

DETC 2000/DAC-14228

DECOMPOSITION-BASED ASSEMBLY SYNTHESIS BASED ON STRUCTURAL CONSIDERATIONS

F. Arkin Yetis and Kazuhiro Saitou*

Department of Mechanical Engineering and Applied Mechanics
University of Michigan
Ann Arbor, MI 48109-2125, USA
E-mail: {ayetis,kazu}@engin.umich.edu

ABSTRACT

This paper presents a method for decomposition of structural products in order to provide the product designer with choices for feasible assemblies. The synthesis of assemblies is done by decomposing a complex structure obtained via structural topology optimization into an assembly of multiple structural members with simpler geometries. The aim is at providing a systematic approach to explore a large number of decompositions prior to the detailed component design phase. Initially, the structure, which is represented as a bitmap image, is transformed to a graph with equivalent topology through application of image processing algorithms. Then, the obtained graph is decomposed by a genetic algorithm into subgraphs using strength-based criteria. Results for an example structure are given to clarify and discuss the method.

INTRODUCTION

Most structural products, from a simple chair frame to an automobile body, are manufactured through assembly of various components which have simpler geometries than the end product. Assembly synthesis is the decision of which components to assemble together to achieve the end product and is done by decomposition of the end product design prior to the detailed component design phase. From a structural point of view, it is not desirable to introduce joints into the structure since they will decrease the strength of the product. However, most designs are impossible to manufacture as one single

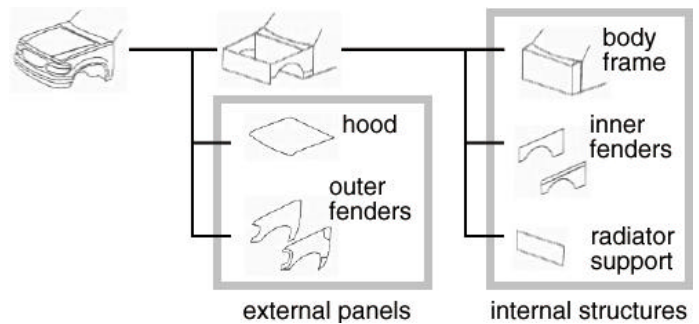


Figure 1. Example of decomposition of an automobile body front.

component. Therefore, assembly synthesis is an inevitable step in most design cases.

Figure 1 shows an example of the decomposition of an automotive body front consisting of the external panels (hood and outer fenders) and the internal structures (inner fenders and radiator support) that connect to the rest of the internal body frame. In industry, such decompositions are typically accomplished *ad hoc* prior to the detailed design of individual components. However, such decompositions might overlook one or more of the following criteria which cause problems in the detailed design phases costing time and money. The important criteria during the decomposition are:

- **Structural strength:** The joints introduced to the structure due to decomposition decrease the strength

* corresponding author

of the initial design. This effect has to be kept to a minimum.

- **Manufacturability and assembleability:** The decomposition should result in components and joint features/methods which are physically and economically feasible.

By observing these criteria during the assembly synthesis, problems in subsequent design phases can be substantially reduced. The method presented in this paper aims to achieve a systematic decomposition process with major emphasis on structural characteristics and minor emphasis on assembleability. Other criteria will be included in future work. The presented approach intends to provide the designer with feedback about possible decompositions prior to the detailed design phase. By applying the presented method prior to the detailed design phase, changes in detailed design due to structural and other issues are minimized which may be costly and time consuming.

In our method, a structure obtained via structural topology optimization is decomposed into an assembly consisting of multiple structural members with simpler geometries. There are two main steps in the process developed:

1. A two-dimensional bitmap image of a structure obtained via structural topology optimization is transformed to a product topology graph through application of image processing algorithms.
2. The product topology graph is decomposed into subgraphs by using a genetic algorithm which results in a decomposition of the product with chosen mating features.

Finally, results for an example structure are presented to demonstrate the method and to discuss the results.

RELATED WORK

Design for assembly (DFA) is a class of design methodologies for improving product design based on assembly considerations to realize easy and low-cost assembly (Boothroyd and Dewhurst, 1983). Based on the results from the seminal work on assembly modeling and sequence generation (Bourjault, 1984; Fazio and Whitney, 1987; de Mello and Sanderson, 1991; Lee and Shin, 1990), a number of researchers attempted the integration of assembly planning and DFA (de Fazio *et al.*, 1993; Lee *et al.*, 1993; Hsu, Lee and Su, 1993). Although proven effective, these approaches require detailed component geometry as input, hence limiting their application to early phases of the design process.

To overcome this limitation, Mantripragada *et al.* (Mantripragada, *et al.*, 1996; Mantripragada, 1998) developed a method to predict the propagation of dimensional variations based on Datum Flow Chain, a logical relationship among the

component dimensions. Although this approach does not require detailed component geometry, the product decomposition must be specified a priori by a designer.

Graph decomposition (Bosák, 1990) has been applied to mechanical design, such as model-based decomposition of design problems (Michelena and Papalambros, 1995), and manufacturing feature recognition (Rosen *et al.*, 1994). Among the most relevant is the application to automatic assembly sequence generation, where a disassembly sequence (assumed to be the reverse of an assembly sequence) is generated by sequential binary decompositions (*i.e.*, cuts) of a graph of connections of an assembled product (Bourjault, 1984; Fazio and Whitney, 1987; de Mello and Sanderson, 1991; Lee and Shin, 1990). Feasibility of each binary decomposition is determined by checking the precedence relationship of two subassemblies subject to partition using human input (Bourjault, 1984; Fazio and Whitney, 1987) or geometric reasoning (Lee and Shin, 1990).

