of components, extending previous technology that requires fixed-size components. The method is used to conceptualize the layout of the cross-section of an automatic transmission.
[DOI: 10.1115/1.1637663]

Layout Optimization of Shapeable Components With Extended Pattern Search Applied to Transmission Design

Su Yin
e-mail: yin@andrew.cmu.edu
Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Jonathan Cagan
e-mail: cagan@cmu.edu
Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Peter Hodges
e-mail: phodges1@ford.com
Product Design Engineer, Vehicle Systems Engineering, Ford Motor Company, Dearborn, MI 48121, USA

An extended pattern search algorithm is presented for the placement of shapeable components within a product layout. Shapeable octrees are used to evaluate the amount of overlap between pairs

1 Introduction

In previous work, Yin and Cagan [1] introduced an Extended Pattern Search (EPS) algorithm for component layout that demonstrated a marked improvement over the previous state-of-the-art. Component placement or intelligent packaging finds application in many different applications. A summary of those applications and a review of the literature in this area can be found in the survey by Cagan, et al. [2]. In the previous work on the EPS method, all components had fixed-sized geometry, required to create octrees used for hierarchical geometry representation of fixed shapes. Many applications, however, require geometries to change shape during their placement. One such example is in automatic transmission layout where clutches must be sized for each location based on performance requirements.

In this note, the EPS algorithm is extended to include shapeable components. This note also introduces a new problem to the layout literature with the conceptual design of automatic transmissions. First the paper briefly summarizes the EPS algorithm. Next *shapeable octrees* are introduced. Finally the automatic transmission problem is presented and solved using the shapeable component EPS algorithm.

2 The Extended Pattern Search Layout Algorithm

2.1 The Extended Pattern Search Algorithm.
The basic pattern search algorithms (Hooke and Jeeves [3]; Torczon and Trosset [4]) search for optimal solutions in the design space following a series of exploratory moves that rely exclusively on the direct comparison of function values, thus providing a tool suitable for the exploration of a design space that is nonlinear and discontinuous. Yin and Cagan [1] extended the basic PS algorithm to a stochastic method and applied it to the layout problem.

In the Extended Pattern Search layout algorithm, the order of component selection is randomized and each component is moved through a translation pattern until improvement is found. The same is repeated for each component for the rotation patterns. If at the end of this process no improvement is found, one or more components are swapped, assuming that they may improve the objective function. The step size is reduced and the process iterates through again, until a minimum step size is reached. Occasionally throughout the algorithm, the step size is increased based on the algorithm's progression and performance.

The randomization of component ordering, the judicious use of the swapping move, and the occasional step jumps all create a stochastic algorithm able to jump out of local, inferior optima. Details can be found in Yin and Cagan [1].

3 Octree Representation for Shapeable Components

Overlap between components is allowed and penalized during the layout since the presence of overlap is found to be essential for obtaining good solutions to the layout problem (De Bont et al. [5]). The exact interference calculation is time consuming when components are of complex geometry and in close proximity, typical for mechanical devices.

Within the EPS algorithm, an octree representation is used to model components of complex shapes and to evaluate interference between components. The octree-based interference detection is much simpler than the exact overlap evaluation. Multi-resolution octrees (Kolli et al., [6]; Cagan et al., [7]) are used in the layout
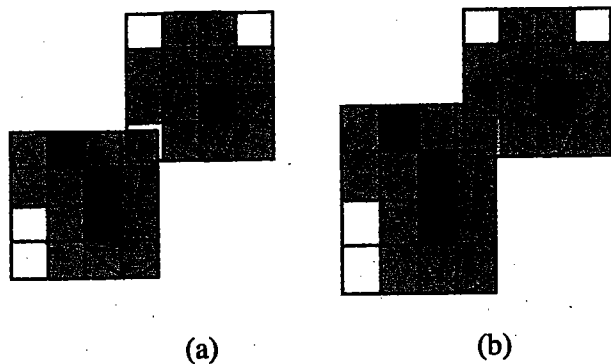
Fig. 1 Shapeable Component Quadtree Representations: (a) No Interference before Scaling; (b) Interference after Scaling.

Fig. 3 A Stick Diagram of Component Connections in a Transmission

for interference evaluation with different levels of accuracy; the resolution level is increased as the algorithm progresses.

