(B245)}$$

$$s_{\text{final}} = (a_{\text{final}} + b_{\text{final}} + c_{\text{final}})/2 \quad \text{(B246)}$$

$$\text{triangle}_{\text{initial}} = (s_{\text{initial}} \cdot (s_{\text{initial}} - a_{\text{initial}}) \cdot (s_{\text{initial}} - b_{\text{initial}}) \cdot (s_{\text{initial}} - c_{\text{initial}}))^{.5} \quad \text{(B247)}$$

$$\text{triangle}_{\text{final}} = (s_{\text{final}} \cdot (s_{\text{final}} - a_{\text{final}}) \cdot (s_{\text{final}} - b_{\text{final}}) \cdot (s_{\text{final}} - c_{\text{final}}))^{.5} \quad \text{(B248)}$$

if $\text{line}_{\text{bot}} = 11$

{

$$\text{sweep area} = \Sigma(\text{integral}_{\text{top}} - \text{integral}_{\text{bot}} + \text{triangle}_{\text{initial}} - \text{triangle}_{\text{final}} + \text{end\_area}) \quad \text{(B249)}$$

}

if $\text{line}_{\text{bot}} = 12$

{

$$\text{sweep area} = \Sigma(\text{integral}_{\text{top}} - \text{integral}_{\text{bot}} + \text{triangle}_{\text{initial}} + \text{triangle}_{\text{final}} + \text{end\_area}) \quad \text{(B250)}$$

}

if $\text{line}_{\text{bot}} = 22$

$$\text{sweep area} = \Sigma(\text{integral}_{\text{top}} - \text{integral}_{\text{bot}} - \text{triangle}_{\text{initial}} + \text{triangle}_{\text{final}} + \text{end\_area}) \quad \text{(B251)}$$

}

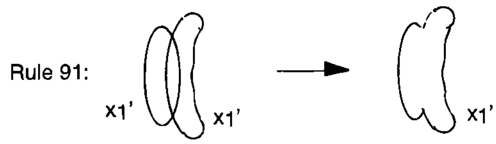


Fig. B10. Representative merge rule.

$$\text{sweep perimeter} = \Sigma\{\text{int}(ds_{\text{top}}, \phi_{\text{top}} = \text{lower}_{\text{top}}..\text{upper}_{\text{top}}) + \text{int}(ds_{\text{bot}}, \phi_{\text{bot}} = \text{lower}_{\text{bot}}..\text{upper}_{\text{bot}})\} \quad \text{(B252)}$$

Water storage unit area and volume equations

$$\text{water storage unit area} = \Sigma\{\text{sweep area}_i\} - \Sigma\{\text{sweep area}_i \cap \text{sweep area}_k\} - \Sigma\{\text{sweep area}_i \cap \text{filter area}\} + \text{filter area} \quad \text{(B253)}$$

$$\text{water storage unit volume} = [2 \cdot \Sigma\{\text{sweep perimeter}_i\} - 2 \cdot \Sigma\{\text{sweep perimeter}_i \cap \text{sweep perimeter}_k\} - 2 \cdot \Sigma\{\text{sweep perimeter}_i \cap \text{filter perimeter}\}] \cdot |y'_b - y'_c| + 2 \cdot \text{filter area} + \text{filter perimeter} \cdot |y'_r - y'_s| \quad \text{(B254)}$$

(i from 1 to number of sweeps $-$ 1; k from $i+1$ to number of sweeps)

Base unit area and volume equations

$$\text{base unit area} = 2 \cdot \Sigma\{\text{sweep perimeter}_i\} - 2 \cdot \Sigma\{\text{sweep perimeter}_i \cap \text{sweep perimeter}_k\} - 2 \cdot \Sigma\{\text{sweep perimeter}_i \cap \text{base perimeter}\} + \text{base area} \quad \text{(B255)}$$

$$\text{base unit volume} = [2 \cdot \Sigma\{\text{sweep perimeter}_i\} - 2 \cdot \Sigma\{\text{sweep perimeter}_i \cap \text{sweep perimeter}_k\} - 2 \cdot \Sigma\{\text{sweep perimeter}_i \cap \text{base perimeter}\}] \cdot |y'_c - y'_z| + \text{base volume} \quad \text{(B256)}$$

(i from 1 to number of sweeps $-$ 1; k from $i+1$ to number of sweeps)

Rule 97: Rule 98:

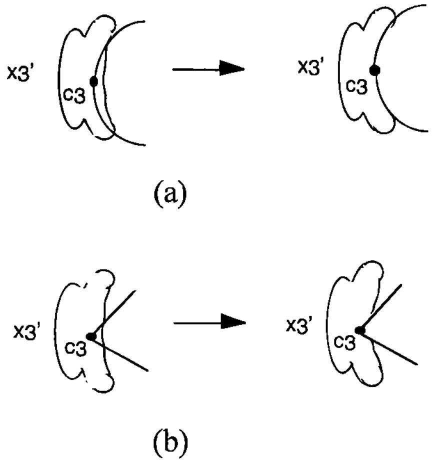


Fig. B11. (a) and (b) Representative rules depicting the merging of the base unit with the water storage unit.