Recently, Wang *et al.* (Wang and Bourne, 1997; Wang, 1997) developed a system which decomposes an unfolded sheet metal product based on the decomposition of a spanning tree of the face-adjacency graph of the product. Although their focus is similar to the presented research, the system does not consider the structural issues of the product. Also, the approach does not address the issue of dimensional errors in cutting, bending and joining.

ASSEMBLY SYNTHESIS METHOD

The method consists of two major procedures. First, the topology graph of a structure, which is obtained via structural topology optimization, is constructed by application of image processing algorithms on the two-dimensional bitmap image of the structure. The next step is the decomposition of the topology graph using a genetic algorithm.

Structural topology optimization

Structural topology design methods, such as the homogenization design method (Bendsøe and Kikuchi, 1988) and the genetic algorithm based (Chapman *et al.*, 1994), enable top-down synthesis of an optimal structure topology that fits within a specified design domain from the specification of loading and boundary conditions. As illustrated in Figure 2, these methods take as input the design domains and the loading and boundary conditions, and then produce through finite element analyses a discretized image (bitmap or grayscale) of an optimal material distribution in the design domain which, for example, maximizes stiffness at the loading point subject to weight constraints. In most cases, each pixel in the output image corresponds to a finite element. The methods also allow the design domain to be multiply-connected (*i.e.*, to have holes).

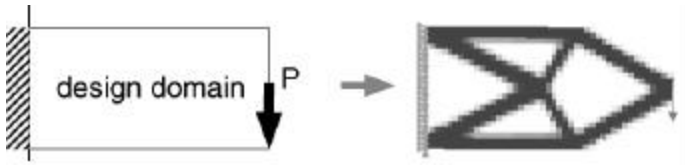


Figure 2. Structural topology design method: The right figure shows a structure with maximum stiffness occupying 40% of the design domain. The result is obtained by using Topology Optimization Web site at the Technical University of Denmark¹.

Construction of product topology graphs

Figure 3 outlines the flow of the transformation process. First, topology of the output image is extracted by identifying the distinct segments in the output image (Figure 3 (b)). Next, the resulting segments are labeled, and their connectivities are checked to produce a product topology graph (Figure 3 (c)). The extraction of product topology is accomplished by the successive application of standard digital image processing algorithms such as dilation, skeletonization, and the Hough transform (Gonzalez and Wintz, 1987), as well as non-standard algorithms such as primary line extraction and topological segmentation.

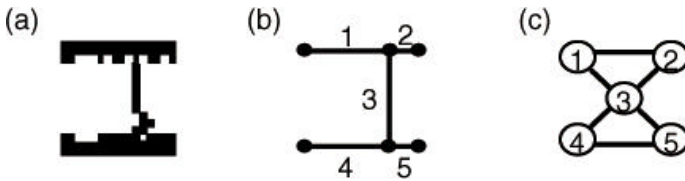


Figure 3. Transformation of a structural topology optimization output to a product topology graph. (a) output image, (b) extraction of product topology, and (c) resulting product topology graph. The I-beam like image was adopted from (Chapman *et al.*, 1994).

Figure 4 illustrates a sequence of these image transformations applied to an example I-beam like image (Chapman *et al.*, 1994) using the preliminary implementation of the transformation algorithms. A brief description of each step is given in the following. Although the description assumes a bitmap image as an input, it can be easily generalized to a gray scale image with a prior application of an appropriate thresholding method.

Dilation

It fattens the image by filling small, isolated holes and expanding the image boundary (Figure 4 (b)). It scans the image and turns a pixel on if a majority of the neighboring pixels are also on. Definition of majority and neighbor

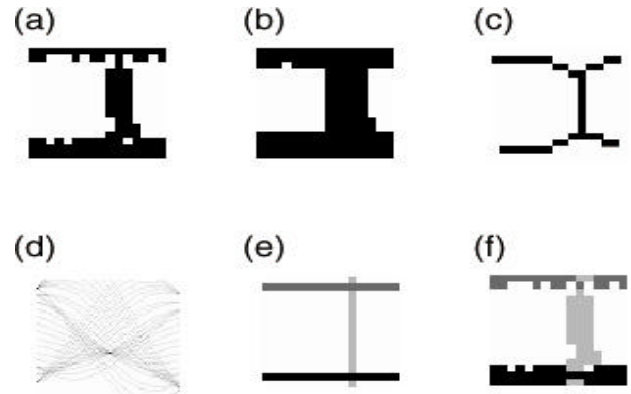


Figure 4. An example of product topology extraction: (a) original image, (b) dilation, (c) skeletonization, (d) initial Hough transform (shown in $q-r$ space), (e) primary line extraction, and (f) topological segmentation.

determines the effects of fattening. Since dilation is to eliminate small voids or non-smooth edges prior to skeletonization, it is unnecessary for smooth images such as the structure in Figure 2.