A shapeable octree representation is introduced to evaluate the amount of interference between shapeable components. Shapeable components with base geometry of any default size can be imported from commercially available CAD packages. Octrees are generated from components of the base geometry. Each octree has a local coordinate frame attached to it. By keeping track of the movement of coordinate frames and mapping one to the other, we can perform interference analysis of components under translation and rotation moves in much th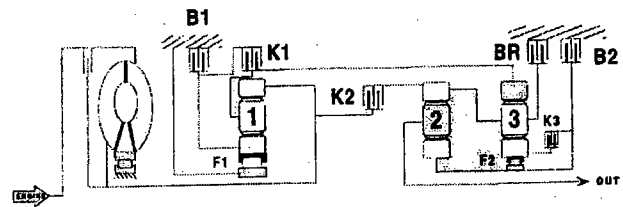e same way as done with fixed-size octrees. A 3-D vector is associated with each octree model to provide the scaling along local axes for shapeable components. The algorithm resizes a shapeable component whenever it is moved and provides the updated scaling factor for the interference analysis.

Figure 1 illustrates the shapeable component representation and the interference calculation process using quadtree models (the 2-D equivalent of 3-D octrees). In Fig. 1a the components are represented with quadtree models of level 2 using the default sizes and there is no overlap detected. In Fig. 1b one of the components is resized and overlap is detected between the partially full (gray) quadrants of the components.

In the EPS algorithm, after every component move, the sizes of shapeable components are adjusted and the starting and the ending points on routes are updated as necessary. The shapeable component representation can be used in layout applications where the parametric design of the components is necessary.

4 Automatic Transmission Design Example

The layout of the cross-section of an automatic transmission for a motor vehicle is a complex problem which involves the design and placement, in the cross-section, of many components including a system of clutches, and one or more planetary gear sets.

Automated cross-section layout tools remain unavailable for the *layout* of the cross-section of an automotive automatic transmission. Specialized tools have automated the development of stick diagrams to represent kinematics (Chatterjee and Tsai, [8]) or the imaging of planetary gear trains in 3D directly from the topological graph representation (Ramakrishnan and Olson, [9]), but none have addressed the automated placement of components within the transmission box.

Figure 2a illustrates the cross-section of a typical automatic transmission assembly and Fig. 2b shows the components and the container space used in the layout problem discussed in this paper. Only half of the cross-section needs to be modeled because it is axially symmetric.

Of the components included in the layout, planetary gear sets are considered to have pre-defined shapes and sizes that do not change during the layout. Multiple-disc clutches, however, are shapeable components whose sizes are related to their positions in the assembly and are to be determined during the layout.

The number of frictional discs and the size of the discs in a clutch are determined by the torque capacity, thermal capacity, and gain requirements, and they are related to the location of a clutch in the transmission assembly. The design requirements have to be checked and the clutch sizes adjusted accordingly during the layout process. The design considerations for clutches are from industry transmission design practices and can be found in Yin [10].

5 Problem Formulation

The transmission layout problem is formulated as a multi-objective optimization problem. The goal is to find a suitable sizing and placement of the components and the shell connections between them that satisfy functional requirements and achieve a compact package.
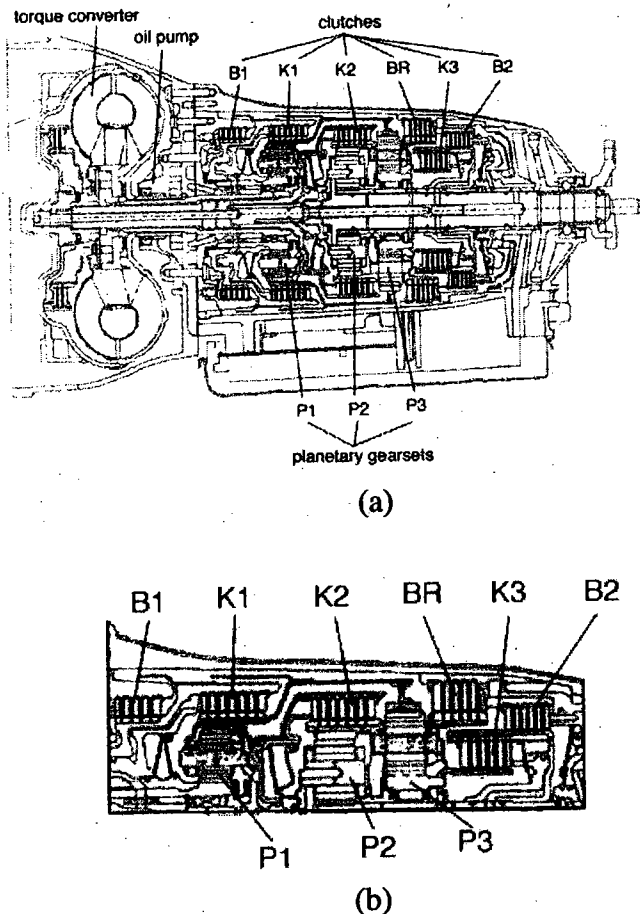