Lid area and volume equations

$$\text{grate area} = 0.5\cdot(\Sigma\{\text{sweep area}_i\} - \Sigma\{\text{sweep area}_i \cap \text{sweep area}_k\}) \quad \text{(B257)}$$

$$\begin{aligned}\text{lid area (grate)} = {} & \Sigma\{\text{sweep area}_i\} \\ & - \Sigma\{\text{sweep area}_i \cap \text{sweep area}_k\} \\ & - \Sigma\{\text{sweep area}_i \cap \text{filter area}\} \\ & + \text{filter area} - \text{grate area} \end{aligned} \quad \text{(B258)}$$

$$\begin{aligned}\text{lid volume (grate)} = {} & [2\cdot\Sigma\{\text{sweep perimeter}_i\} \\ & - 2\cdot\Sigma\{\text{sweep perimeter}_i \cap \text{sweep perimeter}_k\} \\ & - 2\cdot\Sigma\{\text{sweep perimeter}_i \cap \text{filter perimeter}\}] \\ & \cdot|y_r' - y_f'| + 2\cdot\text{lid area} \end{aligned} \quad \text{(B259)}$$

(i from 1 to number of sweeps $-$ 1; k from $i+1$ to number of sweeps)

$$\begin{aligned}\text{lid}_1\text{ area (hinged)} = {} & \Sigma\{\text{sweep area}_i\} \\ & - \Sigma\{\text{sweep area}_i \cap \text{sweep area}_k\} \\ & - \Sigma\{\text{sweep area}_i \cap \text{filter area}\} \end{aligned} \quad \text{(B260)}$$

$$\begin{aligned}\text{lid}_2\text{ area (hinged)} = {} & [2\cdot\Sigma\{\text{sweep perimeter}_i\} \\ & - 2\cdot\Sigma\{\text{sweep perimeter}_i \cap \text{sweep perimeter}_k\} \\ & - 2\cdot\Sigma\{\text{sweep perimeter}_i \\ & \quad \cap \text{filter perimeter}\}] + \text{filter area} \end{aligned} \quad \text{(B261)}$$

$$\text{lid}_1\text{ volume (hinged)} = 2\cdot(\text{lid}_1\text{ area}) \quad \text{(B262)}$$

Rule 95:

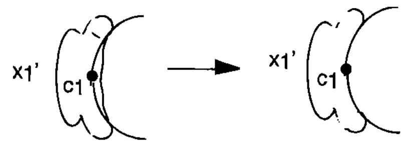


Fig. B12. Representative rule depicting the merging of the filter unit with the water storage unit.

$$\begin{aligned}\text{lid}_2\text{ volume (hinged)} = {} & [2\cdot\Sigma\{\text{sweep perimeter}_i\} \\ & - 2\cdot\Sigma\{\text{sweep perimeter}_i \cap \text{sweep perimeter}_k\} \\ & - 2\cdot\Sigma\{\text{sweep perimeter}_i \cap \text{filter perimeter}\}] \\ & \cdot|y_r' - y_f'| + 2\cdot\text{filter area} \end{aligned} \quad \text{(B263)}$$

Manish Agarwal is a Ph.D. candidate in the department of Mechanical Engineering at Carnegie Mellon University. His research is in the area of design theory and methodology including product design, qualitative optimization and formal design techniques. He has been involved in developing shape-grammar-based design representations for a number of application domains. Manish received his B.Tech. from the Indian Institute of Technology, Kanpur, India in 1995 and his M.S. from Carnegie Mellon University in 1997, both in Mechanical Engineering. He is a member of Sigma Xi National Honor Society and the ASME and SDPT Professional Societies.

Jonathan Cagan is a Professor of Mechanical Engineering at Carnegie Mellon University, with appointments in Biomedical Engineering and Computer Science. His research, teaching, and consulting are in the area of design theory, methodology, automation, and practice. He received his B.S. in 1983 and M.S. in 1985 from the University of Rochester, and his Ph.D. in 1990 from the University of California at Berkeley, all in Mechanical Engineering. Dr. Cagan is the recipient of the National Science Foundation's NYI Award and the Society of Automotive Engineer's Ralph R. Teetor Award for Education. He is a member of the Phi Beta Kappa, Tau Beta Pi, and Sigma Xi National Honor Societies, and the ASME, AAAI, SAE, and ASEE Professional Societies. Dr. Cagan is a registered Professional Engineer.

Kathy Constantine received a B.S. in Mechanical Engineering from Cornell University in 1992. She spent several years working in various manufacturing plants and completed assignments in process engineering, quality control, safety, and product design. In 1997 she received a M.S. in Mechanical Engineering from Carnegie Mellon University. Since then she has been developing and building manufacturing equipment.