Skeletonization

It has an opposite effect to dilation. It thins the image by expanding small, isolated holes and shrinking the image boundary (Figure 4 (c)). It scans the image and turn off a pixel if a majority of the neighboring pixels are also off. Definition of majority and neighbor determines the effects of thinning. Applying dilation before skeletonization prevents the resulting skeleton from being affected by noises in the original image.

Hough transform

It detects lines in the skeletonized image by mapping the image in the $x-y$ space to a parameter space (the $q-r$ space) using the normal representation of a line in $x-y$ space:

$$x \cos q + y \sin q = r \quad (1)$$

Since a pixel (x_i, y_i) in $x-y$ space corresponds to a sinusoidal curve $x_i \cos q + y_i \sin q = r$ in the $q-r$ space, collinear pixels in the $x-y$ space have the intersecting sinusoidal lines in the $q-r$ space. Conversely, an intersection point (q_n, r_n) in the $q-r$ space corresponds to a line in the $x-y$ space. Therefore, all lines passing through arbitrary pairs of pixels in the image are found by checking the intersection points in the $q-r$ space. Discretization of the $q-r$ space for computing the sinusoidal lines determines the accuracy of the detected lines.

The Hough transform is repeatedly applied in the primary line extraction algorithm described below. Figure 4 (d) shows the $q-r$ space from the initial application to the skeletonized image. A generalized form of the Hough transform uses a

¹ <http://www.topopt.dtu.dk/>

spline representation of a curve to detect arbitrary curves in the image (Ballard, 1981), which will be incorporated in the future implementation.

Primary line extraction

This algorithm abstracts the topology of the skeletonized image by selecting primary lines in the x - y space based on the number of pixels they pass through. Basic procedure is as follows.

1. Do the Hough transform of the image.
2. Select an intersection point in the q - r space with the maximum number of intersecting lines. If the maximum number is below a prespecified value, return.
3. Remove pixels in the image corresponding to the intersecting lines in 2 and goto 1.

Figure 4 (e) shows the extracted three primary lines shown in a different gray scale. The prespecified value in the step 2 to cut off the iteration determines the “level of abstraction” of the extracted topology. The repeated application of the Hough transform is needed since the pixels removed in step 3 may also have been on the lines not selected in step 2. However, it does not generally add a significant computational overhead since the number of unremoved pixels rapidly decreases after a few iterations.

Topological segmentation

It associates each pixel in the original image to each primary line identified above. For each pixel in the original image, it calculates the distances to all primal lines, and associates the pixel with the primal line with the minimum distance. Figure 4 (f) shows the original image shaded with three distinct segments corresponding to the three primal lines in Figure 4 (e).

Occasionally, however, the topological segmentation yields a segmented image with stand-alone (disconnected) pixels at the intersection of multiple segments. In such cases, the following post-processing is necessary to re-assign these pixels to another primal line so they can be connected.

Pixel re-assignment

1. Select a segment in an image obtained by the topological segmentation. If all segments have been checked, return.
2. Count the number of disconnected sub-segments. If the number is one, go to 1. Otherwise let the largest sub-segment be the primal segment.
3. For each pixel not belonging to the primal segment, re-assign the pixel to the primal line to which the majority of the surrounding pixels belong.
4. Go to 1.

After the completion of the extraction of product topology, a product topology graph can be easily constructed by labeling each edge of the primary lines and each intersection point among these edges, with a node and an edge in a graph data structure, respectively.

Decomposition of product topology graphs

The structural members are synthesized by decomposing the product topology graph and the corresponding product geometry as illustrated in Figure 5.

The following assumptions are made at this stage:

- Joining method at every joint is spot weld; hence joints are strong for shear and compressive loading, but weak in tensile loading.
- The only joint feature considered is the weld angle which is chosen from discrete set of possible values.
- Number of components desired is specified by designer.
- The structure can be decomposed only at the location corresponding to the edges of the topology graph.

The graph decomposition problem is treated as a discrete optimization problem and solved using a genetic algorithm (GA). The following sections formulate the mathematical model of the optimization problem and describe the method used to solve it which is GA.

Mathematical model

Definition of the design variables

The optimal decomposition can be posed as a graph partitioning problem (Bosák, 1990). The members of the structure are mapped to nodes and the intersections are mapped to multiple edges since they can be joining more than two members. Therefore, the whole structure can be represented as a graph $G=(V, E)$ with node set V and edge set E . The problem of optimal decomposition becomes one of finding a partition, P , of the node set V such that the objective function, $c(P)$, is maximized.

Note that the number of non-empty subsets, k , of V , which constitute the optimal partition, is not defined. Mating features at the joints are major factors affecting structural strength therefore a set, F , of mating features must also be defined to be able to evaluate different decompositions. As based on previously stated assumptions, F is the set of possible mating angles at the welded joints, *i.e.*, the only mating feature considered is the weld angle.

The optimal partitioning of G can be represented mathematically by a vector $x=(x_i)$ where x_i is a binary variable representing the presence of edge e_i in the decomposition defined by the partitioning P . From the above, it is obvious that $i=1, \dots, |E|$ since there are $|E|$ edges in the topology graph.