Fig. 2 An Automatic Transmission Assembly: (a) The Cross-Section View; (b) The Layout Space and Components.
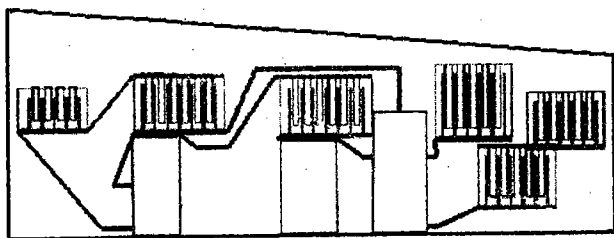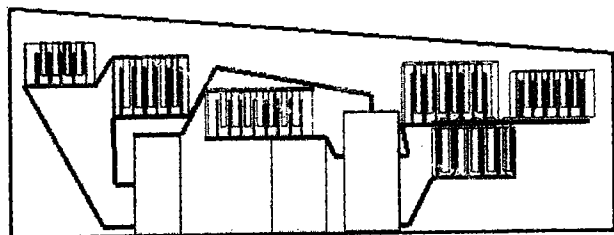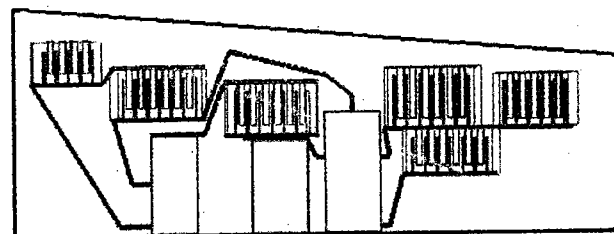
Fig. 4 The Original Transmission Layout—A Simplified Representation

(a)

(b)

Fig. 5 Two Solutions to the Transmission Layout Problem

The design variables include positions of components, locations of intermediate points on the shell connections, and sizes of shapeable components—the clutches.

The objective function is formulated as a weighted sum of individual goals:

$$\text{Minimize} \quad f = w_1 f_{placement} + w_2 f_{routing} + w_3 f_{penalty}. \quad (1)$$

The placement objective $f_{placement}$ is composed of attraction terms for components to stay inside the container, penalty terms for container protrusion and component overlap, and the bounding-box area as an indication of the compactness of packing. The routing objective $f_{routing}$ consists of routing costs in terms of the shell connection length and the number of bends, and also a penalty term for the number of crossings between different shells. The penalty objective $f_{penalty}$ specifies constraint violations and thus drives the design into feasible ranges as the algorithm progresses. As presented in Szykman and Cagan [11], a penalty approach instead of a constraint propagation approach is taken since invalid designs with overlap and constraint violations are allowed during the optimization for a better exploration of the design space. The objective function is evaluated after each exploratory move and the resizing of shapeable components.

A simplified clutch representation is used in the layout here, but the shapeable octree can handle more complex geometries as more detailed representations are used downstream.

Topological connections between components are defined in a stick diagram, for this example shown in Fig. 3. Figure 4 trans-

Table 1 Comparing the Newly Generated Layouts with the Actual Design

# of Discs	K1	K2	K3	B1	B2	BR	OBJECTIVE
Figure 4	6	6	5	4	5	5	280.9
Figure 5(a)	5	7	5	4	5	6	240.7
Figure 5(b)	6	6	6	4	5	6	228.1

lates the original layout (Fig. 2b) into a simplified representation appropriate for conceptual layout design using our clutch and routing models.