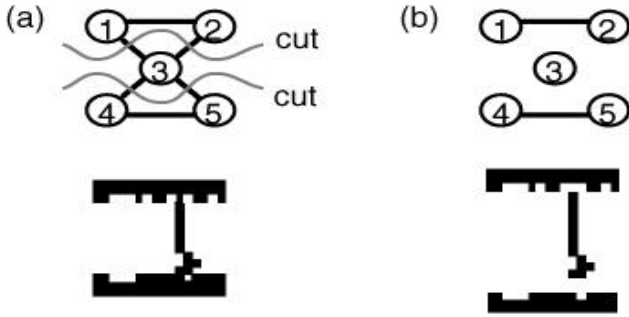


Figure 5. Decomposition of a product topology graph and the corresponding product geometry before decomposition (a) and after decomposition (b).

Finally the problem is defined as:

Given G , the topology graph of the structure, F , set of mating features and k , number of substructures, find vector x , partition representing the optimal decomposition, and vector y , the mating feature for each joint subject to $c(x,y)$, which is a cost function evaluating the decomposition quality.

Definition of the constraints

The constraint on the vector x , which represents the presence of edges, is as follows:

$$\text{COMPONENTS}(\text{GRAPH}(x)) = k, \quad (2)$$

where

- $\text{GRAPH}(x)$ returns the graph after the edges with $x_i = 0$ in vector x , have been removed from the original topology graph
- $\text{COMPONENTS}(G)$ returns the number of disconnected components in graph G

This constraint translates into the fact that the problem asks for a decomposition consisting of k different structural members.

The constraint on vector y is as follows:

$$y_i \in F \quad (3)$$

where F is the set of mating angles at which spot welds can be applied at the joints. One element of set F represents the case for no weld at the corresponding joint.

The last constraint is imposed on the combination of the vectors x and y in the following way:

$$\text{CHECK_CONNECTIVITY}(\text{COMBINED_GRAPH}(x, y)) = 1 \quad (4)$$

where

- $\text{CHECK_CONNECTIVITY}(G)$ is a function which returns 1 if the graph G is connected and returns 0 otherwise.

- $\text{COMBINED_GRAPH}(x,y)$ is a function that returns a graph which consists of the nodes of the original graph and the edges in vectors x, y .

This constraint ensures that the combination of the decomposition given by vector x and the mating angles given by vector y constitutes a structure which has the same connectivity as the original disconnected structure.

Definition of the objective function

Objective function will evaluate each decomposition according to the following criteria:

- Reduction of structural strength due to introduction of joints
- Assembleability of the decomposed structure

To evaluate the decomposition according to the structural strength criteria, the normal stress at the joints and the area on which the normal stress acts are calculated. The evaluation is based on the difference between the angle, at which the normal stress is minimum, q_i^{ideal} , and the chosen welding angle given by vector y because deviation from the ideal angle means higher normal stress. The stress at the chosen angle multiplied by the weld area provides a measure of force acting on the weld which is also used in evaluating the decrease in strength. A weld with larger area introduces a higher amount of decrease in strength than a weld with smaller area.

To evaluate the decomposition according to the assembleability criteria, the similarity of weld angles and the number of welds in the decomposition are taken into account. Obviously, lower number of welds and similar weld angles result in higher assembleability.

These criteria result in the following objective function:

$$f(x, y) = w_1 \sum_{i=1}^{N_{welds}} (q_i - q_i^{ideal})^2 + w_2 \sum_{i=1}^{N_{welds}} (s_i(q_i) A_i(q_i)) + w_3 \sum_{i=1}^{N_{welds}} \sum_{j=i+1}^{N_{welds}} (q_i - q_j)^2 + w_4 N_{welds} \quad (5)$$

The variables in the objective function are defined as follows:

- $x=(x_i)$ x_i is a binary variable representing the presence of edge e_i in subset x
- $y=(y_i)$ y_i is discrete variable representing the choice of weld angle at joint i
- w_i weight of i th criteria in the objective function
- N_{welds} total number of welds in the decomposed structure
- q_i weld angle with respect to vertical direction at joint i
- q_i^{ideal} angle of minimum normal stress at joint i

$s_i(\mathbf{q}_i)$ normal stress at joint i at angle \mathbf{q}_i
 $A_i(\mathbf{q}_i)$ weld area at joint i (function of \mathbf{q}_i)

The constraints and objective function combine to give the following optimization problem:

minimize $f(x, y)$ (as defined above)
 subject to
 $x_i \in \{0,1\}$, $i = 1, \dots, |E|$
 $y_i \in F$, $i=1, \dots, |E|$ where F is the set of mating angles
 $\text{COMPONENTS}(\text{GRAPH}(x)) = k$,
 $\text{CHECK_CONNECTIVITY}(\text{COMBINED_GRAPH}(x,y)) = 1$

Optimization method and implementation

Genetic algorithm

Graph partitioning problem is NP-complete (Garey and Johnson, 1979) even with simple linear criteria, which is to say that no polynomial-time algorithm is likely to exist. As a consequence all known algorithms that solve graph partitioning problems exactly have run-times exponential to the size of the graph. In this case the cost function is nonlinear and since we cannot afford exponential computation, heuristic algorithms are found to be suitable. More specifically, a genetic algorithm has been used to solve the problem approximately, i.e., the algorithm may not give an optimal solution all the time.

The decomposition problem has been solved by using a steady-state genetic algorithm. Genetic algorithms (Goldberg, 1989; Holland, 1975) are a compromise between random and informed search methods, where variables are mapped to chromosomes and new generations are created by cross-over (combining portions of two chromosomes) and mutation (randomly changing the values in each gene in a chromosome) until a termination condition is satisfied. In this project, the termination condition is satisfied when a given number of generations have evolved. A steady-state genetic algorithm (Davis, 1991) has been used and the following is the basic flow of steady-state GA:

1. Randomly create a population P of n chromosomes (an encoded representation of design parameters \mathbf{x} and \mathbf{y}) and evaluate their fitness values and store the best chromosome. Also create an empty subpopulation Q .
2. Select two chromosomes c_i and c_j in P with probability

$$\text{Prob}(\text{chromosome } c_i \text{ is selected}) = \frac{f_i}{\sum_{k=0}^n f_k}$$

where f_i is the fitness value of chromosome c_i

3. Crossover c_i and c_j to generate two new chromosomes c_i' and c_j' .
4. Mutate c_i' and c_j' with a certain low probability.

5. Evaluate the fitness values of c_i' and c_j' and add them in Q . If Q contains less than m new chromosomes, go to 2.
6. Replace m chromosomes in P with the ones in Q and empty Q . Update the best chromosome and increment the generation counter. If the generation counter has reached a pre-specified number, terminate the process and return the best chromosome. Otherwise go to 2.

Known empirical advantages of steady-state GA (Davis, 1991) are that SSGA prevents premature convergence of population and reaches an optimal solution with fewer number of fitness evaluations.

Chromosome representation of the problem

Each solution is encoded in a chromosome of the following nature. The chromosome is of length $2|E|$. First $|E|$ genes carry binary information about which edges of the topology graph are kept and which are removed to produce a decomposition. If the i^{th} element of the chromosome is 0, it means that this edge has been cut in this particular decomposition represented by this chromosome.

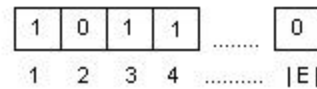


Figure 6. First half of chromosome with binary information

The second half of the chromosome carries the information about which discrete choice of possible mating angles is chosen for a given joint. The $(|E|+i)^{\text{th}}$ element carries the choice of mating angle for the i^{th} joint(edge in the graph).

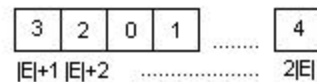


Figure 7. Second half of chromosome with mating angle information

For this project, the possible mating angles have been chosen as $-45, 0, 45, 90$ degrees from the vertical and map to gene values of 1, 2, 3, 4, respectively. A gene value of zero means no weld at that intersection.

Please note that the information carried by the first two half of the chromosome maps to vector \mathbf{x} and the second half maps to vector \mathbf{y} in the mathematical model. It should also be noted that not every chromosome configuration will give a “real” cut. Leaving out of some edges will still keep the whole structure connected, therefore, every decomposition has to be checked whether it results in a graph with at least two disconnected components. Obviously, ideal case is k -disconnected components.

Since chromosomes representing the decompositions carry two different kinds of information (x_i is binary and $y_i \in F$) the cross-over and the mutation operators have been customized. The cross-over operator treats the first and second halves of the chromosome separately. It implements the cross-over in two steps in which the first halves of chromosomes are crossed with first halves, and second halves are crossed with second halves which is practically a multi-point cross-over operator. Figure 8 illustrates the cross-over of two chromosomes in the way described.

The mutation operator is different from traditional mutation operators in that it increases the probability of mutation for the second half of the chromosome.

Fitness evaluation of chromosomes

Since genetic algorithms do not handle constraints directly, the constraints in the mathematical problem formulation have to be translated into penalty terms. Therefore, the fitness function will consist of two main terms which are the objective function value $f(x,y)$ of the decomposition and the penalty term which imposes the constraints of the mathematical model.

$$\text{Fitness} = f(x,y) + \text{Penalty terms} \quad (6)$$

Recall that the objective function has the following form:

$$f(x, y) = w_1 \sum_{i=1}^{N_{welds}} (\mathbf{q}_i - \mathbf{q}_i^{ideal})^2 + w_2 \sum_{i=1}^{N_{welds}} (\mathbf{s}_i(\mathbf{q}_i) A_i(\mathbf{q}_i)) + w_3 \sum_{i=1}^{N_{welds}} \sum_{j=i+1}^{N_{welds}} (\mathbf{q}_i - \mathbf{q}_j)^2 + w_4 N_{welds} \quad (5)$$

The mating angle from the vertical direction, \mathbf{q}_i , is calculated using the value in the $(|E|+i)^{th}$ gene in the following way:

$$\mathbf{q}_i = (\text{gene}(|E| + i) - 2) \times 45^\circ \quad (7)$$

where $\text{gene}(|E| + i) \in F = \{0, 1, 2, 3, 4\}$ where 0 means no weld.

Ideal angle at the i^{th} joint, \mathbf{q}_i^{ideal} , and stress tensors for each weld are obtained through an initial finite element analysis and stored in lookup tables. Normal stress at joint i at angle \mathbf{q}_i , $\mathbf{s}_i(\mathbf{q}_i)$, is calculated using the stress tensor and the mating angle corresponding to the weld. A_i is the weld area which is calculated using the weld dimensions and the mating angle, *i.e.*, it is a function of the mating angle, \mathbf{q}_i . Weld dimensions are calculated initially and stored in lookup tables.