6 Results

The automated layout tool using the EPS algorithm and a shapeable octree representation is coded in C++ and runs on major UNIX and PC platforms. The algorithm was run multiple times with consistent objective function values; each run takes an average of 5 seconds on a Silicon Graphics Indigo2 (195MHz, MIPS R10000 CPU, 128MB RAM). Two alternative solutions to the transmission layout problem (Fig. 4) are shown in Fig. 5. The solutions demonstrate similar configurations to the actual layout but with somewhat different positions and sizes for the clutches.

Table 1 summarizes the number of discs chosen for the six clutches in the actual and the newly generated layouts and compares their objective function values. As well, disk radii are sized dynamically as they are moved during layout optimization. The two alternative solutions are effective designs since the objectives are minimized, clutches properly sized, and routing restrictions from Fig. 3 are satisfied. The two newly generated solutions are better than the actual design *according to our problem formulation* as indicated by the objective function values shown in Table 1, although the design objectives used by the original designers are unknown.

7 Conclusions

The extended pattern search layout technique combined with the shapeable octree representation provides an effective tool for the automatic transmission layout and other design where components must be dynamically sized. Although the example used only two dimensions, the method is applicable for general 3D design. Future work on the transmission layout problem includes using the actual clutch geometry with exact piston and spring designs instead of the simplified models.

Acknowledgments

References

[1] Yin, S., and Cagan, J., 2000, "An Extended Pattern Search Algorithm for Three-Dimensional Component Layout," ASME J. Mech. Des., **122**(1), pp. 102–108.

[2] Cagan, J., Shimada, K., and Yin, S., 2002, "A Survey of Computational Approaches to Three-dimensional Layout Problems," Comput.-Aided Des., **34**(8), pp. 597–611.

[3] Hooke, R., and Jeeves, T. A., 1961, "Direct Search Solution of Numerical and Statistical Problems," J. Assoc. Comput. Mach., **8**(2), pp. 212–229.

[4] Torczon, V., and Trosset, M., 1997, "From Evolutionary Operation to Parallel Direct Search: Pattern Search Algorithms for Numerical Optimization," Computing Science and Statistics, **29**.

[5] De Bont, F. M. J., Aarts, E. H. L., Meehan, P., and O'Brien, C. G., 1988, "Placement of Shapeable Blocks," Philips J. Res., **43**, pp. 1–22.

[6] Kolli, A., Cagan, J., and Rutenbar, R. A., 1996, "Packing of Generic, Three Dimensional Components Based on Multi-Resolution Modeling," Proceedings of the 22nd ASME Design Automation Conference (DAC-1479), Irvine, CA, August 19–22.

[7] Cagan, J., Degentesh, D., and Yin, S., 1998, "A Simulated Annealing-Based

Algorithm Using Hierarchical Models for General Three-Dimensional Component Layout," Comput.-Aided Des., **30**(10), pp. 781–790.

[8] Chatterjee, G., and Tsai, L. W., 1996, "Computer Aided Sketching of Epicyclic-Type Automatic Transmission Gear Trains," ASME J. Mech. Des., **118**(3), pp. 405–411. Ramakrishnan, B., and Olson, D. G., 1989, "Efficient Approach for Generating Solid-Shaded Images of Planetary Gear Trains," ASME Computers in Engineering Conference and Exposition, Jul 30–Aug 3, 1989, pp. 111–116.

[9] Ramakrishnan, B., and Olson, D. G., 1989, "Efficient Approach for Generating Solid Shaded Images of Planetary Gear Trains," ASME Computers in Engineering Conference and Exposition, Jul 30–Aug. 3, 1989.

[10] Yin, S., 2000, "A Pattern Search-Based Algorithm for Automated Product Layout," Ph.D. Dissertation, Carnegie Mellon University, Pittsburgh, PA.

[11] Szykman, S., Cagan, J., and Weisser, P., 1998, "An Integrated Approach to Optimal Three Dimensional Layout and Routing," ASME J. Mech. Des., **120**(3), pp. 510–512.