Number of welds, N_{welds} , is the sum of edges for which the corresponding gene in the first half is zero and the gene in the second half is non-zero. Note that a zero in the first half means

that the intersection has been cut and a non-zero value in the second half means that there is a weld.

The constraint on vectors \mathbf{x} and \mathbf{y} are imposed simply by the chromosome representation of the problem, *i.e.*, genes in the first half of the chromosome are binary values imposing the constraint $x_i \in \{0, 1\}$ and genes in the second half of the chromosome can only have values from 0 to 4 imposing the condition $y_i \in F$, where F is the set of possible mating angles.

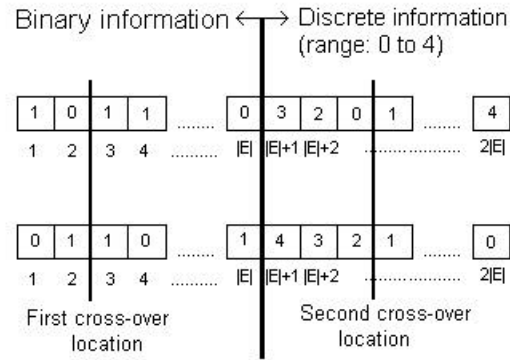


Figure 8. Cross-over of two chromosomes

The constraint on the number of components is imposed as a penalty in the fitness term. Recall that the constraint has the following form:

$$\text{COMPONENTS}(\text{GRAPH}(\mathbf{x})) = k \quad (2)$$

The corresponding penalty term is formulated is as follows:

$$\text{Penalty} = w (\text{COMPONENTS}(\text{GRAPH}(\mathbf{x})) - k)^2$$

The last constraint on the whole decomposition configuration,

$$\text{CHECK_CONNECTIVITY}(\text{COMBINED_GRAPH}(\mathbf{x}, \mathbf{y})) = 1$$

is implemented by returning a fitness of infinity (very large number in the software implementation) for decompositions lacking connectivity, *i.e.*, returning 0 when passed to the CHECK_CONNECTIVITY function. Structurally disconnected decompositions, which are not feasible, are eliminated by this constraint implementation.

The resulting fitness function has been realized by the implementation of the following algorithm:

Fitness (x, y)

```

If genome results in disconnected structure
    return infinity (very large number)

```

Else

$$\begin{aligned} \text{Fitness}_{\text{total}} = & w_1 \sum_{i=1}^{N_{\text{welds}}} (\mathbf{q}_i - \mathbf{q}_i^{\text{ideal}})^2 \\ & + w_2 \sum_{i=1}^{N_{\text{welds}}} (\mathbf{s}_i A_i(\mathbf{q}_i)) \\ & + w_3 \sum_{i=1}^{N_{\text{welds}}} \sum_{j=i+1}^{N_{\text{welds}}} (\mathbf{q}_i - \mathbf{q}_j)^2 \\ & + w_4 N_{\text{welds}} \\ & + w_5 (\text{COMPONENTS}(\text{GRAPH}(\mathbf{x})) - k)^2 \end{aligned}$$

return Fitness_{total}

End If

Software implementation

The implementation has been done on the Unix platform using the C++ programming language. For graph representation and related algorithms the LEDA library² developed at the Max-Planck Institute of Computer Science has been used. For GA implementation the Galib library³ developed at the Massachusetts Institute of Technology was used. For the finite element analysis of the structure Abaqus⁴ was used.

EXAMPLES

This section describes the topology extraction and decomposition of an example structure. The example structure (right image of Figure 2) is a cantilevered structure fixed at a vertical boundary subject to downward load P. The image is obtained by using the Topology optimization Web site at the Technical University of Denmark¹ and converting gray-scale information to binary information.

Construction of Product Topology Graph

As shown in Figure 9 (f), all distinct “members” in the input image are successfully segmented as a result of the image transformation. Note since the original image in Figure 9 (a) is already smooth, dilation has little effect as evident from Figure 9 (b). The dilation algorithm has been applied using a 3x3 window and the Hough transform has been applied using 64 discretization levels for both dimensions of the $\mathbf{q}-\mathbf{r}$ space.

Figures 10 (a) and (b) show the extracted product topology with the label for each member and the resulting product topology graph, respectively. On the topology graph, the intersections, which are candidates for joints are also labeled.

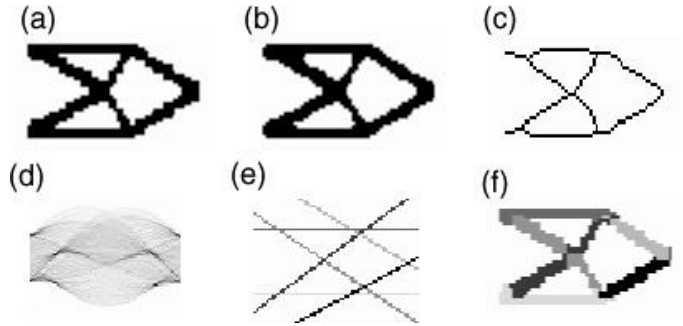


Figure 9. Result of topology extraction (a) original image, (b)dilation, (c) skeletonization, (d) initial Hough transform (shown in $\mathbf{q}-\mathbf{r}$ space), (e) primary line extraction, and (f) topological segmentation.

The next stage involves the application of the optimization method, namely GA, to the obtained product topology graph. This stage is preceded by the intermediate stage which is standard finite element analysis of the structure. The results of the finite element analysis are stored in a look-up table to be used during chromosome evaluation.

The following GA parameters were used:

- Population size = 500
- Number of generations = 1000
- Replacement percentage = 30%
- Cross-over probability = 0.9
- Mutation probability = 0.04 (double for second half of chromosome since it carries non-binary information)

Two cases with different desired number of components have been solved for this structure. In the first case the number of desired components was selected as three, whereas in the second case the number of desired components solution was selected as four.

Three-component decomposition

The solution that the optimization algorithm found when three components were specified was represented by the chromosome shown in Table 1. Figure 11 illustrates the decomposition of the product topology graph that this chromosome maps to and the corresponding product decomposition.

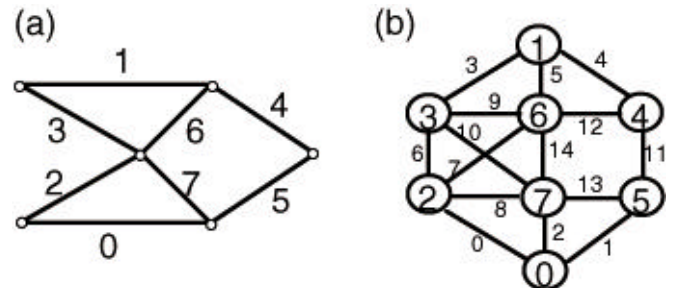


Figure 10. Construction of the product topology graph: (a)extracted product topology and (b) the resulting product topology graph.

¹ <http://www.topopt.dtu.dk/>

² <http://www.mpi-sb.mpg.de/LEDA/>

³ <http://lancet.mit.edu/ga/>

⁴ Abaqus by Hibbitt, Karlsson and Sorensen, Inc.

Table 1. The resulting chromosome for the 3-component solution

Edge no	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1st half of chromosome	1	1	1	1	0	1	0	0	1	1	0	0	0	1	0
2nd half of chromosome	4	1	4	1	0	4	0	0	0	4	0	3	3	3	3

The dashed lines in the graph represent the edges which have been cut during the decomposition and have been selected as the welding points. It is noteworthy that only some of the edges in the graph, which have been cut, have been selected as welding locations which is a result of the objective to minimize number of welds. However, the connectivity of the initial structure is still preserved.

The results obtained satisfy the defined criteria for a good decomposition. Desired number of components of decomposed structure has been achieved, *i.e.*, the decomposition has three components. Chosen weld angles eliminate tensile stress on welds completely whereas the objective was to minimize tensile stress. All welds are under compression. For three components a decomposition with three welds has been found where three is the minimum number of welds needed to preserve connectivity. The objective to have similar weld angles has also been achieved. This can be observed by examining columns with zero in the first row (meaning that the edge has been cut and a non-zero value in the second row (meaning a weld has been placed). All welds are chosen to be at angle three which maps to 45° from the vertical direction.

Four-component decomposition

The solution that the optimization algorithm found when four components were specified was represented by the chromosome shown in Table 2. Figure 12 illustrates the decomposition of the product topology graph into four components that this chromosome maps to and the corresponding product decomposition.

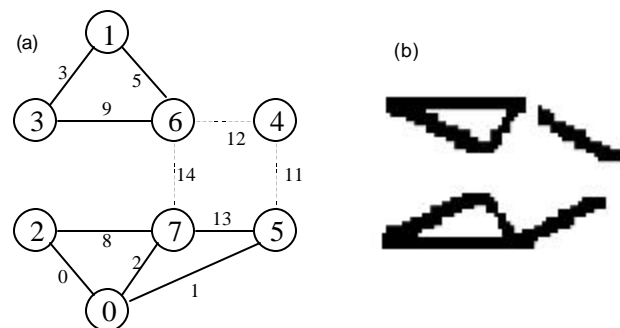


Figure 11. Resulting decomposition of (a) the product topology graph, and (b) the structure decomposed into 3-components

As in the three-component decomposition case, desired number of components of decomposed structure has been achieved, *i.e.*, the decomposition has four components. Chosen weld angles eliminate tensile stress on welds completely whereas the objective was to minimize tensile stress. All welds are under compression.

The minimum number of welds needed to preserve connectivity has been achieved which is four. The objective to have similar weld angles has also been achieved, *i.e.*, all welds are at angle three which maps to 45° from the vertical direction

Table 2. The resulting chromosome for the 4-component solution

Edge no	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1st half of chromosome	1	0	1	1	0	1	0	0	1	1	0	0	0	0	0
2nd half of chromosome	3	0	3	3	0	3	0	0	3	4	0	3	3	3	3

DISCUSSION AND FUTURE WORK

Both results obtained satisfy the defined criteria for a good decomposition. Desired number of components of decomposed structure has been achieved in both results. Chosen weld angles eliminate tensile stress on welds, *i.e.*, all welds are under compression. Number of welds has been minimized in all solutions. The objective to have similar weld angles has also been achieved. In both solutions all welds are at the same angle.

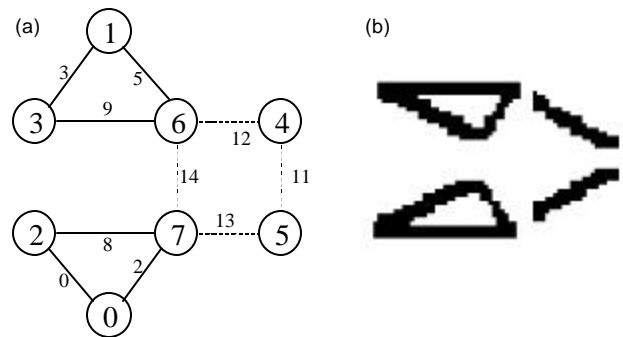


Figure 12. Resulting decomposition of (a) the product topology graph, and (b) the structure decomposed into 4-components

The presented method proves to be valuable in evaluating candidate decompositions according to the defined criteria. With extension of criteria used in evaluation, the method can be employed as a tool in the decomposition phase of the design process which takes important aspects into account that may be overlooked by *ad hoc* decomposition by designers.

As future work, most important improvement will be incorporating additional objectives into the problem formulation and as a result in the fitness evaluation. Additional criteria can include manufacturability and assembleability based on geometry of each component. Furthermore, complex

modeling of joint features can be done to achieve more accurate evaluation of effect of joints on structural stiffness and strength. Ultimately, extension to 3-D structures is aimed to extend the application area of the method devised in this research.

ACKNOWLEDGMENTS

Partial support for this work has been provided by Toyota Motor Corporation and the Horace H. Rackham School of Graduate Studies at the University of Michigan. These sources of support are gratefully acknowledged.

REFERENCES

- D. H. Ballard, 1981, "Generalizing the Hough transform to detect arbitrary shapes", *Pattern recognition*, 13(2):111-122, 1981.
- M. P. Bendsøe and N. Kikuchi, 1988, "Generating optimal topologies in structural design using a homogenization method", *Computer Methods in Applied Mechanics and Engineering*, 71: 197-24.
- G. Boothroyd and P. Dewhurst, 1983, *Design for Assembly Handbook*, University of Massachusetts, Amherst.
- J. Bosák, 1990, *Decompositions of Graphs*, Kluwer Academic Publications, Boston.
- A. Bourjault, 1984, "Contribution a une Approche Méthodologique de L'Assemblage Automatisé: Elaboration Automatique des Séquences Opératoires", Ph.D. Thesis, Université de Franche-Comté, Besançon, France.
- C. D. Chapman, K. Saitou, and M.J. Jakiela, 1994, "Genetic algorithms as an approach to configuration and topology design", *ASME Journal of Mechanical Design*, 116:1005-1012.
- L. David, 1991, *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York.
- T. de Fazio, A. Edsall, R. Gustavson, J. Hernandez, P. Hutchins, H.-W. Leung, S. Luby, R. Metzinger, J. Nevins, T. Tung and D. Whitney, 1993, "A prototype of feature-based design for assembly", *ASME Journal of Mechanical Design*, 115:723-734.
- T. de Fazio and D. Whitney, 1987, "Simplified generation of all mechanical assembly sequences", *IEEE Journal of Robotics and Automation*, RA-3(6):640-658. Corrections on same journal, RA-4(6):705-708, 1988.
- M. R. Garey and D. S. Johnson, 1979, *Computers and Intractability, a guide to the theory of NP-completeness*, W. H. Freeman and Company, New York.
- D. Goldberg, 1989, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Massachusetts.
- R. C. Gonzalez and P. Wintz, 1987, *Digital Image Processing*, Addison Wesley, Reading, Massachusetts, second edition.
- J. Holland, 1975, *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, Michigan.
- L. Hommem de Mello and A. Sanderson, 1991, "A correct and complete algorithm for the generation of mechanical assembly sequences". *IEEE Transactions on Robotics and Automation*, 7(2):228-240.
- W. Hsu, C. Lee and S. Su, 1993, "Feedback approach to design for assembly by evaluation of assembly plan", *Computer-Aided Design*, 25(7):395-410.
- S. Lee, G. Kim and G. Bekey, 1993, "Combining assembly planning with redesign:an approach for more effective DFA", In *Proceedings of 1993 IEEE International Conference on Robotics and Automation*, pages 319-325.
- S. Lee and Y. Shin, 1990, "Assembly planning based on geometric reasoning", *Computer and Graphics*, 14(2):237-250.
- R. Mantripragada, 1998, "Assembly oriented design: Concepts, Algorithms and Computational Tools", PhD Thesis, Massachusetts Institute of Technology, Cambridge.
- R. Mantripragada, T. Cunningham and D. Whitney, 1996, "Assembly oriented design: a new approach to designing assemblies", In *Proceedings of IFIP Workshop on Geometric Modeling and CAD*, pages 308-324, Airlie, VA.
- N. Michelena and P. Papalambros, 1995, "Optimal model-based decomposition of powertrain system design", *ASME Journal of Mechanical Design*, 117(4):499-505.
- D. Rosen, J. Dixon and S. Finger, 1994, "Conversions of feature-based design representations using graph grammar parsing", *ASME Journal of Mechanical Design*, 116:785-792.
- C.-H. Wang, 1997, "Manufacturability-Driven Decomposition of Sheet Metal Products", PhD thesis, Carnegie Mellon University.
- C.-H. Wang and D. Bourne, 1997, "Concurrent decomposition for sheet-metal products", In *Proceedings of 1997 ASME Design Engineering Technical Conference*, Sacramento, CA.